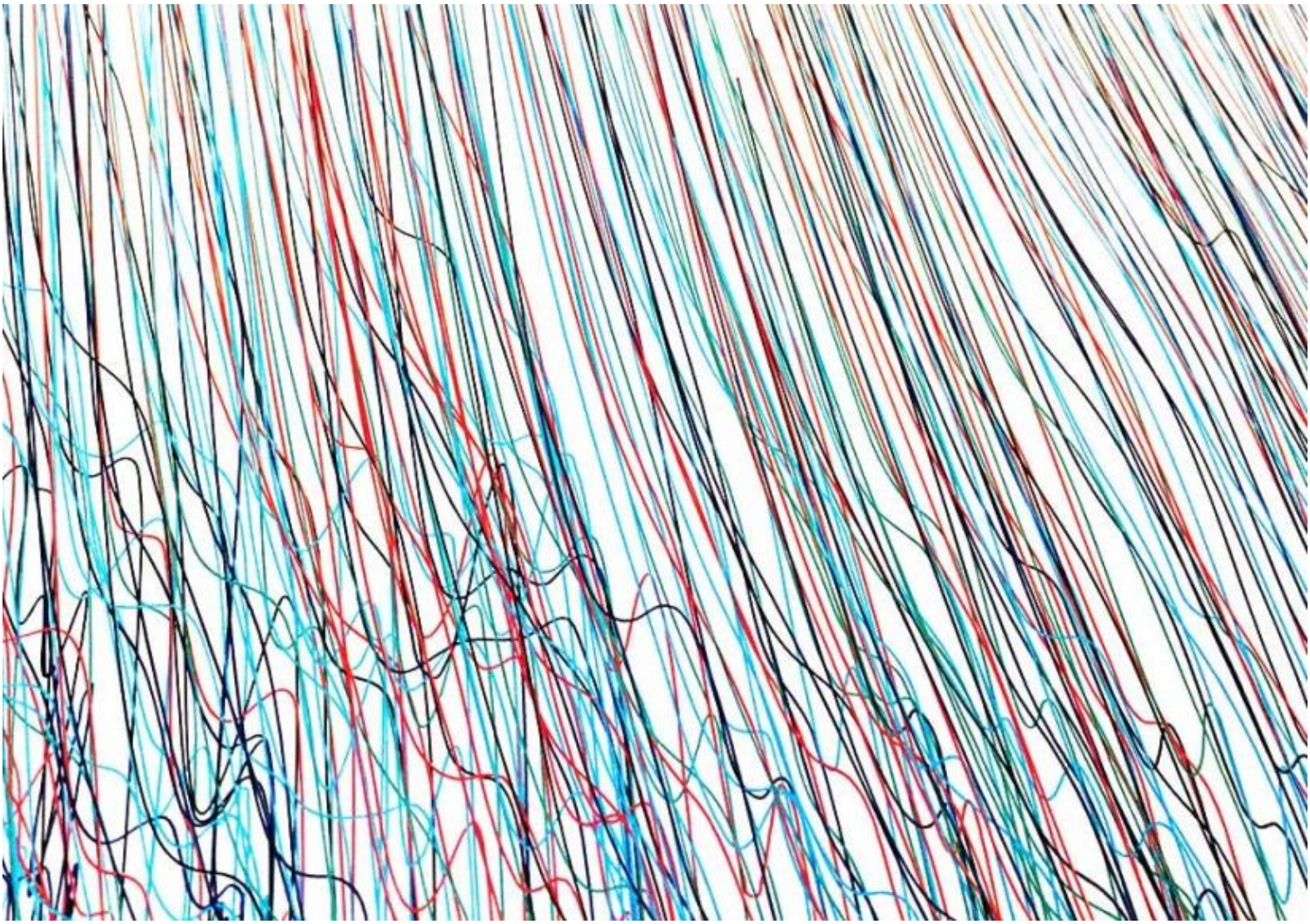
UNIVERSITAT DE BARCELONA

Computational Infrastructures for biomolecular research

Laia Codó Tarraubella



Computational infrastructures for biomolecular research

Laia Codó Tarraubella

2019

Als meus pares

UNIVERSITAT DE BARCELONA

FACULTAT DE BIOLOGIA
DEPARTAMENT DE BIOQUÍMICA I BIOMEDICINA MOLECULAR

PROGRAMA DE DOCTORAT EN BIOMEDICINA. LINIA DE RECERCA BIOINFORMÀTICA

Computational infrastructures for biomolecular research

Memòria presentada per Laia Codó Tarraubella per optar al grau de doctor/a per la Universitat de Barcelona

Laia Codó Tarraubella — Doctorant

Josep Lluís Gelpí Buchaca — Director de tesis

Acknowledgments

Voldria fer menció de tots aquells que han fet possible que aquest treball vegi la llum, però això seria impossible, en són massa.

Així que em limitaré a unes poques paraules d'agraïment, primer i sense lloc a dubte, dirigides al director d'aquesta tesi, i que ves per on, va ser la persona que fa 12 anys m'explicava com funcionava això de les adreces IP. Gràcies pels sempre savis consells, la inestimable i constant ajuda, i les ingents dosis d'optimisme.

Tampoc puc obviar als meus companys, dels que tant he après, i tan estoicament m'han aguantat durant aquests anys. Un gràcies molt gran per les meves dos "pachis" preferides, Romina i Montse, pel bo del Dmitry, per la resta de nens del Gelpí, Pau, Lluís i Dani, pels "Bencheros", i com no, pels de sota la Diagonal, Adam, Genís i cia. Diu que a la feina no s'hi va fer amics, però de vegades te'ls hi trobes!

I com aquest ha estat un llarg camí, voldria agrair l'ajuda d'aquells que ja estan en altres contrades, com el Jose, la Chiara, l'Alexis, el Víctor, el Brian, la Laura ... Com passa el temps. A més, no puc deixar de nomenar els Javis de la Rosa, sempre al peu del canó, i la gent de Support i Sysadmin del BSC.

Finalment, voldria agrair el suport incondicional dels de casa, en especial, al Josep, als meus pares i a la xica, els quals de tant en tant encara em pregunten de què va la tesi.

Moltes gràcies a tots!

Abstract

In the last decade, research processes in Life sciences have evolved at a rapid pace. This evolution, mainly due to technological advances, offers more powerful equipment and generalizes the digital format of research data. In the data deluge context, we need to overcome the current tsunami of data. Making it available raises important challenges in terms of scalability, computing needs, or complexity and costs of storage. At the same time, privacy and security are important issues to be also taken into account. Hence, the current model, consisting to regularly add hardware resources into centralized core facilities without global coordination, is no longer sustainable.

Scientific data management and analysis processes should be enhanced in order to offer services and developments corresponding to the new e-Science uses, and infrastructures are the vehicles to achieve so. They are defined as the combination of technologies, computational resources and support operations that facilitate the collaboration among research communities by sharing resources, analysis tools and data. We propose and implement our research support infrastructures in line with this new science directives, adapting them to the scenarios presented by the divergent use cases. Three different domain-specific infrastructures framed in three different scientific projects are assembled and introduced in this dissertation.

The first case is framed in the clinical data management field, and focuses on the data platforms build around two epidemiologic case studies on Immune Mediated Inflammatory diseases (IMIDs), IMID-clinica and IMID-longitudinal.

Making the leap to infrastructures more oriented to analysis process support, the transPLANT infrastructure represents a first intrusion into the topical cloud computing model. It is focused on plant genomics and offers portable, tailored and elastic compute services based on virtual machines provisioned on-demand.

The design became the seed for a more comprehensive and integrative solution, MuG Virtual Research Environment (MuGVRE). The web-based framework seamlessly integrates cloud-based resources for a selection of data, visualization, and analysis services relevant for the 3D/4D genomics community. Designed as an expandable framework, research software developers might contribute to the platform by adding their own analysis methods in an easy and efficient way.

Becoming obvious the transversal potential of cloud-based computational infrastructures as virtual research environments, openVRE is implemented as an abstraction of MuGVRE. It offers a vanilla platform encompassing computation, data and administration services ready to be adopted and customized by other scientific communities.

They all represent an opportunity to establish better research processes through enhanced collaboration, data management, analysis practices and resource optimization.

The thesis dedicates a separated section at *4*. Results chapter for each of these platforms, prior introduction to the overall ecosystem of related research practices and technologies at 1. Introduction chapter, which are further explored in *3*. Software, Data & Methodology in case of being directly involved in the work. Finally, in chapter *5. Discussion* is gathered a global analysis of the exposed infrastructures with some final remarks and considerations in *6. Conclusions*.

Contents

Index

ABSTRACT ..... **IX**

CONTENTS ..... **XIII**

1 . INTRODUCTION **1**

1.1 BIOINFORMATICS EVOLUTION 3

1.2 THE CREATIVE CHAOS 4

1.3 E-SCIENCE 7

1.3.1 Transversal Infrastructures *8*

1.3.2 New science needs *10*

1.4 ENABLING TECHNOLOGIES 13

1.4.1 Distributed computing *13*

1.4.2 Virtualization *17*

1.4.3 Cloud technology *23*

1.4.4 Data Management *38*

1.5 SCIENCE ON THE WEB 43

1.5.1 Service-Oriented Science *43*

1.5.2 Web applications *44*

2 . OBJECTIVES **49**

3 . SOFTWARE, DATA & METHODOLOGY **53**

3.1 SOFTWARE COMPONENTS 55

3.1.1 Databases technologies *55*

3.1.2 Cloud-related software *57*

3.1.3 Job Managers *63*

3.1.4 Authentication *65*

3.1.5 Summary of web applications *66*

3.2 OPEN STANDARDS 66

3.2.1 Open Cloud Computing Interface (OCCI) *67*

3.2.2 Basic Execution Service (BES) *68*

3.2.3 Job Submission Description Language (JSDL) *69*

3.3 USE CASES 70

3.3.1 Use case: plant genome annotation pipeline in MAKER *70*

3.3.2 Use case: Nucleosome Dynamics *71*

4 . RESULTS **73**

4.1 DATA MANAGEMENT'S INFRASTRUCTURE FOR IMIDS' RESEARCH 75

4.1.1 Context *75*

4.1.2 Data Management *77*

4.1.3 IMID-clinica *79*
4.1.4 IMID-Longitudinal *81*

4.2 transPLANT: trans-national Infrastructure for Plant Genomic Science 87
4.2.1 Context *87*
4.2.2 Cloud architecture *89*
4.2.3 Data Management *98*
4.2.4 Use case: plant genome annotation pipeline in MAKER *100*

4.3 MuG: Multiscale Complex Genomics VRE 104
4.3.1 Context *104*
4.3.2 Cloud architecture *106*
4.3.3 Data management *130*
4.3.4 Use case: Nucleosome Dynamics *140*

4.4 Open Virtual Research Environment 149
4.4.1 Context *149*
4.4.2 openVRE *149*

5 . DISCUSSION 155

6 . CONCLUSIONS 171

7 . REFERENCES 175

8 . ANNEXES 189

8.1 Participant centers of IMID's clinical studies 191

8.2 transPLANT tools 194

8.3 MuG tools & visualizers 194

8.4 MuG data models 196
8.4.1 Data Model: "File" *196*
8.4.2 Data Model : "Tool" *199*
8.4.3 Job Auxiliary Files *204*

8.5 OpenVRE classes 205

8.6 PMES documentation 206
8.6.1 PMES server and dashboard *206*
8.6.2 PMES REST API *209*

8.7 Publications 212
8.7.1 Data management infrastructure for IMIDs' research *212*
8.7.2 MuG: Multiscale Complex Genomics VRE *213*
8.7.3 Thesis not related papers *214*

List of figures

Figure 1.1: Cost per Genome. Extracted from https://www.genome.gov....4
Figure 1.2: The accumulation of unique databases. 5
Figure 1.3: Summary of e-Science requirements and key enabling technologies. Source: [31] 11
Figure 1.4: Subsets of distributed systems 14
Figure 1.5: Virtualization representation in a cloud system. 17
Figure 1.6: Differences between hosted and bare metal virtualization. 18
Figure 1.7: Difference between full- and para- virtualization. Adapted from[257] 19
Figure 1.8: Block vs. Object storage. 21
Figure 1.9: Software defined Network (SDN) architecture. 22
Figure 1.10: Hype cycle for cloud and cloud-related technologies. 24
Figure 1.11: NIST cloud taxonomy. 25
Figure 1.12 Cloud service models: SaaS, PaaS and IaaS 26
Figure 1.13: Stack under user's control on the different cloud service models. 27
Figure 1.14: PaaS model motivation on research. 28
Figure 1.15: Classical cluster-like cloud architecture 29
Figure 1.16: Classification of open source cloud management platforms 30
Figure 1.17: Cloud stack classification 32
Figure 1.18: Research data life cycle 39
Figure 1.19: Database technologies into the CAP theorem 41
Figure 1.20: Web services architecture (a) versus REST services (b) 43
Figure 1.21: Web applications architectures. 45
Figure 3.1: Elastic virtual clouds in COMPS and PMES 64
Figure 3.2 r-OCCI server in a typical setup with OpenNebula. 67
Figure 4.1: General process data flow in IMIDs project 78
Figure 4.2: Snapshot of IMID-clinica online form. 79
Figure 4.3: Snapshot of IMID-Longitudinal online form with un error. 83
Figure 4.4. Snapshot of query issue registry in IMID-Longitudinal for a donation. 84
Figure 4.5: IMID-Longitudinal database design 86
Figure 4.6: tranPLANT cloud architecture 89
Figure 4.7: Process flow on the transplant cloud. 90
Figure 4.8: PMES SOAP operations to submit a new PMES job 94
Figure 4.9 Screenshot of PMES dashboard. 95
Figure 4.10: ScreenShot of DataManager workspace 96
Figure 4.11: "Data Manager" provides de data location URL. 96
Figure 4.12: Scheme of the transPLANT authentication system 97
Figure 4.13: transPLANT application: MAKER2 pipeline 100
Figure 4.14: MuG cloud architecture 106
Figure 4.15: Diagram flow of the MuG cloud 107
Figure 4.16: Two MuG scheduling engines: i) PMES and ii) SGE & OneFlow 109
Figure 4.17: Elastic job scheduling engine based on SGE and Oneflow. 110
Figure 4.18: MuG Tool virtual machine image. 113
Figure 4.19: MuGVRE home page 114
Figure 4.20: MuGVRE workspace 116
Figure 4.21: Selection of input data after tool selection. 117
Figure 4.22: Snapshots of the three MuGVRE visualizers 117
Figure 4.23 : Example of custom visualizations for three different analysis tools. 118
Figure 4.24: MuGVRE Tool's administration panel 119
Figure 4.25: Pop-up windows display information about a finished job. 119
Figure 4.26: MuGVRE Developer's workspace 120
Figure 4.27: New "Tools" integration protocol. 120
Figure 4.28: MuGVRE panel for administering platform's job 121
Figure 4.29: Main Mongodb collections of the MuGVRE database. 123

Figure 4.30: MuGVRE diagram flow for job processing 124
Figure 4.31: Users' access Tokens are displayed at MuGVRE 129
Figure 4.32: MuG token-based authentication across several MuG services 130
Figure 4.33: PMES data management based on a shared NAS. 131
Figure 4.34: Browsing and selection of data from ArrayExpress. 133
Figure 4.35: Tool web form in MuGVRE....138
Figure 4.36: Metadata flow among MuG elements during a tool life cycle. 139
Figure 4.37: MuGVRE wrapper for Nucleosome Dynamics build Rscript calls 143
Figure 4.38 Nucleosome Dynamics Tool submission on MuGVRE. 144
Figure 4.39 Nucleosome Dynamics Worflow in Galaxy....147
Figure 4.40: Home snapshot of a plain openVRE instance....150

List of Tables

Table 1.1: Standards for cloud portability and interoperability. 31
Table 1.2: Public cloud providers. 35
Table 1.3 Analysis data platforms. Source [259] 48
Table 3.1: OpenNebula cloud infrastructures 59
Table 3.2: Cluster details of MuG development's cloud 60
Table 3.3: List of developed web applications and its dependencies 66
Table 3.4: Operation descriptions of PMES web services 68
Table 3.5: MAKER software dependencies 70
Table 4.1: Summary of IMID-clinica case study 79
Table 4.2 Rules applied for the Quality Control data 81
Table 4.3: Summary of IMID-longitudinal case study 82
Table 4.4: MuG data models 134
Table 4.5: Implementation models for Nucleosome Dynamics 142
Table 8.1: TransPLANT tools 194
Table 8.2: MuGVRE visualizers 195
Table 8.3: MuGVRE analysis tools list 196
Table 8.4: File attributes 197
Table 8.5: MuGVRE file types 199
Table 8.6: Attributes description of Tool data model 204

List of Snippets

Snippet 3.1: Simple VM template for OpenNebula....58
Snippet 3.2 : Example of OpenNebula context information prepared to be consumed by cloud-init....62
Snippet 3.3 Virtual machine creation using rOCCI client....68
Snippet 3.4 Minimal JSDL file....69
Snippet 4.1: Directories hierarchy for transPLANT user's data....99
Snippet 4.2 Queue hosts (compute VMs) are enabled/disabled by OneFlow....111
Snippet 4.3: Process submitted by MuGVRE to the corresponding Tool VM....125
Snippet 4.4: BASH script prepared by MUGVRE to run a standalone application....126
Snippet 4.5: MUGVRE job petition for PMES createActivity endpoint. Specification : JSDL....127
Snippet 4.6: Directories hierarchy for MuGVRE user's data....131
Snippet 4.7 : Example for "File" data model in MuGVRE....135
Snippet 4.8: Example for "Tool" data model in MuGVRE....137
Snippet 4.9: Sample configuration file for bootstrapping openVRE....151
Snippet 8.1 Example of MuGVRE tool: pyDockDNA....201

Acronyms

AAI	Authentication and Authorisation Infrastructure
ACID	Atomic, Consistent, Isolated, Durable
Ajax	Asynchronous JavaScript and XML
API	Application Programming Interfaces
AWS	Amazon Web Services
BES	Basic Execution Service
BSC	Barcelona Supercomputing Center
CAP	Consistency, Availability, Partition
CDMI	The Cloud Data Management Interface
CGI	Common Gateway Interface
CMP	Cloud management platform
COTS	Commercial Off-The-Shelf
CRF	Case report form
DaaS	Data as a Service
DevOps	Development Operations
DOI	Digital object identifier
EC	European Commission
EC2	Amazon Elastic Cloud Compute
EDC	Electronic data capture
EGI	European Grid Infrastructure
EMBL	European Molecular Biology Laboratory
EOSC	European Open Science Cloud
EOSC	European Open Science Cloud
ERA	European Research Area
ESFRI	European Strategy Forum on Research Infrastructure
F.A.I.R	Find, Access, Interoperate, Reuse
FC	Fiber Channel
GA4GH	Global Alliance for Genomics and Health
GDPR	General Data Protection Regulation
GGF	Global Grid Forum
GPU	Graphics processing units
GWAS	Genome wide association study
HPC	High performance computing HPC
HTC	High throughput computing
HTTP	HyperText Transmission Protocol
IaaS	Infrastructure as a Service
IaC	Infrastructure as a Code
IAM	Identity Access Management
INB	National Institute of Bioinformatics
IRB	Institute for Research in Biomedicine
iSCSI	Internet Small Computer Systems Interface
IT	Information technologies
JSDL	Job Submission Description Language
JSON	JavaScript Object Notation
LDAP	Lightweight Directory Access Protocol
LXC	Linux Container
MD	Molecular dynamics
MPI	Message Passing Interface
MTC	Many task computing
MVC	Model-View-Controller
NAS	Network attached storage
NIST	National Institute of Standards and Technology
NREN	National Research and Education Networks
NV	Network virtualization
OA	Open Access
OCCI	Open Cloud Computing Interface
OGF	Open Grid Forum
OIDC	OpenID Connect
OVF	Open Virtualization Format
P2P	Peer-to-peer
PaaS	Platform as a Service
PMES	Programming Model Enactment Service
QoS	Quality of Service
RCP	Remote Procedure Call
RDA	Research Data Alliance
RDBMS	Relational database management system
REST	Representational State Transfer
RI	Research infrastructure
SaaS	Software as a Service
SaaS	Software as a Service
SAN	Storage area network
SDN	Software-Defined Networking
SGE	Sun Grid Engine
SME	Small and medium-sized enterprises
SNP	Single nucleotide polymorphisms
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SPA	Single-Page Application
TOSCA	The Topology and Orchestration Specification
UDDI	Universal Discovery, Description and Integration
UI	User interface
VLAN	Virtual Local Area Network
VM	Virtual machine
VMI	Virtual machine image
VMM	Virtual Machine Monitor
VPN	Virtual private networks
VRE	Virtual research environment
WebDAV	Web Distributed Authoring and Versioning
WSDL	Web Services Description Language
WWW	World Wide Web

1. Introduction

The goal of this introduction is to identify current IT standards that are directly relevant in the bioinformatics field, while understanding how they might help to enable computational infrastructures. The text does not intend to provide an extensive review, but a bird's view of current tendencies and practices in research.

1.1 Bioinformatics evolution

The fields of genetics and biology are becoming increasingly dependent on computers and specialized software. After 60 years since the first use of computational resources for *de novo* assembly of protein sequences [1], **life sciences rely on computer programs** on a daily basis, they are an essential toolkit for the analysis, information retrieval and visualization of life sciences data. Biology is being transformed into a data science.

Bioinformatics can be defined as the application of computer science, mathematics, and engineering to the study of biological data [2]. All over, it intends to:

I. organize the data in a way that allows researchers to access and contribute new data,
II. develop new tools and resources to analyze such data, and
III. use the tools to analyze and interpret the data in a biologically meaningful manner

Born more as a means than an end, bioinformatics discipline has evolved at the pace of molecular biology and computer science advances. After the paradigm shift from protein to DNA as 'information carrier', the evolution of DNA manipulation techniques enhanced the emergence of sequencing methods. It resulted in more and more available sequence data, which, coupled with the rise of increasingly small and more powerful computers, drove to the implementation of specialized software for performing computer-assisted analyses: assembling, aligning, pattern matching, tree building, (energy) function minimization, diffraction pattern resolution, etc. Then, the promotion of the free software movement (GNU manifesto was read in 1985) led to the introduction of scripting languages, which would abstract significant areas of computing systems to simplify the process of developing a program. The appearance of the **World Wide Web** communication system in the early 1990s promoted a rapid proliferation of these bioinformatics tools. The access to publications, tools, methods and public databases was simplified, like EMBL [3], GenBank [4] or PDB [5] that went online in that time.

The draft publication of the human genome in 2001 marked the beginning of the new bioinformatics generation [6]. Nowadays, the genomics era permits the technology and the resources not only for sequencing whole genomes, but for doing so at feasible costs - since 2008, Moore's Law stopped being an accurate predictor of sequencing costs (Figure 1.1).

With the arrival of **high throughput sequencing** and omics techniques, like the massively parallel sequencing (the so-called next-generation sequencing - NGS), or the single-cell analysis techniques, the challenge is back on the computer engineering field, who ought to meaningfully handle a huge amount of data and metadata. With Big Data, more computational resources for storage and processing are required, and the need for new and consistent ways to generate, access and digest this data are underlined. Scalability, scheme-less and readiness are requirements that may imply revising present computational distribution schemes.

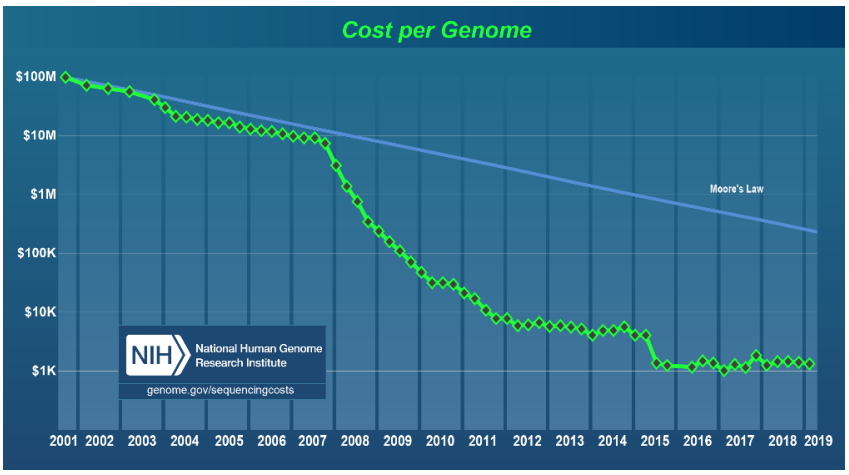


Figure 1.1: Cost per Genome. Extracted from https://www.genome.gov

Biology and bioinformatics are becoming so interlinked and mutually reinforcing that probably, one day there will be no need to distinguish one from the other. Both would be simply known as biology [7].

1.2 The creative chaos

With the exception of a few big institutes, the bioinformatics community is **segregated** in thousands of independent research groups, typically co-located with diverse related disciplines like biology, medicine, biochemistry or biotechnology. Unlike other computationally assisted academic communities (*i.e.*, astrophysics, high energy physics, climate researchers), bioinformatics has not grown around centralized core facilities (*i.e.*, synchrotrons, observatory centers), but rather just the opposite. Because of the intrinsic variability of the bio-molecular data, a wide range of bioinformatics domains has emerged, each with their own data resources, methods and standards.

Nowadays, bioinformatics exhibits huge **heterogeneity** of data types. Based on their functions, bioinformatics data sources can be classified into (i) sequence databases, *e.g.*,

GenBank [4], RefSeq [8]; (ii) functional genomics databases, *e.g.*, ArrayExpress [9], GEO (Gene Expression Omnibus) [10]; (iii) protein-protein interaction databases, *e.g.*, BIND (Biomolecular Interaction Network Database) [11], MINT (Molecular Interactions Database) [12]; (iv) pathway databases, *e.g.*, KEGG (Kyoto Encyclopedia of Genes and Genomes) [13]; (v) structure databases, *e.g.*, CATH [14], PDB (Protein Data Bank) [5]; (vi) annotation databases, *e.g.*, GO (Gene Ontology) [15], NCBI Taxonomy [16]. Biological data type diversity is evolving every day, together with experimental methodologies and their analysis. The direct consequence is a fragmentation of bioinformatics resources which is neither optimal nor sustainable. Nowadays, a large number of analysis tools, databases and Web services are available. The Molecular Biology Database Collection published by the Nucleic Acids Research journal currently lists 1737 publicly-available unique databases [17] with a linear growth trend of around one hundred databases per year [18] (Figure 1.2).

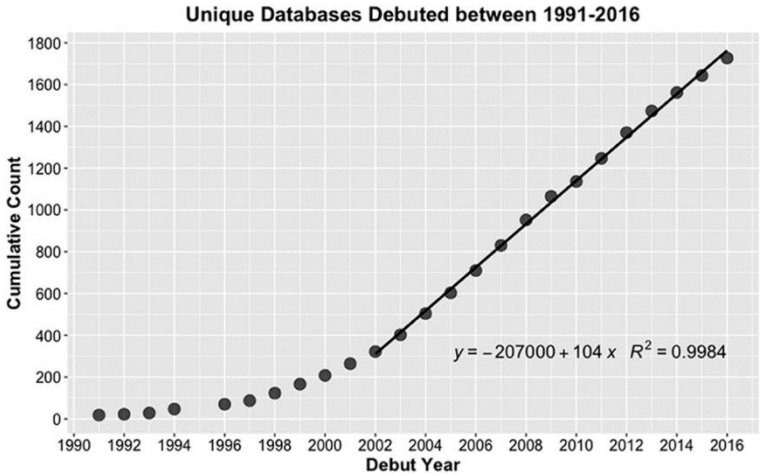


Figure 1.2: The accumulation of unique databases.
Plotted are the number of databases debuted in NAR Database Issues between 1991 and 2016

In terms of methodology, the initially diverse and loosely defined community, led software developers to mostly implement in-house methods as installable applications for their personal computers or institutions. Moreover, biologists had traditionally coded their specialized software themselves, either because a certain understanding of the biology is required prior implementation, or because is only lately that life-sciences research demands high-end computational resources. Mostly under free and open-source distribution terms, bioinformatics software tools and packages have shown rapid **proliferation** in number and complexity, encapsulating diverse novel algorithms for the processing of the ever-evolving biological data types. Currently, the ELIXIR bio.tools registry [19] has up to 14,000 entries annotated as Web applications or command-line tools. And the integration of diverse software tools, computational resources, services and databases into an analysis workflow still adds another tier to the complex bioinformatics “resourceome”.

Novel techniques have created not only a great variety of data, but also a huge **volume** of them. Methods for rapid genomic and RNA sequencing, mass spectrometry, microarray,

yeast two-hybrid assay for protein-protein interactions, X-ray crystallography and NMR for protein structures generate an enormous amount of biological data. In fact, to have an idea of the volume of data storage that these techniques have to deal with, we can just observe that, for the genomics alone, the first 20 of the largest biomedical institutions currently consume more than 100 petabytes of storage. Moreover, it is estimated that by 2025 the data-storage demands for this application could run to as much as 2–40 Exabyte [20]. It raises important challenges in terms of scalability, complexity and costs of big data storage infrastructure, but also demands new scientific methodology paradigms.

This profusion of databases and resources has fueled the scientific knowledge and the innovation in bioinformatics for many years, shaping the so-called "**creative chaos**" [21]. Yet, it also has created some significant obstacles.

The **maintenance** and hosting of the massive number of bioinformatics tools, databases and other services, demands a tremendous effort from the large number of autonomous service providers, who confront the consequent challenges in sustainability and long-term persistence. Several studies make apparent the number of discontinued published resources [18]. As areas of research become more dependent on access to data and method collections, more urgent become the need for sorting out the community's capacity to maintain them. **Redundancy** is another undesired side-product of community fragmentation that has led to functionally duplicated resources. Additionally, while the resources are mainly freely available to the whole community, they are not necessarily easy to find, use, compare, evaluate, and integrate with each other in order to find the best and most appropriate solution. **Discovering** and using new resources became one of the major issues and bottlenecks in bioinformatics. Gaps exist between the quality of documentation, ways of distribution, or degrees of **usability**. Some computational tools are available with various interfaces (*i.e.* graphical user interface, command-line, web application, plugin, programming library), while others offer only one, hence, an additional wrapping effort is required if the user is willing to use them in a different way. On top of that, there is a lack of **standards** on bioinformatics software in terms not only of programmatic interfaces, but also data formats and types. As a result, input data, output data or data access from distinct databases vary hugely in the format in which they are represented. Even when they are represented using the same format, the flexibility on some of the formats allow to use them in very different ways, greatly complicating tool's interoperability. Also, the nomenclature used inside the data may be meaningful only inside certain domains or databases, or could point to not unique data source. And last but not least, the **metadata** accompanying bioinformatics resources is frequently poor, fragmented, domain-specific, outdated, or simply absent. A poor qualitative description of the data strongly affects the practical reliability and reusability of it, as no knowledge discovery can be applied to that. Efforts in standardizing the representation of data, operations, and services is one of the most used strategies to achieve portability and interoperability, yet, specifications are numerous and

few *de facto* standards exist, especially in task interoperability, where adoption of common strategies is residual.

The efficiency of research does not rely only on data and software qualities – such as heterogeneity, accessibility or usability–, but also on the actual users, as tool end-users, data producers, or tool providers. Enabling an efficient **collaboration** between scientists and specialists geographically distributed or of diverse disciplines is the key point to address scientific challenges of the largest scale. As a consequence, institutional structures are now fostering the adoption of new research practices based on collaboration and supported by the development of advanced technological innovations oriented to the adoption of global and interconnected solutions for computing models, storage and communication. The concept is known as e-Science.

1.3 e-Science

The number of research areas becoming reliant on new ways of collaborative and multidisciplinary working promoted the emergence of new modes of doing research, all they encapsulated under the term "e-Science". At 2002, John Taylor, the former Director of the UK Research Council Office of Science and Technology (OST) firstly introduced it as follows:

> *"e-Science is about global collaboration in key areas of science and the next generation of infrastructure that will enable it". (John Taylor* [22]*)*

The new research practice brings together information technologies (IT) and scientific methodologies in order to allow scientific challenges on previously unreachable scales. e-Science has **revolutionized the scientific method**: empirical studies relies on big data studies across geographically distributed resources, while the scientific theory is now based on the use of computer simulation models [23]. e-Science, named by equivalence 'e-research' when applied to other disciplines like social or humanity sciences, is an abstract vision, at times misused or even abused, which is sustained by two fundamental concepts:

I. the enhancement of collaborative research,
II. the generation of new technologies to enable it.

Providing computational and data services to tackle large scale challenges is precisely one of the pillars of e-Science. **e-infrastructures** are those hardware and middleware enabling the supportive technologies to implement research capabilities. Hardware is wide-ranging and may include high-speed networks, large-capacity data storage facilities, supercomputers, clusters of workstations, or shared experimental facilities. Meanwhile, middleware permits a dynamic interoperability and virtualization of these IT systems, implemented as diverse software applications, data management tools, security protocols,

algorithms, data transfer protocols, ontologies, and web resources and services. Depending on the scope and focus, these infrastructures might be classified:

- *Networking infrastructures*: hardware, software, services and facilities that enable connectivity, communication, operations and management of devices.
- *Computational infrastruct*ures: software basis for building, configuring, running, and analysing on a global network of computational resources and data archives.
- *Data infrastructures*: compilation of hardware and software for acquiring, using, transporting, storing and securing data for promoting an efficient and adequate data consumption and sharing.

In general terms, all these technologies together with equipment, services, computational resources and domain tools make up **research infrastructures (RIs)**.

1.3.1 Transversal Infrastructures

The European Commission (EC) defines, evaluates and invests on strategies and tools to provide Europe with sustainable research infrastructures, the key enablers for implementing a single and open European space for online research, the so-called European Research Area (ERA). In order to support convergent and coherent strategies, several coordinated initiatives have been pushed since the emergence of e-Science [24].

The European Strategy Forum on Research Infrastructure **(ESFRI)** develops strategic roadmaps identifying investment priorities in RIs at long-term. It was firstly established in 2002 under Framework Program 7, and extended in 2016 under Horizon 2020. ESFRI fosters the development of reliable services for networking, computing, data resources, as well as standards and procedures for open access to e-Science environments. In tune with ESFRI roadmap, in 2017 the European Open Science Cloud (**EOSC**) was established for coordinating a harmonized contribution of RIs initiatives. Its goal is to assemble a European trusted virtual environment to deposit, access, analyze and re-use scientific data. The project defined two key enablers, firstly, cloud-based services and data for open and multidisciplinary science, and secondly, policy development for global data stewardship and long-term funding.

Following these umbrella initiatives, complementary and very diverse e-infrastructures has been funded, covering both, horizontal e-infrastructures providing transversal services, and thematic ones, focused on specific disciplines. Also, they include infrastructures based at a single location, distributed across several sites, or provided via virtual platforms. A summary of the transversal e-infrastructure ecosystem serving life sciences research in Europe is following [25].

HPC infrastructures

The pan-European e-infrastructures for networking, high-performance computing (HPC) and high throughput computing (HTC) are already well-established in Europe, and act as a supporting layer for other domain-specific e-infrastructures. National Research and Education Networks (NRENs) have been connecting universities and research institutes for few decades, each country using their technology and access management. **GÉANT** has federated these networks (some 60 NRENs across 43 European countries) providing a high-performance communication platform that offers private service for IP, multi-domain virtual private networks (VPNs), point-to-point connectivity, Wi-Fi access points (Eduroam), and also online identities technologies like eduGAIN and eduPKI. The ESFRI initiative **PRACE** (Partnership for Advanced Computing in Europe) is the best-known e-infrastructure in the HPC area. It offers world-class HPC facilities access through scientific project proposals, as well as educational training (PATC courses) and HPC services to SMEs (small and medium-sized enterprises). In the HTC area, where clusters are built from more commodity-type hardware, **EGI** (European Grid Infrastructure) is the most well-established e-infrastructure. It provides a large-scale HTC data analysis infrastructure, federating a large number of national independent organizations and delivering computing resources with high scalability.

Data infrastructures

Data infrastructures and services are developing fast, evolving along with the open research data requirements (see *1.4.4* Data Management). Scientific communities have been building their own e-infrastructures for data sharing - formats and metadata - often at the international level, (*e.g.* MIAPPE: Minimum Information About a Plant Phenotyping Experiment), taking into account their specific needs. The Research Data Alliance (**RDA**), a global organization started by the European Commission and other American and Australian institutions, holds up that community specifics are to be interfaced with generic components, such as those provided by **EUDAT**. EUDAT has started with the aim to store researchers' data at European level and provides services covering the full lifecycle of research data: access and deposit of informal data sharing (B2DROP service) and long-term archiving (B2SHARE); data identification (B2ACCESS); data discoverability (B2FIND and B2HANDLE); and data storage for both long-tail (B2SAVE) and big data (B2STAGE). Other European data e-infrastructures are particularly focused on research publication data, like **OpenAIRE**, the European Open Access (OA) infrastructure that enables researchers to deposit publications and data into OA repositories such as **Zenodo**, a research publication and archival service hosted by CERN that already houses thousands of datasets and software packages.

Cloud infrastructures

Unlike networking or HPC/HTC e-infrastructures, pan-European cloud initiatives are not yet so well established. EGI offers a federation of cloud infrastructures, and GÉANT a cloud

services platform to deploy on-demand IT services. **Helix Nebula** science cloud has been the first European initiative where private cloud service companies have worked with public research organizations (*e.g.* CERN, EMBL and ESA) to offer a federated, multi-tenant and multi-supplier hybrid cloud relying on open cloud standards and open-source software. The Open Clouds for Research Environments project (**OCRE**), is the recent initiative that takes off after Helix Nebula conclusion with the mission to accelerate cloud adoption in the European research community, by bringing together cloud providers and education community. Public cloud providers are further discussed in *1.4.3.5 Bioinformatics in the cloud*.

Life European e-infrastructures

ELIXIR [26] is a ESFRI pan-European distributed infrastructure that brings together life science resources from across Europe. It connects national bioinformatics centers, like the **National Institute of Bioinformatics (INB)** [27] in Spain, into a single virtual infrastructure for biological research. ELIXIR initiatives include the designation of fundamental data resources (Elixir Core Data [28]) to be long-term preserved, the development of transversal supportive platforms (*e.g.* benchmarking or training platforms), or the creation of a common Authentication and Authorisation Infrastructure (Elixir-AAI) [29] for single sign-on and authorization into ELIXIR services. Importantly, the organization acts as the life science interlocutor in Europe to establish collaboration with other international initiatives. Examples are the strategic partnership with Global Alliance for Genomics and Health (**GA4GH**) [30] that sets the technical standards to facilitate the responsible sharing of genomic data across national borders, or the collaboration with the **FAIRsharing**, the RDA working group dedicated to stablish data and connection standards for bio-data sharing.

1.3.2 New science needs

e-infrastructures are motivated by the promotion and development of the new research conducts, characterized by large-scale data-driven simulated analyses undergone on geographically scattered laboratories. Such features impose a set of **requirements over IT** systems used to build, configure, run and analyse such data. The following representation is proposed [31] to summarize the e-science requirements and its enabling technologies:

I. Resource sharing
II. Data-driven science
III. Collaborative Research
IV. Reproducible Research

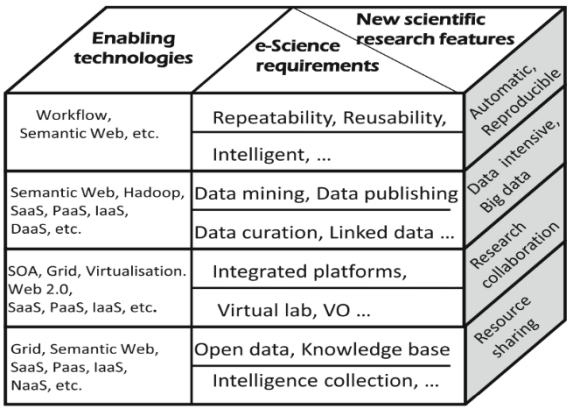


Figure 1.3: Summary of e-Science requirements and key enabling technologies. Source: [31]

Following are presented these four differential features of e-Science that computational infrastructures are to promote and preserve. At *1.4* Enabling technologies sections, the combination of IT solutions aiming to make possible each of these features is introduced.

Resource sharing

Infrastructures are to support Open Science, not a new concept, but definitively pushed forward by the emergence of data-driven research. Open access (OA) aims free **access to data, tools, methods and results** in academia, following the rational that scientific knowledge is a product of social collaboration and its ownership belongs to the community. Indeed, many public funding agencies have data sharing policies and require researchers to include compliant data management plans when applying for funding.

Most academics recognize the vital importance of data sharing and the imperious need of distributing it with high-value curated metadata [32] – data about data. Yet, they admit not always adopting OA practices [33]. There are a number of barriers to the full-scale adoption of data sharing practices, and these are not only technical, like data volume or security, but social or funding-related, like the **lack of recognition** of data providers or the **long-term sustainability** of data infrastructures [34]. Research infrastructures aim to adequately handle these limitations, for instance, proposing fine-grained authorization protocols [35], adopting open protocols and standards, distributing datasets under digital object identifiers (DOIs) for easing being citable, etc. Also, many data repositories have emerged in the last years, including pan-European archives like those just above presented. Some initiatives [36] [37] keep a registry of the long-term data infrastructures that make available scientific records or published data.

Data-driven science

Infrastructures are often generators of large volumes of data which have motivated the appearance of both, technical and policy solutions in order to properly curate, document, process and preserve that data.

At the policy level, academia is widely accepting and progressively supporting the definition of a minimal set of community-agreed guiding principles and practices for data handling. These correspond to the so-called **F.A.I.R. principles** [38], by which data providers and data consumers - both machine and human - could more easily (F)ind, (A)ccess, (I)nteroperate and (R)euse the vast quantities of information. Eventually, it would become accessible to the new and powerful plethora of semantic and machine learning algorithms. Compiled as concise and measurable metrics [39], turning these principles into a reality is proving to be challenging as it involve important **efforts on data stewardship** and data structures – while Findable and Accessible principles relay mainly on appropriate metadata, Interoperability and Reusability depends on data governance, services and stewardship. In spite of that, the principles are being extended to research methods – i.e., **F.A.I.R. software.** Their key requirements are already available: registries, standards, software managers and open repositories. Several organizations including the Software Sustainability Institute [40] or the ELIXIR itself are participating actively in the discussion.

Regarding technical specifications, infrastructures have to cope with the needs arising from handling Big Data and their five main characteristics: **Volume, Velocity, Variety, Value and Veracity** [41] (known as the "5 Vs"). Correspondingly, they refer to the huge amount of data volume, the fast analysis speed shifting from batch to real-time stream processing, the heterogeneous data sources with complex formats and unstructured models, the low density of values into the data, and the tolerance of errors within this data. Such features oblige the adoption of adequate computing models, processing algorithms and database engines into any infrastructure dealing with data-intensive tasks.

Collaborative Research

Knowledge diffusion and resource sharing are the cornerstones of multi-center and multidisciplinary projects, which nowadays are making possible to tackle large-scale scientific analyses otherwise unattainable. **Integrative solutions** and supporting applications are essential to enhance collaboration and personalization, and are especially relevant in multi-branched dynamic areas of research like bioinformatics and the omics era we are immersed in. Moreover, though collaboration, scientific communities are endorsed and better positioned to agree on standards, align scientific goals and practices, etc.

Additionally, infrastructural resources such as storage, computing facilities, and other services are increasingly used and shared by researchers at **geographically distributed locations**. In involves innovative applications for the exchange, selection and aggregation of distributed resources. Thus, collaborative research has raised security, trust and privacy concerns, along with other technical requirements stressed in data-intensive environments, like data transfers.

Reproducible Research

The result of the scientific method must be reproducible in that when carrying out the documented procedure, the same result is to be obtained every time. Today, this procedure nearly always involves software and digital data, typically known to have short lifespans. Because of this alone, traditional results' documentation, even when good practices are adopted – *i.e.* code and data availability, control version, configuration parameters specification, etc. –, might not be sufficient to achieve reproducibility, neither for validation, nor reusability. The truth is that inadequate reproducibility efforts are not the only cause of the so called "**reproducibility crisis**" [42], which indicates for instance, the need of more robust experimental designs and better statistics.

Nevertheless, the global network of infrastructures can play an essential role to foster reproducibility by building provenance mechanisms, stateless encapsulations of data and workflows, or simply using standard and agreed protocols and technologies. Considering computational infrastructures are the software basis for building, configuring, running, and analyzing data, these are responsible to encompass these practices, and do so in an automatic and deterministic manner in order to achieve reusable and repeatable research processes.

1.4 Enabling technologies

Research infrastructures capabilities are based on various technologies aimed to support the new generation of scientific research. The following sections provide introductions of relevant methodologies, architectures, tools, systems, services or frameworks that are designed to address the **new science needs**.

1.4.1 Distributed computing

The growing popularity of the Internet and the availability of powerful computers and high-speed networks as low-cost commodity components have changed the way computing is done. Distributed computing has been an essential component of scientific computing for decades. It consists on networked independent computer **nodes that cooperate** to achieve a common goal at task level. Several distributed models have appeared along the years diverging on the geographical node's location (sparse or centralized), their interconnection (loose or high-speed), their management (centralized or distributed) or their disparity (homogeneous or heterogeneous nodes). A classification of them is represented in the following figure:

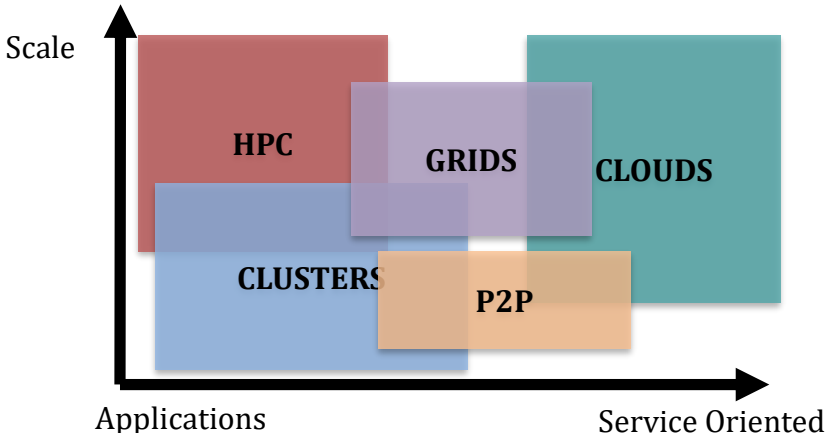


Figure 1.4: Subsets of distributed systems
Classified according its centrality and scalability. HPC: High performance Computing. PHP: Peer-to-peer. Adapted from [256]

1.4.1.1 Decentralized systems

Peer-to-peer (P2P) systems are those where nodes (peers) are simply client computers connected to the internet loosely interlinked and with a distributed self-organized administrative control that handle the requests to the resources, *i.e.* peers are both resource's providers and consumers. Rosetta@home [43] and Folding@home [44], for protein structure prediction, are examples of P2P volunteer distributed computation applied to bioinformatics. Yet, a server-less and self-controlled network involves some security challenges that may limit its application in some areas, while enhancing it in others, like file-sharing or anonymous network browsing. Lately, **blockchain** technology has promised to ensure security and privacy, enabling projects for rewarding computation (or storage) sharing (*e.g.* CureCoin [45] as Folding@home sequel), but also enabling researcher identification systems, or facilitating decentralized genomic data distribution (Nebula Genomics [46]) [47]. On top of that, rapid advancement in mobile devices and wireless technologies is enabling the use of Personal Digital Assistances (PDAs) and mobile phones in several ways (edge computing), including the idea of using such devices to expand P2P-based systems (ubiquitous computing).

1.4.1.2 High Performance Computing

Cluster computing is another distributed system that consists of a set of independent but local and homogenous computers (nodes) interconnected through a fast local area network. Subsystems are supervised by a single administrative domain, integrating the system as a single computing resource. Thanks to the availability of low-cost

microprocessors, high-speed networks, and software for high-performance distributed computing, these systems have emerged as a cost-efficient parallel computing solution.

Traditional **High Performance Computing** (HPC) systems are focused on parallel architectures aligned with distributed systems to achieve the petaflop scale. Software optimizes the speed-up of multi-core CPUs or graphics processing units (GPUs) using (massive) parallel programming techniques - *i.e.* OpenMP for shared memory machines, Message-Passing Interface (MPI) for more loosely connected, or CUDA/OpenCL for GPUs. Highly demanding bioinformatics applications have largely benefited from supercomputers or GPU clusters, for instance, molecular dynamics (MD), virtual screening software or NGS, which even adopted novel algorithms approaches like those inspired by Genetic algorithms [48] to better exploit parallel resources. More and more, the tendency of using inexpensive general-purpose COTS-based (Commercial Off-The-Shelf) components is being imposed over on-purpose designed processors. In 2011, the fastest supercomputer, Tianhe-1A, was based on NVIDIA TeslaTM M2050 general-purpose GPUs. Marenostrum 4, placed at the Barcelona Supercomputing Center, is composed by Intel Xeon Platinum. Main bottlenecks for HPC adoption are its high power consumption, especially for GPUs, the lack of flexibility and accessibility, the need of advanced or even on-purpose designed software, and still, hardware costs, which indeed are COTS components, albeit with high-end.

Nowadays, the focus has shifted to programming models like MapReduce, more centered on data analysis and manipulation, and not that much on the algorithm optimization. Unlike more traditional HPC systems, the data is not stored at a single point but divided and replicated across the distributed computer system, and the application only deals with its local subset of relevant data (further discussed in *1.4.4.2 Data-storage solutions*). These new requirements make HPC data systems easy to deliver, even over the cloud. **HPC clouds** tries to address some of the bottlenecks mentioned above offering HPC-as-a-Service. It may permit benefiting from multi-tenancy, resources on-demand, and scalability, which translates into a better resource utilization. The economic and technological barrier to high performance clusters is being lowered, as typically are in-use facilities. However, other challenges have arisen, like network latency, configuration and optimization issues, or virtualization overhead, actively researched nowadays. Just in 2016, first commercial HPC clouds were launched, offering infrastructures only partially virtualized (*e.g.* virtualized login nodes but bare-metal computing nodes) or with high-speed networking (10 Gbit/s Ethernet or InfiniBand)[47][49].

1.4.1.3 Grid computing

Cloud computing can be perceived as an evolution of grid computing, with the inclusion of virtualization and sharing of resources [50]. The architecture is similar to the clusters, except that the computational nodes are spread over the world and connected through a combination of local area networks to the Internet [51]. Grid applications implement

parallel and distributed computing models where the program can be split into many independent pieces and computed independently among the nodes, as communication latency among nodes is orders of magnitude higher than in cluster's backend. Since the late 90s, distributed and grid computing have been extensively employed in science, and particularly for those applications where inter-process latency was not critical, *i.e.* **high throughput computing** (HTC) (*e.g.* embarrassingly parallel applications) or many task computing (MTC).

Grid initiatives, like Open Source Grid (OSG) in the USA or the European Grid Infrastructure (EGI) have played a crucial role in developing and adopting standards and protocols for grid computing, being able to coordinate thousands of processors located in several countries into aggregated grids for large-scale scientific computation. The **middleware** required for such distributed executions typically uses a (i) distributed storage system for sharing data between application tasks running on different nodes, and (ii) resource managers for scheduling tasks to nodes. Both technologies are fundamentals for today's cloud technologies. In deep, these tools and libraries have been updated and adapted to serve cloud elasticity and virtualization. Under OSG, it was first developed the Globus grid middleware [52], where libraries for remote process management, service discovery, monitoring or security are packed. Other middleware appeared afterward [53]. Additionally, distributed systems fostered the development of other hardware and software stack, like **Resource and Job Management Systems (RJMSs),** which aim to assign jobs upon the available resources in an efficient way. The assignment involves (i) job management, (ii) scheduling and allocation, and (iii) remote resource processing. Examples for open-source RJMS are TORQUE [54], SLURM [55], CONDOR [56] or Sun Grid Engine (SGE) [57], systems widely used in all distributed environments like clusters or HPC.

OSG also established **standards and specifications** for enabling interoperability among grid components, most part of related software is compliance with:

- Job Submission Description Language (JSDL) [58]: provides an XML schema to describe a single job submission, including pre- and post-staging of data files.
- Basic Execution Service (BES)[59]: defines an interface to monitor and manage remote computational executions.
- HPC Basic Profile: coordinates the use of the JSDL and BES, along with an interoperable security mechanism in order to compose a batch job.

With grid technology, computational intensive challenges are accomplished, but the system has rigid constraints, since there is only **limited access** to the servers, with little or no interactivity, due to remote batch scheduling of the jobs on the remote clusters. To mitigate these problems, the solution might be the creation of flexible virtual environments on top of them. Virtualizing resources is critical for making grid computing able to address the complex workflow analysis systems that are necessary for Bioinformatics.

1.4.2 Virtualization

Virtualization is the act of creating logical instances of computing hardware, storage resources or network devices on a physical hardware resource (Figure 1.5). The virtualized system is called the guest, while the physical system, the host.

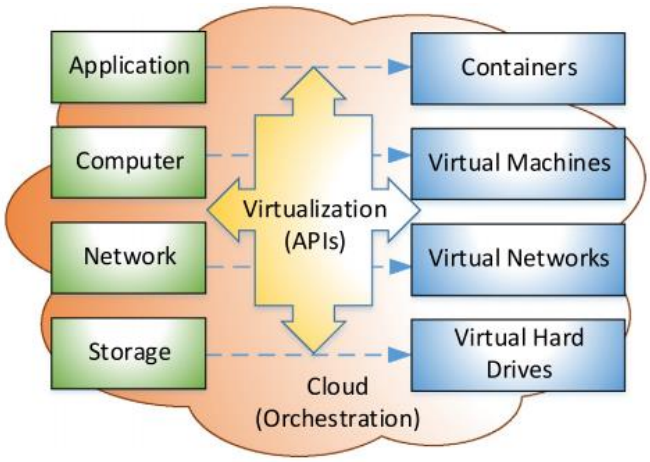


Figure 1.5: Virtualization representation in a cloud system.
Three main cloud computing resources, i.e., computing, networking, and storage, and applications can be virtualized through virtual machines, virtual networks, virtual hard drives, and containers, respectively. Source [257]

Following sections present the virtualization on each of these four resources.

1.4.2.1 Compute Virtualization

Virtualization is generally realized by a **hypervisor** or Virtual Machine Monitor (VMM), a piece of specialized software that emulates a hardware by enabling host resource sharing with the multiple guest OSs - or virtual machines (VMs). Hence, a VM corresponds to an isolated environment with access to a subset of the physical resources. Virtualization requires not only powerful processors and fast interconnection with the host hardware, but also an architectural control of the shared resources, which affects multiple levels: CPU cycles, memory management, storage, I/O operations, communication bandwidth, etc. Besides, the hypervisor controls the assignment of resources among VMs, not an easy task for some of them – while scheduling for sharing CPU cycles is implemented for most hypervisors (virtual CPUs), allocation of cache, memory or I/O bandwidth is more intricate.

Hypervisor types

IBM released its VM 370 in the 1970s, yet was VMware who brought virtualization to commodity x86 platforms, followed by Xen and a variety of other **virtualization platforms**, each using different underlying techniques. The techniques can be classified based on different criteria. Focusing on where the virtualization takes place, a second classification is proposed:

- Instruction Set Architecture (ISA) level: the ISA of the host processor is fully emulated on the guest processor. A virtual ISA requires a processor-specific software that translates the instructions issued by the guest OS, which leads to an interpretation latency *e.g.* [60].
- Hardware Abstraction Layer (HAL) level: exploits the resemblance between guest and host platforms to cut down the interpretation latency. It is performed on top of the bare hardware via a kernel extension or from a device driver *e.g.* [61], [62],[63] ,[64].
- OS level: the host OS shares not only the hardware but also the OS to create an OS interface instead of a stand-alone guest OS *e.g.*[65],[66] ,[67]; section *1.4.2.2*).
- Library level: system calls are performed through libraries programmed using a set of APIs at user level *e.g.* [68] for Windows on top of UNIX or [69] for GPU support.
- User-application level: virtualizes an application as a VM, which executes as a process in traditional OS through a high-level language (*e.g.* Java Virtual Machine).

Finally, focusing on the type of hypervisor employed, there are two more virtualization options presented in Figure 1.6:

- "**bare metal**" (or "Type 1") hypervisors are a thin software sitting directly on top of the physical hardware without needing any operating system. Their main advantage is performance. Examples: RedHat [70], many **Xen** variants, VMWare [71] and [72].
- "**hosted**" (or "Type 2") hypervisors run as an application on the host OS that is installed on the physical hardware. Hence, the guest OS calls need to traverse through the host OS before reaching the bare metal. Advantages are that the VM is easier to build and install, and additionally, they are lighter, as they can use several host OS components rather than providing their own. A price to pay is the increased overhead because the host OS act as an intermediate, and sharing OS rises challenges to support complete isolation of VMs. VirtualBox and **QEMU** are examples of hosted hypervisors.

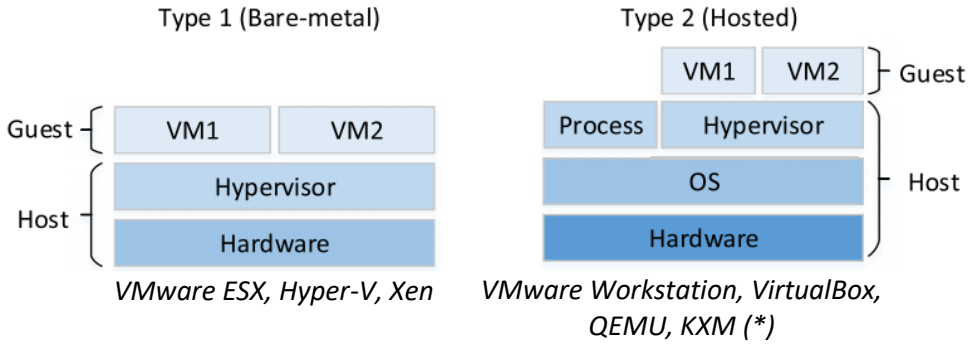


Figure 1.6: Differences between hosted and bare metal virtualization.
(*) KVM is not a clear case as it could be categorized as either one. The KVM kernel module turns Linux kernel into a type 1 bare-metal hypervisor, while the overall system could be categorized to type 2 because the host OS is still fully functional and the VM's are standard Linux processes from its perspective

Such hypervisors can map resources in two different modes:

- Full virtualization: the guest OS stays unmodified, so hypervisor needs to fully virtualize each guest's resource petition, a task that entails performance consequences. Yet, as the hypervisor is an exact replica of hardware, all system functionalities are covered. (*e.g.* **KVM**).
- Paravirtualization: the guest OS is modified to run on the hypervisor, who performs the tasks directly on the guest kernel. Is limits support to open source OSs like Linux or Solaris, yet, the ability of the kernel to directly communicate with the hypervisor results in great performance levels (*e.g.* Xen).

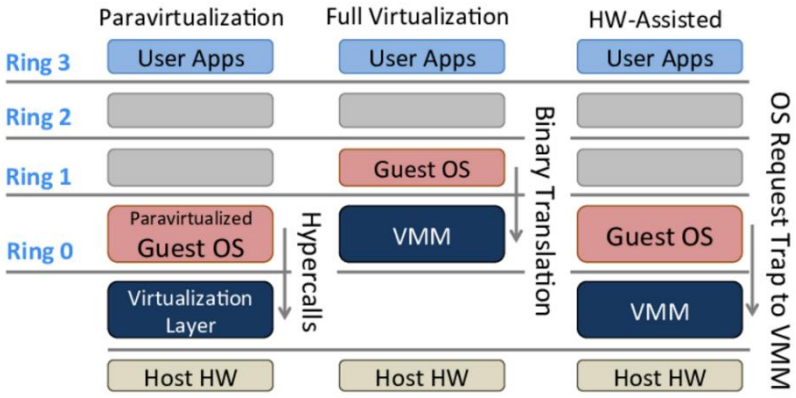


Figure 1.7: Difference between full- and para- virtualization. Adapted from[257]

Benefits

The important benefits of virtualization make the technology applicable over a wide range of areas. Primarily, the virtualization is used to offer configurable and **portable computing environments**, presenting the illusion of an OS fully controlled by the user. In there, the environment is controlled, constrained and **replicable**, what promotes automatic application executions and facilitates software development, testing and debugging. The environment portability is especially relevant for software migration and sustainability, indeed, cloud services allow (live) migration from host to host. Virtual environments are isolated from the host system. It presents important security advantages, as the user's code can run **sandboxed** without risking harm to the OS. Multiple execution environments are also supported, with no - or minimal - affectation on the performance of other virtualized runs or OS applications. In fact, this last feature, called crosstalk, is an important cloud QoS (Quality of Service) as nowadays is still not fully resolved. VMs allow for resource optimization and server consolidation by effectively scaling the resources to the user's real needs. Moreover, such flexibility also enables VMs to work as plug-and-play appliances that can be easily enabled and disabled, and easily provides **disaster-recovery** functionalities to the system.

Limitations

Limitations of virtualization are mainly those derived from the layer of **overhead** that represents being constantly allocating resources to the virtual environment. Hence, the performance degradation is more present in hypervisor-based virtualizations than in OS or application virtualizations, where the virtual environment does not include their own OS. System functions are not all virtualized equally efficiently, critical for the performance of a VM are the cache and memory management, and I/O handling. For instance, CPU scheduler generally down perform at a 10-20% rate, yet the overhead is much more noticeable as the guest workload increases (particularly, in KVM). Memory overhead also fluctuate around these values for most of the platforms, yet, the performance variation can be as high as 140% for network or I/O activities, showing that all techniques outperform others in particular operations and conditions. Over-commitment (allocating more resources in the guest than the physical host has available) for CPUs is a feature supported for most of the hypervisors at a reduced performance cost. However, for memory (RAM) costs are higher (*e.g.* VMware supports it), and other memory management techniques have been developed, like memory compression or memory "ballooning", by which the hypervisor dynamically rebalances RAM reallocating idle memory from some VMs to others (*e.g.* VMware, Hyper-V, KVM).

Cloud platform management

Cloud platforms make use of different virtualization techniques. For instance, Amazon EC2, the largest infrastructure cloud, uses Xen as a hypervisor, but Microsoft Azure uses Hyper-V and VMware partners use ESX. Recently, Google launched its own IaaS cloud that uses KVM as a hypervisor. On the other hand, open-source platforms tend to offer support for more than one hypervisor (see section *1.4.3.2* Cloud Management Platforms).

1.4.2.2 Operation System Virtualization

Hypervisor-based virtualization consumes part of the resources to maintain a separated Operating System (OS) for the guest. If guests use the host's OS, important resource consumptions are achieved. Recently, OS-based virtualization – i.e. **containers** – captured the attention of industry and academy because they accelerate the development process, eases distribution, and applications' deployment. Containers subtract the hypervisor layer and rely on namespaces and cgroups in order to provide isolation and resource management. Called also lightweight VMs, containers are virtual space instances that field like a complete and isolated OS to the user, yet, they run as applications with extended chroot permissions into the host.

Leaders of such development are **Docker** [73] and **Linux Containers** (LXC) [74]. Indeed, Docker is basically an additional layer on top of LXC exposing additional features such as mounted storage, network port redirection, and container catalogue management.

Singularity [75] is a container-based approach more focus on application portability than on host virtualization. Hence, it only virtualizes what is necessary for achieving a runtime application container: user escalation is not supported. Unlike Docker, Singularity inherits permissions of the user who runs that container, being able to work within multi-user or multi-tenant environments (typically HPC).

Compared to VMs, OS-level virtualization usually imposes little to **no overhead**, offers the highest performance, specially Singularity, and reach the highest density of virtual environments. Memory allocation is more flexible and unused memory in one virtual environment can be used by another. Obviously, it implies that the container's isolation degree is lower than in VMs. Another disadvantage is containers' reduced flexibility, as they only support the virtualization of the host OS server, Windows or Linux. Moreover, we have mentioned already the security implication that have some of the OS-level implementations.

1.4.2.3 Storage Virtualization

Storage can be virtualized too, and that it is referred to as **Software Defined Storage (SDS)**. Storage virtualization means decoupling storage volumes, from the underlying physical hardware. Thus, features like caching, snapshotting, high availability, etc. are managed in the software layer and are not dependent on a specific hardware brand. Storage systems can provide either block accessed (*i.e.* storage area networks (SANs)) or file accessed (*i.e.* network attached storages (NASs)).

In **block virtualization,** multiple physical disks are abstracted into a single logical storage resource creating a SAN, typically accessed over Fibre Channel (FC), iSCSI (Internet Small Computer Systems Interface) or other protocols. Files are split in chunks into volumes known as blocks, which are individually mounted with the underlying file system protocol (such as NFS, CIFS, ext3/ext4 and other). In cloud computing, most providers offer virtual block storage (*e.g.* Amazon Elastic Block Store, Cinder blocks from OpenStack; Datablocks from OpenNebula).

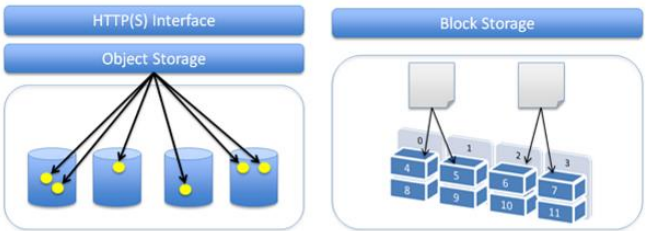


Figure 1.8: Block vs. Object storage.
Block virtualization provides good I/O performance compared to Object storage virtualized solutions

On the other hand, in **file virtualization,** the abstraction layer is between the file server and users accessing these files, typically accessed over NAS protocols like NFS or CIFS. The result is that multiple storage devices are grouped together to form a single, logical file mount.

1.4.2.4 Network Virtualization

Network virtualization (NV) is defined by the ability to create logical, virtual networks that are decoupled from the underlying network hardware. From the perspective of any application residing on the host, the principal functions of physical networks involve switches and routers (at Layers 2 and 3), or load balancers and firewalls (at Layers 4 - Layers 7). With virtual networks, these functions are accomplished with pieces of software that frees physical devices from their one-to-one association with hardened addresses and endpoints. This software-based overlay conforms the so-called **Software-Defined Networking** (SDN) technologies.

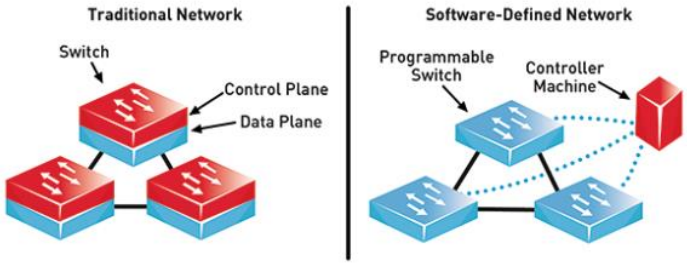


Figure 1.9: Software defined Network (SDN) architecture.
SDN attempts to centralize network intelligence in one network component by disassociating the forwarding process of network packets (data plane) from the routing process (control plane).

Lately, the development of these techniques has come to focus thanks to the popularity of distributed computing and storage paradigms. SDNs aims to centralize the management of networks to enable a flexible and programmatic network configuration. This strategy enables networks to stage multiple workloads, especially micro-services, and scale them dynamically on demand. Some other popular SDNs focused on attaching networks to VMs or containers are [76], [77] or the Kubernetes propose-solution [78] - all of them compliance with the [79] specification. Examples of more general-propose network management solutions are [80] or [81].

Fast flat and composable networks provide agility and automation, yet, they come with their **challenges**, mainly security considerations. Trust zones frontiers are blurred in virtual networks, and current network topologies make difficult to establish the appropriate level of trust that is required for each individual system. The dynamic real-time changing capability of networks is also raising severe visibility issues, making evident the need for new monitoring and tracking tools. Also, as any composable environment, standards and compliances are important aspects to consider for SDNs.

1.4.3 Cloud technology

Technologically speaking, the cloud is more an evolution that a revolution [82], as it has emerged as a **convergence** of several computing trends. It seeks to achieve scalability/elasticity, availability and optimal resource utilization, key aspects only partially addressed for the individual supporting technologies. Because in fact, cloud computing borrows the best of grid computing, virtualization and Service Oriented Architecture (SOA).

Virtualization is not a requirement to create a cloud environment (e.g. Ironic [83] provisions bare-metal machines to OpenStack clouds), but it enables a rapid scaling of resources in a way that non-virtualized environments find hard to achieve.

Computation and storage are delivered as services rather than as resources. Thus, web services technologies and SOA are architectural components of cloud (see section *1.5.1* Service-Oriented Science*)*.

Distributed computing legacy is also a fundamental part of the new paradigm, where tools for remote provisioning, security, compliance, privacy and "pay per use" models are inherited from and extended.

Other IT systems are also part of the new computing model, like metering systems, advanced data management system, task orchestrators, etc.

1.4.3.1 Cloud Fundamentals

A well-accepted **definition** of cloud computing comes from the National Institute of Standards and Technology (NIST) that presents the computing as a utility:

> *"Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" (NIST 800-145* [50]*)*

According to this definition, the five key characteristics of cloud services are:

- *on-demand self-service*: access to computing power without human intervention.
- *broad network access*: access to computing power on the Internet.
- *resource pooling*: virtual resources dynamically assigned without having control of the exact location of them.

- *rapid elasticity*: resources are able to be quickly allocated by increasing/decreasing both, the number of virtual appliances deployed (*i.e.* horizontal) and resources dedicated within each (*i.e.* vertical).
- *measured services*: control and optimization of resource use by evaluating storage, processing power, bandwidth, active user accounts, etc. These measures are the metrics used for achieving a Service Level Agreement (SLA), basic for customer's confidence.

Cloud computing has developed greatly since 2006 when the Amazon Elastic Cloud Compute (EC2) was first launched, and currently, it is reaching its productivity plateau after positioning itself beyond a disrupting IT technology, to become the basis for most future IT models [84] (Figure 1.10).

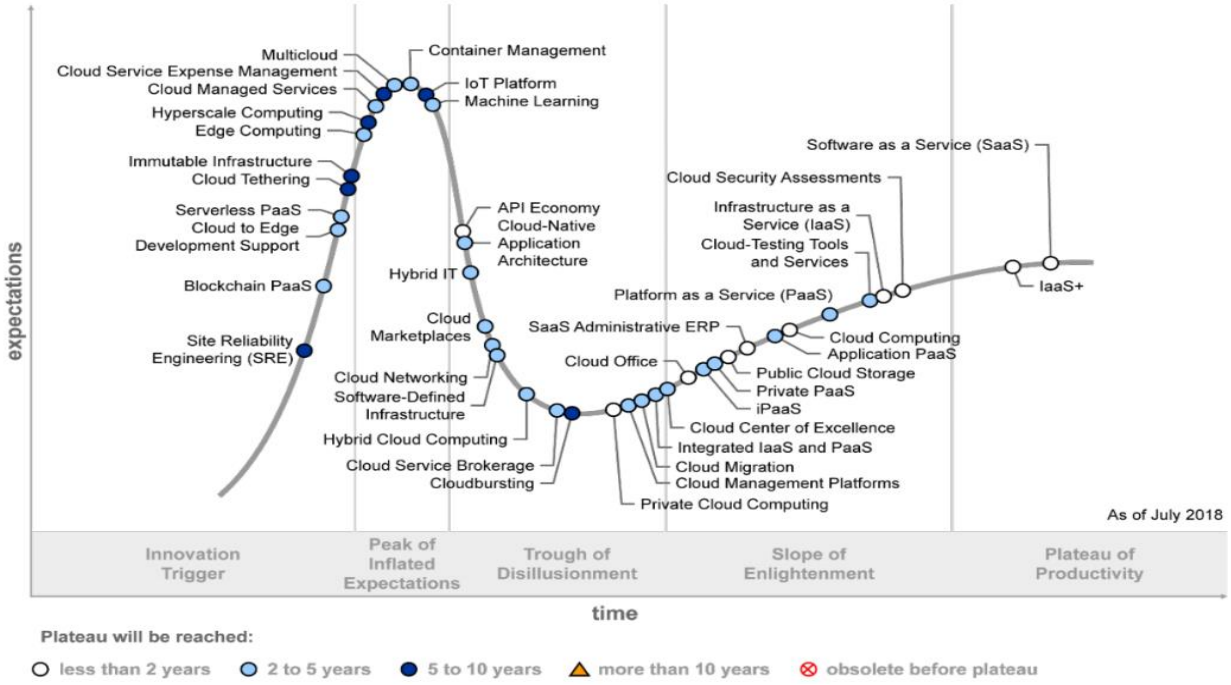


Figure 1.10: Hype cycle for cloud and cloud-related technologies.
The hype cycle [258] *graphically represents the maturity and adoption of specific technologies. "Cloud Computing" is already in the enlightenment slope and is predicted to be in the productivity plateau in less than 2 years.* [84].

Cloud computing is a large-scale distributed computing model, providing computing, storage, networking, and other resources to many users in service mode. The cloud offers high scalability (*i.e.* horizontal, vertical elasticity), high availability and reliability (*e.g.* through replication), while improving resource utilization efficiency (*e.g.*, over virtualization techniques like over-commitment or pre-emptible virtual machines). These **key capabilities** is motivating the migration of a large number of companies to cloud computing, but also scientists, who point out the new paradigm as a solution for big data storage and large-scale analysis in bioinformatics, thanks to the virtualization that may reduce data transference and improve computation accessibility to medium or small laboratories, meanwhile promoting data sharing and collaborative work.

Besides the five essential characteristics, NIST describes three cloud service models, and four cloud deployments that are introduced in the following to subsections.

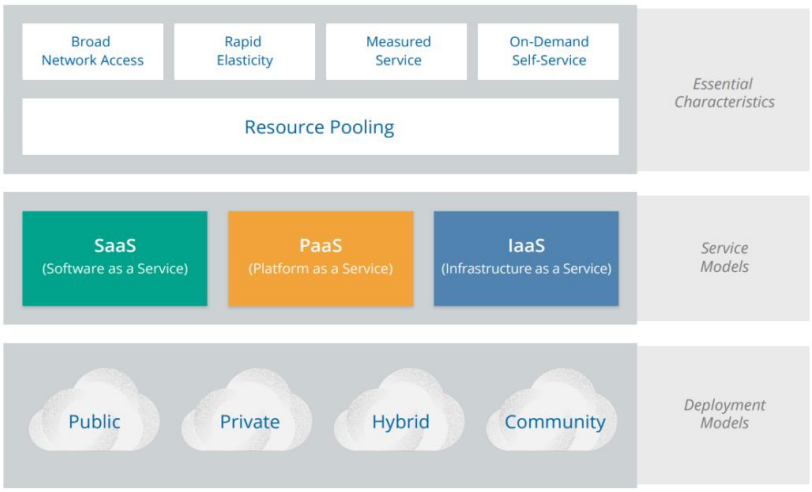


Figure 1.11: NIST cloud taxonomy.

Deployment Models

Clouds can be classified based on their deployment model, that is the architecture and purpose of the underlying infrastructure in terms of cloud location and management. NIST identifies four different deployment models for cloud computing: private clouds, community clouds, public clouds, and hybrid clouds.

Private Clouds

They are intended for the exclusive use of a **single organization** with multiple users and emulate cloud computing on private networks, either owned, managed and operated by the same organization (on-premises), or by an external cloud provider (off-premises). Hence, regarding its physical location, private clouds are classified into "on-site" or "outsourced" clouds. "On-site" private clouds are secured and controlled inside the organization and clients access it from within the security perimeter or through firewalls and a Virtual Private Network (VPN). In "Outsourced" private clouds, the cloud provider permits the client access into its premises and separate the private cloud resources from the other cloud resources by different mechanisms such as Virtual Local Area Network (VLAN), VPN, separate network segments or clusters. As discussed below at *1.4.3.2*Cloud Management Platforms, several open-source packages are available to install private clouds.

Community Clouds

They are intended for the exclusive use of a **specific community** of users from several organizations that have shared objectives (*e.g.*, security requirements, compliance considerations), who can reduce costs by sharing the infrastructure. Like in private clouds,

the infrastructure can be owned and managed by a third party, or be operated within the community on-premises on-site scenario, where each organization may provide cloud infrastructure, consume services, or both.

Public Clouds

They correspond to the traditional and mainstream definition of cloud computing, where resources are dynamically provisioned on a self-service basis over the Internet. The infrastructure, owned, managed and operated by a business, academic or governmental organization, is intended for **open use** by the general public. Cloud provider is responsible for the security management. Compared to private clouds, public cloud services offer a lower degree of control and oversight of the physical and logical security perimeters to separate computational resources (usually present in the outsourced private cloud model). Amazon Web Services (AWS) is the leading company, representing almost 50% of the market share in 2018. Together with the rest of the main public cloud providers, Microsoft Azure, Google Cloud, Alibaba Cloud, and IBM, they dominate 80% of the market share [85].

Hybrid Clouds

They are a **composite** of any of the other deployment models. The individual clouds remain unique entities and are aggregated by standardized or proprietary technologies that enable data and application portability (see also section *1.4.3.3*). Some use cases for this complex model are the access to external clouds during periods of high demand (called cloud bursting), or to run some applications into public clouds while maintaining sensitive data in an on-premises private cloud.

Service models

Cloud Computing enables hardware and software to be delivered as services following the SOA principles (see also section *1.5.1* Service-Oriented Science). These services are usually described on a Something as-a-service (XaaS) taxonomy. In this context, the term service not only refers to SOA precepts, but also reflects that the service is provided on-demand as a utility - very convenient for cloud vendors. According to NIST and as depicted in Figure 1.13, these services can be classified into one of three delivery models: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

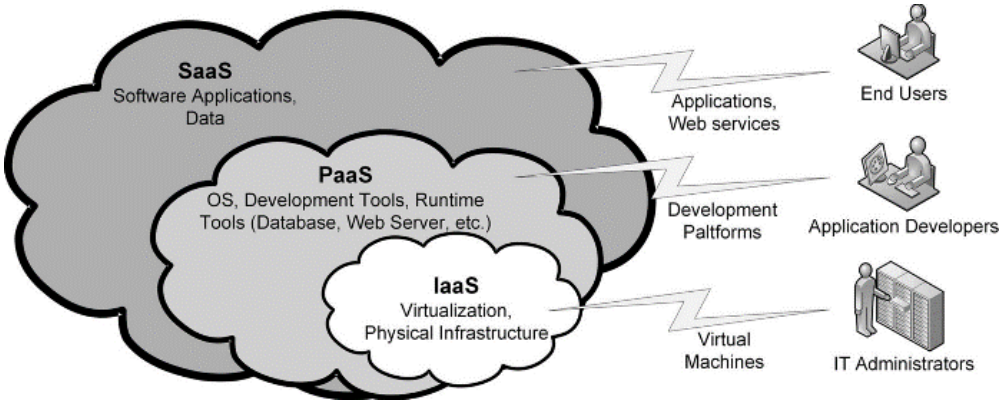


Figure 1.12 Cloud service models: SaaS, PaaS and IaaS

Each model allows the user a different degree of control over the cloud stack:

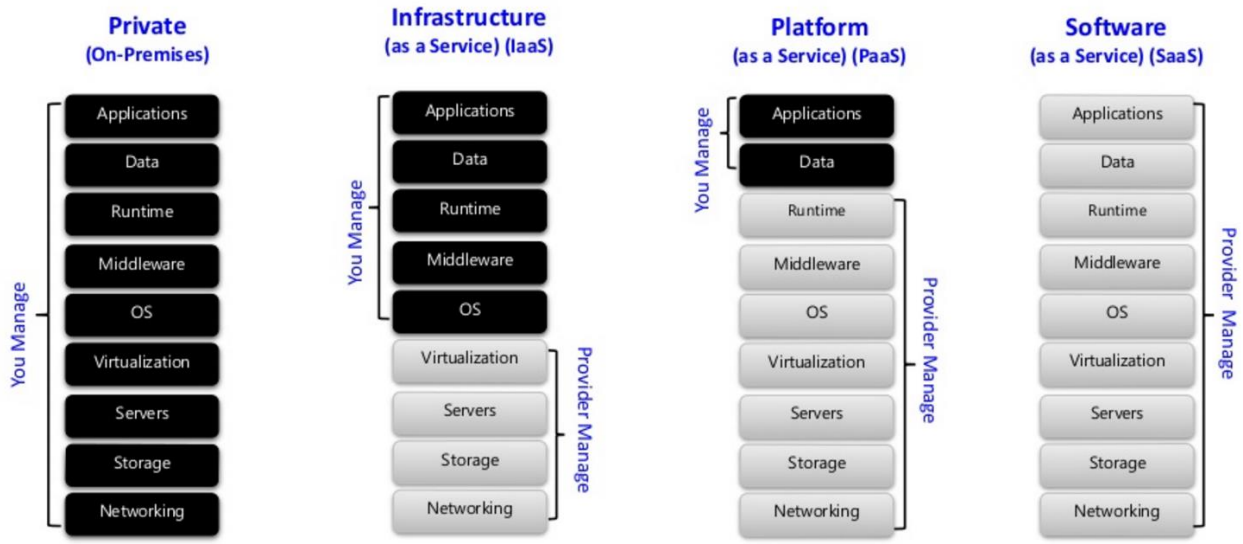


Figure 1.13: Stack under user's control on the different cloud service models.

Software as a Service (SaaS)

This service mode delivers **application** services online and facilitates remote access to available bioinformatics software tools through the internet. The user typically accesses to a web application and is totally unaware of the infrastructure lying behind. A simple and widely used SaaS is Google Docs. SaaS is actually the most usual approach to publish bioinformatics tools on the cloud. CloudBlast [86] would be an example, but there are many others, some introduced at *1.4.3.5* Bioinformatics in the cloud.

Platform as a Service (PaaS)

Allows the development, installation and execution of **user-developed applications** on the cloud infrastructure. Applications must be created using specific programming languages, API libraries or tools supported by the PaaS provider which constitute the development environment. The clients have full access and control over the created tools and the developmental languages, yet not over service run times, web server, or storage network. Offered services rank from user's application hosting, development and testing, to extensive integrated services with scalability and maintenance. Examples for PaaS generic vendors are Heroku [87] or openShift [88], where the user is able to execute in the cloud their applications (called "Dynos" or "Gears" respectively) as containers in real time and shying away the deployment details of the infrastructure. The main benefits of these services include that users can focus on high-value software rather than infrastructure. Additionally, to these stand-alone PaaS, if user's applications are meant to be exposed to online through a certain SaaS, the developer's platform is called "PaaS from SaaS".

As illustrated in Figure 1.14, in the **bioinformatics** domain, some platforms have been designed following the same philosophy, as a new strategy to outsource infrastructure management while keeping control of the scientific code running there. Eventually, such code can even be offered as SaaS to researchers. Section *1.4.3.5* Bioinformatics in the cloud shows some examples following this approach.

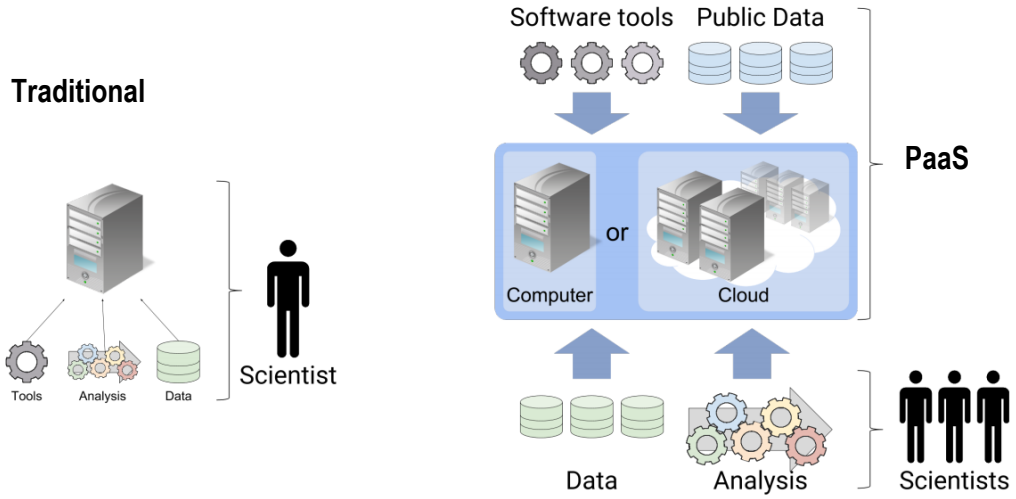


Figure 1.14: PaaS model motivation on research.
Scientists are not anymore responsible for computing hardware, software installation, yet they only produce and consume applications and pipelines.

Infrastructure as a Service (IaaS)

It refers to the group of high-level APIs that control low-level details of the underlying infrastructure, like physical computing resources, data partitioning, scaling, security, backup, etc. In this way, the user controls the **virtualized servers** and the specific computational capabilities and storage of the same: VLANs, access to raw block or file storage, load balancers, IP addresses, etc. However, he/she has limited control over the network settings. The most popular example is Amazon EC2, which allows the user to create virtual machines and manage them, or Amazon Storage Service (S3), which allows to store and access data through a web-service interface.

1.4.3.2 Cloud Management Platforms

In the classical cluster-like cloud architecture, a front-end executes the core management services (*e.g.* schedulers, APIs), hypervisor-enabled hosts laying behind provide VMs' resources, and datastores hold VM base images. They all are connected by at least one physical network to such support basic services, while VLANs are set for the VMs (). Thus, three main components conform a **cloud architecture**, *i.e.* storage, networking and virtualization, and cloud management platforms (CMPs) provide software solutions to administrate them in an integrative way.

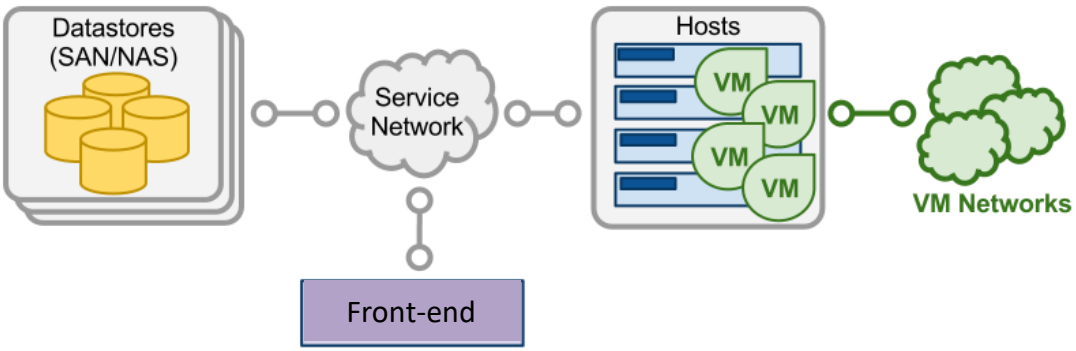


Figure 1.15: Classical cluster-like cloud architecture

CMPs are software products that provide management of cloud infrastructure to facilitate organizations the deployment, management, monitoring and control of virtual applications in an agile and cost-effective way. Necessary capabilities include:

- Infrastructure management: resource management across the infrastructure/s to allow the user configure and control life-cycle's virtual appliances (virtual machines or containers), storage and networks, always mediating with the local virtualization tools.
- Automation: improve agility and resource utilization with load balancing tools, automatic booting and contextualization systems, live migration tools, automatic snapshotting or backups, etc.
- Compliance and security: ensure compatibility with other cloud system adopting unified configurations, while protecting the user's environment.
- Accounting and cost management: log and monitor activity to not only optimize resource consumption, but also costs.

Depending on which of these features are bold, the platform management is more oriented to offer "infrastructure provisioning" clouds (also called "consumer" clouds) or "datacenter virtualization" clouds (also called "enterprise or organization clouds"). "Infrastructure provisioning" clouds are mostly designed for public clouds offering a simplified view of life-cycle's virtual resources and its underlying infrastructure, while "datacenter virtualization" clouds are targeting private clouds, with more transparent view of physical resources, and easy to adapt into existing infrastructure. Amazon AWS is the reference for "infrastructure provisioning", while VMware vCloud lays on the opposite side.

Open-source Cloud Management Platforms

After the success of the public cloud providers based on proprietary CMPs, many open-source CMP solutions appeared in the market during the last 10 years. They offer other deployment models like **on-premises or hybrids** clouds, motivated by particular enterprise or institution computing needs, hypervisor-agnostic requirements or simply wriggling away

from highly-priced advanced cloud services. Following, a brief introduction to some of the best known open-source CMP. Figure 1.16 classifies them according to their cloud model and flexi

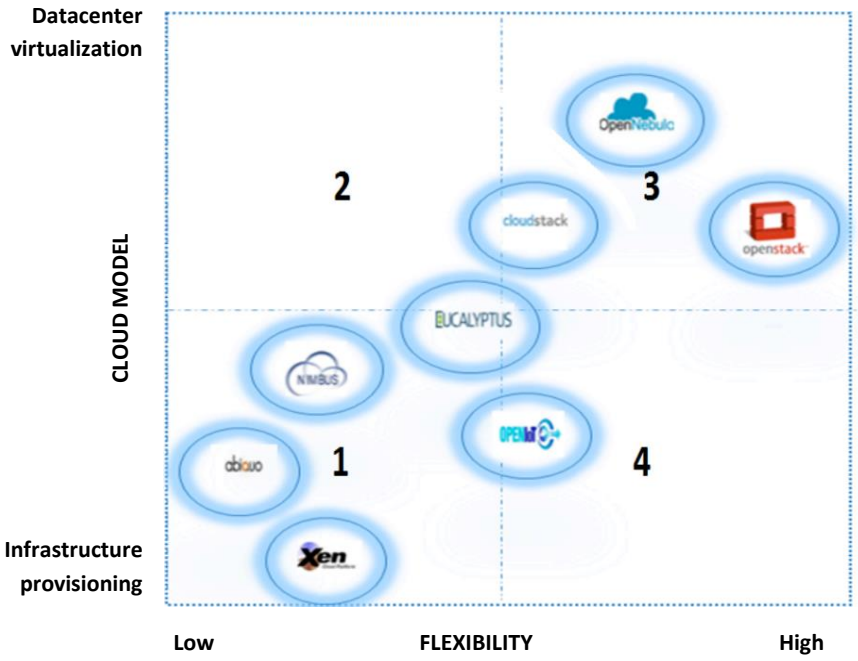


Figure 1.16: Classification of open source cloud management platforms In terms of flexibility and cloud strategy. Adapted from [95]

- Nimbus (2009) [89]: targets the scientific community by support for instance proxy credentials and rich batch schedulers. Also compatible with EC2/S3, it does not support VM migration.
- Eucalyptus (2008) [90]: designed primarily for private cloud, it supports Amazon EBS (elastic block storage) and AWS, but not VM migration.
- Opennebula (2008) [91]: can deal with dynamic resource needs, snapshotting, live migration, fault tolerance and load balancing. Besides, features good interoperability, supporting different access interfaces, including OCCI service interface, REST-based interfaces and emerging cloud API standards.
- Openstack (2008) [92]: designed for private and public clouds, it provides very good scalability, and flexibility. It exploits various open-source projects and manages VM migration, load balancing and fault tolerance.
- Cloudstack (2010) [93]: beside supporting most common hypervisors and operating systems, it can deal with snapshotting and supports high availability. However, shows reduced flexibility and lacks support for shared access.
- Abicloud (2009): manage and integrate clouds on virtualized heterogeneous environments. However, it does not support VM migration, fault tolerance, nor AWS.
- Xen Cloud Platform (XCP) (2012) [94]: a light platform that concentrates on provisioning XEN VMs, but user access is only via CLI.

A number of papers analyze and qualitatively compare each platform [95] [96] [97]. However, it is a challenging mission, as the previous figure illustrates, each platform is designed to offer certain cloud models, and institutions should decide which is the one that better fits their needs.

1.4.3.3 Cloud interoperability and portability

Portability is the ability of software (application cloud portability) and data (data cloud portability) to be transferred from one cloud system to another. Interoperability is the ability of two or more cloud infrastructures to exchange information in a common model. Addressing cloud interoperability and portability issues surrounding cloud infrastructures are one of the major discussions in the field. They have received considerable attention as standards are the key enabler for **cloud brokering** and **multi-cloud architectures**. Being able to easily reuse data among different cloud DaaS (Data as a service) or use your appliances across clouds, would be the solution to one of the cloud computing limitations: the vendor lock-in. If interoperability is achieved, several cloud infrastructures could be federated.

Several European and international research projects (like OASIS, SNIA, VENUS-C) have designed and implemented various cloud brokers, - *i.e.* single point entries to manage multiple cloud services, which are based on a number of specifications that [98] summarizes in the following table:

Activitiy	Context	Layer	Developing standards
Interoperability	Management	IaaS	OCCI, CIMI, UCI
	Platform	PaaS	Stub
	Application	SaaS	mOSAIC
Portability	Platform	IaaS (components)	Stub
		IaaS (image)	OVF
	Application	PaaS	CAMP
		IaaS	OVF, TOSCA, mOSAIC
	Data	SaaS	OData
		DaaS	CDMI

Table 1.1: Standards for cloud portability and interoperability.
They define standard operations for managing an infrastructure, image packing files, or application deployment procedures.

We cite here the more relevant:

- Open Cloud Computing Interface (OCCI) [99], Cloud Infrastructure Management Interface (CIMI) and Unified Cloud Interface (UCI) offer standard APIs for all kinds of IaaS management tasks (*e.g.* instantiate a VM).
- mOSAIC and jClouds are complete brokering APIs for resource discovery and usage.

- Open Virtualization Format (OVF) is a standard platform-independent packaging virtual images.
- Topology and Orchestration Specification for Cloud Applications (TOSCA) [100] standardize application description and orchestration to automatize its deployment.
- Cloud Data Management Interface (CDMI) [101] offer an interface for creation, retrieval, update and deletion of data elements from the Cloud

Unfortunately, none of these ongoing standards is widely adopted. In general, public cloud computing providers **resist adopting open standards**, and in fact, are open source solutions that incorporate compliance with them (*e.g.* via AWS API). Adoption examples are VirtualBox and VMware, that support OVF images, or multiple open-source CMPs (OpenStack, Microsoft Azure, jClouds, EMOTIVE cloud, Amazon EC2) that support OCCI. In fact, OpenNebula supports well open standards (CDMI, CIMI).

1.4.3.4 Cloud software

Cloud architecture layers as shown in Figure 1.17 is a good way to classify cloud-related software, which grows in diversity and number by leaps and bounds. Cloud-Native Computing Foundation shows in a complete diagram the major cloud-native technologies of the commercial and open-source scope [102].

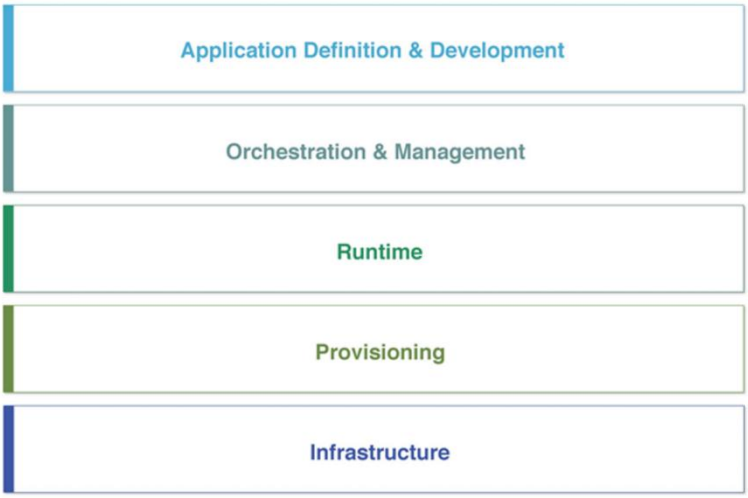


Figure 1.17: Cloud stack classification

In the following lines, a short introduction to some of these well-known pieces of software. They all operate on top of the cloud infrastructure, and are here classified according to the layer they run at.

Provisioning

Provisioning refers to the allocation of the user's system into the cloud up to the point of having a **functional virtual appliance**. It implies cloud services such as image repository, storage or networking. In case the user's system is built from several virtualized components, provisioning also implies orchestration and interconnection management. Provision might be based on manual CLI or GUI processes when directly interacting with CMPs; a semi-automatic process based on scripts or APIs, or a fully automated process following "Infrastructure as a Code" (IaC) practices.

Semi-automatic approaches include the use of cloud-vendor provisioning tools, for instance, OpenStack **Nova** scripts [103] for VM scheduling or load balancing, or OpenNebula **OneFlow** [104] for self-provisioning in front of conditional triggers like workload, or OpenStack **Heat** for orchestration deployment and scalability. In any case, they are dependent on the underlying CMP, so other cloud-agnostic tools are developed based on open cloud standards. For instance, the Barcelona Supercomputing Center developed a programming model enactment service (**PMES**) [105] that permits the programmatic deployment of VMs on OCCI compliant CMPs in an elastic manner.

Good examples of automatic provisioning of infrastructures are Packer and Terraform. **Packer** automatically builds from a coded recipes VM images for several platforms (*e.g.* OpenStack, Amazon). **Terraform** composes from recipes VM instances or sets of them – *i.e.* virtual clusters. The mature and widely-used implementation supports the majority of cloud vendors, including OpenNebula and OpenStack. Additionally, it interacts with software provisioning tools, discussed hereafter, to further configure the virtual instance.

Contextualization

Contextualization refers to the cloud context information passed to newly created VMs that enables a **first customization** of the virtual instances – *e.g.* the network configuration of the VM, user credentials, initialization scripts, or some free form data. Contextualization packages are installed as part of the VM stack and are executed upon boot time on different "runlevels". Cloud providers pass the information to the virtual environment using different strategies, like ISO file systems attached to the VM, file injection into the VM, or through a metadata server at a known location.

Nowadays, *de-facto* standard for VM contextualization is **Cloud-init** [106], available for many popular operating systems and supported by a good number of open-source CMPs. It enables advanced options like configuration of trusted CA certificates, creation partitions and file systems, configuration of cloud data stores, or execution of user-defined scripts. There exist other implementation-specific tools like **One-context** [107] for OpenNebula, or Amiconfig [108] for Amazon EC2.

Typically, contextualization is useful for setting configurations not known until instantiation, like DNS, hostname, private information like user's SSH keys or personal data volumes, or software that changes frequently with discrete user-defined commands.

Software provisioning

The philosophy of Development Operations (DevOps) is pushing towards the full automation of provisioning, deployment and configuration of software. In this context, runtime environments are build starting with vanilla hosts and dynamically installing in there all the necessary software required to run the applications at scale. Under such interest, lots of **configuration management tools** have arisen. Differences aside, they all

model specific actions on a local or remote machine in order to prepare and fully configure their applications according a given recipe. As such, developers and administrator tasks are considerably reduced. Popular examples are **Ansible** [109], **Chef** [110], Saltstack [111] or **Puppet** [112], each with their pros and their cons [113].

Orchestration & Management

Orchestration tools automate the deployment, management, scaling and networking of **coordinated applications**. Such tools, are everyday more used in the cloud environment because of the rise of microservices architectures, frequently implemented as container-based applications. Some examples would be Apache **Mesos** [114] or Nomad [115], workload orchestrators for distributed environments, or Docker Swarm [116] and **Kubernetes** [117], more focused on container-based clusters.

1.4.3.5 Bioinformatics in the cloud

Over the last 10 years, there have been numerous efforts in developing cloud-based tools to support very diverse bioinformatics tasks, including new processing algorithms ready to exploit horizontal scalability, portable compute infrastructures to outsource computing or exploit data/compute colocation, and secure remote data frameworks with flexible computing backend. These are approaches well positioned to exploit cloud differential benefits respect to other distributed computational infrastructures, such as its intrinsic availability, portability and elasticity. Following are mentioned several of cloud-based efforts, yet, it is a fast-moving field and as diverse as the bioinformatics field itself [118].

Cloud infrastructures

During recent years, an increasing number of commercial and academic **cloud vendors** are offering services for science. Although cloud computing has not been initially designed for science but for business, the model has grown increasingly popular among scientists, and commercial providers offer not only general-propose IaaS (*e.g.* Amazon EC2 or Microsoft Azure), but on-purpose infrastructures specialized or with particular high-end services, for instance, **GENESIS** [119], a cloud-based on GPU accelerated nodes, or **HPC Google** cloud [120]. Meanwhile, the number of public providers is increasing. Table 1.2 lists the most representative for life sciences domain, some with co-located public databases like the ***Embassy Cloud*** [121] hosted at the EMBL-EBI, other production clouds developed for university- or institute-based projects. Resource services are provided free of charge, although under their own access policy, usually bound to scientific projects, collaborations, or institutional agreements.

Rodeo	www.tacc.utexas.edu/systems/rodeo	Part of Texas Advanced Computing Center (TACC), allocating the central Galaxy public server. Comprises 256 processing cores and 2 TB of memory
Corral	www.tacc.utexas.edu/systems/corral	Part of TACC and also allocating Galaxy public servers. Comprises 20 PB of storage
XSEDE	xsede.org	Supported by NSF
Jetstream	jetstream-cloud.org	Science cloud in XSEDE located at TACC and Indiana University's Pervasive Technology Institute
Bionimbus Protected Data Cloud	bionimbus-pdc.opensciencedatacloud.org	Science cloud associated with Open Source Data Center (OSDC) that permits secure analysis of protected health information
Compute Canada	computecanada.ca	High-performance Canadian computing network spanning ACENET, Calcul Québec, Compute Ontario and WestGrid
de.NBI	www.denbi.de	Bioinformatics service provider in Germany spanning education, consulting, computing and storage, as well as databases
Embassy Cloud	embassycloud.org	Science cloud for EMBL–EBI affiliates including direct access to public genomics data sets
Helix Nebula	helix-nebula.eu	European open science partnership across industry and academia to provide cloud computing infrastructure
Nectar Cloud	nectar.org.au	Self-service Australian science cloud
Broad FireCloud	software.broadinstitute.org/firecloud	Part of Cancer Genomics Cloud (CGC) by the NCI. Offers collocated genomic data and analysis tools in the cloud
ISB-CGC	http://cgc.systemsbiology.net/	Part of Cancer Genomics Cloud (CGC) by the NCI
Seven Bridges CGC	cancergenomicscloud.org	Part of Cancer Genomics Cloud (CGC) by the NCI

Table 1.2: Public cloud providers.
Adapted from [122]

In such a way, the execution of bioinformatics applications and pipelines is moving from the traditional "download and analyse" paradigm to the “compute on-demand” model. Some IaaS are delivered as **pre-build infrastructures**. A classic example would be **CloVR** [123], a publicly available VM with a desktop interface and a CLI with configured pipelines for microbial analysis. CloVR runs on Amazon EC2 the pre-packed applications as a cluster of VMs, where data is transferred via SSH. Similarly, **Cloud BioLinux** [124] is another popular distributed virtual machine for bioinformatics computing on Amazon EC2. There exist other domain-specific workbenches, like **BioVLab** [125] for sequence mapping, **ProteoCloud** [126] for proteomics, or **BioVLAB-MMIA** [127] for transcriptomic. Most of them are compatible with Amazon EC2, some even feature GUIs or include workflow orchestrators, and all of them are delivered as deployable infrastructures.

Cloud Platforms

Another delivery model would be that of serving customizable **platforms** on the cloud (PaaS). Here, the most popular solution is **Galaxy** [128], an online user-friendly workbench

for bioinformatics tools, where such tools are plug in and out to achieve a tailored framework. With thousands of users, Galaxy was first born as a centralized analysis platform with a cluster-based backend behind. Nowadays, Galaxy interface can run atop different computing infrastructures, and the community has set up over 80 public servers besides the main server. Moreover, users can also install Galaxy on cloud clusters using **CloudMan** [129], and Galaxy containerization made possible a Kubernetes-based deployment to outsource jobs from public servers towards other clouds [130].

Alternative include stand-alone PaaS, like **Eoulsan** [131], a package developed to easily set up a cloud with a Hadoop implementation of the MapReduce algorithm. It includes a number of NGS tools. Another platform offering tailored research environments, in this case for metabolomics analysis, is **Phenomenal** [132]. Applications are presented to the researcher through a Galaxy server, or, in case of advanced users, a Jupyter Notebook [133] – *i.e.* an interactive development environment. Interestingly, the whole infrastructure deployment is customizable and automatically provisioned: Terraform deploys a virtual cluster with Kubernetes or Mesos, who in turn installs the chosen containerized metabolomics applications and launch the corresponding front-end and auxiliary services. Finally, **Indigo Data cloud** [134] does not offer virtual workbenches in particular, but any type of application the user might configure Indigo to compose. Indigo is a set of PaaS components (including authentication, data or orchestration modules) that automates the deployment and configuration of virtualized services given a tool configuration recipe – specified using TOSCA standard. The resulting virtual environment could be any, a Galaxy server, or simply a web application with a MySQL database.

Cloud Software

Regarding **standalone cloud-native** applications in bioinformatics, most solutions are focused on large-scale sequence processing, mainly exploiting the benefits of the Hadoop implementation of MapReduce. Examples are **CloudBurst** [135] for short sequence alignment, **CloudAligner** [135] for long reads, or **Balaur** [136] for mapping on hybrid clouds while preserving data privacy, **Crossbow** [137] for single nucleotide identification (SNP) identification, **PeakRanger** [138] for ChIP-seq peak calling, or Myrna [139] for RNA expression analysis, or **Falco** [140] for single-cell RNA-seq processing, amongst others. Additionally, a number of programming libraries that facilitate sequence manipulation and processing have arisen, such minimizes Genome Analysis Toolkit (**GATK**) [141] and **Hadoop-BAM** [142]. Hadoop implementation is also present in other areas, yet more residual. Works in other areas like molecular dynamics (MD) or medical imaging are in their infancy. There are few examples of cloud-native developments like **AutoDockCloud** [143], a workflow system that enables distributed screening. Some clustering methods [144] or search engines for proteomics [145] are also being explored.

Applications or frameworks may not be distributed but directly offered as **SaaS**, each bound to a particular IaaS. The list of commercial products is long, mostly targeting the genomics

sector, but telemedicine and medical imaging are increasingly popular fields. By way of example, **DNAnexus** [146] is an NGS analysis framework used by ENCODE and running on Amazon EC2, Globus Genomics [147] includes Globus toolkit with its GridFTP backend, and **CycleCloud** [148] runs on Azure a virtualize Gromacs and NAMD on GPUs for MD. Publicly available platforms are not that abundant. **FireCloud** [149] makes the perfect example of a SaaS offering an analysis platform. Hosted at the Broad Institute, FireCloud runs on a Google powered cloud and offers cancer genome pipelines while hosting public datasets (*e.g.* The Cancer Genome Atlas (TCGA) data). Similarly, other infrastructures dealing with data-intensive tasks like imaging data, either from electronic microscopy [150] or biomedical images [151] could well benefit from this computing model

Data services

Cloud data services have also been developed to approach researchers' requirements. Public funded scientific datasets are being made available in commercial clouds, like Amazon Web services (*e.g.* Ensembl, GenBank) [152]. Repositories with sensitive data are also exploring how to securely access and distribute it on the cloud. For instance, the European Genome-phenome Archive (EGA) is providing a **local-EGA** [153] implementation by which a private repository is stored on-premises installations, though it is findable through the central EGA and accessible upon the right credentials. **Beacon** [35] is a platform for global discovery and sharing of genetic variants on distributed and private clinical repositories.

1.4.3.6 Cloud open challenges

Despite the many benefits associated with cloud computing, it currently presents some issues and open challenges that prevent research institutions and companies from the systematic adoption of the popular model, particularly in those case dealing with sensitive data.

Security is probably one of the most important issues that providers must confront. Biomedical research, with omics data extracted from patient's samples, as in pharmacogenomics studies, as well as other clinical and epidemiologic studies, are imposing specific constraints in terms of privacy and security. Risks are mainly related to the failure of mechanisms for separating storage, memory, and routing between different tenants in a shared infrastructure. Centralized storage solutions and shared tenancies of physical storage space mean that the cloud users are at a higher risk of disclosure of their sensitive data to unwanted parties. Other threats on transferring data to third-party platforms include user's identity spoofing, data malicious alterations or information disclosure to authorized parties. Lately, there is considerable work aiming to adopt secure protection schemes and cryptographic techniques to protect the data. Even so, there are some **legal-related issues** in open discussion. The usual lack of transparency and visibility on IaaS,

especially in public clouds, prevents a user from controlling the actual data location, which might entail contractual issues, particularly if cross-border transfers occur, as data privacy laws are different in each country. Discussions over legal agreements for multi-cloud or federated strategies at European level are taking place [154]. Indeed, the first step has been the General Data Protection Regulation (GDPR) that came into force in 2018 and set a uniform data protection law for European members. Meanwhile, international certificate agencies like ISO (e.g. IEC27018, IEC27017) are auditing infrastructures and establishing standards to grant data protection as much as possible. At the same time, however, the possibility of deploying virtualized tools in private clouds running under controlled conditions provides a seamless solution for such concerns.

Another point to consider is **reliability and performance**. Data is to be accessible anytime and anywhere, thus, high-availability cloud strategies like load balancing cluster services or live migrations from host to host are every day more optimized and automatized. Performance is mostly bounded to virtualization or networking latency issues, but as mentioned, specialized cloud providers are working to offer HPC clouds [83].

Issues on interchanging data and cloud APIs standardization are also becoming relevant. Common standards are still immature and lead to **vendor lock-in** [155], as most cloud providers still use proprietary APIs, for both application development and data storage. The community is working in pushing vendor-agnostic applications relying on open standards to recover portability and flexibility among cloud providers.

Finally, computing's **pay-as-you-go model** of public clouds is proving to be a barrier to adoption of cloud computing in research [149], especially for publicly funded research. This is a quickly evolving field and with rapid reductions in cost, but in parallel, agencies are adapting requirements, more and more public providers are available, and resource consumption is also being optimized.

1.4.4 Data Management

Data is undoubtedly the cornerstone of new scientific research. In the last years, data stack has suffered structural changes, mainly driven by virtualization and its dynamicity, which has driven storage systems from static COTS hardware into software delivered. As discussed above, computing and applications rapidly adopted the new paradigm, which means that users can compute anywhere, rapidly accessing huge resources, while data could be fragmented over different infrastructures. However, data management shows important difficulties in this paradigm, and these are only stressed by big data. From the data policies to novel storage systems or strategies, data management has responded to the new scientific research.

1.4.4.1 Data governance

Until recent years, a focused and centralized strategy for the annotation, storage, and curation of research data is something that has not been widely considered within academic communities. The majority of research data sits, fragmented, on a variety of disk structures and is usually managed locally. In front of new research practices, data policies governing how data is backed up, disseminated and organized for short or long term reuse are gaining relevance.

Infrastructures intend to assist researchers with the digital data cycle. It implies (i) acquitting the data, (ii) validating it for the analysis, (iii) transforming it into understandable results, (iv) and preserving them in long-standing conditions, (v) to enable share and reuse.

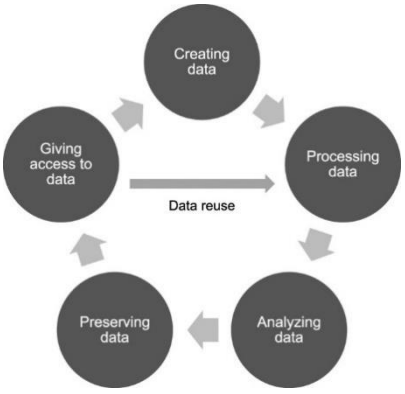


Figure 1.18: Research data life cycle

With new technologies, the implementation of these stages has not come without challenges, and infrastructures need to include a set of services, quite interconnected with FAIR principles, that aim to maximize resource interoperability:

- the use of services for assigning permanent identifiers to datasets and accessing data using these IDs. In such a way, stored data is more easily findable and accessible.
- metadata services for assigning metadata to a digital object identified by identifiers and accessing the metadata. Good quality of data annotation ease reusability and interoperability, especially if controlled vocabulary or data models are used.
- authentication services for identifying researchers and authorization services for determining which datasets researchers can access. As such, platforms can support controlled access data.
- standardization, encapsulation, and versioning of processing services for executing bioinformatics tools so that data can be traced and harmonized.

1.4.4.2 Data-storage solutions

The number of data-access solutions increases as much as the diversity of scenarios where they are intended to be implemented: easy anytime/anywhere data access, simplicity for data sharing or for making it public, efficient access to large volumes of data, long and secure archiving, strict and granular access, changeable data, etc. Hence, according to the system's purposes, we could define data-access solutions as one of the following:

Typical grid and cloud lower level data-access tools

Lustre (Intel) and **GPFS** (IBM) are parallel file systems often used as high-performance scratch systems for computational clusters. They can only be shared within the local cluster and has not transparency, although they show high efficiency, high availability and low metadata-access latency. **GlusterFS** is an interesting alternative that does not need metadata servers to locate data across nodes, which ensures good performance scaling, yet affects elasticity and reconfiguration. Block storages, either as NAS or SAN, can be built on top of any of them. SAN keeps high performances, yet NAS provide shared access with limited scalability. Alternatively, scalability can be maximized using object storages that manage data as objects, but there is no hierarchical structure nor block access like in traditional file systems, and no file edition is permitted. OpenStack **Object Store** (known as Swift) and Amazon S3 are the most popular object storages.

Tools for any time/anyplace data access

They focus on the ease of access. Widely known examples are **Dropbox** and Google Drive. Most popular OS have clients for transparently mounting them and as virtual file systems, letting the user to manually handle discrepancies in case of desynchronization. Furthermore, they impose rigorous limits on the storage size and transfer speed, which become an obstacle for conducting research in a geographically distributed manner. **ownCloud**, or his successor **NextCloud**, offer a more flexible solution, transferring data over HTTP via WebDAV, allowing users to maintain full control over their data location, and providing file sharing using public URLs. However, it might not be sufficient for data-intensive applications.

Tools for distributed data processing

One of the most prominent tools for remote data access is **Globus Connect**, built on the GridFTP protocol to provide fast data movement and data-sharing capabilities inside an organization. Another possibility is the use of parallel file systems like those typically set for grid nodes, mentioned above. Solutions built on top of object storage systems are equally valid. For instance, **CephFS** shows good performance and features simple POSIX-compliant file systems instead of object storage delivery. Alternatively, on another direction, map-reduce paradigm effectively process distributed data, and Hadoop Distributed File System (**HDFS**) is designed to support it, as it permits to stream large data sets at high bandwidth for the user's processes.

Tools for a unified view of multi-organizational data

They aim at providing an abstraction layer on top of the storage resources across multiple organizations – *i.e.* across multiple clouds. Files have a unique namespace across data servers and data-management rules can be configured to control the data location policy across servers. An exemplary tool is **iRODS**, used in EUDAT project. Data can be simply stored at designated folders on any data server and its metadata is centralized in a

database. POSIX interface is available upon FUSE mount, and several storage systems are supported and integrated as plug-ins (GridFTP, Amazon S3). Importantly, multiple iRODS installations can be federated into a "Zone". OneData represents an IRODS alternative, as it features very similar characteristics. Yet, **oneData** [156] provides location transparency for data stored across federated servers, and importantly, it offers as build-in "copy-on-read" policy that transfers data among Zones only when at the moment it is required and transparently. Indigo data cloud is using oneData. Another slightly different perspective is given by CernVM file system (**CVMFS**) [157] originally designed for efficient software distribution. It provides a FUSE-based POSIX interface for local users, but all writes occur at a central repository server, whose data is distributed with an efficient caching system around nodes. Additionally, CVMFS has no data privacy mechanisms at the metadata level, so the system is more used for multi-organizational data distribution than for data exchange.

1.4.4.3 Databases

Relational database management systems (RDBMSs) has been the primary database technology used since the beginning of data warehousing. They are well-matched to client-server programming and today they are the predominant technology for storing structured data in Web. RDBMS stores data values in fixed tables and logical relationships among them, and ensures data integrity using a strict transaction model called **ACID** (Atomic, Consistent, Isolated and Durable) by which once a transaction is complete, its data remains consistent and stable in disk and memory. Moreover, most relational engines adopt Structured Query Language (SQL) as their querying language, a very powerful and flexible standard. When considering the CAP theorem [158], which states that a distributed system cannot simultaneously provide (C)onsistency – all reads get last write –, (A)vailability – clients always can read and write – and (P)artition Tolerance – system up despite network drops), RDBMS falls into the CA category (Figure 1.19), with a poor scalability with vast volume of data. Popular implementations are **MySQL** (currently owned by Oracle), MariaDB or PostgresSQL.

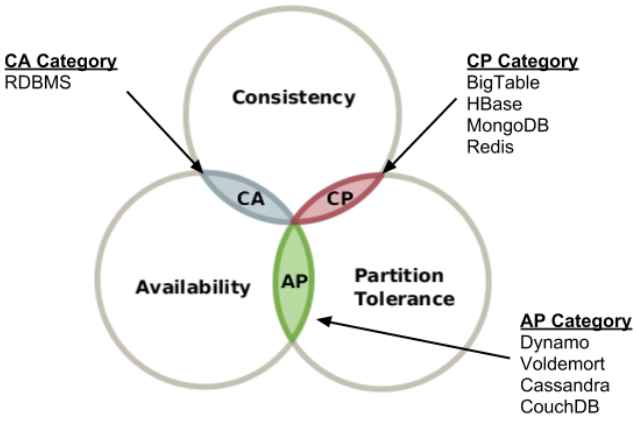


Figure 1.19: Database technologies into the CAP theorem

However, there are various types of data that may not require a fixed schema, avoid some form of join operations, and typically scale horizontally – these are known as unstructured data and RDBMS fails to handle so. New omics technologies and biomedical machinery are massive producers of such type of data. **NoSQL** encompasses a wide variety of different database technologies developed in response to that complex data and the big data requirements (the "5 Vs"). There are several categories of NoSQL databases, each with their own specific attributes and limitations, as they are designed to solve specific problems:

- Key-value stores: are the simplest NoSQL database type, storing attribute names (the "key") with values. They are good for anything requiring in-memory computing, in-memory analytics, real-time communications, and large amounts of streaming data (*e.g.* **Redis**).
- Document databases: store documents, which can be structured or unstructured, making them ideal for unstructured and semi-structured data (*e.g.* **Mongodb**).
- Column family stores: are optimized for querying large datasets, storing data as columns, not rows, in tables that can have millions or billions of columns (*e.g.* **Cassandra**, **HBase** (Hadoop DB)).
- Graph databases: store information on objects and their relations to one another. Social networks are built on graph databases (*e.g.* **Neo4j**).

Other NoSQL strategies consist on optimizing the search engines, usually supporting complex search statements, streaming, and full-text search (*e.g.* **Elasticsearch**), or optimizing the handling of certain data types, for instance, time-series databases (*e.g.* InfluxDB), or RDF databases for storing subject-object-predicate triples (*e.g.* MarkLogic).

NoSQL systems are designed for specific scenarios with different prioritizations, although generally speaking, they properly manage fault tolerance, scalability, and availability, in exchange of not immediate read and write consistency (AP category from CAP theorem). **NewSQL** is a new branch of relational databases that aims to regain this consistency returning to ACID transactions while preserving good scalability capabilities typical from NoSQL systems. NewSQL systems can be loosely grouped into:

- In-memory databases: stores data in RAM with durability and infrequent in-disk storage (*e.g.* **VoltDB**)
- Optimized storage engines for SQL: data is partitioned 'shards', capable of running on a large number of nodes without suffering bottlenecks (*e.g.* **Spanner**).

1.5 Science on the Web

The take-up of Web technologies in bioinformatics is ubiquitous. Since its creation, the World Wide Web (WWW) has established itself as a *de-facto* standard for information sharing changing the way users and developers disseminate, discover information, collaborate, and distribute data and computation. The Web is the biggest, most successful, and most programmable distributed system architecture ever [159]. Its decentralization avoids social and technical bottlenecks, and its openness permits information reuse and fairness [160]. Such features made the Web the natural gateway for science since the very beginning. And resource digitization did not only change how we do research but also who does so. e-infrastructures are meant to bring together those high-end researchers used to work in high-performance environments, with the long tail of researchers using the Web. Several strategies are employed for exploiting the Web potential in online research.

1.5.1 Service-Oriented Science

Service Oriented Architecture (SOA) is a model that looks for efficient and productive coding by splitting application architecture into single-purpose reusable units, rather than building them within each individual application. SOA is typically implemented through **web services**, that communicate among each other and with clients using standard protocols [161]. In life sciences, web services, originally designed to support machine-to-machine interaction over the network, have become a road for systematically standardizing the multitude of bioinformatics tools available [162]. Three standards recommended by international consortiums (W3C [163], OASIS [164]) define the most basic framework for representing web services: XML (Extensible Markup Language) as the format used to contain the data and provide metadata around it; SOAP (Simple Object Access Protocol) is used to transfer the data; WSDL (Web Services Description Language) is used for describing the services available; and UDDI (Universal Discovery, Description, and Integration) lists what services are available.

Figure 1.20: Web services architecture (a) versus REST services (b)

However**, services on the web** are not limited to these recommendations. Some emerging standards are widely used, like JSON (JavaScript Object Notation) for data structure, and RESTful web services are very popular. These are implemented directly on the HTTP protocol and follow the principles of Representational State Transfer (REST), where a

unique URL is a representation of a unique object. Other data transfer protocols still in use include RPC (Remote Procedure Call), again XML-based (XML-RCP) and with a similar SOAP request-response, but supporting only a subset of his functionalities (only over HTTP(S), with basic AUTH, and with no support for WSDL specification). Other protocols like WebDAV (Web Distributed Authoring and Versioning) enables the use of web services as generic file servers. WebDAV is an HTTP extension that implements extra methods like COPY and MOVE.

Web services are widely used in the context of the life sciences applications, and several **registries** and repositories have appeared along the years, for instance, BioMOBY [165], EMBRACE [166], BioSWR [167] or BioCatalogue [168]. However, ELIXIR bio.tools [19] is positioning itself as the most promising repository for bioinformatics tools with more than 14,000 services annotated from more than 7,000 providers. And web services approach is an up and coming model whose **benefits** are every day best exploited by the scientific community. The architecture enables cross-domain expertise integration, by establishing a common information framework between different bioinformatics areas, also allows data access in a systematic and standardized manner regardless of the underlying data retrieval mechanism, and by design and as discussed above, the native interoperability of web services make them very suitable for composing scientific workflows.

Lately and particularly after the emergence of cloud technologies in research, the most important principles of SOA - service-oriented, autonomy, reusability, composability, discoverability and statelessness - have been extended and applied to heterogeneous storage and compute virtualized components able to interoperate with each other at the web service level by following the standards and protocols. These cloud-delivered services are popularly described under the term **'as-a-service'** (platform as a service, workflow as a service, container as a service, etc), and the "servicification" is reaching infrastructure devices as these become more powerful and complex (security as a service, network as a service, authentication as a service, etc.).

Modern e-infrastructures and middleware use a SOA approach on a cloud-based infrastructure, *i.e.*, combine web services interoperability and virtualization technologies [159]. This computing paradigm frees the researchers from dealing with the maintenance of the physical infrastructure and provides many potential benefits, including scalable IT infrastructures, QoS (Quality of Service) assured services and customizable computing environment. Yet, as discussed previously, the dynamic service composition needs to comfort data format compatibility and other standing integration challenges.

1.5.2 Web applications

A web application enables information processing functions to be initiated remotely from a browser and executed partly on a web server, application server and/or database server.

Through the use of a set of standard languages governed by the World Wide Web Consortium [163] - primarily Hypertext Mark-up Language (HTML), but also CSS, JavaScript - web applications ensure the interoperability with web browsers, as well as top-level tools like multimodal devices (*e.g.*, voice browsers) or semantic web models.

Although a number of web application architectures are proposed, bioinformatics applications typically adopt the **classic thin client-server** architecture, (Figure 1.21) with a client part communicating via the request-response HTTP cycle (HyperText Transmission Protocol), with a server web application on the other side, with access to the computational and data resources. WWW technologies like Common Gateway Interface (CGI) and FastCGI support the execution of scripts (*i.e.* in Python or Perl) on the web server, as well as Java Servlets and other programming environments. But as the complexity of the operations or the web frontend grows, the application quickly gets heavy and requires a complete implementation on the backend of the server-side for managing high-end computations or interacting with the data tier (Database Management System (DBMS)). Some common backend languages are Ruby, PHP, Java and Python, although often are run on frameworks that simplify the web development process mostly providing templates, data structures and component repositories inspired by the Model-View-Controller (MVC) UI pattern (*e.g.* Ruby on Rails [169] for Ruby; Django [170], Flask [171] or Web2py [172] for Python; Laravel [173], Symfony [174] or Slim [175] for PHP). Frontend development is based on W3C1 standard languages, but there are also frameworks like Bootstrap [176] or Angular [177], as well as JavaScript libraries like JQuery [178] or React [179] and CSS extensions like Sass [180] and Less [181].

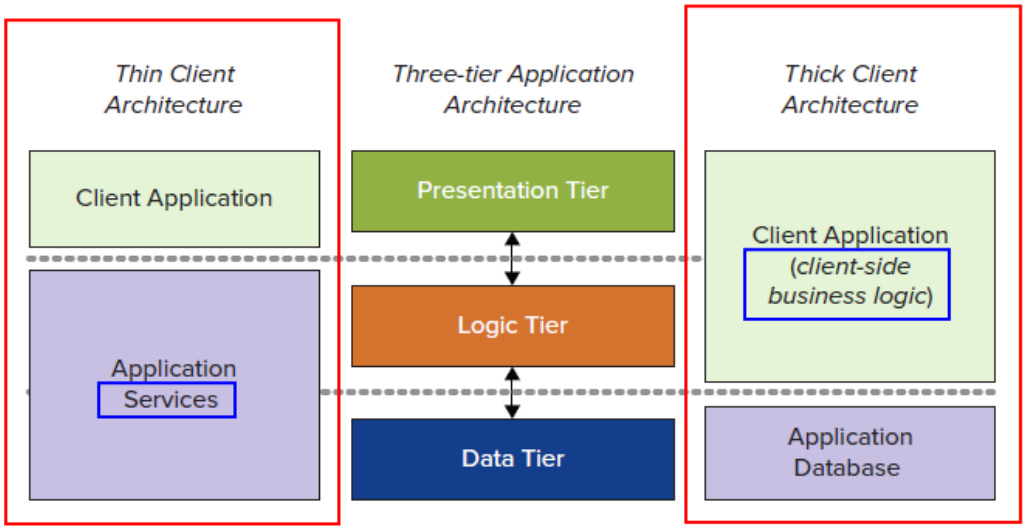


Figure 1.21: Web applications architectures.
Out of the three application layers (presentation, logic and data), thin clients reserve only the presentation tier for the client side, while thick clients implement also the logics in them.

Now, and increasingly, those web applications where the client-side performs most of the operations are becoming much more common, *i.e.* they run within the web browser on increasingly powerful user's devices with less or no support at all from a remote server. These **thick clients** (or Fat clients) might be even installed and administered locally on the user's computer, but run in a web browser in order to achieve independence from hardware platforms and operating systems. In other models, applications are automatically downloaded from a web server when a user starts them but do not communicate with the server while running. Another type of thick clients is "server-agnostic", *i.e.* serverless architectures able to connect to multiple (cloud-based) remote servers and data stores, which offer great flexibility and scalability. Such applications are often connected to the so-called web services or microservices (section *1.5.1* Service-Oriented Science). Other graphical tools are also available as Java applets, which are usually server-less and can be embedded inside web applications. For security reasons, and for avoiding the plugin installation required for Java Applets or Flash, rich web applications are today being totally developed with the web frontend languages mentioned above, taking profit of the web standards like HTML5 and JavaScript and supplemented with CSS, SVG and Ajax (Asynchronous JavaScript and XML). These methodologies enable Single-Page Applications (SPAs), which allows a dynamic content update to the current page instead of loading completely new pages.

New features in browsers allow for clients to move **beyond the client-server pattern**. For instance, web sockets provide direct communication between a client and some other server - no longer relying on browser's HTTP requests [182], and there examples of libraries allowing browsers to execute peer-to-peer file sharing protocols [183], or turning browsers into nodes of a distributed data store [184]. In other words, web application clients seem to be evolving into something more like a node of a distributed system, following the trend of the distributed data and computing paradigms.

1.5.2.1 Data Portals

Web portals were among the first type of bioinformatics web applications and nowadays are broadly implemented. They are specially designed to serve as **information aggregators**. As the number of distributed data resources available on the Web increases, more imperative becomes the development of central web portals imposing an integrative view over third party resources. Research portals aim a one-stop-shop feel for researchers with easy navigation and seamless access to information and authentication for underlying research e-resources.

Shortly after WWW was created, ExPASy [185] was released as the first web server in life sciences offering data access to multi-databases in an integrated way, a model still up-to-date today. Others like Entrez [186] or SRS [187] appeared shortly after. Although the use of the web standards in the recent web applications hardly resembles that from the 1990s,

the conceptual remain unaltered. They do contain more than hyperlinks to other related sites, as to some extent, they integrate the resources per se, either consuming them from data providers, primarily using APIs (Application Programming Interfaces), or elaborating processed cross-linked data to offer a seamless aggregation exploiting, for instance, web semantic strategies. Shortly after WWW is created, ExPASy [185] is released as the first web server in life sciences offering data access to multi-databases in an integrated, a model still up-to-date today. Others like Entrez [186] or SRS [187] appeared shortly after. Although the use of the web standards in the recent web applications hardly resembles that from the 1990s, the conceptual remain unaltered. They do contain more than hyperlinks to other related sites, as to some extent, they integrate the resources per se, either consuming them from data providers via APIs (Application Programming Interfaces), or elaborating processed cross-linked data generated, for instance, using web semantic strategies – an approach every day more relevant with the huge among of unstructured e-resources that digitalization has brought to the surface.

1.5.2.2 Workbenches

A natural evolution of web portals is going from promoting resource accessibility over the Internet, to actually utilizing them. Firstly, it appeared applications dedicated to executing particular programs on the Web, and the plethora of bioinformatics tools quickly made obvious the need for **integrated analysis platforms**. With the primary objective to enable a convenient data analysis that minimizes user's effort, workbenches might include not only access to computational tools and data, but also and to a certain extent, visualization, storage, automated workflow design, and execution, or data management services. In this manner, data process transparency and reproducibility are promoted from the infrastructure. Modern workbenches are expandable and multi-functional rather than narrowly specialized for a certain research domain. The computational infrastructure hidden behind the web application might correspond to any of the computational models above presented or simply be a local installation.

Popular web-based workbench frameworks have already been implemented. Examples are Chipster [188], Omics Pipe [189], GeneProf [190], GenePattern [191] or Galaxy [192]. They all let users run individual steps or entire pipelines on a remote computing system with the framework keeping track of the executed analysis. They all offer graphical user interfaces for making the tools accessible to scientists without extensive skills in programming. Yet, Galaxy is the solution more widely adopted, as many of these frameworks are exclusively focused on specific NGS analyses, or are simply are not designed in an expansible and modular way to support the plug in and out the new analysis tools. Omics Pipe would be another modular workbench, and both are implemented as to run on top of several distributed computational back-ends. Table 1.3 summaries the features of some of these

workbenches, including those focused on scripting frameworks, like Snakemake [193] or Nextflow [194]

Tool	Web page	GUI	Command line	Parallelization	Portability on cluster	Programming skills	Database integration	Integration of scripts	Workflow sharing	Visualization tools
Biokepler	http://www.biokepler.org	Yes	No	Yes	Yes	No	No	No	No	Yes
Bpipe	*Sadedin, Pope & Oshlack, 2012* https://github.com/ssadedin/bpipe	No	Yes	Yes	Yes	Yes	No	Yes	No	No
Chipster	*Kallio et al., 2011* http://chipster.csc.fi	Yes	No	Partial	Yes	No	No	No	Yes	Yes
ClusterFlow	http://clusterflow.io	No	Yes	Yes	Yes	Yes	No	No	No	Yes
Dagr	https://github.com/fulcrumgenomics/dagr	No	Yes	Yes	Yes	Yes	No	Partial	No	No
Galaxy	*Giardine et al., 2005* https://usegalaxy.org	Yes	No	Partial	Yes	No	No	No	Yes	Yes
GenePattern	*Reich et al., 2006* http://software.broadinstitute.org/cancer/software/genepattern	Yes	No	Yes	Yes	No	No	No	Yes	Yes
Kronos	*Taghiyar et al., 2017* https://github.com/jtaghiyar/kronos	No	Yes	Yes	Yes	Yes	No	No	No	No
Loom	https://github.com/StanfordBioinformatics/loom	Yes	Yes	No	Yes	Partial	Yes	No	Yes	No
Moa	https://github.com/mfiers/Moa	No	Yes	Yes	Yes	Yes	No	No	Yes	Partial
NextFlow	http://www.nextflow.io	No	Yes	Yes	Yes	Yes	No	Partial	Yes	No
PipEngine	https://github.com/fstrozzi/bioruby-pipengine	No	Yes	Yes	Yes	Yes	No	No	Yes	No
QuickNGS	*Wagle, Nikolić & Frommolt, 2015* http://bifacility.uni-koeln.de/quickngs/web	Partial	Yes	Yes	Yes	Partial	Yes	Yes	No	Partial
Rubra	https://github.com/bjpop/rubra	No	Yes	Yes	Yes	Yes	No	Yes	No	No
SnakeMake	Köster & Rahmann, 2012 https://bitbucket.org/johanneskoester/snakemake/wiki/Home	No	Yes	Yes	Yes	Yes	No	Yes	Yes	No
Toil	https://github.com/BD2KGenomics/toil	No	Yes	Yes	Yes	Yes	No	No	No	No

Table 1.3 Analysis data platforms. Source [259]

1.5.2.3 Virtual Research Environments

Virtual research environments (VREs) represent a step beyond workbenches. The term is born together with the e-science concept, as part of the technologies seeking to support a vision of large-scale collaboration. Capable of doing any of the tasks usually assigned to workbenches, VREs are presented as the **interface of a infrastructure** that allows access to data and services in an environment focused on a particular research activity [195]. Thus, VREs are part of an infrastructure more than a free-standing product as workbenches. In line with current research practices, VREs usually integrate community-building components, like forums, comprehensive views of data and information or other support tools. The ultimate goal is be able to enroll researchers from any discipline that seek to bring together despair expertise to collaboratively work on a multidisciplinary problem.

Multiple and very variated are the interpretations and implementations of VREs in bioinformatics. In the context of **LifeWatch** [196] project, the European e-infrastructure focused on biodiversity, a full catalogue [197] of VREs dedicated to diverse subjects (e.g. marine metabolomics, eco-diversity, etc.) has been developed, each with their own logistics, interface, datasets, etc., but all operated on LifeWatch grid resources. Another approach is that of **Workflow4Metabolomics** [198] and **Phenomenal** [132], both dedicated to comprehensive metabolomics data processing.

2. Objectives

In line with the evolution of computational research infrastructures in the field of bioinformatics the following main questions arise:

- Which are the most appropriate technological solutions to fulfill user needs in modern bioinformatics
- How that technology can be abstracted in a way that researchers can concentrate on scientific questions.

To focus on specific objectives for this doctoral work we benefit from a serial of bioinformatics projects that will provide the appropriate use cases. Following such projects, the specific objectives of the work are:

1. The development of a clinical data management system appropriate for a multi-centric biomedical research project. Such a system should provide secure and efficient data storage and management, based on data capture approaches accepted at the clinical side.
2. The development of cloud-based solutions for the establishment of production-level virtual research environments, adaptable to a series of scientific use cases. Such infrastructures should combine state-of-the-art technical solutions, and the appropriate interfaces to data management, and to final users and developers.

3. Software, Data & Methodology

The present chapter includes a selection of IT solutions and methodologies used during the implementation of the infrastructures presented in the dissertation.

3.1 Software components

3.1.1 Databases technologies

The data tier of the web applications developed in the present work is mainly based on databases, which vary in architecture and model depending on the nature and architecture of the data they are mean to store. Two different technologies have been used, MySQL and Mongodb, and they are compared in the following table, that summarize engines' features:

3.1.1.1 Relational databases, MySQL

MySQL[199] is the most popular open-source database. Presently owned by Oracle, the relational database management system (RDBMS) employs the concept of storing data in rows and tables, thus, a **rigid structure** is to be clearly defined in advance and kept essentially unaltered. Changes in data model usually imply the database schema modification, yet, data consistency is easily granted.

MySQL's limitations are common to most relational databases: millions of read/write highly affects the performance and hence horizontal scaling is not quite easy. The constrain inherits from the strict transaction model in use**, ACID**. "Write consistency" is quite suitable for applications which can't bear data loss or inconsistency. Under the CAP theorem, MySQL opts for Consistency and Availability (*i.e.* **CA**), so data will be consistent between all nodes as long as nodes are online.

Even with relational design limitations, **replication and clustering** are available for MySQL to some extent. Replication, either master-slave or master-master replication, quite easily add reading scalability, and multi-master replication writing scalability, yet, the latter only for separate applications, each using its master. MySQL Fabric and Cluster [200] are not standard implementations that support data sharding for MySQL.

For the **present work**, the MySQL server corresponds to a standard installation hosted at the (BSC) Life Cluster front-end providing service to several projects. Here, it is being used to store phenotypical data: up to 200,000 records of clinical variables in a key-value pair pattern, together with well-structured data of clinical cases and donations. Although clinical fields feature a very changing nature, for legacy and technology maturity reasons, MySQL is the chosen system. MySQL is around the block for a long time and supported by a wide

community. Flexibility is achieved using a particular database design further discussed at *4* . Results. Furthermore, clinical studies have a very clear structure easily to be defined in advance. Apart from offering strong security, MySQL servers have easy and low-maintenance implementations.

3.1.1.2 Mongodb

Mongodb [201] is one of the most popular document-oriented databases under the banner of NoSQL databases. It was first released in 2010 and employs the format of key-value pairs, here called document store, in collections as BSON files (binary version of JSON files) which facilitates data exchange between web applications and servers in a human-readable format.

Moreover, Mongodb offers great efficiency and reliability while meeting high storage capacity and speed demands thanks to the adoption of the **BASE consistency model** (Basic Availability, Soft-state and Eventual consistency), popular among NoSQL databases, which generally have loosened their requirements for immediate consistency, data freshness and accuracy in order to gain other benefits, like scale and resilience. The database does not guarantee consistency of replicated data at write time, but it does eventually, for instance, at read time. In this way, Mongodb supports atomic updates on a single document level. In summary, Mongodb system has opted to provide Consistency & Partition tolerance sacrificing availability so that users of one node will have to wait for any other nodes to come to an agreement before being able to read or write to the database.

Mongodb provides also **high scalability** and service availability as it supports auto-sharding and on-board replication. Its sharding breaks up a collection into subsets of data and distributes it, with redundancy and automatic failure recover, and transparently to the application. Data replication is also supported, only in a master-slave model**.**

But on top of all that, the **schema-free** implementation is one of the best Mongodb features. It eliminates the prerequisites of defining a fixed structure which provides high flexibility and facilitates the ability to change the structure of a record. However, due to the absence of joins and transactions, the user needs to frequently optimize the schema based on how the application will be accessing the data. Moreover**,** schema-less documents might cause problems with data consistency, which should be resolved by the application. These are reasons might motivate the use of JSON schemas ($jsonSchema) for validating document collections, an option appeared in Mongo 3.6.

Mongodb results in a good solution for handling unstructured data in a cost-effective way. In the **present work**, stored data corresponding to:

- Metadata (operational and functional) accompanying file system entries.
- Tools registry repository (software developers' applications and visualizers)

- Application's management data (job registry, user administration, etc)

It is convenient to loosely define such data at the database level, in particular for the two first cases, as they are very changing data models, with complex, nested content, and hierarchical data. Furthermore, they are easily abstracted as programming objects at the application's library, and the JSON format is very handy to directly map them into the database model. Even offering such flexibility, Mongodb is not comparable to other NoSQL systems more target for big data, but still.

Furthermore, the in-built sharding solution of Mongodb offers good horizontal scalability very suitable for cloud-based services as ours, as it aligns with the horizontal elasticity and agility provided by cloud resource pooling. The expansion might be flexible, easily achieved by adding more machines and RAM to the system.

3.1.2 Cloud-related software

3.1.2.1 Cloud management platforms

Cloud management platforms (CMP) are the suite of integrative software tools used to monitor and control cloud computing resources. They incorporate interfaces for self-service resource management and more-advanced offerings, like workload optimization, support for configuration of storage and network topology, shortly introduced at *1.4.3.2* section. Here, the two major open-source IaaS CMPs are used: OpenNebula and OpenStack.

OpenNebula

OpenNebula is designed to help building simple and reliable, datacentre-like clouds on existing IT infrastructure in a cost-effective way. OpenNebula can be managed through a Web-based UI (Sunstone) that provides access to all features, yet it also has a CLI and a powerful RCP-XML API, essential to programmatically manage the cloud from other systems. Other main components of OpenNebula are the nodes, the image repository, the daemon, and the drivers. OpenNebula is structured as the classic **cluster-like architecture** with a front-end, where the API, Sunstone and other services are located, and a set of hypervisor-enabled cluster nodes, where VMs are instantiated. Such model is very convenient for our HPC cluster installations, as it permits to transfer compute nodes from the HPC to the cloud with a certain flexibility.

In terms of management, OpenNebula is organized as other CMPs: *VM images > VM templates > (VM services) > VM instances:*

- *VM images* refer to the disks images containing either file systems or operating systems. Supported formats are QCOW2, RAW, and VMDK.

- *VM templates* correspond to YAML-like files defining the set of attributes required to compose a VM instance. Apart from specifying one or more VM disks, it includes information on CPUs, virtual CPUs, RAM memory, network interfaces with their gateway or IP, contextualization details, and many other attributes. Snippet 3.1 shows a simple VM template.
- *VM services* are multiple and varied, in fact, these are one of the differential traits among CMPs. OpenNebula offers self-provisioning tools for managing clusters, auto-scaling services (like OneFlow, below discussed), network definition tools, or VM authentication management, among others. However, the platform's philosophy is offering good connectivity and interoperability capabilities while allowing new services implemented as add-ons.
- *VM instances* are the result of deploying a VM template, either manually or via one the advanced services.

```
-   DESCRIPTION = "TADBIT FOR MUG"
    .
    CPU = "8"
-   CPU = "12"
    MEMORY = "189152"
    MEMORY_UNIT_COST = "MB"
    DISK = [
      IMAGE = "mg-tool-tadbit"
-     TARGET = "hda"
-     DRIVER = "qcow2" ]
    FEATURES = [
      ACPI = "yes" ]
    GRAPHICS = [
      LISTEN = "0.0.0.0",
      TYPE = "VNC" ]
    HYPERVISOR = "kvm"
    NIC = [
      NETWORK = "kvm-servers-mmb"
-     IP = "range" ]
    OS = [
      ARCH = "x86_64",
      BOOT = "" ]
-   CONTEXT = [
-     NETWORK = "YES",
      SSH_PUBLIC_KEY = "$USER[SSH_PUBLIC_KEY]" ]
```

Snippet 3.1: Simple VM template for OpenNebula

Images, templates, instances, networks and other services, are all managed either through the Command Line Interface (CLI) or the Sunstone UI. Additionally, OpenNebula supports several cloud client interfaces, such as Amazon EC2, Google Cloud or vCloud, demonstrating being vendor-neutral. And importantly, the **Open Cloud Computing Interface (OCCI)** (based on draft 0.8) is natively supported, providing a standard endpoint to create, control and monitor VMs. Indeed, the analysis platforms here presented interact with OpenNebula via the OCCI connector.

OpenNebula supports XEN, KVM, and VMWare ESX hypervisors, yet our installations are all based on the popular **KVM** virtualization platform, operationally managed via the open-source virtualization library Libvirt. KVM is outperformed by purely bare-metal hypervisors like Xen, yet, it is widely used because of its simplicity and ease of use, running directly on Linux's kernel and being delivered by default in most Linux distributions.

We have had access to two cloud infrastructures managed by OpenNebula, one at the Barcelona Supercomputing Center (BSC), the other at the Institute for Research in Biomedicine (IRB). Both are based on cluster-like architectures and correspond to two private on-premises deployments operated and managed by us from the respective institutions. Details on the underlying hardware are found in Table 3.1.

- The **"*INB Cloud*"** is hosted at the Barcelona Supercomputing Center (BSC) and installed on top of on a computational cluster It includes an externally accessible front-end which acts as a gateway for the internal cloud VLAN on HTTP(s), FTP(s) and SSH. A shared common storage system of some TBs is accessible to both, frontend and compute nodes. Accessed as a Network Attached Storage (NAS) on a high-speed network, disks export several NFS (v3) endpoints on top of a GPFS distributed file system.
- The **"*MMB Cloud*"** cloud is hosted by the Molecular Modeling and Bioinformatics Unit of the Institute for Research in Biomedicine (IRB). Likewise "*INB Cloud*", OpenNebula manages a series of homogeneous nodes virtualized in KVM and the storage is also block-accessed over the network, yet in this case, the NAS protocol is CIF, while the distributed storage solution is based on NetApp.

Cloud Name	Cores	RAM Memory	Storage System	OpenNebula	Project
INB Cloud (BSC)	4 x 12 cores Intel Xeon E5649	4 x 96 GB RAM	Shared GPFS of 78TB (NFS)	OpenNebula (v5.2.1)	transPLANT MuG
MMB Cloud (IRB)	3 x 58 cores Intel Xeon E5-2640	3 x 442 Gb RAM	Shared disk 7.5TB (CIF)	OpenNebula (v4.4.1)	MuG

Table 3.1: OpenNebula cloud infrastructures

OpenStack

OpenStack is designed to control large pools of computing, storage, and networking infrastructure resources to create a massively scalable, flexible cloud platform of private and **public IaaS** architectures. It consists of a series of interrelated open source software projects that provide both, core services, and a catalog of pluggable advanced cloud tooling. A minimal OpenStack installation could include: Keystone (Authentication Service), Glance (Image Service), Nova (Compute Service), Neutron (Network Service), Horizon (Dashboard Service), Cinder (Block Storage) and Swift (Object Storage).

OpenStack offers a full ecosystem of RESTful APIs to programmatically control these services, and likewise OpenNebula, some non-proprietary interfaces are also supported, including the **OCCI standard**. The graphical UI is organized in "tenancies", groups of users and resources that manage their own templates, images and networks. They even might have a separated pool of resources. In terms of virtualization, KVM, QEMU, Xen, LXC, VMware vSphere, Hyper-V are supported.

Part of the work presented here is underpinned by an **OpenStack tenancy** requested to the *Embassy Cloud*. Hosted by the European Bioinformatics Institute at the European Molecular Biology Laboratory (EMBL-EBI), the infrastructure is externally managed, unlike the above-mentioned infrastructures. The *Embassy Cloud* provides IaaS to multiple tenants (*e.g.* researchers, scientific projects and institutions, etc.) from the life science domain. EMBL-EBI features more than 4,000 cores and three petabytes of storage, with the interesting added value of being co-located with some of the most relevant European biomolecular databases (*i.e.* most of ELIXIR core data resources are physically hosted there, for example, ArrayExpress [9], PDB [5], etc.). So, even considering that the network cloud is logically isolated from EBI's LAN, the mobilization of such reference data could be efficiently handled. Storage is also based on a shared NFS file system (under tenants' petition), but some quota for object store is also offered (provided by OpenStack Cinder storage [202]). The following table details the specifications of our EMBASSY tenancy:

Name	Cores	Memory	Storage System	Network	Project
EMBASSY Tenancy (EMBL-EBI)	16Gb	64 Gb RAM	- 1Tb Object storage (CINDER) - NFS under request	- 2 floating public IPs - Internal VLAN	MuG

Table 3.2: Cluster details of MuG development's cloud

3.1.2.2 Provisioning Tools

Provisioning implies the deployment and configuration of virtual appliances in a cloud environment. CMPs are those above-presented permit serf-service provisioning, although there is an increasing number of tools automatizing the process (see *1.4.3.4* introduction's section). For the infrastructures presented here, provisioning is managed by one of the two tools hereafter described.

PMES

Programming Model Enactment Service (PMES) allows the remote deployment and job management of computing virtualized services in distributed infrastructures. PMES offers a standard REST interface to manage VM provisioning in a uniform and elastic manner, hiding in this way the heterogeneities of the underlining cloud stack. Indeed, it goes beyond and

controls the full-service lifecycle of **pre-emptible stateless VMs** that act as compute units. Sequentially, stages are the (i) deployment of the appliance, (ii) stage in data, (iii) execution invocation and monitoring of the application of interest, (iv) stage out data, and eventually (v) un-deployment of the VM(s). Cloud provisioning tasks are undergone via the cloud standard OCCI (below detailed at *3.2* Open standards), which mediates PMES interactions with the cloud infrastructure. Thus, OCCI compliant CMPs are a requirement. The communication of PMES with the transient VMs is enabled via SSH using a pair of public keys adequately contextualized at boot time. FTP is used for data transferences, which permits not only the dispatching of input and output files, but also the transference of a JAR file containing the actual application to be executed. As such, VMs can automatically be provisioned with the software at boot time.

Here, PMES is installed in two different cloud middlewares, OpenNebula and OpenStack. For each IaaS a stand-alone server is installed as a permanent virtual machine, accessible for outside the local VLAN. Two PMES components are part of that server:

- *PMES service:* deployed as a Tomcat 7 [203] application
- *PMES dashboard* service: NodeJS [204] application deployed using mp2 [205]

The installation requires the specification of the compute node resources available and importantly, the cloud rOCCI endpoint. Full documentation of the installation process can be found annexed at *8.6.1 PMES documentation*. PMES service exposes two APIs implementations, one SOAP-based (documented online [206]), used in the transPLANT infrastructure, the other enabled in REST and used in openVRE platforms (documentation annexed at *8.6.2*PMES documentation). APIs are compliant with two well-known open standards inherited from grid computing, BES and JSDL (further discussed below at *3.2*).

OneFlow

OneFlow is the key component of OpenNebula to build scalable services. It creates, controls and monitors services – composed by a single VM or a group of them – self-provisioning them in basis to (i) deployment dependencies between the group of VMs, (ii) elasticity policies. These policies establish the triggering parameters that make OneFlow scale up or down the number of running VMs for a given service. Interestingly, the CPU or MEM load of the VMs can be set as triggering parameters, thus, in front of load burst, OneFlow automatically makes available extra instances. OneFlow is installed as part of OneNebula, implemented as one of its add-ons.

3.1.2.3 Contextualization

A context is a small, usually human-readable, snippet that is used to apply a role to a VM. It contains the relevant information from the cloud manager to configure that role at the “first

boot". Hence, a context allows having a single VMI backing many different VM instances as long as the instances vary on settings able to be configured in the context. Considering we have a catalog of VMIs pre-packing bioinformatics tools, we can concurrently instantiate a VMI several times, each contextualized with the particular needs of that run. Typically, contextualization is useful for setting configurations not known until instantiation, like DNS, hostname, private information like user's SSH keys or personal data volumes, or software that changes frequently with discrete user-defined commands.

```
CONTEXT=[
  PUBLIC_IP="192.168.11.247",
  DNS="192.168.0.202",
  GATEWAY = "192.168.11.202",
  MAC = "02:00:c0:a8:0b:f7",
  MASK = "255.255.255.0",
  SEARCH_DOMAIN = "vm.mmb.pcb.ub.es",
  SSH_KEY="ssh-rsa AAAAB3NzaC1yc2EAAAA[...]Q/I0w== pmes@mmb",
  HOSTNAME = "mg-192.168.11.247",
  USER_DATA="
    #cloud-config
     packages: []
     mounts:
     - [vdc,none,swap,sw,0,0]
     runcmd:
     - echo 'Success' | wall"
]
```

Snippet 3.2 : Example of OpenNebula context information prepared to be consumed by cloud-init

CMPs, either OpenNebula or OpenStack, are responsible to **pass the context information** into the newly created VM. Each CMP enables different methods to do so. OpenNebula parametrize VMs by a iso 9660 CD-ROM image, which contains a shell script (context.sh) with custom variables defined on virtual machine start. *Snippet 3.2* is an example of that information transmitted by OpenNebula and ready to be consumed by the contextualization package installed at the VM. There are no fixed contextualization variables, and are in charge of integrating them and locally configuring the instance accordingly. OpenStack sends the information to a metadata server configured to serve up vendor data in a JSON object. Contextualization packages will reach it from there.

During the present work, two different contextualization systems are used to achieve very similar parameterization tasks, yet One-context is specific for instances deployed with OpenNebula, while Cloud-init supports several CMPs.

One-context

One-context is an add-on that provides contextualization packages for most Linux distributions (and, other Unix-like) on guest VMs or Linux containers running in the OpenNebula cloud. Packages are able to read context variables and prepare, accordingly, the networking in the running guest operating system, configure SSH keys, set passwords, run custom start scripts, and many others functionalities. Some of these functionalities are

OpenNebula-specific, like passing token-based credentials of OneGate, an OpenNebula module to enable authentication among VMs.

We used One-context as part of the transPLANT VMIs (see *4.2*. Results). We installed the corresponding DEB package (v. 5.7.0) into Ubuntu 14.04 LTS images, before importing them into the OpenNebula as QCOW2 images. On the other hand, OpenNebula templates need to be configured as *Snippet 3.2* exemplifies. The system is easy to be installed and configured, and USER_DATA is passed as opaque key-value metadata to the VM with no major restrictions. However, One-context can only be used for OpenNebula IaaS.

Cloud-init

Although few cloud open standards are still well-established, cloud-init is becoming the standard *de facto* for contextualization. Similarly to One-context, Cloud-init is used for actions on "first boot", pre- and post- networking, and USER_DATA section permits also some post-boot configuration by means of YAML scripts. However, the most remarkable feature of Cloud-init is its interoperability. It provides connectors ("datasources") for the most popular CMPs, including OpenNebula, OpenStack, Amazon EC2 or Azure.

As such, VMIs with Cloud-init are ready to be portable among different cloud providers with no need of further configurations or conversions, a relevant feature exploited at MuG infrastructures, as discussed at *4.3*. Results. Cloud-init (v. 0.7.5.) is installed in MuG VMIs and configured accordingly, choosing the correct "Datasource" and configuring some network details as Cloud-init demands them, like determined interface names or settings.

3.1.3 Job Managers

Presented infrastructures propose several strategies on how to distribute and manage jobs across the available resources. Following are described two different components involved in job management: COMPSs, a job orchestrator for distributed environments, and SGE, a classic batch queue system.

3.1.3.1 COMPSs

COMP Superscalar (COMPSs) is a **programming model** designed to ease the development of applications for distributed infrastructures (e.g. Clusters, Grids, and Clouds). To this end, COMPSs features a runtime system able to discover applications' parallelism of at execution time and dynamically distribute the tasks.

COMPSs hides parallelization complexities to developers so that programmers do not need to deal with the duties of **parallelization**, such as thread creation and synchronization, data distribution, messaging or fault tolerance. Instead, COMPSs is based on user's sequential

programming, which makes it appealing to users that either lack parallel programming expertise or are looking for better programmability. User's Java applications are directly supported by COMPSs runtime, while exists binding for other languages like Python, pyCOMPSs. Although programming is sequential, execution is parallel, since at run time COMPSs builds a workflow composed by the tasks of the application, which are connected through edges that denote data dependencies between them, and determined by annotations specifying the needs of each task. From this workflow, the COMPSs runtime is able to execute different application tasks at a time within a master-workers architecture (Figure 3.1). In this scenario, the user submits its application to the master node, which orchestrates the parallelization and launches the tasks in the available resources, distributing the input data and collecting the results.

PMES and COMPSs form a good tandem for enabling **virtual elastic compute clusters**. At execution time, when COMPSs tasks demand extra resources, the master schedule extra jobs in a queue system, or provisions extra VMs on the cloud, elastically demanding the resources according to the specific needs of the execution. PMES server provisions these VMs, which become COMPSs workers and remotely start user's execution.

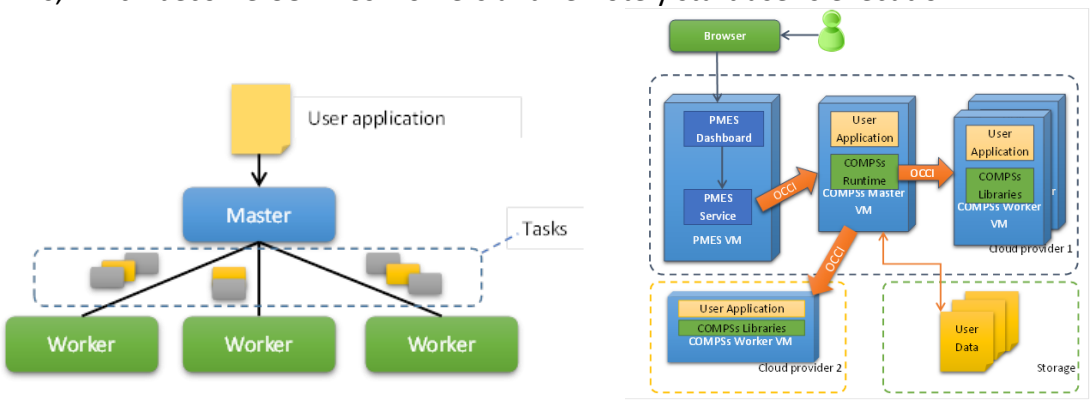


Figure 3.1: Elastic virtual clouds in COMPS and PMES
(a) Architecture of task distribution in COMPSs. (b) COMPSs deployment on cloud with PMES deployments

3.1.3.2 Sun Grid Engine (SGE)

Sun Grid Engine, originally developed by Sun Microsystems and now maintained by Oracle, is a resource management system for executing UNIX batch jobs (shell scripts) on a pool of cooperating resources, usually distributed and heterogeneous. SGE is typically used on **HPC or cluster** infrastructures, and is responsible for accepting, scheduling, dispatching, and managing the remote and distributed execution of large numbers of standalone, parallel or interactive user jobs. The workload is distributed among the resources based on the load situation of each machine and the resource requirements of the jobs.

A typical Grid Engine cluster consists of a master host and one or more execution hosts accessible over the network. Multiple shadow masters can also be configured as hot spares, which take over the role of the master when the original master host crashes. In cloud

computing, execution hosts correspond to VMs, thus, SGE acts as **local on-demand resource VM scheduler**. In this way, a virtual SGE cluster is built on top of the cloud, connected over the private cloud VLAN. SGE has an implementation in Nimbus and Amazon EC2 cloud on Hadoop nodes.

3.1.4 Authentication

3.1.4.1 Keycloak

Keycloak [207] is an open-source Identity Access Management (IAM) server developed and maintained by the RedHat Community**.** It offers a broad set of authentication and authorization services to make easy to secure applications and services. Most relevant services are:

- centralized authentication and authorization using a local MySQL or LDAP
- centralized user management for the multi-tenant environment – (multiple realm user spaces)
- two-factor authentication using a one-time password
- brokering identity with OpenID Connect (OIDC) or SAML Identity Providers (e.g. Elixir AAI).
- Single-Sign-On using OIDC

OIDC is built on top of OAuth2, a set of token-based authentication flows widely used for REST services (implicit flow) and Web applications (authorization code flow). In short, at the implicit flow, the authentication server, here Keycloak, issues an access token to the user, which is submitted as part of the request header to the RESTful server. The latter validates the token against the Keycloak to ensure user's identify. If it results validated, the REST server returns the appropriate response to the user. In the Authorization Code flow, the OAuth2 dialog is slightly different, as first the web application needs to get a code to interact with the Keycloak server on behalf of the user.

For those applications or services serving protected resources, Keycloak supports fine-grained authorization policies and is able to combine different access control mechanisms such as attribute-based, role-based and group-based.

3.1.4.2 LDAP

One of the *de facto* standards for storing and querying authentication and authorization data is the Lightweight Directory Access Protocol (LDAP) — it can conveniently store passwords and handle authentication data. LDAP is a lightweight client-server authentication protocol used to access centrally stored information over any location in the

network. It is mostly used as the data backend for centralized authentication systems to consolidate information of the entire organization into a central repository, which additionally, supports TLS certificates for data protection.

The solution is able to integrate and centralize user account management of Linux POSIX or Samba accounts. In transPLANT, the LDAP open source OpenLDAP [208] implementation is used to centralize authentication for transPLANT services (further discussed in Results section at *4.2.2.4* Authentication).

3.1.5 Summary of web applications

The following table summarizes all web applications implemented as part of this work and will be discussed in context with the infrastructure they are designed for.

	Project	Frontend	Application	Dependencies	Data tier
IMID-clinica	*4.1 Data management's infrastructure for IMIDs' research*	HTML, PHP, JavaScript	PHP (v5.2)	- Egroupware 5.2	MySQL
IMID-Longitudinal		HTML, PHP, JavaScript	PHP (v5.6)		MySQL
DataManager	*4.2 transPLANT: trans-national Infrastructure for Plant Genomic Science*	HTML, PHP, JavaScript	PHP (v5.6)		File system/FTP
MuGVRE	*4.3 MuG: Multiscale Complex Genomics VRE*	HTML, PHP, JavaScript	PHP (v5.6)	- League/oauth2-client 2.4 - Justinrainbow/json-schema 5.2 - Multiscalegenomics/mg-rest-dm - Mongo PHP driver 1.3.x	Mongodb
openVRE	*4.4 Open Virtual Research Environment*	HTML, PHP, JavaScript	PHP (v7.2)	- Auth0/auth0-php 5.0 - Justinrainbow/json-schema 5.2 - League/oauth2-google 2.0 - slim/slim 3.0 - Mongo PHP driver 1.3.x	Mongodb
OEB-VRE		HTML, PHP, JavaScript	PHP (v7.2)		Mongodb

Table 3.3: List of developed web applications and its dependencies

3.2 Open standards

Standardization of data formats, protocol messages, and interfaces is an important aspect of execution environment interoperability. For task-based frameworks on distributed environments, the following three proposed standards are relevant.

3.2.1 Open Cloud Computing Interface (OCCI)

The Open Grid Forum (OGF) has proposed the Open Cloud Computing Interface (OCCI) [99] as an open standard defining a RESTful API for **managing cloud appliances**. It has been one of the first standards in the cloud ecosystem and covers the basic management tasks for Infrastructure as a Service (IaaS) providers by means of a set interoperable REST endpoints that remove any kind of vendor lock-in effect in the infrastructure. In summary, supported operations consist on managing the compute instances (create, delete, start, etc.), network basics (attach/detach network) and block storage (create, delete, attach/detach) [209].

Currently, some OCCI implementations already exist for several cloud vendors like OpenStack, OpenNebula, Jcloud, Eucalyptus or EMOTIVE. However, OCCI is implemented in the form of general-purpose frameworks that can be extended with several back-ends, for instance, OCCI-Libvirt, which implements the interface on top of the virtualization toolkit, o pyOCNI, which implements OCCI with a networking extension. The most popular implementation is **rOCCI** [210]. Written in Ruby, it delivers an rOCCI-server and a r-OCCI-cli implementation. The rOCCI-server component stands as a standalone server that proxies the requests to the underlying cloud management framework (Figure 3.2). Indeed, some OCCI-enabled stacks like OpenNebula and OpenStack took such approach. On the other hand, rOCCI-cli is the client component that makes possible to interact with any OCCI-enabled framework. PMES makes use of it.

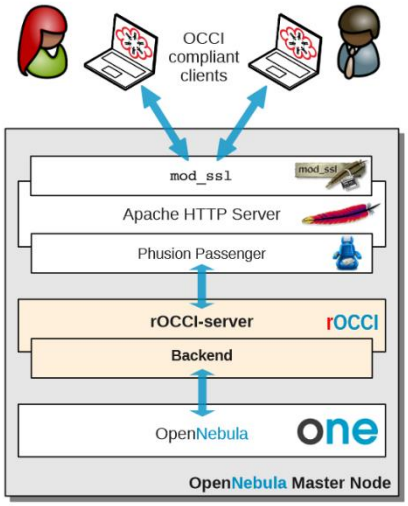


Figure 3.2 r-OCCI server in a typical setup with OpenNebula. PMES server is a OCCI client

The following snippet is a simple example of how to create a compute instance using the rOCCI client. It indicates the cloud OCCI endpoint and user's X509 credentials, together with two templates containing the details of the instance:

- *Image template* (os_tpl): corresponds to VM template as registered in the CMP (see above *3.1.2.1* OpenNebula and OpenStack).

- *Resource template* (resource_tpl): corresponds to a simpler template indicating only (i) cores, (ii) memory and (iii) local disk (*e.g.* "small" template - 1 core and 2 GB RAM). OCCI server maintains the registry of these templates. If the image template contains any of these parameters, they are ignored.
-

```
occi --endpoint https://extcloud05.ebi.ac.uk:8787/occi1.1
     --action create
     --resource compute
     --mixin os_tpl #54_mugTool
     --mixin resource_tpl #small
     --context user_data= file://$PWD/tmpEBI.json
     --auth x509
     --user-cred /tmp/pmes.x509

> RESPONSE
> https://extcloud05.ebi.ac.uk:8787/occi1.1/compute/2048be92-dce8-1fd7babcd1ba .
```

Snippet 3.3 Virtual machine creation using rOCCI client

3.2.2 Basic Execution Service (BES)

The OGF has purposed the Basic Execution Service (BES) [59] specification as the standard interface for **job management** across different distributed environments. This specification is defined as a Web Service implementation in terms of a WSDL document for creating, monitoring and managing computational jobs, called activities. There is already some solid implementation of BES, all belonging to the Grid computing environment like Globus Toolkit [52] or UNICORE [53], and EGI is using a RESTful implementation of it [211].

Activities are represented using JDSL (described in the following section) and are identified using Web Services address endpoint references. When a client submits an activity to the BES server, an endpoint reference is obtained in result, which is later used in invocations to refer to that particular activity. BES also defines an extensible state model defining the states that an activity can pass through during execution: pending, running, terminated, failed, finished. Briefly, the basic operations considered are:

Operation	**Description**
Create activity	Creates and submits an activity to the infrastructure.
Terminate activity	Terminates a previously created activity.
Get activity status	Retrieves the status of the activity (*e.g.* running, failed, etc.)
Get activity documents	Gets the activity documents of an activity giving the: JSDLs [58], jobs status, execution progress, elapsed time and error messages.

Table 3.4: Operation descriptions of PMES web services

3.2.3 Job Submission Description Language (JSDL)

Job Submission Description Language (JSDL) [58] is an extensible XML specification from the Global Grid Forum (GGF) for the **description of simple tasks** to non-interactive computer execution systems. Currently at version 1.0, it is a verbose, yet straightforward language for defining application's requirements and the submission preferences. It includes aspects of a job as:

- Job name, description
- Resource requirements that computers must have to be eligible for scheduling, such as total RAM available, total swap available, CPU clock speed, number of CPUs, Operating System, etc.
- Execution limits, such as the maximum amount of CPU time, wallclock time, or memory that can be consumed.
- File staging, or the transferring of files before or after execution.
- Command to execute, including its command-line arguments, environment variables to define, stdin/stdout/stderr redirection, etc.

The vocabulary and normative XML schema in use include this data, as exemplified in the following snippet:

```
<JobDefinition
 xmlns="http://schemas.ggf.org/jsdl/2005/11/jsdl"
 xmlns:posix="http://schemas.ggf.org/jsdl/2005/11/jsdl-posix">
  <JobDescription>
    <JobIdentification>
      <JobName>Test job</JobName>
    </JobIdentification>
    <Application>
      <posix:POSIXApplication>
        <posix:Executable>/path/to/executable</posix:Executable>
        <posix:Argument>arg1</posix:Argument>
        <posix:Argument>arg2</posix:Argument>
        <posix:Output>stdout.txt</posix:Output>
        <posix:Error>stderr.txt</posix:Error>
      </posix:POSIXApplication>
    </Application>
  </JobDescription>
   <DataStaging>
      <FileName>input_file.txt</FileName>
      <CreationFlag>overwrite</CreationFlag>
      <Source>
        <URI>file:///local/path/to/input_file.txt</URI>
      </Source>
    </DataStaging>
    <DataStaging>
      <FileName>output_file1.txt</FileName>
      <Target>
        <URI>file:///group_workspaces/jasmin/some/path/to/save/outputs/</URI>
      </Target>
    </DataStaging>
</JobDefinition>
```

Snippet 3.4 Minimal JSDL file

JSDL is a well-established standard, particularly in grid computing, and systems like UNICORE, EMOTIVE cloud or IBM Tivoli support it. With these middlewares, interoperability with job schedulers and distributed resource managers.

3.3 Use Cases

A couple of use cases are presented here to illustrate the functionalities of our infrastructures. Hereafter, the implementation details of the same are described.

3.3.1 Use case: plant genome annotation pipeline in MAKER

transPlant infrastructure is populated with analysis tools and pipelines of interest for the plant genomics' community. MAKER is one of them. Pre-packed in a VM, the following table describe the relevant software included as part of the pipeline.

Software	Description
Maker (v2.28) [212]	Genome annotation pipeline. Their tools identify repeats, aligns ESTs and proteins to a genome, produces ab initio gene predictions, and automatically synthesizes these data into gene annotations having evidence-based quality indices
Exonerate (v2.2.0) [213]	Splice-site aware alignment algorithm to realign, or polish, sequences following filtering and clustering
Snap (v2006-07-28) [214]	Ab initio gene prediction modelling
Augustus (v2.5.5) [215]	Gene-prediction algorithm that can be used to produce either ab initio or evidence-based predictions when provided with an external GFF3 file of EST and protein alignment data
Blast (v2.2.28+) [216]	Algorithm for comparing primary biological sequence information, such as the amino-acid sequences of proteins or the nucleotides of DNA and/or RNA sequences
RepeatMasker (v1.295) [217]	Screens DNA sequences for interspersed repeats and low complexity DNA sequences
Tandem Repeats Finder (4.07b) [218]	Finding tandem repeats which work without the need to specify either the pattern or pattern size

Table 3.5: MAKER software dependencies

Additionally, some test data is provided along with the implementation. It can be found at the transPLANT's documentation page[1] together with other tool's documentation.

- Maker configuration files:
  - maker_exe.ctl - contains the path information for the underlying executables.

[1] http://transplantdb.bsc.es/documents/samples/

  - maker_bopt.ctl - contains filtering statistics for BLAST and Exonerate
  - maker_opt.ctl - contains all other information for MAKER, including the location of the input genome file.
- Files referred into the configuration files:
  - dpp_contig.fasta: main pipeline input containing DNA contigs in FASTA format
  - dpp_est.fasta: expressed sequence tags (*ESTs*) in FASTA format

Full use case's details are found at *4.2.4* Use case: plant genome annotation pipeline in MAKER.

3.3.2 Use case: Nucleosome Dynamics

A second use case implemented as part of the MuG infrastructure is Nucleosome Dynamics [219], one of the tools populating the platform. For demonstrating how the method works in several genomic scenarios, three different MNAse-seq datasets extracted from the literature are used. These are made available at MuG platform under "Example Datasets".

- Cell-cycle dataset: MNase-seq data for *S. cerevisiae* cells synchronized in G1 and S phase as described at [220]. Raw data available under accession number SAMEA2698380.
- Yeast metabolic dataset: MNase-seq data for two time points of the yeast metabolic cycle (YMC): T9 and T12, as described at [221]. Raw data downloaded from GEO under accession number GSE77631.
- Nutrients change dataset: MNase-seq data for *S. cerevisiae* grown in different media (YPD, Gal and EtOH) [222]. Data aligned to sacCer1 downloaded from GEO using accession numbers GSM351492, GSM351493, and GSM351494.

Full use case's details are found at *4.3.4* Use case: Nucleosome Dynamics

4. Results

This chapter presents in four separated sections the infrastructures developed along with the thesis under the frame of four scientific projects. Projects are shortly summarized to provide a context and motivation for the designed platforms. Architectures and components are described, stating for each case the candidate's leading role in the design and/or implementation process, although, all the presented infrastructures are the result of group cooperation and team working.

4.1 Data management's infrastructure for IMIDs' research

IMIDs stands for Immune-mediated inflammatory diseases. The work presented in the present section is focused on the electronic data capture (EDC) systems developed for two biomedical research studies of IMID patients: IMID-clinica and IMID-Longitudinal.

The doctoral candidate has participated at the later stages of IMID-clinica project, focused on the extension of new features on the, at the moment, newly implanted system, as well as his administration and maintenance tasks to the day. In IMID-Longitudinal study, the candidate has been involved in the design of the clinical data acquisition system, the database modeling and the implementation of the EDC.

4.1.1 Context

IMIDs are a group of **inflammatory chronic diseases** of unknown etymology highly prevalent in the general population (5-6%) and with a great impact on patient's quality of life. Thus, IMIDs treatment incurs in elevated socio-sanitary expenses. Yet, they are multifactorial (environmental and heritable predisposition) diseases and with a very heterogeneous molecular pathogenesis: the most representative pathologies are systemic lupus erythematosus (LES), psoriasis (Ps), psoriatic arthritis (APs), arthritis rheumatoid (AR), ulcerative colitis (UC), Chron's disease (EC). In the last few years, with the increasing availability of genomic data, Genome-Wide Association Studies (GWAS) have led to the identification of patient's subgroups with specific molecular patterns and expression profiles. Moreover, association studies have shown predictive power over treatment's responses, which more and more are biological therapies specifically designed for molecular targets that seeks the total disease's remission.

Such objectives were pursuit by the two clinical studies presented in this work and led by the group of Immune-Mediated Diseases & Innovative Therapies at the Vall d'Hebron Institut de Recerca (VHIR):

- IMID-clinica: Development of a kit diagnostic for IMID (PSE -10000-2006-6/PSE-10000-2008-9).
- IMID-longitudinal: IMID biomarkers identification and new therapies (INNPACTO: IPT-010000-2010-36.)

Both projects (sequential in time) have proposed the **data acquisition** and curation of clinical, epidemiologic and serologic data in a multi-center approach across Spain. Our team at BSC has been in charge of developing and maintaining the clinical data infrastructure to support data gathering, storage, and validation. Corresponding biological samples are extracted to IMID patients by the Spanish IMID-Biobank, for later genomic analysis (mainly array-based genotyping). Such analyses, based on GWAS studies, have produced a number of fruitful results published on several research publications (see annex *0*).

4.1.1.1 Motivation

In genetic epidemiological research, the sample size is critical to the success of GWAS when detecting causal genes of common and complex multifactorial diseases like IMIDs. GWAS aims to identify the DNA allele variants (single nucleotide polymorphisms (SNPs)) that contribute to the disease's risk, even those with low penetrance, shown abundantly in IMIDs. Yet, sufficient statistical power is to be achieved in order to obtain significant signals. The collaboration and coordination among multiple health centers and hospitals have been a commonly chosen strategy to overcome the sample size effect. However, **cross-organizational data integration** raises up standardization, validation and curation challenges that need to be accurately addressed for an agreed clinical data management plan among the participant centers to ensure true progress monitoring and high-quality and reliable research data. That is particularly relevant when potential data providers may not be familiar with data curation best practices, as they mainly belong to the assistance services of the participant health centers and hospitals.

he diverse nature of the different IMIDs adds an extra layer of **data heterogeneity**, requiring not only the integration of different medical sources (*e.g.* images, raw medical record files) but also a set of very diverse clinical variables difficult to be compared among them, with different biomarkers, dictionaries, etc. However, one of the requirements of the project was precisely the integration of such diverse data types in a single platform.

Furthermore, the implemented strategy should allow a rapid and **flexible** edition of case report forms (CRFs), as clinical variables are in constant change and reedition – new therapies, new serology tests, new formats, etc. The system should also enable easy

integration of completely new CRFs additively. Furthermore, the nature and purpose of data collection in clinical research can easily evolve during the analysis phase, and the data acquisition process can be costly and slow, so it is important to maximize the amount of collected data, for instance, by setting quality controls with protocols that do not interfere or prevent in the registration process.

Safeguarding **data privacy** in another critical requirement for infrastructures dealing with biomedical data, still, maximizing research results to the possible extent. IMID-clinica and IMID-Longitudinal are cohort studies that translate data from the healthcare system to the research environment, so the framework has to adopt the Ethical Legal Social Implications (ELSI) and contractual aspects described in the data management plan.

4.1.2 Data Management

Traditionally, the standard method of data capture in clinical and translational projects includes two steps. In the first step, data is collected using paper-based **case report forms (CRFs).** In the second step, data is manually entered into electronic databases from paper-based CRFs for the purpose of data analysis. This standard paper-based data capture method is inefficient in terms of data accuracy and timeliness, so as an alternative, the electronic data capture (EDC) method propose to directly enter into electronic databases using electronic CRFs. CRF define the model of data and other information collected on each subject participating in the study. These are paper-based and agreed among the IMID consortium medical committees. They include epidemiological data (*e.g.* lifestyle variables), clinical data (*e.g.* family history) and serological data (*e.g.* antigenic serology).

The **clinical data management plan** (DMP) of the IMID's consortium covers from the Case Report Form (CRF) design phase to the clinical database locking after data assessment. Certification authorities granted it ISO 9001:2008 and ISO:2012. Both studies, IMID-clinica and IMID-Longitudinal, follow similar governance's plans (Figure 4.1)., primarily differentiated on the actors entrusted to perform some of the defined tasks. The online system developed here is designed to support clinicians, data managers and epidemiologic researchers in the different steps defined at IMID's DMP.

Our team at BSC was in charge of designing and providing the online systems to efficiently support the clinical data acquisition process from the different healthcare centers, *i.e.* developing the electronic data capture (EDC) software and the clinical database. The first task is the creation of the **electronic CRF** translated from the paper-based model to become part of the database. EDC's administrator and maintainer are in charge of registering the clinical variables and preparing the corresponding web forms.

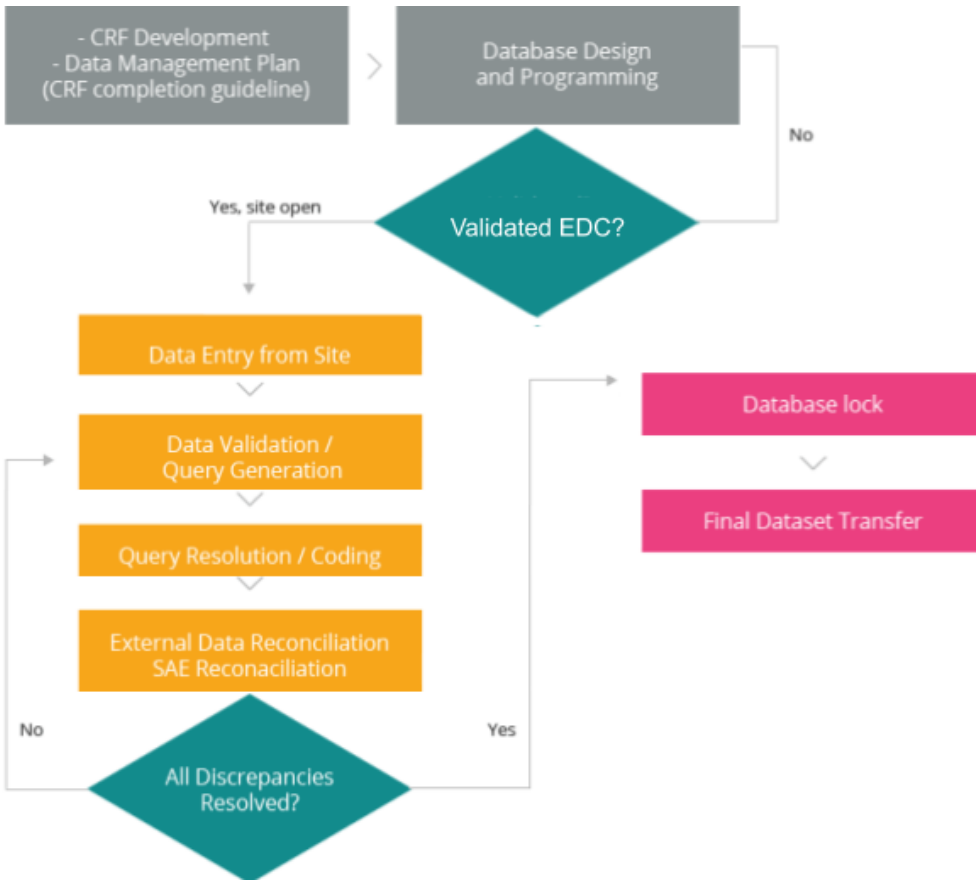


Figure 4.1: General process data flow in IMIDs project

The actual **data entry** of patients' information uses such online web forms, which include validating tools that enable a first automatic quality control to check for data completeness and even data coherence. Yet, the responsibility of such a task falls on different actors when comparing IMID-clinica and IMID-Longitudinal data governance. In the IMID-clinica study, data providers, *i.e.* participating healthcare centers, feed the data into the online system, and even two roles are defined in each centre ("centre coordinator" and "user") to further control data registry. In IMID-Longitudinal, the project designates a centralized group of data curators who gather filled-in CRFs in paper from the participating centres and perform the data registration. Such distinction weights differently the relevance of some EDCs functionalities like data validation assistance. Anyhow, in order to ensure the high-quality of the resulting clinical data, a clinical data manager figure is made indispensable. He/she audits the overall data, case by case, and ensures conflicts resolution, data discrepancy. EDCs also recognize this role and propose a set of tools to aid the coordinator in their monitoring tasks.

Finally, to complete data's cycle, EDCs provides **extraction data** options, at both, case and database level. After obtaining the consent from all the parts, the database is locked, and clean data can be extracted for statistical analysis.

4.1.3 IMID-clinica

IMID-clinica[2] is the web-based EDC system designed to store multi-center clinical data from observational **cohort studies** with cross-sectional data (analysis at a specific point in time). The instance in production hosts the clinical and epidemiological data, and since 2010, when the first prototype was released, has been collecting data from of up to 90 centers of the Spanish National Healthcare System. The data, cross-linked with genotyped samples from IMID biobank, is consumed by the IMID consortium.

IMIDs	Systemic lupus erythematosus, psoriasis, psoriatic arthritis, arthritis rheumatoid, ulcerative cholitis, Chron's disease
Cases	12,173 patients (+ 2,130 controls)
Centers	82 (whole list can be found in *8.1* Participant centers of IMID's clinical studies)

Table 4.1: Summary of IMID-clinica case study

User's perspective

IMID-clinica offers a friendly and intuitive **user interface** (UI) for the CRF data collection and validation from centers and hospitals. IMID-clinica is implemented on top of the Egroupware [223][223][223][222][221][220][219] framework, a collaborative software that provides out-of-the-box plugins (*e.g.* contact list applications, shared calendars, wikis, etc) and allows the integration of additional tools. IMID-clinica PHP web application was integrated into the framework.

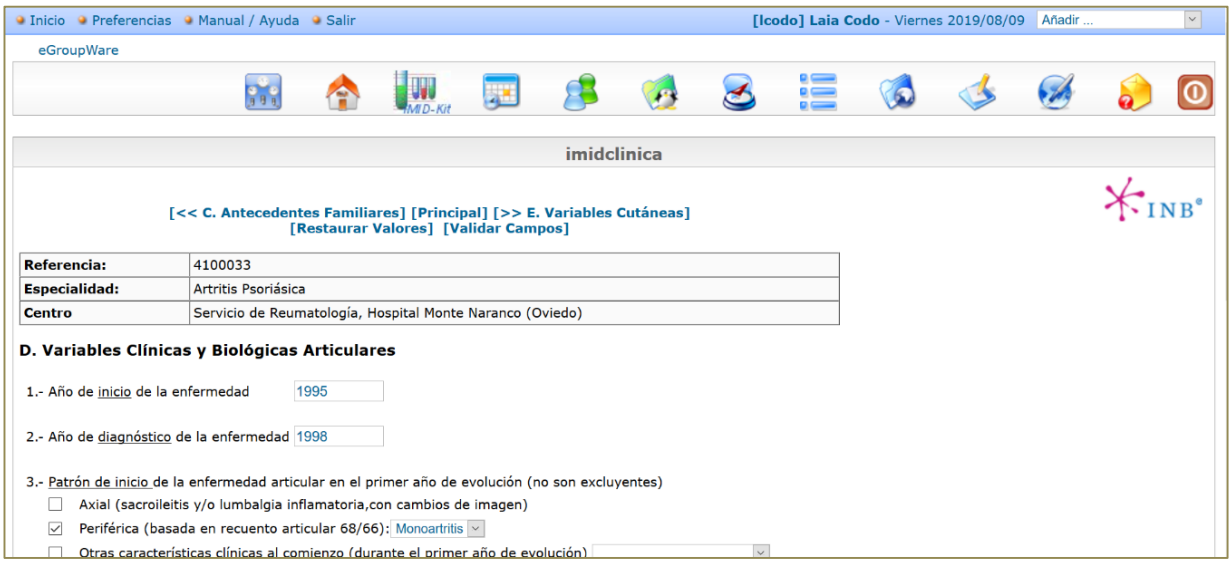


Figure 4.2: Snapshot of IMID-clinica online form

[2] https://IMIDkitclinica.bsc.es *(*) Due to data privacy requirements access is restricted to authorized users*

The application is in Spanish and structured hierarchically, along with the data, as *Pathology > Centre > Clinical Cases > Clinical Data*. From user's perspective, the most relevant views are those displaying the **electronical CRF** of each pathology study (Figure 4.2). These are accessible as online HTML forms when registering a new IMID clinical case on the system. After clinical fields are filled in, an automatic validation at type level (*e.g.* "date" format type or integer/string type) takes place, and it prevents data storage unless the indicated fields are fixed.

The EDC system provides also tools and views designed to support **data manager's tasks**. Tables at case, centre or pathology level are very useful from monitoring. In them, metrics for data monitoring are displayed: statistics on the completeness and unresolved issues per case; plots with number of completed cases per centre or pathology; etc. Moreover, the locking case system (data modification disabled), only permitted to privileged users, allows to have a fine grain control of verified and audited data. Finally, a powerful querying system enables an advanced filtering which allows to set search criteria on every clinical field of the database and even combine them with logical operators.

And last but not least, another relevant aspect of the application is the high degree of adaptability in front of **new CRFs**. When a study for a new pathology is to be included in the system, the administration only needs to prepare a set of HTML templates containing all the new clinical fields as they are to be displayed on the online web form. Templates are pure HTML (including JavaScript snippets when necessary) except for specific tags that are to be placed where the HTML input fields (*e.g.* "text box", "drop down selector") are expected. Such tags should indicate the name of the variable, the type, and the default value (if any). The application features an importing tool to load these HTML templates and creates the corresponding entries on the database.

These were the initially implemented core features of IMID-clinica, which along the project's lifetime have been continuously amended, improved or automatized, thanks to continuous feedback of clinicians and data managers. Additional functionalities have been implemented, for instance, a more exhaustive data **quality control** (QC) system for data discrepancy. Such process is also applied on each "Save" request, but unlike the type-based validation, it allows data registration. To notify discrepancies, the system raises persistent queries (*i.e.* errors) displayed with comprehensive alert messages (Figure 4.3 shows as alert's example) that are accountable and monitored across the platform not only by the user, but also the clinical data manager coordinator. The system reports on non-coherent or uncompleted clinical variables, that are the main data quality affectations. These are detected thanks to an advanced set of dependency and exclusion rules, listed in Table 4.2. These rules are individually set up for each field (or group of fields) when first time annotating the CRF on the system (*e.g.* field "α-TNF" cannot be ">1" if field "anti-TNF therapy" is TRUE). The resolution of such queries is mainly handled outside the system,

establishing a direct communication (*e.g.* email) between the data manager and hospital's data providers. Indeed, the application is able to export query lists (among other data) as DOC or PDF files.

Id.	Rule	Message
1	Variable (or group of variables) is compulsory	- Field [VAR_NAME] is required - The following fields are required: [VAR_NAME], […]
2	Variable(s) is compulsory if its Boolean master variable is evaluated as true. Reverse applicable.	- Field [VAR_NAME] is required in [MASTER_NAME] - Field [VAR_NAME] cannot without [MASTER_NAME]
3	At least one variable in the group is compulsory if its master variable is defined	- Section [MASTER_NAME] requires more specifications
4	Special rule applying to a particular group of interconnected variables conveniently defined together	- At therapy [VAR_NAME] table [VAR_A] requires to be defined while [VAR_B] or [VAR_C] are set
7	Variable is required if a logical expression is evaluated as TRUE. Reverse applicable	- Field [VAR_NAME] is compulsory because [VAR_A] has a value greater than [VAR_C]

Table 4.2 Rules applied for the Quality Control data

4.1.4 IMID-Longitudinal

IMID-Longitudinal is a complete web-based EDC system designed for observational cohort studies with follow-up data – those longitudinal analyses that collects data on the same individual at different times along treatment. The system was first released at 2012, and since then, it is being used to collect data at the VHIR. The inclusion of time-based data itemsis where the analysis model most differs from IMID-clinica's. Data from the same six IMIDs is collected at **different disease stages**: first visit (S0), week 12 after first visit (S12), disease remission (RE), treatment withdraw (SRTB), etc. (Table 4.3).

Differences aside, most of **core functionalities of IMID-clinica's** system are also in used and adapted for IMID-Longitudinal study, like data QC toolkit, or template's strategy for CRF integration that avoid a study-specific database design. Application's model is also structured hierarchically, here as: *Study > Pathology > Health Centre > Disease Stage > Clinical Cases > Donations > Clinical Data*. A "Case" refers to a patient's data, while a "Donation" to a patient's data on a given time checkpoint.

The data collection period is still active, yet, already 103 centres of the Spanish National Healthcare System are engaged. The web application is only accessible inside VHIR lab premises, where data entry is centralized. Interestingly, the distributed approach used in IMID-clinica was changed to an centralized one in this second project. Yet, the testing prototype, with demonstration data, can be visited[3].

[3] https://inb.bsc.es/Innpacto/

IMIDs	Systemic lupus erythematosus, psoriasis, psoriatic arthritis, arthritis rheumatoid, ulcerative cholitis, Chron's disease
Study Stages	- First Visit (S0), Week 12 (S12), Week 24 (S24), Week 104 (S104), … - Disease relapse (SE), remission (SRM), 4 weeks after remission (SRM4), therapy Withdraw (SRTB), Reactivation (SF), …
Cases	983 patients. Still collecting data
Centers	93 (listed in *8.1* Participant centers of IMID's clinical studies)

Table 4.3: Summary of IMID-longitudinal case study

User's perspective

The EDC for IMID-longitudinal features a completely new **user interface** (UI). It is implemented as a server-side PHP web application that enables the data entry from the different centers and hospitals, while keeping in view the flexibility and control restrictions stated by project's data management. It permits an agile navigation through the different data structures.

IMID-Longitudinal's main page offers a menu for either inserting a new donation or navigating among registered data (*e.g.* other donations, cases, centers, stages, pathologies). When **creating a new donation**, clinician already knows its identifier (which is assigned at consortium's level), and the insertion creates an empty record in "on tracking" mode, for the selected clinical stage (*e.g.* first visit – S0). The CRF is displayed online, segmented in sections that are gradually stored while navigating from one division to the next (Figure 4.3). CRF donations become locked when the data coordinator sets them as monitored (see Figure 4.4*, shown* them as green balls). If all the donations of a clinical case are positively monitored, the full case can be locked (mode "closed"), although other modes are contemplated to have a better control of case's statues (*e.g.*" discontinued case", "exitus case", etc.).

When displaying the general **view of a patient's case** and its donations, only those clinical stages relevant for the given case are shown as available on the application. Such availability depends on each case and pathology. Thus, each pathology evaluates their patients at different time checkpoints (e.g. Week 12 (S12)). Besides, during these routine checkpoints, clinicians might discover a clinical change (e.g. disease's remission RE) or decide on another therapy (*i.e.* therapy withdraw (SRTB)). In such cases, routine CRFs are disabled in favor to the specific CRFs for these clinical stages, which might have exclusive or dependence relationships (e.g. a case cannot have a "Reactivation" stage prior a "Remission" stage). Figure 4.4 shows case donations as a collection of colored balls coding availability and monitoring state.

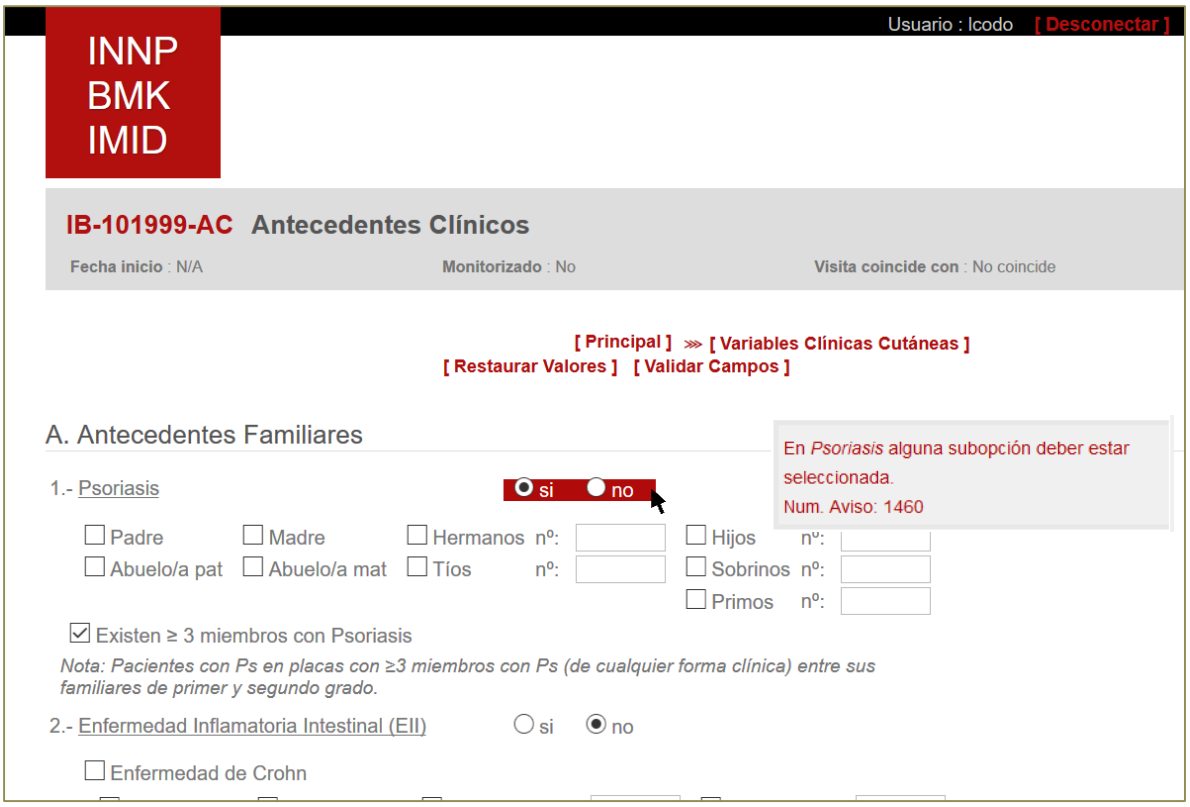


Figure 4.3: Snapshot of IMID-Longitudinal online form with un error.
The data belong to donation code IB-101999-AC. The alert reflect that at least one option in section A.1 is to be selected.

Data validation tools were extended and further improved respect to IMID-clinica in order to offer a complete **query's resolution pipeline**. Now, the validation based on data types includes allowed ranges for numeric and date values, while strings can be evaluated against a simple pattern. Truly missing values (i.e not evaluable) are also now tagged to differentiate them from not completed fields. And the platform-wide alerts emitted when a query (i.e. error) is detected (Figure 4.3), are now fully registered to build provenance on issue's resolution (Figure 4.4*)*. Queries might be "closed" when resolved, or "re-opened" if needed, and data coordinators can "send" them to the health center. Besides, data managers can also create queries not associated with faulty clinical fields, but with custom messages that enable a dialog channel on the system with data providers.

The EDC system provides also several **overviews** and metrics to control the data status at different levels (*e.g.* donations positively monitored for a certain case, or overall completed cases per center). Such views are controlled according to the user's role. There are also views for comorbidity cases, as it is a frequent scenario among IMID pathologies. Finally, the integration of new CRFs follows the same strategy applied in IMID-clinica as described above (Section).

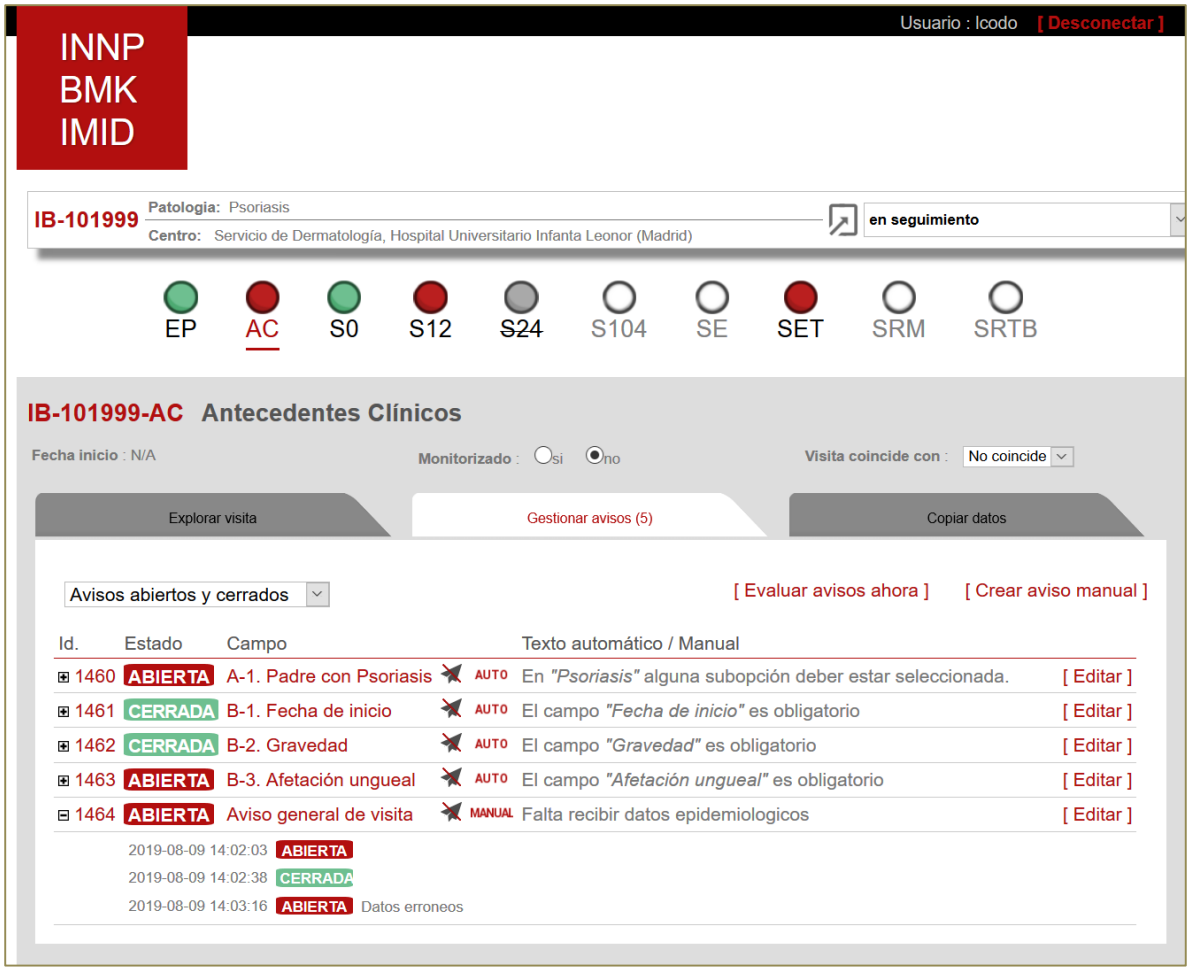


Figure 4.4. Snapshot of query issue registry in IMID-Longitudinal for a donation. The donation (coded IB-101999-AC) has 5 queries, 3 of them unsolved.

Database and data model

The project aimed to collect data from a large number of patients, during a long-term period, and of great diversity, as data comes from six different diseases with a large number of fields of different nature. Therefore, the project was articulated towards the generation of a **generic phenotypic database**, based on the relational database technology (MySQL). Opposite to the usual strategy in relational databases, the nature of the information stored in it is not part neither of the database's structure nor the associated web application, allowing the storage of any clinical questionnaire without changes in the database structure.

The database design implemented in IMID-Longitudinal is depicted in Figure 4.5. As indicated, flexibility and adaptability in front of CRF changes is achieved by storing data definitions as part of the database content. The **storage of the CRF** for a pathology study corresponds mainly to two tables: "Defdato" and "Formulario". The first contains all the defining information about clinical fields (*e.g.* field named "α -TNF" is of type "float" and belongs to form's section "A" of disease "Artritis Reumatoide"), including dictionary's

references if the field can only take a fixed list of values. The latter, "Formulario", simply groups the fields in sections (*e.g.* form section "A- Family History" belong to pathology "Artritis Reumatoide"), to match the structure of paper clinical questionnaires. This overall strategy allows to easily integrate new CRFs (*i.e.* new pathologies) into the system and to carry out the posterior analyses quite independently from the data structure.

The overall study structure follows the **study data model**: Study > Pathology > Clinical Cases > Disease Stage > Clinical Data, which dictates the main data structures in the system. "Casos" retains the identification of the stored data for a certain patient, as well as the confirmation of its status (*e.g.* under track, complete) and external database identifiers (*e.g.* IMID-clinica). Each clinical case may contain one or more "Donaciones" (sample donations) extracted on a certain time point (disease stage) and associated to a BioBank's sample. Finally, the "DatosClinicos" table hosts the actual clinical values introduced by the participating centers as simple key-value triples (*e.g.* in donation "00000", the variable "anti-TNF" takes the "0.568" value). Such data is going to be the raw data for the statistical analyses.

Additionally, the EDC relies on the database for other data or **operational services**. Discrepancy data management is based on the "oblig_field" table, where internal rulings (listed in Table 4.2) express fields dependencies and collisions. User's administration is centred on "users" table, and keeps the registration ownership, along with the access control derived on user's roles: "user", "centre coordinator", "coordinator", "administrator". Finally, the querying system also employs the database for storing user's search parameters and results.

Security and Authentication

Authentication and authorization are internally managed at the application level, based on the data stored in the MySQL database. Different data controllers with different permissions on the data are set up, so that users are granted rights to either access, edit or block groups of data, i.e. at pathology or center level. As such, the following roles are created: administrator, data coordinator, center coordinator and plain user.

The **security** in the IMID-longitudinal system comes with the design, as given the sensitive nature of the data, it is fragmented into several databases belonging to IMIDs' consortium. IMID-longitudinal stores already anonymized epidemiological data. Only IMID-clinica identifiers and a correspondence with the BioBank identifiers are retained. Participant health centers are the ones that keep an internal registry of the relationship between the IMID-clinica identifier and the actual patient's identity. Permissions to pursue the investigative actions of the IMID consortium are assured through a legal contractual consent signed by the patient. The legal document constraint not only the use of the donations and data, but also its accessibility and sharing.

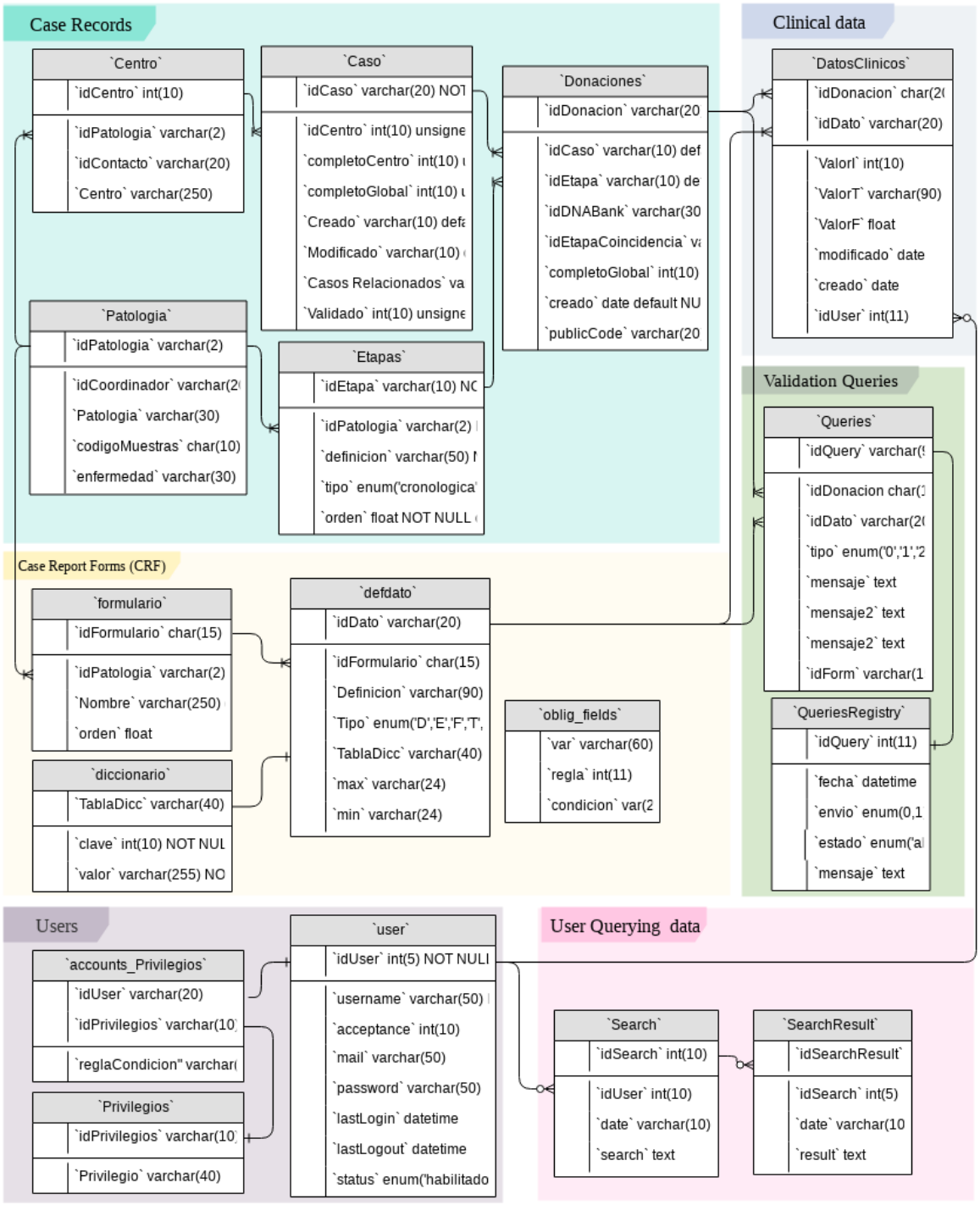


Figure 4.5: IMID-Longitudinal database design

4.2 transPLANT: trans-national Infrastructure for Plant Genomic Science

This section describes the design and development of the research cloud we have built in the context of the transPLANT project, oriented to scientific computing and data analysis.

Doctoral candidate has directly participated in the assembly, configuration and maintenance of the different infrastructure components described in the Cloud Architecture section, also developing complementary tools to ensure the completion of user's data life-cycle. These involved the development of an FTP service, a web-based access for the same, and the configuration of an openLDAP/PAM authentication system. Additionally, the candidate prepared a catalog of Virtual Machine Images (VMIs) with the plant genomic tooling packed in. The last part of this section presents a use case that illustrates the platform's mechanics.

4.2.1 Context

Under the frame of the European transPLANT project (FP7-INFRASTRUCTURES, grant agreement ID: 283496) a complete computational infrastructure for plant's genomics data it has been developed. The infrastructure should address the particular challenges and opportunities of the field. Compared with vertebrate genomes, plant genomes may be large and have complex evolutionary histories, which makes their analysis a hard problem, not only in scientific terms, but also for their compute and storage requirements. Issues include genome size, polyploidy, and the quantity, diversity and dispersed nature of data in need of integration. To address these problems, transPLANT has developed some solutions exploiting the expertise of the project's partners to provide a set of computational and interactive services to the plant research community. The project has built new data repositories, implemented novel querying systems, or contributed to the development of distributed computing tools.

We have designed the cloud environment intended to offer a virtual platform for programmatic and interactive access to applications of interest for plant genomics researchers.

4.2.1.1 Motivation

Data size is definitely one of the challenges the plant genomics community is confronting, like any other community based on the high throughput sequencing techniques that have evocated us to the data-driven science. Hence, special care in the management of such amount of data is to be taken, as it no longer allows individual laboratories to keep their

own databases. Genomics data is normally produced in large scale sequencing centers that act as data providers, together with large computational facilities performing highly demanding operations like mapping or assembling genomes. However, primary analysis is just the initial phase. A fast-evolving domain-specific corpus of analysis tools should be applied to the sequencing data to obtain useful information, a process that usually requires a fairly significant amount of expert's manual operations.

The present paradigm of a large sequencing center side-to-side with large computational facilities cannot fulfill the necessary requirements to complete downstream analyses. Thus, **data transmission** becomes a bottleneck due to two main reasons, the size of the data, and also possible privacy requirements. An obvious strategy to minimize the requirements for data transmission is to move the analysis tools to the same infrastructure that holds the data. As introduced, virtualization and web services are enabling technologies for building remote and decentralized environments. transPLANT have chosen to exploit them to offer a portable and flexible computational platform for the plant's community.

A second requirement for genomics analysis has also been considered here. Day to day, more analytical power is required to process the ever-increasing genomics data. For instance, a sequence assembler designed for processing animal genomes, can become useless for newly sequenced plant genomes, like crops, whose model genomes were only completed during the last 5 years due their complicated scaffolding processes. The reason is not faulty assembly **algorithms**, but the amount of input data they are designed for, which following current trends lead to unrealistic memory or CPU time consumption. As discussed, HPC is the solution that offers a most efficient scalability, however, the development of specific HPC solutions require certain stability in methods and processing, as well as particular expertise on the part of the software developer. transPLANT decided in favor of a distributed system like cloud computing. It offers also access to scalable resources able to smooth the impact of data size, and with no need to use specifically compiled software. On top of that, multi-scalable programming solutions are also employed to optimize the use of such distributed and scalable resources. These are characterized to be able to run executions of the same algorithm in very diverse architectures, from personal workstations to HPC supercomputers or grid-based computers.

Specifications

- Portable and expandable analysis platform able to get adapted to novel analysis tools
- Elastic and distributed computing backend ready to absorb the processing of important amounts of data
- Multi-scale parallel algorithms for better exploiting distributed resources

4.2.2 Cloud architecture

As depicted in the vertical diagram of Figure 4.6, transPLANT computational infrastructure is offered as a PaaS, where the necessary tools to develop new plant genomic applications are provided. Hardware resources are dynamically managed using PMES and COMPSs (see *3.1* Software components) in a way transparent to the user, on top of OCCI compliant cloud management platforms. From the researcher point of view, the platform follows a SaaS approach with a bench of prepacked virtualized applications well known in the plant genomics community. These are implemented as short-lived compute instances that act as private virtual environments automatically contextualized in terms of computational resources and accessible data.

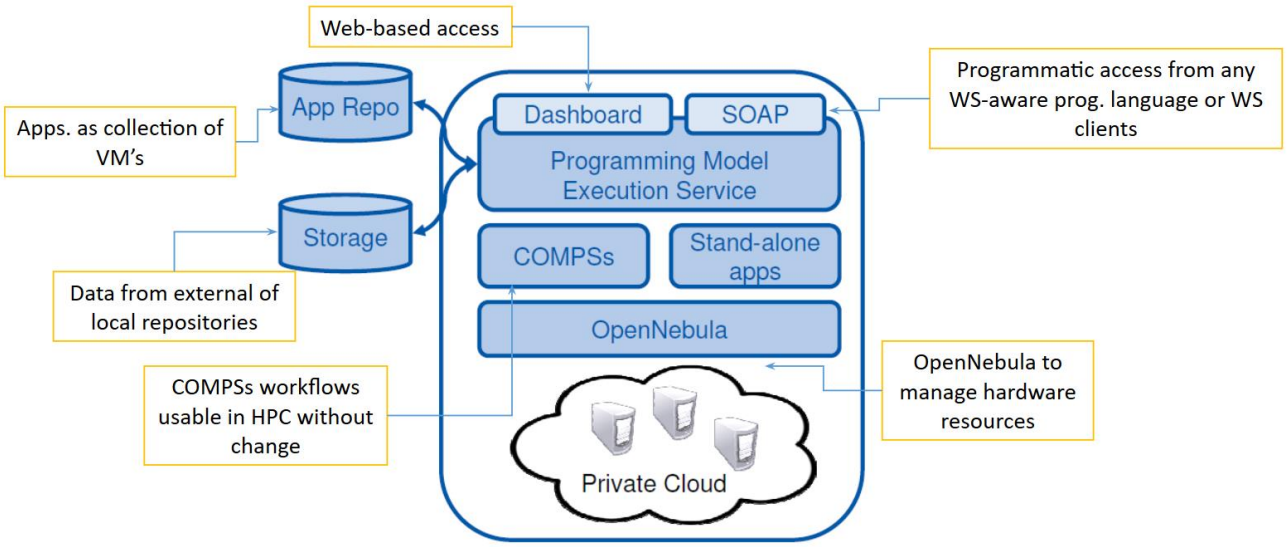


Figure 4.6: tranPLANT cloud architecture

The diagram flow described in Figure 4.7 sequentially describes the natural chain of events occurring across transPLANT components:

1- Researcher accesses transPLANT execution services on the Web through PMES dashboard, or programmatically via PMES SOAP API.
2- Access requires authentication. transPLANT centralize user's management in an internal LDAP server for all transPLANT services.
3- Researcher selects the tool it is interested in and configures the parameters for the run: arguments, location of input data files, resources, etc.
4- PMES enables the deployment of a transient virtual machine via OCCI.
5- PMES controls the full application lifecycle and offers the information at user's interfaces. Job monitoring information is obtained from the underlying CMP, in our testbed, OpenNebula.

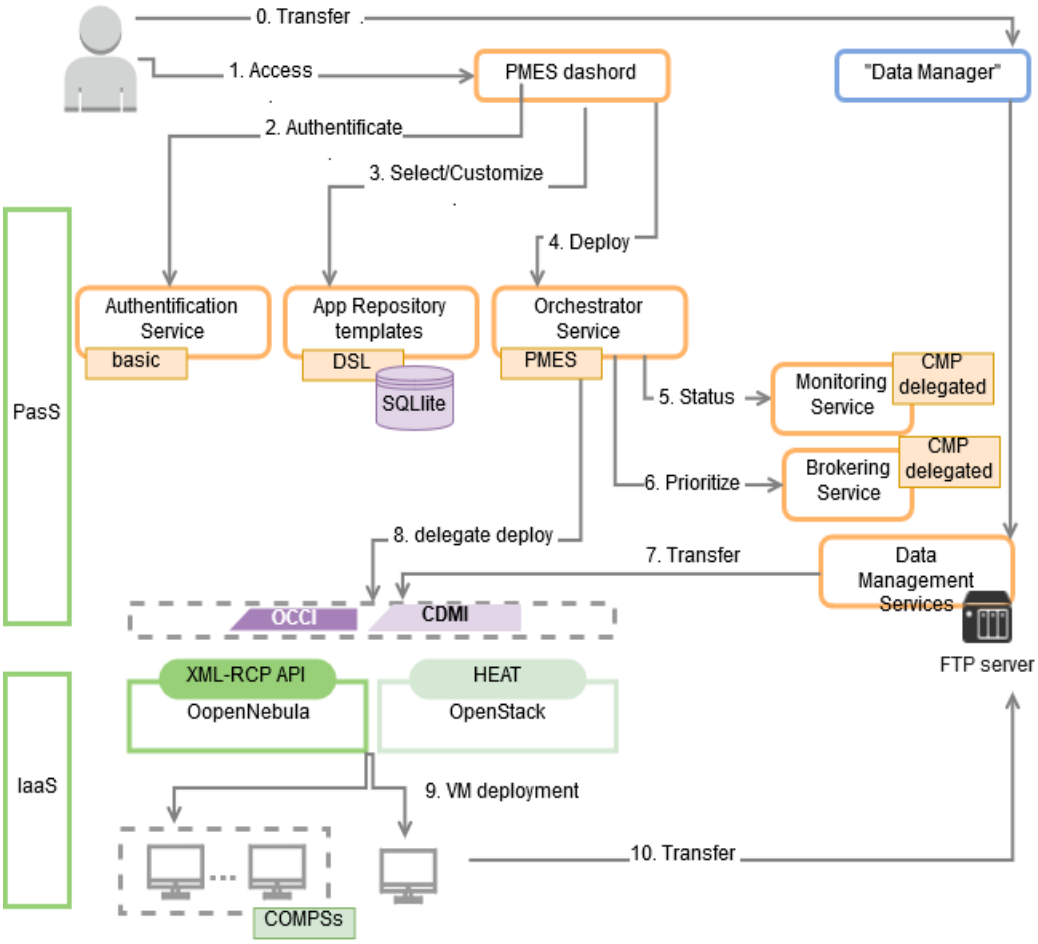


Figure 4.7: Process flow on the transplant cloud.
JSDL: Job Submission Description Language; DSL: Domain Specific Language; CMP: Cloud Management Platform

6- In a similar way, PMES delegates into OpenNebula the job prioritization and allocation on the compute nodes.
7- The actual deployment occurs after provisioning cloud resources. The VMI encapsulating the chosen genomic application is instantiated according to PMES job petition, which in turn, obeys to the configuration set up at PMES dashboard.
8- Defined input data files are loaded via FTP into the VM. For those users without access to FTP storages, data can be loaded into the cloud using a separated web application, the transPLANT “Data Manager” (step 0).
9- Once loaded, the actual execution takes places. PMES calls the application’s main executable, which could be either a standalone or a COMPSs application. The latter is able to create an elastic virtual cluster.
10- After the application’s execution, generated output file data is transferred into the FTP server, and VM(s) are eliminated, freeing allocated resources.

Hereinafter, the section describes one by one infrastructure's components here summarized, and how they interact to assemble the operable compute platform. They are classified as belonging to the IaaS, PaaS or SaaS layer.

4.2.2.1 Infrastructure as a Service

As any cloud-based infrastructure, the platform is not ligated to a specific **hardware** architecture. It can be implemented in any computer cluster of homogeneous or heterogeneous nature. The minimum composition requires a single computer node, acting as front-end and administering the resources through a CMP that implements the OCCI computed standard API, *e.g.* OpenNebula or OpenStack.

Since transPLANT itself did not provide computational resources, a testbed of the infrastructure is installed at the "*INB Cloud*" (see *3.1.2.1* OpenNebula for hardware details), hosted at the **Barcelona Supercomputing Center (BSC)**. There, the infrastructure prototype is developed and tested, yet the resulting products intend to be totally portable.

4.2.2.2 Platform as a service

PMES Server

PMES server (see *3.1.2.2* Provisioning Tools in Methods) is the core of transPLANT's compute service. It enables the deployment of **pre-emptible virtual machines** in remotely distributed infrastructures by interacting with the OCCI interface of the corresponding cloud infrastructure. PMES exposes a REST unified standard interface that hides from the rest of the platform's components the heterogeneity of the underlying infrastructure. PMES sequentially undertakes the following steps:

I. instantiate the virtual appliance according to the requirements set up in the job definition
II. contextualize the VM with Cloud-init;
III. stage in input data, via FTP;
IV. execute the application according to the arguments given by the researcher;
V. stage out result data, via FTP;
VI. undeploy the virtual appliance.

The SaaS layer, either PMES API or PMES Dashboard, compose a job definition including what VMI is to be instantiated, how big it should be, which user or mounting point needs to be contextualized, where are the files to be staged in, and what are the arguments of the application. Along the whole job, PMES controls the appliance lifecycle as well as collects logging information from the guest OS, always via the OCCI interface who mediates the

communication with the CMP. As such, researcher tasks are executed as **batch jobs** remotely and in sandboxed optimized environments where computational resources are offered on-demand and exploited efficiently.

〉 Tool's virtual machines

Virtual Images

The collection of genomic tools, usually complex applications with a multitude of dependencies not always automatable, are encapsulated in **VMIs** fully compatible with most common cloud providers. We have prepared a base image (QCOW2 and OVA formats), with the transPLANT software stack installed in it. The OS guest of choice varies depending on the genomic software to be installed, yet, for most cases, it corresponds to an Ubuntu 14.04 LTS. The disk image includes:

- COMPSs task orchestrator (see *3.1.3* Job Managers)
- One-Context contextualization package (see *3.1.2.3* Contextualization). It is parameterized at boot time to ensure PMES interaction with the freshly created VM. Following, are the required settings:
    - IP-based hostname,
    - local user with an associated home directory,
    - SSH public keys injection to enable password-less access to that user

The whole catalog of transPLANT VMIs is prepared by installing the genomics' tool of choice, mainly assembling pipelines, on top of this base image (tools listed in annex *8.2* transPLANT tools). Corresponding VMIs are imported in OpenNebula, although they are also distributed at the project site[4], together with some documentation.

Templates

To be able to deploy these disk images into VM instances, cloud managers require some extra information. VM templates collect that. We have prepared for OpenNebula a VM template for each VMI tool. In fact, when the OCCI connector is used, VM templates are split into image templates and resource templates (see *3.2.1* Open standards). At the **image template**, the network settings are prepared. A single NIC with a dynamic IP address is configured. The IP belongs to the selected virtual network, a dedicated VLAN that ensures isolation, and also manages the number of available leased addresses. Internet access is routed via the cloud frontend host, that acts as a gateway. If a temporary disk to increase runtime storage is to be attached, it is also defined in this step. On the other hand, at the

[4] http://transplantdb.bsc.es/transPlantCloud_Apps.htm

resource templates, CPUs and RAM memory are set up. The OCCI server of each IaaS offers an extensible list of resource templates (*e.g.* "small" - 1 core and 2 GB RAM). A valid combination of both, resource and image template, is defined when launching a PMES run.

〉 COMPSs

For those applications that natively do not feature **parallelization** (threads, MPI, etc.) or those encapsulating a pipeline potentially parallelizable, COMPSs (see *3.1.3* Job Managers) can be used to orchestrate their tasks and add scalability to the system. Along the execution, the workflow manager transparently and dynamically assigns resources to each of the tasks according to their computational needs. If tasks cannot be locally allocated on the VM, COMPSs runtime deploys worker VMs the allocate them. In this way, a **user's virtual cluster** is dynamically created, the master machine by the initial PMES instantiation, and the worker nodes as replicas of that master (Figure 3.1). COMPSs fulfills the multiscale requirement indicated above, the same workflow definition could be executed in single workstations, in a large HPC facility or in a distributed grid, without modification.

4.2.2.3 Software as a Service

〉 PMES API

In order to programmatically manage the compute services of the transPLANT, PMES offers a set of SOAP-based **web services** (*3.1.2.2* Provisioning Tools in Methods). The Java API implementation is compliant with the Basic Execution Service (BES) standard [59], a well-known open standard for remote job management in distributed environments. BES endpoints (listed in Method's section *3.2.2*) provides a common interface for remote task managing. Each new BES activity corresponds to a batch job petition to PMES server. These petitions, are expected in the Job Submission Description Language (JSDL) specification (see *3.2.3 in method's section).*

Well-known WS clients like Taverna [224], now part of Apache Incubator, can be used to remotely send tasks to the infrastructure. Following snapshot corresponds to the usual operations needed to submit a job using the Taverna client.

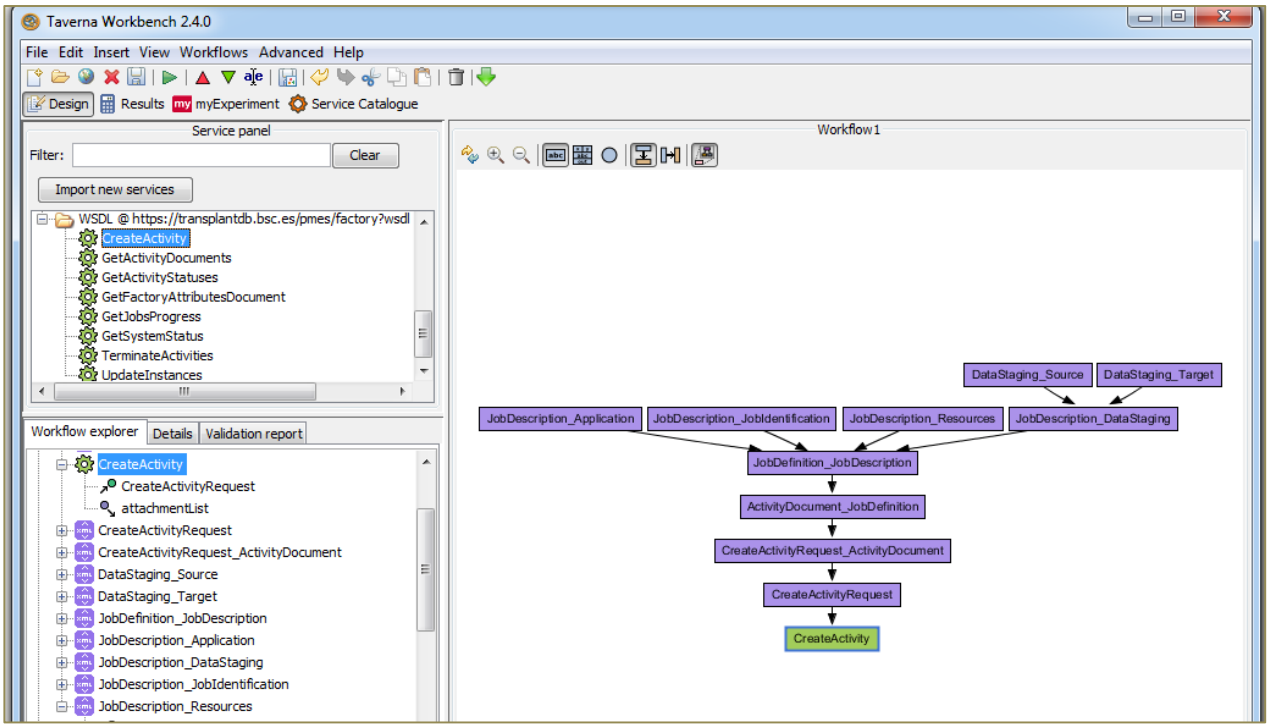


Figure 4.8: PMES SOAP operations to submit a new PMES job

PMES Dashboard

For occasional or developmental uses, PMES offers a **web-based Dashboard**[5]. For not programmatic access, PMES Dashboard is the core manager of the infrastructure as, likewise the API, it allows full control of the PMES layer. It provides a UI where researchers configure, monitor and **execute their application** runs. PMES dashboard holds the registry of transPLANT applications, whose VMIs are in the repository of the underlying CMP. The flow that an end-user follows at the dashboard is essentially:

I. Select the plants genomics' tool of interest. Available applications with a brief description for each is listed.
II. Specify the location of the input files. As will be discussed below, these should correspond to an FTP URLs.
III. Set the values for the application's arguments. They will be part of the executed command.
IV. Specify where the expected output files are to be stored.
V. Launch the job application and monitor its advance in the central panel.

[5] https://transplantdb.bsc.es/pmes/

VI. Check execution's STDOUT and STDERR.

PMES deals with both standalone and COMPSs based applications. The central panel is where user's batch executions are recorded, and enables users to monitor their job progressions. Eventually, PMES stages out the results from the transient virtual machine to the user's FTP selected location.

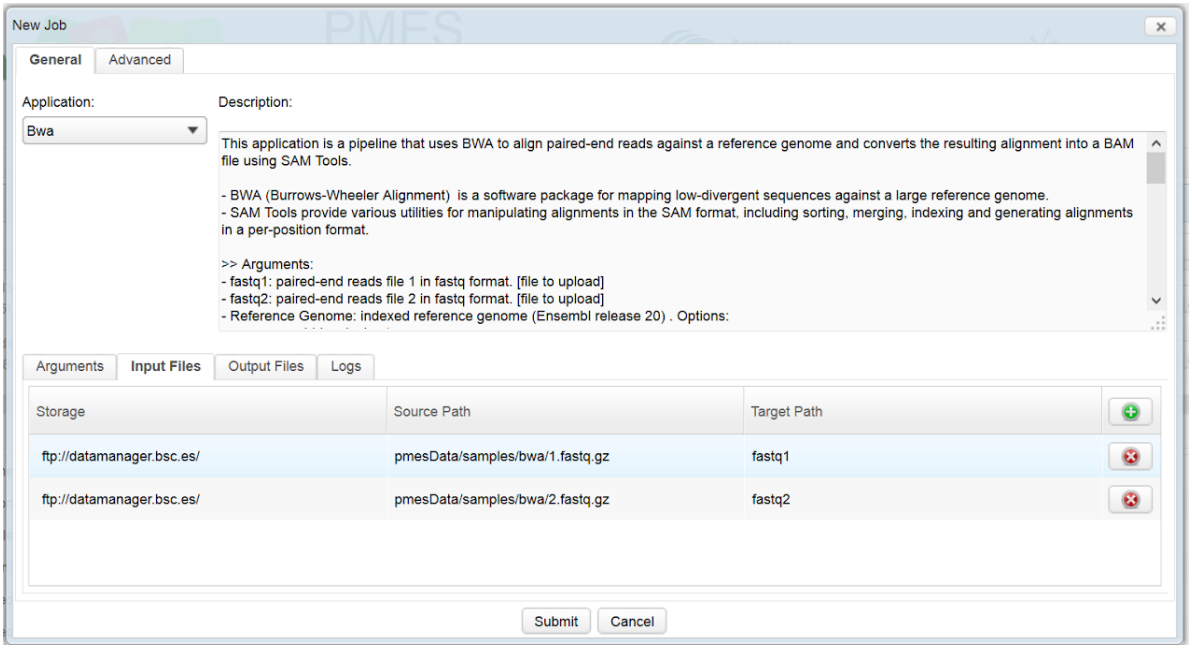


Figure 4.9 Screenshot of PMES dashboard.

Data manager

Although PMES dashboard perfectly manages job-related operations, user's data management operations are not that well resolved, as it assumes that the end-user has access to an FTP server. Thus, a second Web interface is implemented in order to better support user's data transferences to the transPLANT cloud platform and cover that gap. TransPLANT "Data Manager"[6] is a **web-based file server** that enables users to upload and download their data via HTTP(S) onto the platform. Indeed, the application acts as a web frontend of an FTP server set behind BSC premises and configured as the web application's storage.

Behind, the **FTP server**, implemented with vsFTP [225], is configured with custom directories and virtual users accounts, integrated via PAM (Pluggable Authentication

[6] https://transplantdb.bsc.es/uploader/

Module) into the transPLANT authentication system. “Data Manager” populates FTP directories with user’s uploaded data, and thanks to an accurate permission management, they are also accessible by PMES via FTP. The full data flow is further discussed in *4.2.3 Data Management*.

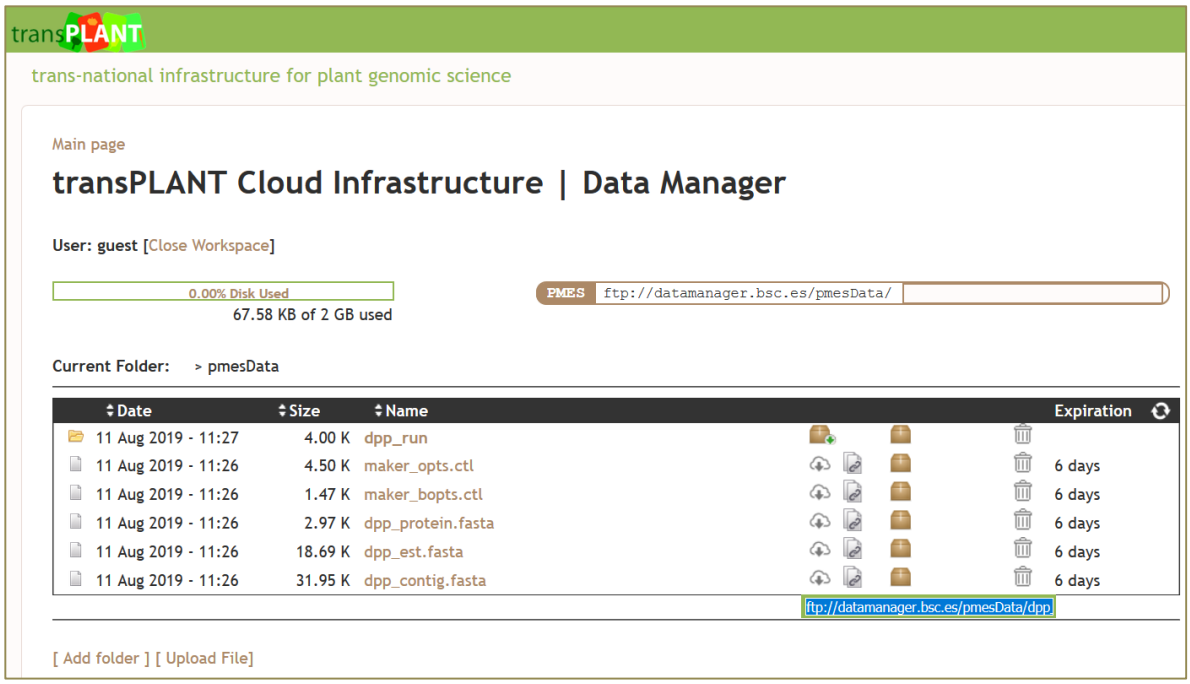


Figure 4.10: ScreenShot of DataManager workspace

Written in PHP, the **user interface** consists of a one-page application that offers an intuitive user’s space where the remote files and directories are displayed under a clean file system layout (Figure 4.10). The central panel displays the user’s current working directory, its space quota, the navigation menu to move around directories’ hierarchy, and some basic file-based operations: new folder creation, file download or deletion, and data compression and archiving (ZIP, TAR). The user selects one or more files from their local system, and these are transferred on the current working directory. All the time, the directory FTP URL (ftp://datamanager.bsc.es/pmesData/$CWD/) is shown on the right-top corner of the central panel, as this is the only piece of information the user requires to submit a job on the platform, either via PMES API or PMES Dashboard (Figure 4.11).

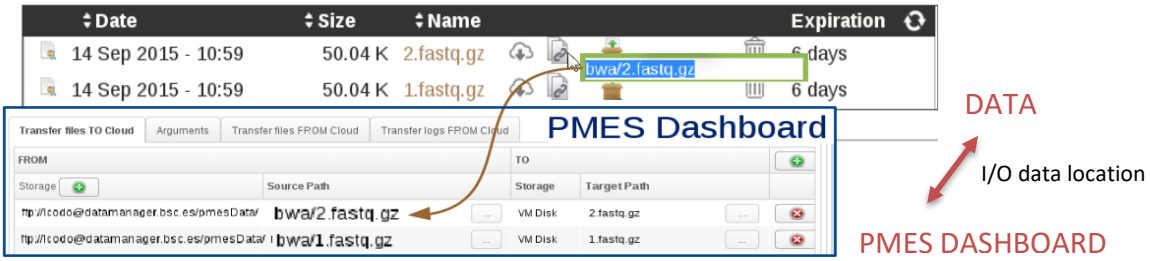


Figure 4.11: “Data Manager” provides de data location URL.
When defining the data location of new jobs in PMES dashboard, we can use the FTP URL given by the “Data Manager”.

"Data Manager" authenticate users against the internal transPLANT LDAP, so it does not handle users accounts on his own. In fact, the web application is so light that it does not even require a database.

4.2.2.4 Authentication

An internal LDAP-based authentication system is set using a **Basic authentication** schema. It centralizes the access control of the different infrastructure components: PMES Dashboard, PMES API, "Data Manager", and transPLANT FTP server.

User accounts are granted on request through the project's website, yet a demo user is published:

- User: guest
- Password: gu3st2014

Accounts are set using the admin section of PMES Dashboard. When creating a new user, credentials are stored at the SQLite DB of PMES Dashboard, and automatically, the new account is also created at the transPLANT LDAP, which is configured as an LDAP POSIX-account. Such type of entries permits a central Linux access management by means of the PAM module. As such, central LDAP user accounts are compatible with transPLANT FTP server authentication, eventually building a **unified account system** for all transPLANT services.

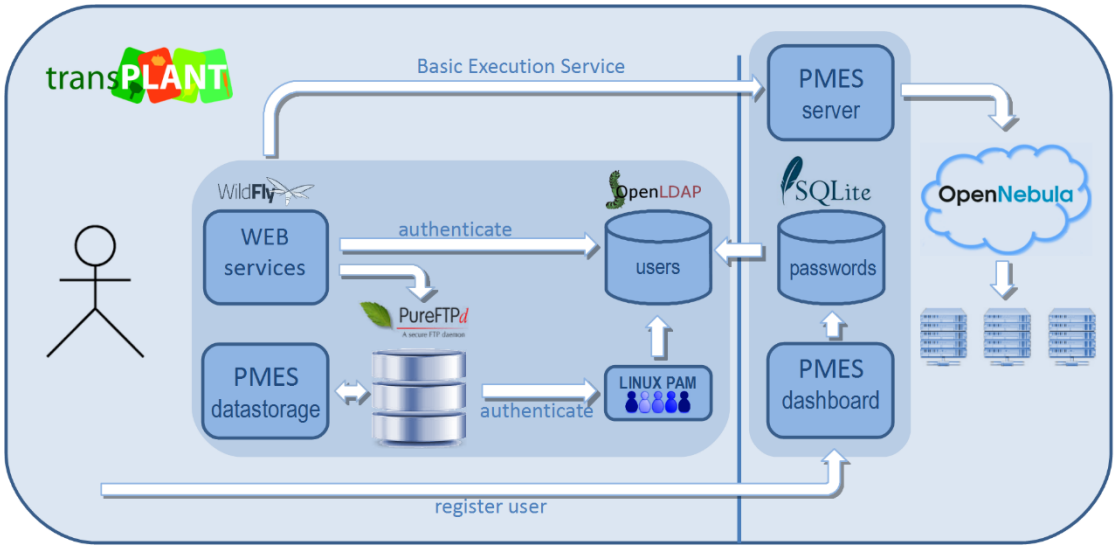


Figure 4.12: Scheme of the transPLANT authentication system

4.2.3 Data Management

Cloud-based infrastructures consist of volumes of data which might be shared among multiple tenants. Thus, data management is an essential aspect for data storage in cloud, both for security and efficiency. Besides, transPLANT, as mentioned above, pretends to deliver a **portable infrastructure**, as hybrid or multi-cloud approaches are desirable in the scientific landscape for flexibly co-locating data and computational resources or simply extending private data centers with on-demand public clouds (hybrid cloud bursting). However, offloading computation not only requires dynamically managing the compute resources, but also moving target data into the allocated VMs.

4.2.3.1 User's data

Although some high-level solutions are lately emerging for multi-cloud environments (see *1.4.4.2* Data-storage solutions), these are still not mature solutions with one limitation or another, and most existing technologies often require researchers to move data explicitly between the processing steps of geographically spread pipelines. Here, PMES is responsible to transfer the data to the local VM storage through the network using the standard **FTP protocol**. Thus, researcher's data is to be stored in a FTP server. During the service life cycle, PMES server stages in input files into the virtual appliance (requesting user's credentials if needed), to subsequently run the application, and eventually stage out user's specified results files into researcher's FTP server before undeploying the virtual instance. Data is accessed locally during the runtime, especially beneficial for processes with high I/O loads. Additionally, to prevent data allocation issues, user's data is located in a separate partition of the VM that might be implemented as a volatile data block sizable during the deployment process by contextualization.

Alternatively, **"Data Manager"** provides a web-based data loader to support those users that may not have access to an FTP server. It acts as a web frontend for the transPLANT FTP service, a server prepared to provide temporary storage of user's data - according to current transPLANT policies, data is stored up to 7 days before it expires.

"Data Manager" finds the data for each user under its corresponding directory, which is created automatically during user's registration process and corresponds to user's FTP root. The application organizes user's data in disk following the "User > PMESdata > Workspace" hierarchy:

```
userData/
+ --- username/                        # user's home directory. FTP user's chroot.
|     + --- PMESdata/                  # Bellow, users freely organize their data
|     |     --- MyInput1.txt
|     |     --- MyInput2.txt
|     |     + ---MyExecution/
|     |     |     --- MyOutput1.txt
|     |     |     --- MyOutput2.txt
|     |     |     --- stderr.log
|     |     |     --- stdout.log
```

Snippet 4.1: Directories hierarchy for transPLANT user's data.
Data organization below user's root is free. User builds it either accessing viat FTP or navigating online with the "DataManager".

The application **manages the permissions** so that data's ownership at the system level is maintained as that of the logged user account. In this way, the "Data Manager" handles cloud's data on behalf of the user. By doing so, FTP credentials secure PMES transferences, regardless whether the FTP site had been populated directly or via the "Data Manager".

4.2.3.2 Reference data

A usual use case in the research community is also the use of reference datasets and databases, which we prepare in order to fulfill the particular needs of the applications populating our testbed installation. For reference data, an architecture based on local data sharing is proposed. OCCI supports the attachment of file-based or block storage devices as part of the contextualization process. In this way, compute VMs mount at boot time the **shared folder** where the reference data is located using a NAS protocol, either CIF or NFS. In the present testbed installation, the shared folder corresponds to an NFS (v3) endpoint on the cluster NAS storage, installed on top of a GPFS distributed file system. Context parameters, primarily the NFS endpoint address and the credentials of the mapped user, are passed as part of the job definition to the PMES server, who in turn set the adequate script for the contextualization package triggered on boot. Only a single and static NFS endpoint address is required, as the storage is configured using clustered NFS (CNFS), so that on node failure, the NFS serving load is transparently moved to another node. Also, to ensure data consistency across VMs simultaneously accessing the reference data, NAS is accessed in read-only mode.

4.2.4 Use case: plant genome annotation pipeline in MAKER

Following lines describe how the different components of the transPLANT cloud interact on the Web with the end-user. The example covers the deployment of a plant genome annotation service on the compute infrastructure.

4.2.4.1 Preparing a transPLANT application

Annotation Pipeline

Genome annotation is still a major challenge in plant biology, far from a straightforward, 'plug-and-play' activity. transPLANT partners designed an annotation pipeline [226] (Figure 4.13) based on **MAKER2** [212], a genome annotation tools, in combination with InterProscan and Uniprot analyses.

MAKER uses a combination of AUGUSTUS, SNAP and Genemark for gene model assessment, using either external training sets, or training sets derived from the data itself. The pipeline takes as input the genome assembly, a configuration file in a YML-like format, and optional evidence files (protein, EST, gene models). MAKER generates three different outputs: a) GFF file with genes and gene structures; b) FASTA file with predicted proteins, and the InterProscan analysis of the same; and c) the CSV alignment output from BLASTP against Uniprot.

Figure 4.13: transPLANT application: MAKER2 pipeline

Creating transplant application

We have encapsulated the pipeline in a **virtual machine image** based on the base transPLANT image. For that, we have configured and installed all the necessary software, and prepared a Python wrapping script that would act as the appliance entry point. Next step consists of configuring the VM template on OpenNebula for the new image. It simply is going to state the VMI identifier, and the usual network configuration, by which the cloud middleware will dynamically assign an IP from a given local VLAN. Data requirements (*i.e.* storage attachments) are not considered at this stage, as PMES is going to make them available on runtime.

Once the VMI is ready, we have configured the PaaS components in order to set how they need to interact with the new VM. PMES will require deployment details (*e.g.* target cloud, demanded resources, VMI OpenNebula identifier), contextualized actions on boot, application CLI details (*e.g.* arguments, input files), FTP data location, etc. If using the PMES API, we would need to compose the corresponding JSDL submission file with all this data. As it could be a tiresome process if the user does not prepare an automatic approach to set it up, all prepared VMIs are **registered into PMES Dashboard**. To do so, the Dashboard includes a web form where the following information regarding the new application is required:

- Name: the name of the application.
- Image: VMI OpenNebula identifier containing the application.
- Location and Path (Optional): URL of the package to be deployed in virtual machines after system boot.
- Executable: Name of the executable to run. In COMPSs applications, the name of the main class (*e.g.* simple.Simple), in standalone applications, the name of the command (*e.g.* /usr/bin/maker).
- COMPSs: If it is a COMPSs application or not.
- Public: If the application can be executed by other users.
- Description (Optional): A brief description of the application.
- Wall Clock Time (Optional): The default execution time of the application.
- Disk Size (Optional): The default disk size for the virtual machines when running the application.
- Cores (Optional): The default number of cores for the virtual machines when running the application.
- Memory (Optional): The default memory for the virtual machines when running the application.
- Max VMs (Optional): The maximum number of virtual machines to run the application. This is used only for COMPSs applications.
- Min VMs (Optional): The minimum number of virtual machines to run the application. This is used only for COMPSs applications.

Additionally, to help PMES build the application CLI, arguments, input files and expected output files are to be specified. Each argument will be optional if specified or mandatory otherwise, and it is defined by a name, an optional default value and an optional prefix. Eventually, the CLI launched by PMES would look something like this:

- COMPSs applications `runcompss executable [[prefix] [argument]]`
- Standalone applications: `executable [[prefix] [argument]]`

4.2.4.2 Running a transPLANT application on the platform

Load user's data using "Data Manager"

If user has no FTP site to expose their data to PMES, first step is to do so by loading researcher's data into the FTP site of the transPLANT cloud. "Data Manager" is the web interface that enables such data **uploading**.

Users are to log in, and given their workspace - *i.e.* their FTP directory -, they upload the required data to run the annotation pipeline via HTTP from their local PC. For MAKER application, the compulsory input file is a FASTA file with the assembly contigs and the YAML configuration. An example for these files is found at the project's site[7]. Once loaded, "Data Manager" shows the FTP URL for each of the files, which are to be used at the PMES Dashboard, when defining run input files.

Configure PMES Dashboard

On first PMES Dashboard login, some user's configurations are to be set, so that the framework knows from/to which FTP server the I/O data is to be transferred, as well as which are the environment variables to be sent to the **contextualization** script, if any. These arrangements are stored in the Dashboard application, thus, they are only configured once. Adding a storage registry only requires the FTP URL of the user's remote directory and a descriptive name, while contextualization variables are given as key-value pairs. For using the transPLANT storages of the current testbed installation, the settings are:

- Storage: The "Data Manager" indicates the URL, *i.e.* ftp://datamanager.bsc.es/pmesData/. Credentials are not stored, only requested when needed, and the unified authorization system permits to share the account between PMES-Dashboard, "Data Manager" and the transPLANT FTP server.
- Contextualization: In the current installation, for PMES mounting the NAS where the reference data is stored, the following variables are to be set:

[7] https://transplantdb.bsc.es/documents/samples/maker

DATA2_USER = guest
DATA2_PWD = guestTransplant01

It indicates PMES to attach DATA2 NFS mounting point after the BOOT. In fact, the credentials correspond to the NFS identity mapping (between the virtual instance and the network local storage).

Run MAKER on PMES Dashboard

MAKER application is part of the collection of tools integrated into the transPLANT platform. Hence, it appears as an option under "New Single Job" creation menu. There, a description of the application is given and users define which are application's arguments and input files. For each input file, a storage (FTP site) is to be set, the information that "Data Manager" provides in case of using the internal transPLANT FTP server. The FTP location for the expected output files is also required. After application details are set, the job can be submitted.

Behind, the magic happens, and the provisioning process is initiated. PMES pulls from the CMP the deployment status, and once the VM is running, the **application life cycle** advances. First, the VM parameterization attaches DATA2 NAS, then, the two input files are staged in at the allocated VM from the local FTP server, the MAKER wrapping script is transparently called using user-defined arguments, and finally, once the execution terminates, the output files are fetched from their expected appliance location and transferred to the user's FTP server before the virtual instance is deleted. Meanwhile, PMES reports the appliance STDOUT and STDERR, enabling real-time control of the execution progression.

Download user's data using "Data Manager"

Before termination user's virtual run, PMES **downloads** the output files into the desired FTP's location, together with logging information. If the transPLANT FTP server has been defined, the "Data Manager" interface is used to check for those files. According to current transPLANT policies, the data is going to be stored for 7 days before it expires.

4.3 MuG: Multiscale Complex Genomics VRE

This section describes the design and development of the cloud-based virtual research environment (VRE) we have built in the context of the Multiscale Complex Genomics (MuG) project.

The candidate has directly participated in the design and development of the overall cloud architecture as presented below, and in particular on the implementation of MuG Virtual Research Environment (MuGVRE), the central platform interface. A manuscript pre-print is attached at annex *8.7.2* Publications

4.3.1 Context

The Multiscale Complex Genomics (MuG) project (H2020 EINFRA-9-2015; grant agreement ID: 676556) is born as a bottom-up approach pushed from some European **3D/4D genomics** research groups that combined three different expertise: (i) biologists with interest in chromatin structure, (ii) method developers, and (iii) HPC facilities with strong history of supporting Bio-computational problems. Genomics is probably the fastest evolving field in current science. A decade ago the main concern was to obtain the sequence (the 1D code) of the genome; today the big challenges are to determine how genotype information is transferred into phenotype. Investigations elucidated that part of the gene expression regulation is implicitly coded in the way chromatin is folded, which lead to this new branch of the genomics called 3D/4D genomics and focused on the dynamics of chromatin structure.

We have contributed to strength this young and dynamics field by implementing the **MuG Virtual Research Environment (MuGVRE)**, a platform that aims to provide the 3D/4D genome community with an adequate combination of relevant data, visualizers, and computational tools to non-programmers. MuG partners contributed by providing specialized visualizers, and a complete catalog of analysis tools (annexed *8.3* MuG tools & visualizers).

4.3.1.1 Motivation

Hundreds of laboratories are right now conforming the 3D/4D genomics field, which, even though eventually concerned with the same scientific problems, employ many different approaches to study it. Individually, they target radically different length and timescale data, coming from biophysics to cell biology disciplines, and use technologies as diverse as

HiC or ChIP-seq or microscopy. The collaboration in a field so notably **multidisciplinary** highly predetermine the framework to be developed. An easy-to-use and supportive environment is an essential feature in order to inclusively integrate the different expertise in MuG represented.

Yet, simplicity in use should not reduce flexibility, as handling **multi-scale data** put very diverse sources together, like atomistic simulations, genome annotation, middle and high scale 3D genomics, and cell biology imaging data. Moreover, the youth of the field exacerbates the profusion of file types and formats, with no clear methodologies nor standards. In this situation, is essential to establish a common and domain-agnostic framework that permits to set relationships among the different data levels, in order to build an integrated view of all the components for a complete 3D/4D genomics study. Additionally, users are expected to be active in generating and analyzing data in a given expertise, and interested in browsing and integrating data from the others. This breaks the common paradigm in bioinformatics where most users are data consumers, while data is produced by a reduced and controlled community or even a single group. As a general rule, all members of the VRE should not only use data and computational resources, but they also need a set of tools to populate the platform with their own analysis. Thus, the computational infrastructure should ensure interoperability of analysis tools and generate an integrated environment with a seamless transition among the available data levels.

From a computational point of view, 3D/4D genomics constitutes also an extraordinary challenge. Very much like plant genomics, data size and transmission are to be carefully considered, as discussed above. Hence, the single user environment is to ensure the best efficiency in both, data mobilization and processing. Finally, MuG infrastructure's design should project the support of some transversal European e-infrastructures, namely, EGI and PRACE for computational capabilities, and EUDAT for shared storage, aligned with global initiatives like EOSC, ELIXIR, and GA4GH.

Requirements

- Flexible environment, able to adapt to the specific needs of the analysis tool, both in terms of software requirements, or computational resources.
- Software scheduler(s), able to manage computational resources in a transparent and adaptable manner. This will be an elastic infrastructure with automatic adaptation to user's loads.
- Web-based centralized access. User access will integrate MuG authentication, data, compute and visualization services.
- Community driven platform. Users are expected to integrate analysis tools and visualizers into the platform in an easy manner.

4.3.2 Cloud architecture

MuG computational infrastructure is using already existing components, which have been adapted to work in an integrated manner at the overall platform, while complemented with new elements when necessary. The layout has been designed as an evolution of the cloud infrastructure presented above for the project transPLANT. Below, Figure 4.14 details the layers of the new architecture.

MuG platform is offered as a PaaS via the project's virtual research environment (MuGVRE), that concentrates most of the infrastructure services following the one-stop-shop approach. Virtualized resources are transparently and dynamically managed using PMES and COMPSs, in a similar way transPLANT appliances were provisioned, though completely revised versions of those are used here. Additionally, a second task management strategy is proposed here, based on a combination of the Sun Grid Engine (SGE) queue system and the OneFlow auto-scaling service.

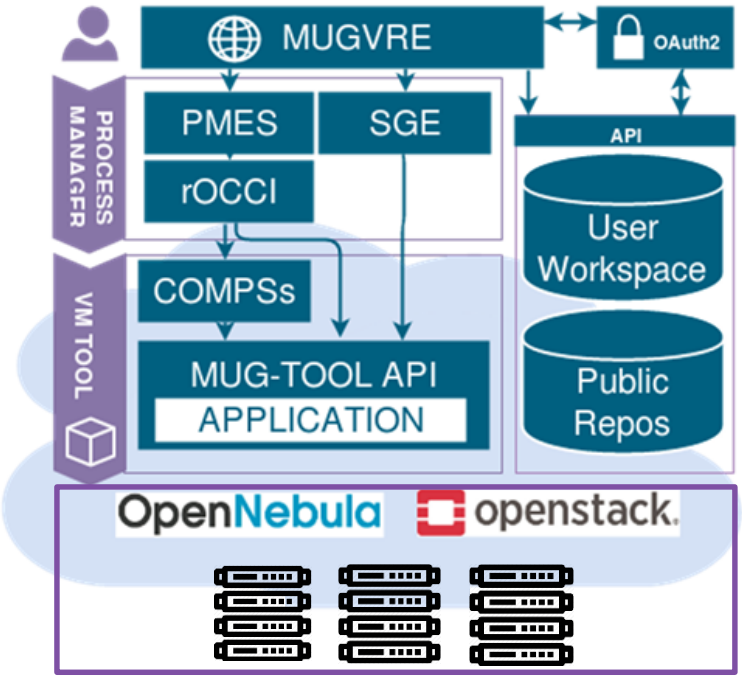


Figure 4.14: MuG cloud architecture

Anyhow, appliance petitions reach the IaaS, layer managed by any OCCI compliant platforms, like OpenNebula or OpenStack, and there, MuG VMs are instantiated (short or long-lived). These encapsulate MuG applications, 3D/4D genomic analysis tools provided by tool-developer users. Transversal to the infrastructure, a centralized authentication system is set up, as well as a shared storage system.

A description of each of these components and their interactions is summarized in Figure 4.15. The diagram enumerates the following stages to briefly illustrate the functioning of the analysis platform:

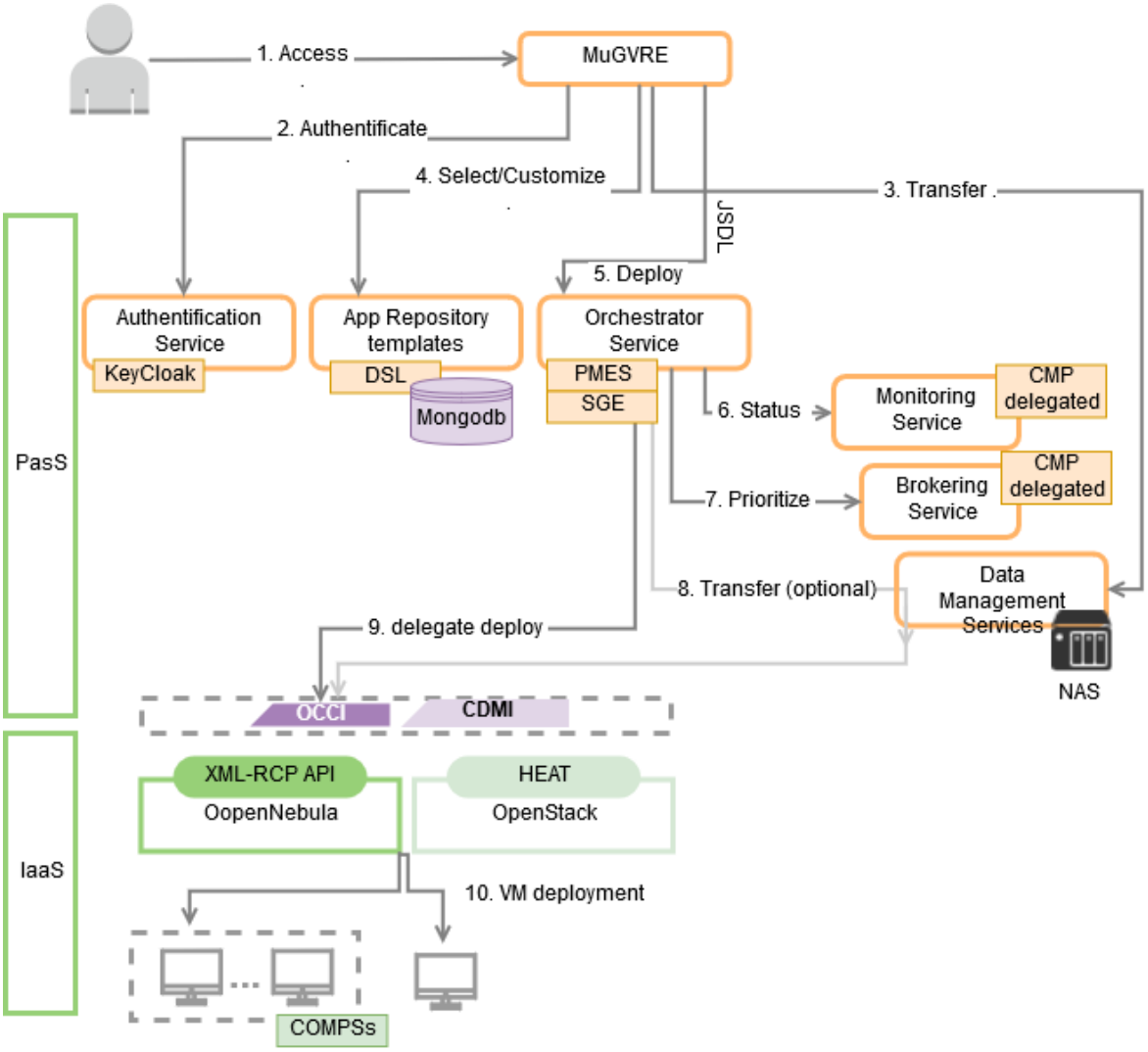


Figure 4.15: Diagram flow of the MuG cloud

1- Users access the cloud via a unique web frontend, MuGVRE, a complete virtual research environment (VRE)
2- Users access their own space after authenticating themselves against an external authentication server based on KeyCloak (OpenID connect with OAuth2 flow).
3- Users load their data into the platform, internally stored in a shared storage accessible over the cloud network
4- Users browse, select, and configure the analysis tool they want to run. These are found registered in the MuGVRE Mongodb using an *ad hoc* data model.
5- Using Tool registry data, MuGVRE composes a job according to JSDL specifications where the virtual machine image encapsulating the selected analysis tool is targeted. The JSDL template is posted to PMES server, which acts as PaaS orchestrator.
6- Using standard BES specifications and a new REST interface, PMES offers job monitoring, information that it retrieves from the underlying infrastructure using the OCCI API. At MuG's production cloud, the CMP is OpenNebula.

7- Similarly, job allocation – *i.e.* computational resources required to instantiate the Tool VM - is delegated to OpenNebula.
8- User's data is sitting in a network-attached storage (NAS), previously loaded by users (step 3), and dynamically attached to deployed VMs, which avoid any extra transfer process. OCCI does accomplish the task, so the CDMI cloud data protocol (see *1.4.3.3* Cloud interoperability and portability) is not implemented.
9- PMES requests the instantiation of the selected image to OCCI server, who in turn interacts with the corresponding CMP provisioning service, for OpenNebula, an XML-RCP-API. Instead, if the IaaS would be provided by OpenStack, OCCI would interact with its HEAT component.
10- After instantiation, MuG VMIs are contextualized with the relevant data and permissions according to JSDL instructions. Afterward, PMES triggers the actual analysis tool. It could correspond to a standalone application or workflow, or a COMPSs pipeline, which would, in turn, deploy a cluster on-demand, again via OCCI.

Following sections describes the components of the MuG cloud platform, here summarized. Firstly, the infrastructure layer, secondly, the PaaS components responsible for orchestrating the deployment process, and eventually, the outer layer, in contact with the researcher.

4.3.2.1 Infrastructure as a Service

The cloud infrastructure currently in production for MuG corresponds to the "MMB Cloud" at the **Institute for Research in Biomedicine (IRB)**, an on-premises cloud managed by OpenNebula. More details on the infrastructure can be found at section *3.1.2.1* in . Software, Data & Methodology .

Additionally, MuG consortium also has access to two other cloud infrastructures, not currently in production but intended for developing purposes, mainly benchmarking of analysis tools under development or the exploration of a multi-cloud approach. These correspond to **two other private clouds** sitting in separate locations:

- "INB Cloud" at the Barcelona Supercomputing Center (BSC)
- Tenancy of the "Embassy Cloud", located at the European Bioinformatics Institute (EMBL-EBI).

Both installations are further detailed at Methods sections, under *3.1.2.1* Cloud management platforms.

4.3.2.2 Platform as a Service

MuG platform is conceived as a dynamic ecosystem where analysis tools are plugged in and out of the infrastructure. Hence, PaaS building blocks are the virtual appliances (VMs) encapsulating such methods. PaaS components are in charge of:

- providing Tool developers with the means and instruments that enable a rapid integration of their methods into the platform
- implementing scalable and atomization capabilities for the deployment and provisioning process of Tools
- interacting with the IaaS layer

MuGVRE communicates with PaaS components to pass through job researchers' petitions collected on the VRE. Transparently and dynamically, PaaS handles IaaS resources. These components are primarily two different scheduling engines for task management on compute virtual machines. Figure 4.16 summarizes them. The first strategy uses PMES provisioning, as transPLANT cloud does, while the second is based in a combination of SGE and OneFlow. Compute VMs are configured one way or the other when integrated into the MuGVRE.

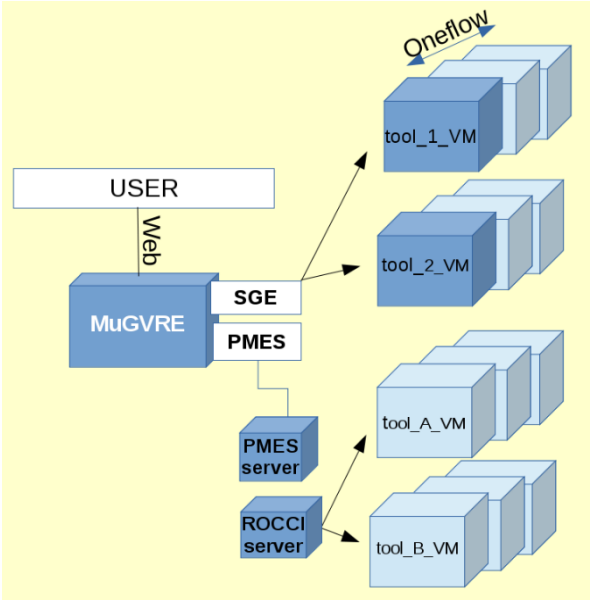


Figure 4.16: Two MuG scheduling engines: i) PMES and ii) SGE & OneFlow

PMES

PMES (see *3.1.2.2* Provisioning Tools in Methods) allows to remotely submit job execution requests to cloud infrastructures via OCCI. How PMES is used in our computational infrastructures is already explained in the transPLANT section (*4.2.2.2* PMES Server). In short, it controls and monitors the complete life-cycle of the virtualized applications

provisioning **pre-emptible VMs**, *i.e.* stateless short-lived VMs acting as transient environments for the analysis tools, which after the execution place, they are switched off.

A PMES server is running on each MuG cloud infrastructure. MuGVRE backend communicates via REST with it (Figure 4.16). Currently, PMES server is accepting petitions on the "MMB Cloud", and a development installation is available at the "Embassy Tenancy". Thus, PMES allows submitting jobs on remote clouds infrastructures, opening the doors to a MuG federated cloud. The major requirement is that cloud providers should support the OCCI connector, like OpenNebula and OpenStack do.

SGE & OneFlow

The second MuGVRE job scheduler is based on the traditional Sun Grid Engine (SGE). SGE is designed to manage distributed software executions in heterogeneous computational environments. MMB-IRB's has been using SGE as a scheduler for web application back-ends. To adapt it to the MuG cloud layout, the backend of the MuGVRE is defined as a master of the SGE system that dispatches job petitions the virtual machines encapsulating the applications, instead of being sent to regular compute nodes. A **long-lived VM** per application exists in the cloud, and the queue system is configured accordingly, with separated queues per each application. Here lies one of the differences with the PMES strategy, where is the job petition that triggers the provisioning of VMs. SGE requires a worker host alive waiting for job petitions.

To provide **elasticity** to the system, a second element is added to the equation, OneFlow. The VM scheduler, part of OpenNebula, is able to dynamically auto-scale the number of VMs instantiated according to workload parameters.

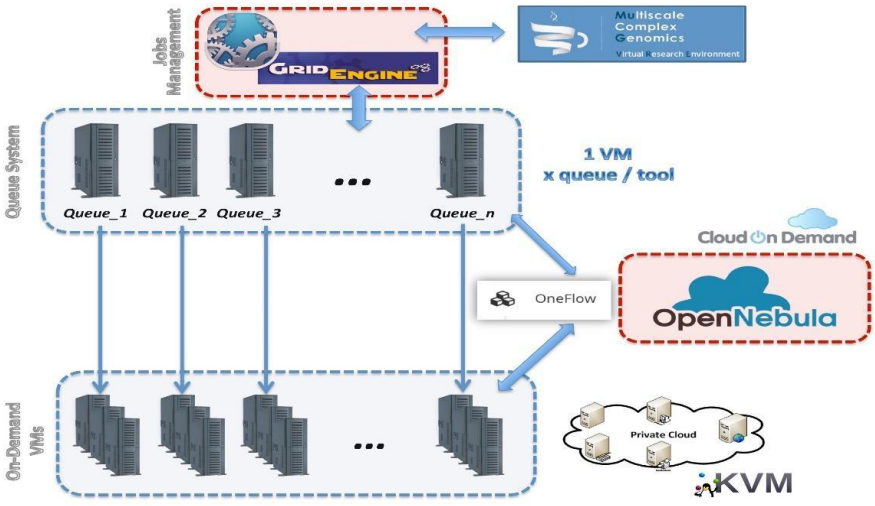


Figure 4.17: Elastic job scheduling engine based on SGE and Oneflow.
Application _1 in encapsulated in VMI _1, which lives in the cloud waiting for a job petition from the SGE master, installed at MuGVRE. At the moment the researcher launches the application _1, the VMI _1 instance receives the commitment and starts working. If more jobs are requested for application _1, VMI _1 will increase its CPU load, which will trigger OneFlow. The latter will instantiate a second instance of VMI _1, so that job petitions can be distributed among the two VMIs _1 and queuing time is reduced.

Figure 4.17 shows a schema of the structure implemented in MuG prototype. In the case of increased demand on a certain application (*i.e.* certain VM), OneFlow instructs OpenNebula to replicate such VM, which is translated into an increased number of hosts available in the queue system for such application. Once the workload for such a group of VMs decreases, OneFlow undeploys them one by one, always keeping at least one instance ready to accept new job petitions. In this way, allocated resources are dynamically and transparently adjusted on-demand.

The following snippet illustrates how SGE detects the VMs hosting the queues. Oneflow deploys and undeploys them as it choice according the workload metrics, while SGE simply distribute job petitions among those hosts available

```
HOSTNAME                ARCH         NCPU  LOAD  MEMTOT  MEMUSE  SWAPTO  SWAPUS
----------------------------------------------------------------------------------
dna-192-168-27-41       lx24-amd64      2 1.86   31.4G       - 1024.0M    5.9M
dna-192-168-27-42       lx24-amd64      2 1.05   31.4G       - 1024.0M    0.86
dna-192-168-27-43       lx24-amd64      2 0.5    31.4G       - 1024.0M    6.8M
dna-192-168-27-44       -               -    -       -       -       -       -
dna-192-168-27-45       -               -    -       -       -       -       -
dna-192-168-27-46       -               -    -       -       -       -       -
nucDyn-192-168-28-11    lx24-amd64      2  0.00   23.6G  138.5M     0.0     0.0
nucDyn-192-168-28-12    -               -    -       -       -       -       -
nucDyn-192-168-28-13    -               -    -       -       -       -       -
[…]
```

Snippet 4.2 Queue hosts (compute VMs) are enabled/disabled by OneFlow.
In red, disabled VMs that will be provisioned on peak demands. dna and nucDyn* are the hosts prepacking 2 tools*

Tool's virtual machines

MuGVRE is designed such that tools' catalog could be extended when relevant new 3D/4D genomics tools or visualizers may appear. **Software developers**, are responsible for building and engrossing the catalog of MuG VMIs MuG Tools include from high computationally demanding applications (*e.g.* molecular dynamics simulations) to high-throughput data analysis applications (like the processing of next-generation sequencing data). One of these tools has been selected as a use case for this dissertation, and it is further discussed at *4.3.4* Use case: Nucleosome Dynamics. Check the complete list of integrated VMIs and the corresponding MuG tools at annex *8.3* MuG tools & visualizers. A VMI can encapsulate more than one MuG tool. This is especially recommended in case of related applications that require a common environment.

The base VMI is prepared here, following a similar procedure to the one already described for transPLANT VMIs (section *4.2.2.2* Tool's virtual machines). It consists of a vanilla Ubuntu 16.04 image, configured to run on PMES-enabled clouds and with all the necessary MuG software stack ready to use:

- COMPSs (see *3.1.3* Job Managers)
- Cloud-init (see *3.1.2.3* Contextualization)
- MG-TOOL (discussed below)

Major differences lay on the contextualization package used, here Cloud-init instead of One-context, and the parameters passed through it, which now allows a job-tailored storage attachment as discussed in the following section focused on data management.

Unlike transPLANT VMIs, the images now include the installation of a platform-specific component, MG-TOOL-API. It works as an adaptor, which does not compromise software portability to other infrastructures, but enhance the interaction with MuG infrastructures.

MG-TOOL API

MuG VMs, as any other compute unit, have a main executable that triggers the whole pipeline, either stand-alone or parallelized with COMPSs. Such an **entry point** has decided to be standardized for all MuG VMIs. With this purpose, a thin python wrapper called MG-TOOL-API[8] is prepared in collaboration with other MuG partners. On one hand, MG-TOOL-API provides a common access interface for MuGVRE to all its VMIs. On the other, it absorbs the unavoidable heterogeneity of the different application's CLIs, which are handled internally and individually as part of the VMI implementation. Thus, as shown in Figure 4.18, MG-TOOL-API works as an **adaptor**. The specifics of the standardized interface is latter discussed when explaining how *4.3.2.3 MuGVRE backend* interacts with MG-TOOL-API on job submission.

MG-TOOL-API is formalized as a MuG tool skeleton with two main classes (Tool and Pipeline) in which the user injects his/her application, extending and coding their own workflow in pure python. MG-TOOL API, designed jointly but implemented by other MuG partners, has a relevant role in the MuG data management plan, as discussed in the following sections. The wrapping library includes methods for:

I. parsing and validating application's arguments and input files received from MuGVRE job petition;
II. easing tool developer task of coding their own pipeline in python, by providing some generic Classes (Pipeline, Tool), skeletons and examples. Some pyCOMPSs ready-to-use skeletons are also prepared for those willing to orchestrate their applications in COMPSs;
III. logging and monitoring to communicate MuGVRE the application progression;
IV. gathering metadata of application's output files

[8] https://github.com/Multiscale-Genomics/mg-tool-api

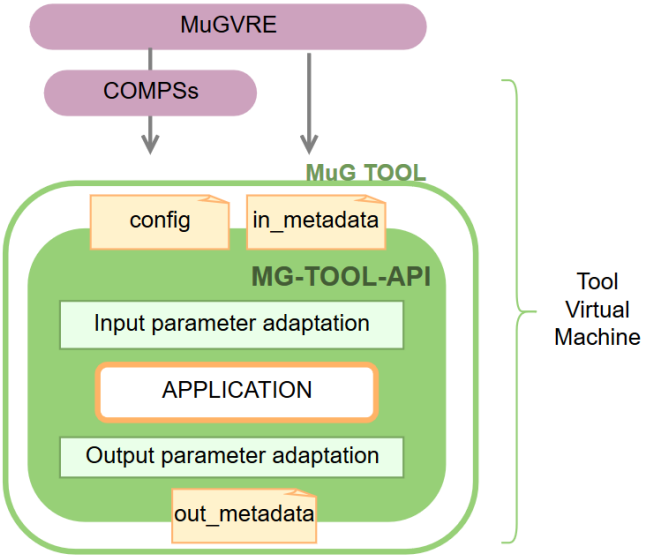


Figure 4.18: MuG Tool virtual machine image.
MG-TOOL-API is a wrapping layer that acts as an adaptor. It enables a rapid prototyping of MuG Tools, and an easy integration into the infrastructure

COMPSs

Just as for the transPLANT cloud, the python binding of COMPSs is used in the MuG infrastructure as a **workflow orchestrator.** It enables the dynamic and transparent deployment of virtual appliances according to the pipeline demands at runtime, providing the system with horizontal elasticity. COMPSs stack is part of MuG VMIs, which as mentioned above, are provided by users themselves like in any other PaaS. Hence, the platform is not aware of whether the application is standalone or COMPSs. If COMPSs creates a virtual cluster, MuGVRE controls the master VM, from where monitoring and logging information is aggregated. Again, as COMPSs workflow management do not interact with MuG central management, VMIs can be configured with any of the two job schedulers enabled in the infrastructure: PMES or SGE.

4.3.2.3 Software as a Service

The infrastructure is consumed in two different ways: as a SaaS or as a PaaS.

- It is offered to the non-programmer 3D/4D genomics researcher as a unified environment of tools, services and data with seamless access to the MuG Computational Infrastructure
- It is offered to software developers as a framework where to rapidly integrate their methods and offer them along other complementary services.

MuGVRE (v. 1.1) is the key component to achieve so.

MuGVRE frontend

MuG Virtual Research Environment (MuGVRE) [227] (MuG Virtual Research Environment (MuGVRE) [227] (Figure 4.19) is the PHP server-side web application that provides a UI that integrates MuG analysis tools, visualizers and data for the end user. Its front-end is conceived as a virtual research environment designed to enhance collaborative research in the 3D/4D genomics community, while behind the scenes, it is responsible to collect high-level deployment requests from the web and coordinate the resource or service deployment over MuG PaaS components.

Figure 4.19: MuGVRE home page
Each box represents one of the tools offered by the platform.

MuGVRE researcher's perspective

MuGVRE[9] aims to offer a responsive, effective, easy-to-use and immersive working environment. The central panel could be either a data-centric workspace that acts as the main landing zone for the users, or an application-centric form allowing to launch specific analysis. It interconnects most of the researcher operations at the platform, which helps covering for the complete user's flow at a computational infrastructure:

I. Authenticate to preserve data across sessions;

[9] http://vre.multiscalegenomics.eu/home/

II. Import the data willing to analyzed in the personal workspace and annotate it;
III. Select the 3D/4D Genomics application of interest;
IV. Tune in tool's arguments, and the job name;
V. Launch the application and monitor the job progression. Output files will eventually appear back at the workspace;
VI. Select the visualizer that can properly display the output data, together with other data if desired;
VII. Feed the results into another tool to further analyze it, or just pack and download the whole project.

Access to MuGVRE is open and does not require registration. Yet, an authentication system is put in place so that registered users can easily regain access to their projects. Unregistered users only can access to the previous session using a recovering link.

Once logged in, the first step is **loading the data** to work with, and the very second step, **annotate** it. Imports might come from researcher's data (either by file streaming on the browser or asynchronously on a given URL), or from one of the embedded public repositories upon one-click (*i.e.* ArrayExpress [9], BigNASim (*8.7* Publications) [228]). Loaded files need to be annotated, as some MuGVRE functionalities are precisely based on file semantic and descriptive metadata. For instance, such curated metadata enables the interoperability among MuGVRE tools and visualizers, and permits to contextualize and guide the researcher offering adapted toolkits, dynamic available operations, etc. Minimal annotation includes file format (*e.g.* FASTQ) and data types (*e.g.* HiC-seq reads) (supported data type and formats listed at annex *8.4.1* Data Model: "File").

All uploaded data is accessible and navigable at the user's **personal workspace**. It is based on a filesystem-based layout (Figure 4.20), where data is centralized and administered regardless it is uploaded by the user, it comes from integrated repositories, or has been produced at the platform by running processes. User's workspace is organized into scientific projects. Each of these projects holds an independent workspace, that in turn is organized in folders:

- Uploads/: uploaded data,
- Repository/: data obtained from public repositories
- "Run" folders: results for pipelines and analyses

File lists can be filtered by any of row fields (*i.e.* name, format, data type, or project) as well as a tool-based filter that select only the valid input files for a given tool. Upon a file (or a group of them), three interactive toolkits display the valid operations:

- File toolkit: Rerun job, download data/folder, edit metadata, delete, pack and compress, rename, move.

- Visualization toolkit: Available visualizers supporting the selected combination of file formats and types.
- Tools toolkit: Available tools supporting the selected combination of file formats and types.

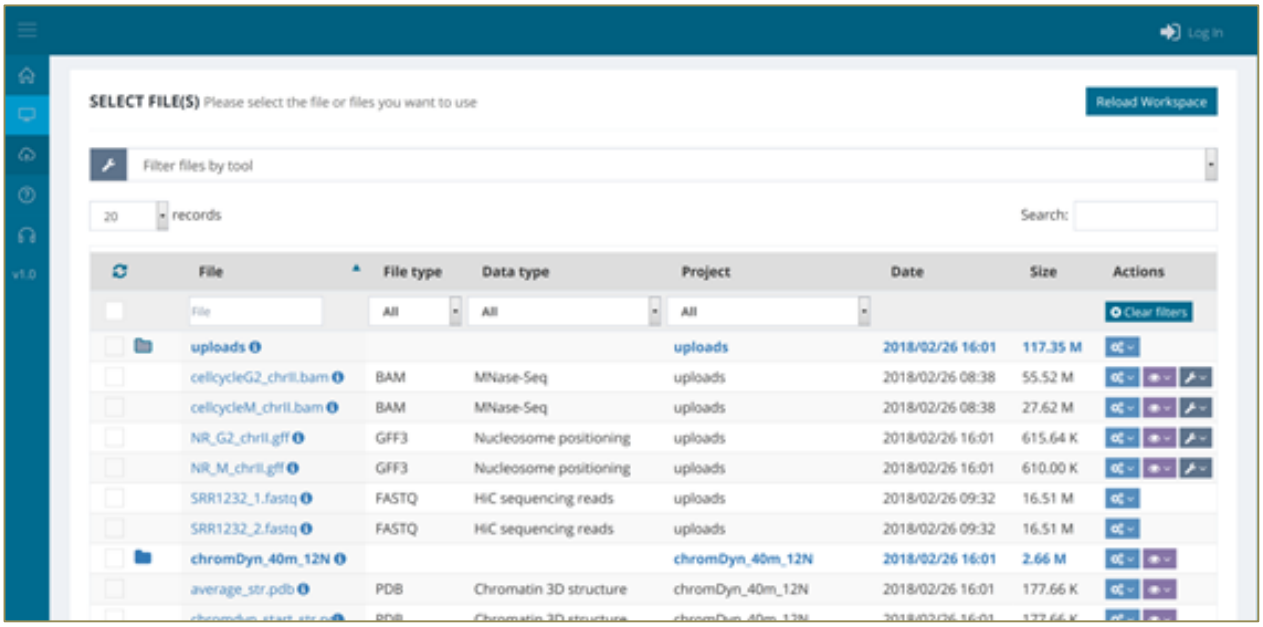


Figure 4.20: MuGVRE workspace

Alternatively, the operation's selection (tool or visualizer) can be performed directly. MuGVRE offers a "Launch" section where the user **browses the catalog of tools**, organized as a searchable table with friendly descriptions, ownership, keywords and supported operations. As usual, upon tool selection, the user fills in the particular application parameters. In case input files have not been selected at the workspace, an additional dialog offers a filtered list of files across workspaces suitable for the given tool input file. This mode is more convenient for non-experienced users, as permits an exploration of all the available tools.

Once launched, user instructions are mapped into effective **job executions**, which are monitored and debugged on the workspace. Each run creates a new "Run" folder that contains all the data belonging to such execution (*i.e.* log file, and job-related metadata), along with its eventual output files. These are automatically annotated (data type and format) so are eligible for visualizers, or other analysis tools.

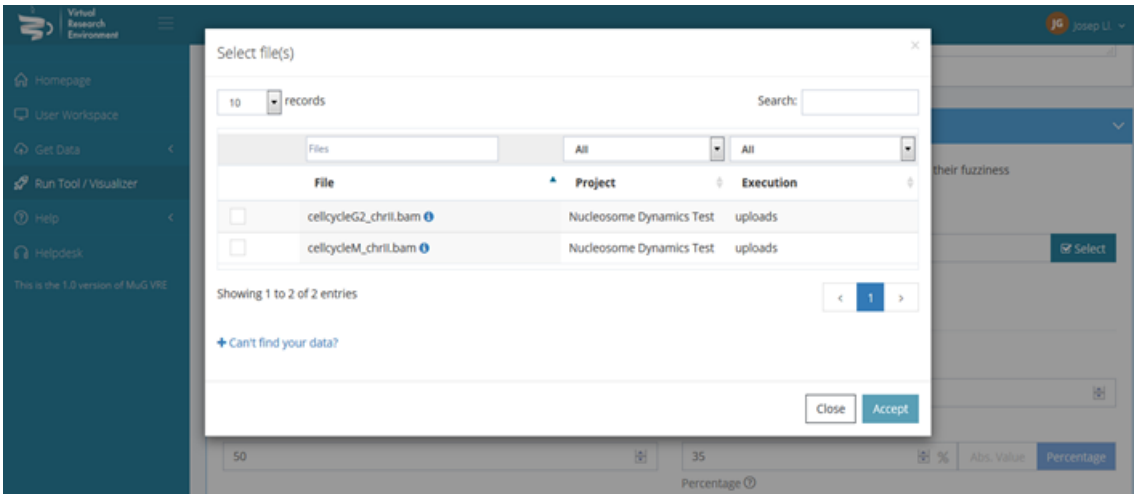


Figure 4.21: Selection of input data after tool selection.
In the example, MuGVRE is selecting all user's file of type "MNase-seq data" and format BAM.

Currently, MuGVRE integrates three different **visualizers** for sequence browsing and visualization of 3D structures, molecular trajectories or chromatin assembly models (check the list at annex *8.3* MuG tools & visualizers). Users choose from their workspaces the different files of interest, even repository's data, and feed it into one of the visualizers.

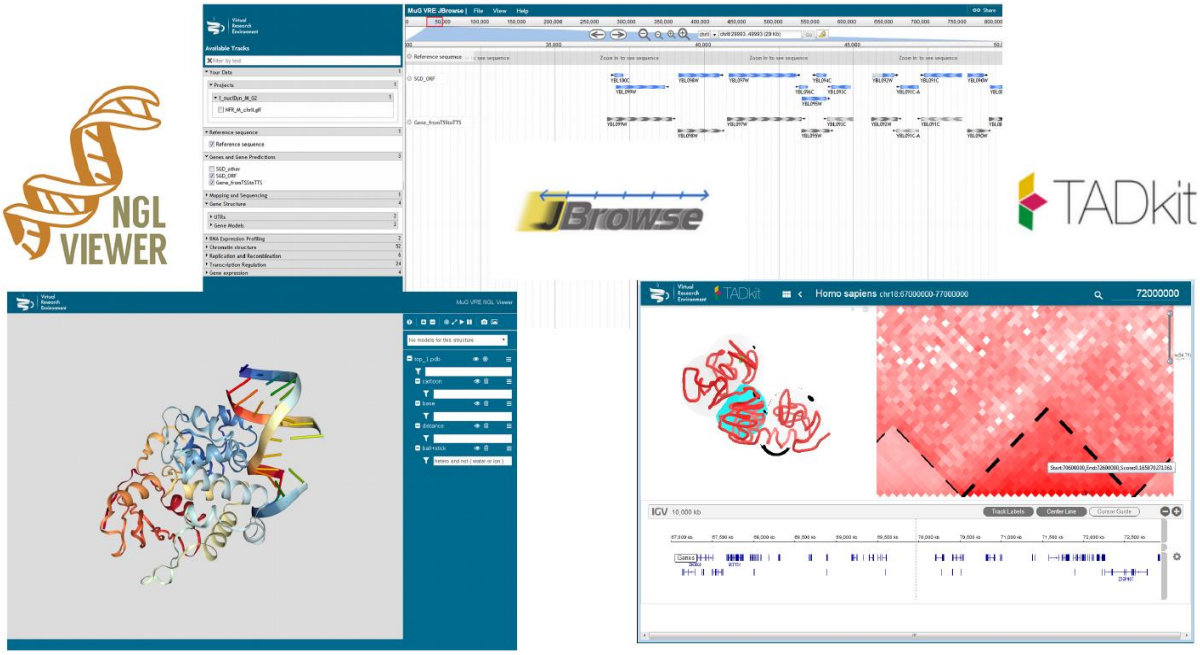


Figure 4.22: Snapshots of the three MuGVRE visualizers

Additionally, analysis tools may offer a **custom viewer** for better displaying their output data. Unlike generic visualizers, these displayers are not meant to interactively analyze and combine data across the platform, but to show the data of a particular run, as to include

statistical reporting, specific plots or applets, etc. (Figure 4.23). In fact, they are provided and plugged-in by software developers along with their methods.

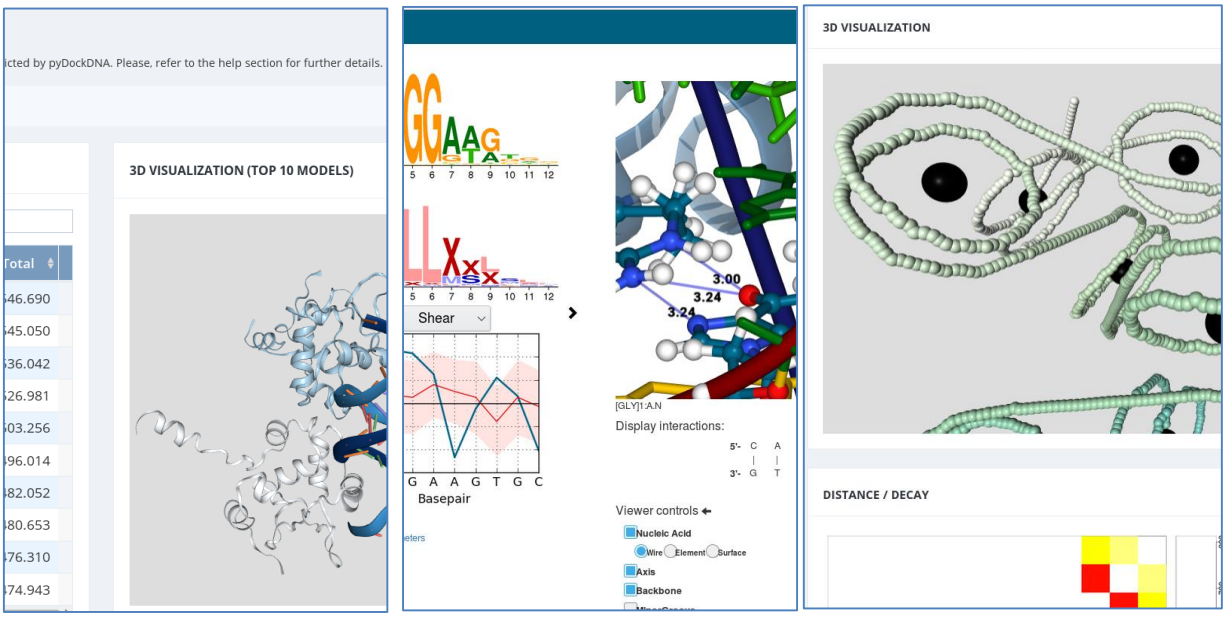


Figure 4.23 : Example of custom visualizations for three different analysis tools. From left to right, "3D-consensus", "MC-DNA" and "pyDockDNA".

To **support researchers** on such an integrative framework, the virtual research environment has made special emphasis on providing documentation pages (editable online for tool developers), abundant help tips, dynamic toolkits, interactive monitoring, verbose error and progress reporting, etc. - without compromising platform's flexibility. Moreover, usability is enhanced by supplying sample dataset for each tool (*i.e.* input and results examples) so that demonstration runs are available on one click. On top of that, a helpdesk service provides a ticketing system to address general MuGVRE petitions (*e.g.* disk quota, bring-your-own-tool support, IT general issues), and analyses related concerns, directly responded by the tool owners.

MuGVRE Software developers' perspective

MuGVRE is not only meant to interact with the researcher, but also with those software developers willing to integrate their methods at the platform. Although the integration *per se* of new tools – *i.e.* new VMs –, involve offline configurations at the IaaS level, MuGVRE includes some panels and utilities to assist in the development process and keep an open channel between the developer and MuG support team.

Users entitled as "tool's developers" (permissions are requested online and require Administrators' approval) have access to the panel controlling **tool's administration** (Figure 4.24). They can manage their own Tools (available statuses: active / inactive / under test / coming soon / obsolete) and check their usage statistics.

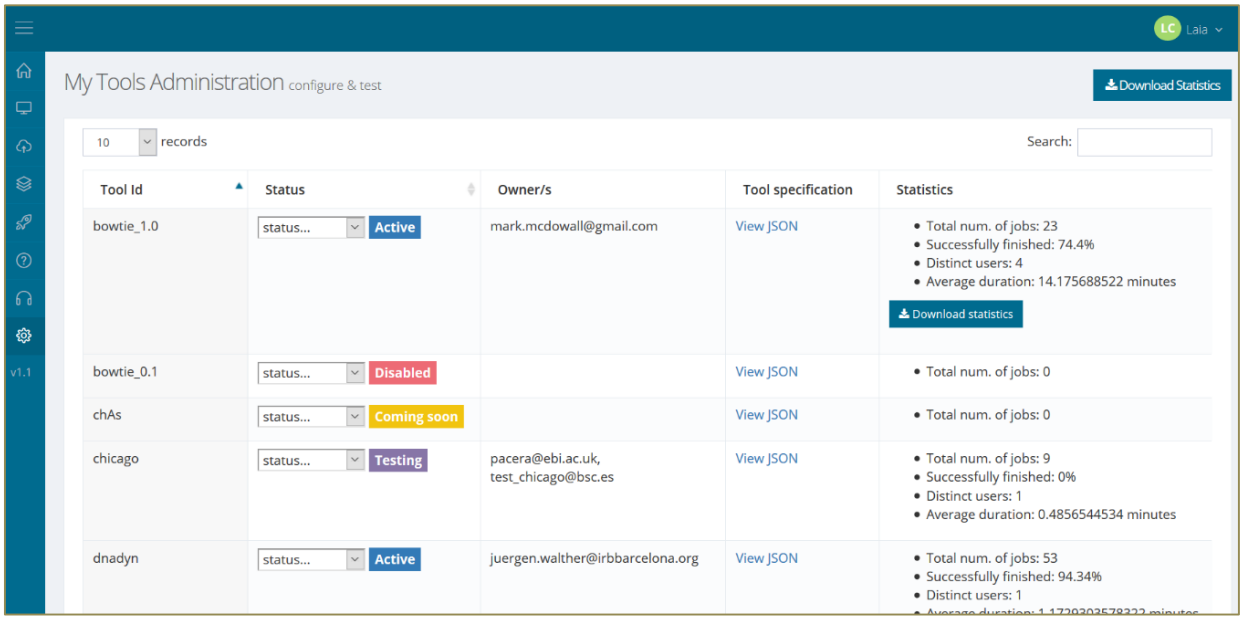


Figure 4.24: MuGVRE Tool's administration panel

Additionally, software developers also have special views designed to **facilitate the testing and debugging** of new tools. For instance, raw view of job auxiliary files is accessible at the workspace per each run, as well as the metadata for each file, which is displayed in the native JSON format as stored in the database (Figure 4.25). Also, online editors for documentation pages that are expected to be filled in when plugging in a new tool in the system.

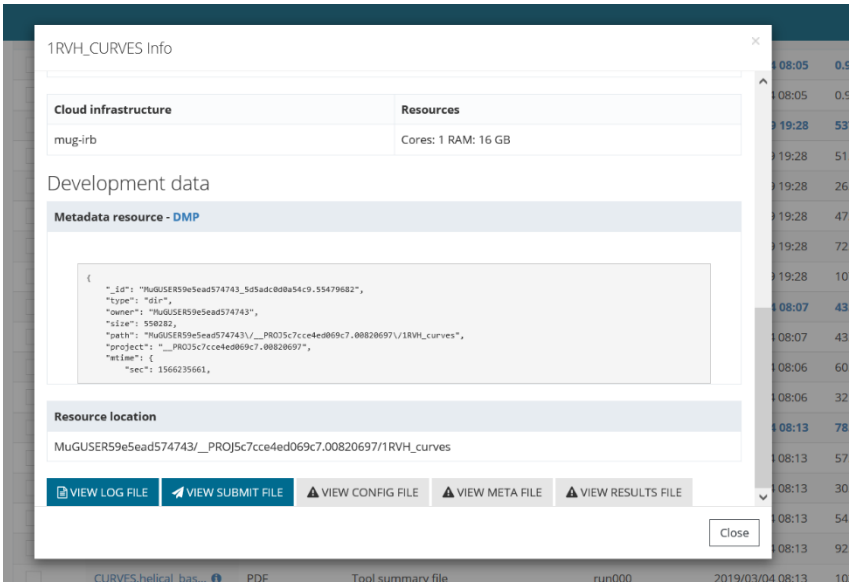


Figure 4.25: Pop-up windows display information about a finished job.
Tool Developers have extended data on the run, like internal identifier, associated files, etc.

And last but not least, a complete workspace (Figure 4.26) is available for assisting during Tool's submission procedure (protocol more detailed in *4.3.3* Data management).

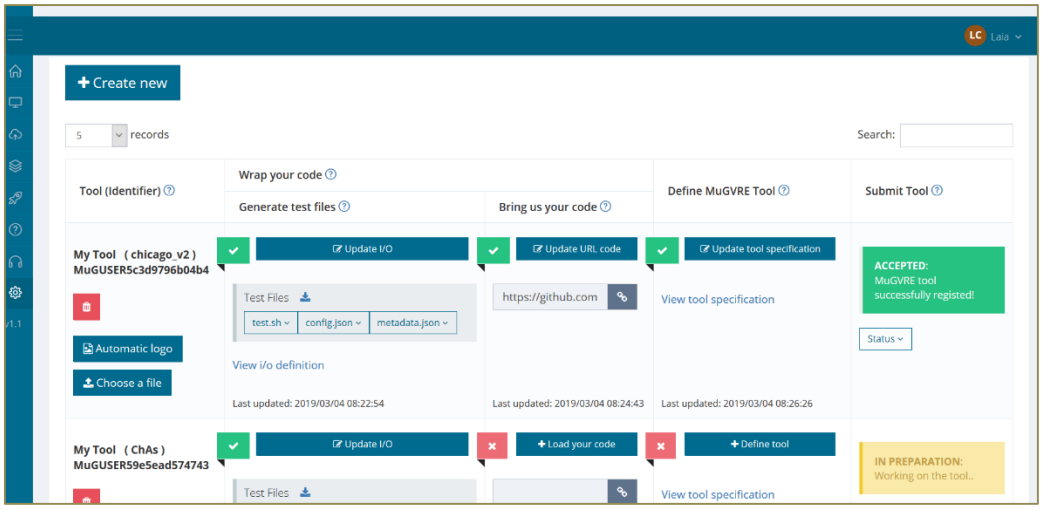


Figure 4.26: MuGVRE Developer's workspace

A complete protocol has been set up to allow the submission of new tools into MuGVRE. The whole process can be divided in two main steps, a wrapping process for preparing the application to be virtualized, and the actual integration of such code into the infrastructure.

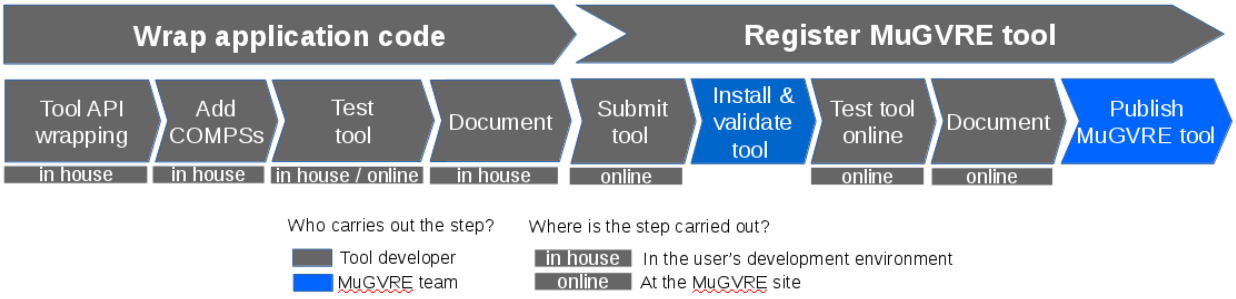


Figure 4.27: New "Tools" integration protocol.
The process involves not only the tool developer, but the MuGVRE support team.

Application wrapping is based on the MG-TOOL. The python library is formalized as a tool skeleton on top of which user adapts their application. The user clones the MG-TOOL repository in their own development environment, and fill in the skeleton according to its pipeline's requirements. If parallelization is to be enabled, PyCOMPSs decorators should also be added here. Once the Tool is set up, we suggest a functional testing by which a MuGVRE execution is emulated at user's in-house installation. On the MUGVRE web interface (developer's panel, Figure 4.26), a testing set job auxiliary files (config.json, in_metadata.json) are created with the developer's help, who defines input, argument and outputs for the new tool. Test files can be downloaded to be used locally. If the in-house test is passed, the code is ready to be made available on a public software repository, as MuG coding guidelines suggest.

The second part essentially implies the **registration and documentation** of such code on the MuGVRE system, which requires to establish a dialog with MuGVRE support team.

Through the developer's workspace, the user provides the code location (URL), tool's descriptive metadata (*i.e.* title, keywords, etc.), deployment details (*i.e.* CPUs, memory, main executable, etc.), logo images, etc. Eventually, the tool can be "Submitted" with a click. At this point, the ticketing system opens a communication channel with the user, and MuGVRE support team is made aware of the Tool proposal. The submission is evaluated and validated by MuGVRE team, who deploys the MuG base VMI including the new code at the MuG cloud. The new tool is activated under the testing mode, and tool-related web pages are generated automatically based on the Tool object. Finally, the tool is debugged, refined and tested on the platform, example data under 'Get Sample Data' menu is prepared, and tool help pages are prepared through the MuGVRE online markdown editor.

The whole integration protocol is fully described and documented step by step on the web[10], including training material and tutorials[11].

MuGVRE Administrators' perspective

Finally, MuGVRE provides some extra sections dedicated to support not the researchers but platform's administrators. Administrators have **privileged access** to a series of operations and views, like (i) the dashboard with platform's metrics (*e.g.* total storage in use), (ii) user's administration to control their connections and profiles (*e.g.* manage user's role, assign tools' ownership), (iii) job management of the whole platform, (iv) run's accounting and logging, and (v) tool's management (*e.g.* activation/deactivation) with access to usage statistics per tool.

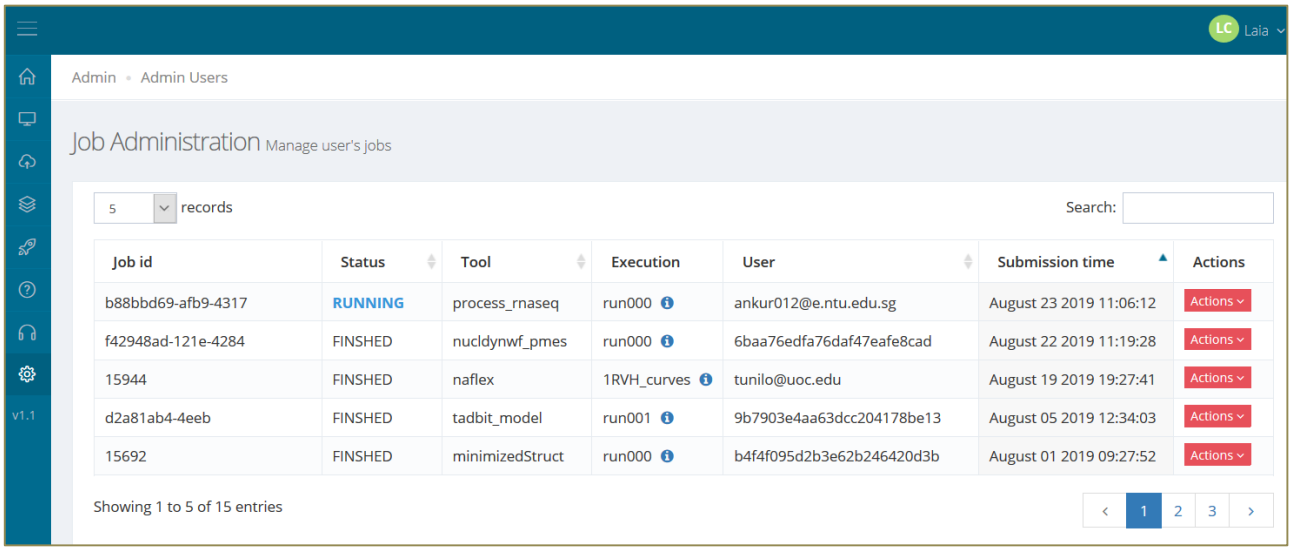


Figure 4.28: MuGVRE panel for administering platform's job

[10] https://www.multiscalegenomics.eu/MuGVRE/instructions/
[11] https://www.multiscalegenomics.eu/MuGVRE/training/

MuGVRE backend

MuGVRE collects high-level deployment requests from MuGVRE fronted and coordinates the resource and service deployments applications. It is based on a **Mongodb database** that maintains the operational data and metadata regarding installed tools, visualizers, users and user's files. Figure 4.29 describes the main collections comprised in the MuGVRE model. Each collection contains a list of records (BSONs, *i.e.* binary JSON documents) that may feature varying sets of fields, with different types for each field (more details at 3.1.1.2 Mongodb***Error! Reference source not found.***). Several PHP libraries have been prepared to populate the DB with instances of tools, users, files or jobs objects. MuGVRE source code is available at the project's GIT repository[12], together with examples and documentation.

Users are authenticated using a central server who manage the credentials, as further discussed in the following section. Usernames have files associated. Each of the **files and folders** displayed at the user's personal workspace corresponds to an entry in the database ("Files" collection), which is a one-to-one match of the data in the disk. The record stores the operational metadata of the file, including a MuG file identifier and the file location, currently resolved as a local file (or directory) path. Like in file systems, "File's" records are a hierarchical system, with cross-references to parent and children records. When a file is loaded, it becomes registered at the "File" collection, and once the user fills in its descriptive and semantic metadata, it is stored in the "File Metadata" collection. Files are fed into and out of the jobs.

[12] https://github.com/Multiscale-Genomics/VRE

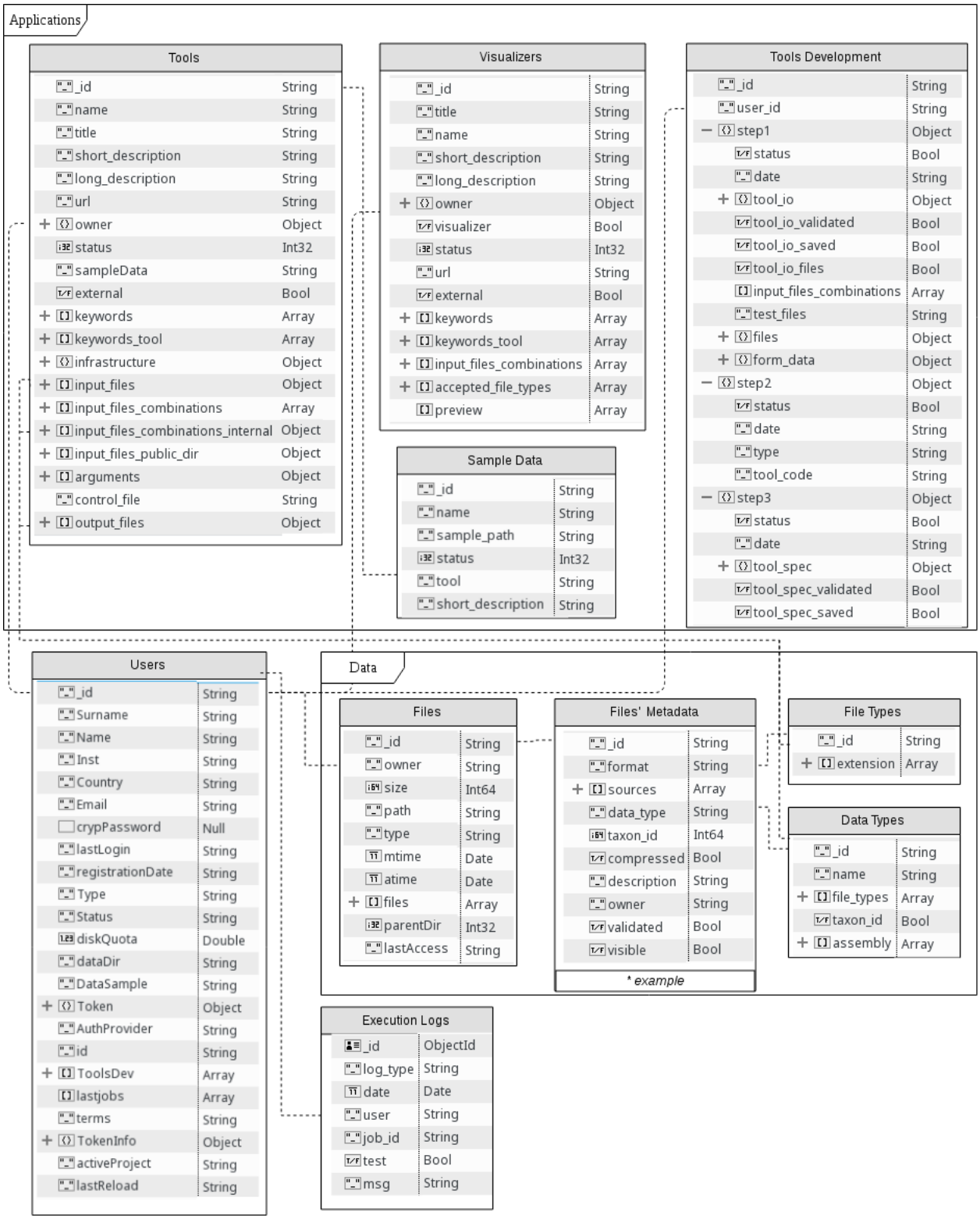


Figure 4.29: Main Mongodb collections of the MuGVRE database.
Records in each collection may present varying sets of fields, so object models described here are indicative, especially in the case of the catchall collection "File Metadata". Dashed lines indicate relationships among collection fields - controlled at the

Tool execution lifecycle

MuGVRE is responsible to compose an operable job for PaaS components and remotely invoke it. Figure 4.30 shows a top-level flow diagram on the actions carried out when the researcher clicks the "Run Job" button, after defining the run parameters for a given tool.

This section explains in detail each of the boxed stages depicted in the diagram.

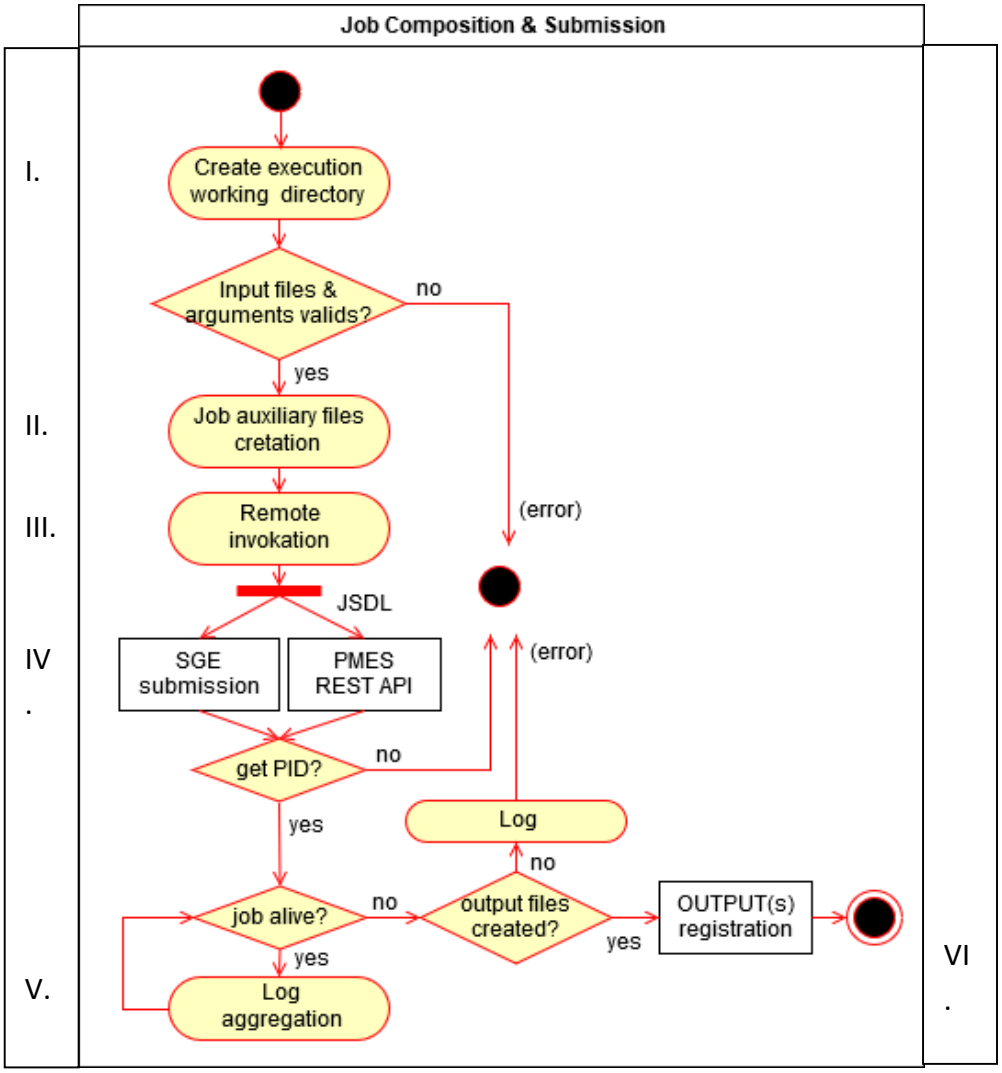


Figure 4.30: MuGVRE diagram flow for job processing

I. When the researcher, at the frontend, starts a Tool execution, behind, a new job management process is triggered and registered. Firstly, a "run" folder, named according to user's preferences, is created in their private workspace, registered in the database, and set as the **execution working directory**. If the action would fail – *e.g.* the directory becomes not accessible – an error is raised until the web frontend.

II. A set of **auxiliary files** are prepared on the fly containing all the application-specific details

- "config.json": contains the values that application's arguments and input files take for that particular run.
- "input_metadata.json": contains all the metadata associated with the selected input files as stored in the metadata database.

As any application-specific information is enclosed into these auxiliary files, MuGVRE can call all the Tools using a single and generic interface. In this way, there is no need to learn how to build the particular CLI of each of the encapsulated applications, such burden is managed internally by MG-TOOL-API in each Tool implementation at the VM level.

III. MuGVRE performs the **remote job invocation** submitting the following agreed BASH command:

```
# (A) # For standalone applications

[APPLICATION_EXECUTABLE]                              \
        --config [config.json]                       \  # IN
        --in_metadata [input_metadata.json]          \  # IN
        --log_file [log.txt]                         \  # OUT
        --out_metadata [results.json]                \  # OUT
```

```
# (B) # For COMPSs applications
runcompss                                               \ # COMPSs runtime
    --summary                                           \
    --jvm_workers_opts="-Xms5120m,-Xmx5120m,-Xmn1024m" \
    --base_log_dir = [WORKING_DIR]                      \
    --lang = python                                     \
    [APPLICATION_EXECUTABLE]                           \
        --config [config.json]                       \  # IN
        --in_metadata [input_metadata.json]          \  # IN
        --log_file [log.txt]                         \  # OUT
        --out_metadata [results.json]                \  # OUT
```

Snippet 4.3: Process submitted by MuGVRE to the corresponding Tool VM.
The executable always responds to usage (A), for standalone applications, or (B) for runs configured with COMPSs.

[Application_Executable] corresponds to the script provided by the software developer that acts as an **adaptor**: interprets MuGVRE job invocation and passes this information to the actual wrapped application. MG-TOOL-API helps developers to build such an adaptor. The arguments are basically the job auxiliary files above mentioned, as well as information about where MuGVRE can find application's logs and output files. As such, application's CLI heterogeneity is handled by software developers when coding their own wrapping script, a strategy that provides total transparency and flexibility.

IV. Job execution continues by submitting the referred command to one of the two schedulers: **PMES or SGE**. The task implies crafting valid job compositions, considering aspects as the following:

Management of application's input requirements, in terms of input files and arguments. parameters incompatibility, dependency or compulsory usage. Initial validation of type and format.

- tailoring and dimensioning job petition according to the defined resource requirements: not only configuring job's cores and memory, but ensuring such requirements could be met by the target IaaS (*e.g.* listing available OCCI resource templates).
- ensuring data availability to the VMs: not only researcher's data but also reference data, as further discussed in the following section.

If SGE is enabled, a **queue submit script** exemplified at Snippet 4.4 is submitted to the virtual machine where the relevant Tool is installed.

```
#!/bin/bash
# Generated by MuGVRE

#$ -q naflex.queue
#$ -N vre2598_naflex
#$ -o /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000/.tool.log
#$ -e /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000/.tool.err

# Moving to working directory
cd /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000

# Running naflex tool ...
echo '# Start time:' $(date)
/home/NAFlex/NAFlex_wrapper.py  \
   --config       /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000/.config.json  \
   --in_metadata  /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000/.in_metadata.json \
   --out_metadata /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000/.results.json \
   --log_file     /shared/MuG/userdata/MuGUSER59e5ead574743/__PROJ5c7cce4ed06/run000/.tool.log \

echo '# End time:' $(date)
```

Snippet 4.4: BASH script prepared by MUGVRE to run a standalone application
Here, NAFlex via SGE

If PMES is enabled, MuGVRE builds at runtime a job definition according to **JSDL specifications** (example at Snippet 4.5) and POST such data to the PMES "createActivity" REST endpoint. Data includes the identifier of the VMI to be deployed, along with other context parameters.

```
{
  "jobName": "G1-S",
  "compssWorkingDir": "/shared/MuG/userdata/__PROJ5c93530a68/G1-S",
  "wallTime": 1440,
  "memory": 50,
  "cores": 2,
  "minimumVMs": 1,
  "maximumVMs": 1,
  "limitVMs": 1,
  "initialVMs": 1,
  "disk": 1.0,
  "inputPaths": [],
  "outputPaths": [],
  "infrastructure": "mug-irb",
  "mountPoints": [
       {
          "target": "/shared/MuG/userdata",
          "device": "/services/MMB/lab/MuG/userdata/MuGUSER5af1b916c27a2",
          "permissions": "rw"
       },
       {
          "target": "/shared/MuG/MuG_public",
          "device": "/services/MMB/lab/MuG/MuG_public",
          "permissions": "r"
       }
   ],
  "numNodes": 1,
  "user": {
      "username": "vre04385",
      "credentials": {
          "pem": "/home/pmes/pmes.pem",
          "key": "/home/pmes/pmes.key",
          "uid": 33,
          "gid": 33,
       }
   },
  "img": {
      "imageName": "uuid_nucldynmgtoolapi_48",
      "imageType": "small-max"
   },
  "app": {
      "name": "nucleosomedynamics",
      "interpreter": "python3",
      "target": "/home/pmes/nucleServ",
      "source": "nucleosome_dynamics.py",
      "args": {
        "config":      "/shared/MuG/userdata/__PROJ5c93530a68/G1-S/.config.json",
        "in_metadata": "/shared/MuG/userdata/__PROJ5c93530a68/G1-S/.in_metadata.json",
        "out_metadata":"/shared/MuG/userdata/__PROJ5c93530a68/G1-S/.results.json"
       },
      "type": "Single"
   },
  "compssLogDir": "/shared/MuG/userdata/__PROJ5c93530a68/G1-S"
}
```

Job resources & definition

Data management

Virtual user setting

VMI template & OCCI resource

MuG universal CLI

Snippet 4.5: MUGVRE job petition for PMES createActivity endpoint. Specification : JSDL

V. Once the job is sent, MUGVRE **monitors** the job and reports its progress pulling the tool raw log file, which is processed to be nicely displayed on the web. The whole execution cycle is controlled via libraries designed to interact with PMES (via REST) or SGE (via local network).

VI. After the job's completion, the main task of MuGVRE is to **register the output files** created during the execution in the "Files" and "Files Metadata" Mongodb collections. This way, they become available in the workspace and eligible for new runs. Apart from receiving task managers' error, MuGVRE detects that an error has occurred during the remote execution based on the fulfillment of output file requirements, which are set up as required or optional during the tool registration process. If a required output file is missing on job completion, MuGVRE considers that an **error** has occurred, and indicates so publishing on the workspace a file with the log information gathered along the execution.

4.3.2.4 Authentication and authorization

Data privacy is maintained using the INB's (National Institute of Bioinformatics) authentication and authorization server[13] to securely handle user access and Web communications. Keycloak implements OpenID Connect on top of the **OAuth2** protocol, a token-based authentication flow widely used for the REST services (implicit flow) and Web applications (authorization code flow). On the web, registered users are redirected to the authorization server following the authorization code grant, which accepts username/password credentials backed up by a local LDAP database. Alternatively, the server may proxy to third-party identity providers, including social IDs like Google or ORCID, as well as the European life research AAI facilitated by ELIXIR.

MuG components like MuGVRE and data MuG RESTful services (*e.g.* DM-API [229]) have integrated OAuth2 clients in order to interact with this MuG authentication server. MuGVRE uses it not only to authenticate their users, but also to display the access token (automatically refreshed) on the web (Figure 4.31). In this way, researchers can consume MuG APIs directly, outside MuGVRE web framework.

[13] http://inb.bsc.es/auth

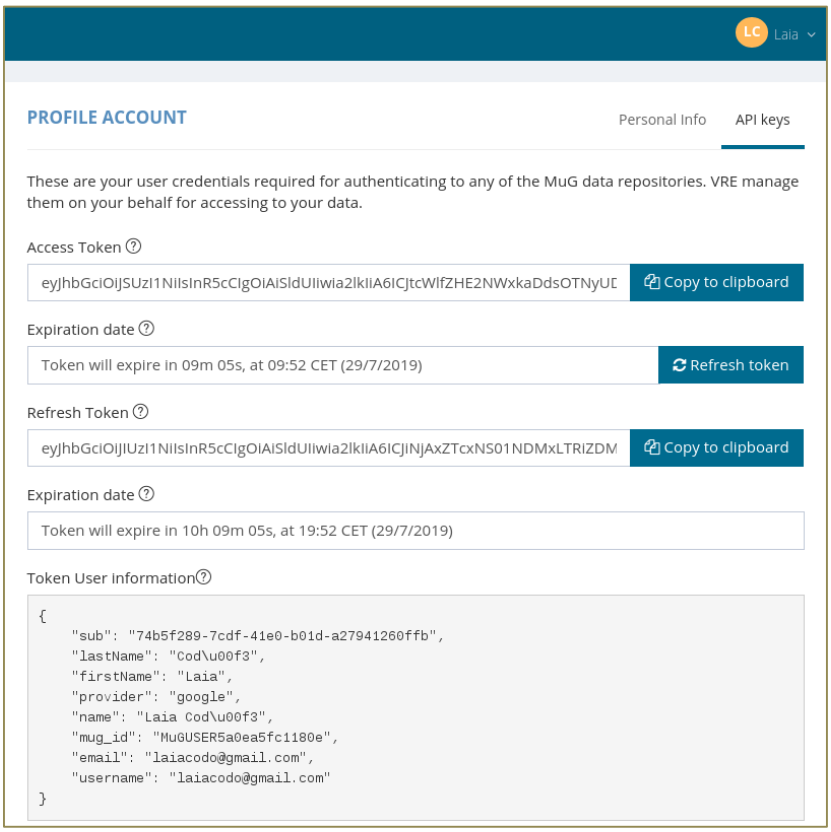


Figure 4.31: Users' access Tokens are displayed at MuGVRE

Regarding the **authorization**, MuGVRE has three types of user accounts in order to have a fine grain control of the authorization roles and permissions over resources (*i.e.* files, jobs, tools):

- Users: own and administrate their data on their personal workspace. They correspond to either registered user accounts, or anonymous records if they have not been logged in. Available functionalities are the same for both user's modes, with the only difference that some tools affected with licensing issues might not be eligible by non-registered users, as MuG agreement terms are only accepted during the registration procedure.
- Tool developers: own data, but also tools. They are authorized to submit new tool petitions, and manager their already integrated tools. They decide over its accessibility and visibility, its configuration, etc. They access their logs, statistics, and other debugging information.
- Administrators: full access to all user accounts. They administrate accounts, their runs and their tools.

As access controls are implemented at the application level, KeyCloak accounts are mapped into MuGVRE accounts, which have MuG identifiers and distinct authorization privileges and roles associated with them. MuGVRE keeps the two systems synchronized by injecting the identifier ("mug_id") into the Authentication Server every time a new user is registered.

Doing so, we manage to securely circulate the “mug_id” as part of the user token among the different MuG services.

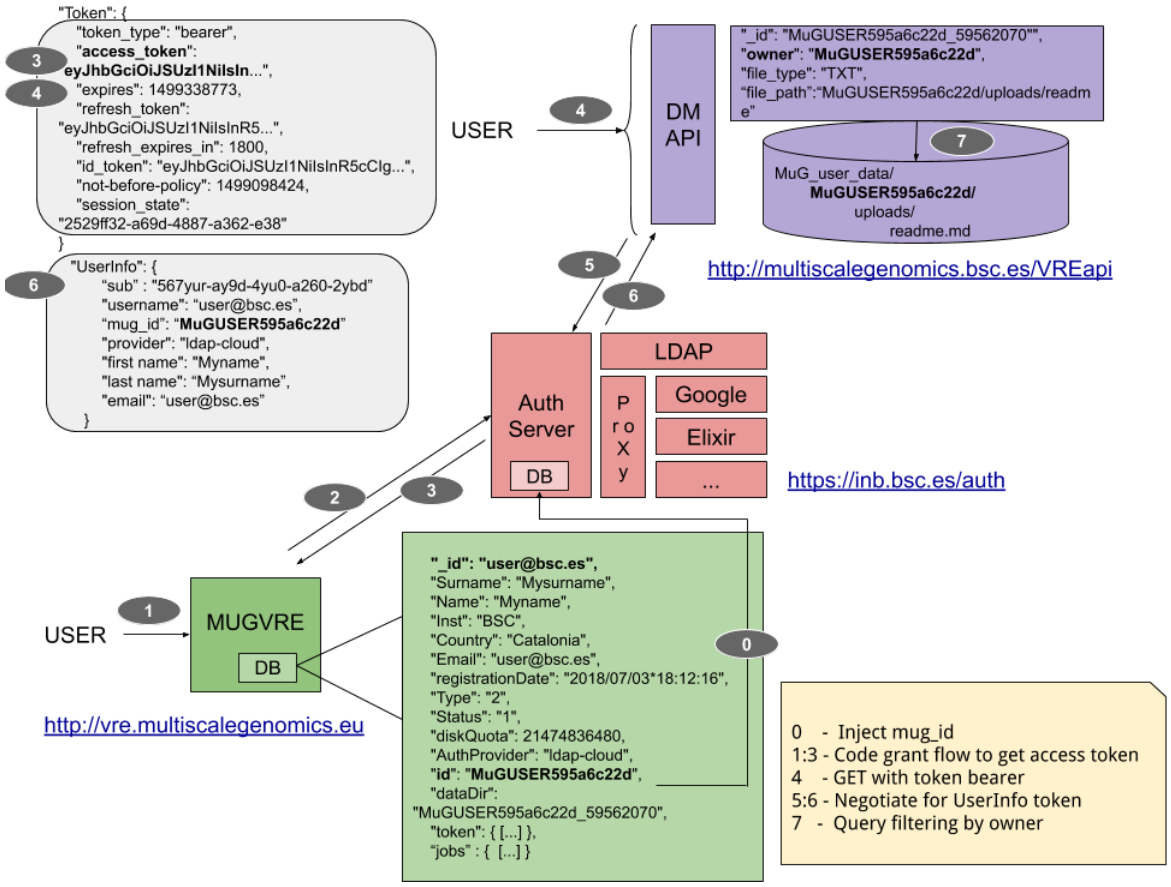


Figure 4.32: MuG token-based authentication across several MuG services

4.3.3 Data management

MuG’s data repository stores both data related to the project, user workspaces and metadata, as well as pointers to relevant data stored in public repositories, either produced by MuG or by reference providers. An accurate management permits to import user’s and reference data in the platform, as well as to keep it private and isolated in the dynamically created virtual executions. Moreover, a metadata-driven system enables not only enhanced user experience but also the automation of the application’s management.

4.3.3.1 User’s Data

Researchers import their data into the platform via HTTP(s) or FTP(s) using the MuGVRE frontend. Either coming from a local user’s file or a public URL (open or protected), an asynchronous data load is initiated using a CURL-based job, visible at user’s workspace. Like for transPLANT public data, user files are saved in a block storage accessible on the cloud

network both by MuGVRE application and the rest of VMs that undergo job executions. Thus, a unique **NAS** per IaaS is the basis of the MuG data cloud - user's data is staged in at the moment user uploads it on the UI.

Analysis tools grant access to user's data differently depending on the job manager that leads to the execution. If SGE is configured for such a tool, the virtual appliance corresponds to a set of long-live VM(s) which trivially access via NFS to the whole MuG shared storage on an auto-mount basis. If PMES is enabled, an extra layer of data privacy is granted, as the only piece of data mounted by contextualization is the one belonging to the particular user launching the job (Figure 4.33). As shown in the JSDL example of Snippet 4.5, on each run, different "mount_points" can be set, and for user's data, the device targets a user's particular directory (*i.e.* /services/[…]/userdata/MuGUSER5af1b916c27a2).

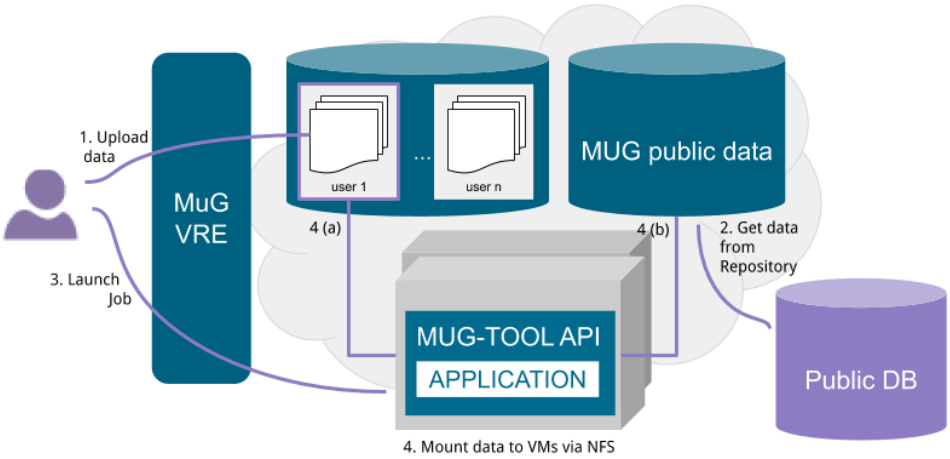


Figure 4.33: PMES data management based on a shared NAS.
Pre-emptive VM's mounts only the data belonging to the user that launched the execution

User's data in disk is organized following the "User > Project > Workspace" hierarchy:

```
MuG_userData/
+ --- user_id1/
|     + --- project_id1/
|     |     + --- Uploads/                # Contains data uploaded by user
|     |     |     --- MyInput1.txt
|     |     |     --- MyInput2.txt
|     |     + --- Repository/             # Contains data sourced from public repositories
|     |     |     --- SRA0000.fastq
|     |     + --- Run000/                 # Contains data generated during a job execution
|     |     |     --- MyOutput1.txt
|     |     |     --- MyOutput2.txt
|     |     |     --- .tool.log
|     |     |     --- .tool.err
|     |     |     --- .in_metadata.json
|     |     |     --- .config.json
|     |     |     --- .results.json
|     + --- project_id2/
```

Snippet 4.6: Directories hierarchy for MuGVRE user's data

MuGVRE sets a working directory - “Run folder” (*e.g.* “Run000” in the previous snippet) – for each job submission, and it does so on the shared disk, so that virtual appliance can directly access it. As shown, the data structure inside a workspace is quite rigid, with thematic directories for “Uploads”, “Repository” data and n “Run” folders. User can create n workspaces, called “Projects”.

Remotely, MuG data can also be accessed using a simple RESTful API[14], in case data is willing to be accessed outside MuGVRE premises. It is implemented as a simple HTTP(S) interface written in PHP using the PHP Slim MVC framework. The API features two routes secured using the MuG central authentication server that returns the actual data given a file identifier. It complements MuG APIs, implemented outside BSC, which expose MuGVRE files’ metadata and offer filtering and navigation options.

4.3.3.2 Reference Data

MuG public data is treated very similarly to transPLANT public data. The globally accessible NAS is the chosen strategy, with the restriction of being accessed under read-only mode.

However, unlike transPLANT, the UI offers a selection of 3D/4D genomics relevant datasets that user can, i) browse the available data in an integrated way, ii) use in MuGVRE analysis infrastructure, and iii) download in the appropriate formats for in-house further analysis. Such repositories currently include:

- ArrayExpress [9]: a selection of HiC, MNase-seq, ATAC-seq and sequencing experiments related to 3D/4D genomics.
- BigNASim (*8.7* Publications) [228]: a selection of nucleic acid trajectories (molecular and coarse-grain) focused mainly on chromatin modeling transcription factors.
- Senescence: chromatin assembly models of HiC senescence human data derived from MUG’s use cases

Data is fetched from the original repository over the network (using annotated URLs or provided REST APIs) and transferred to the MuG cloud. MuGVRE tries to automatically infer and register the minimal descriptive metadata for the sourced files (*e.g.* based on the file extension) before displaying them on the user’s workspace, under the “Repository” directory.

[14] https://github.com/Multiscale-Genomics/VRE_data_api

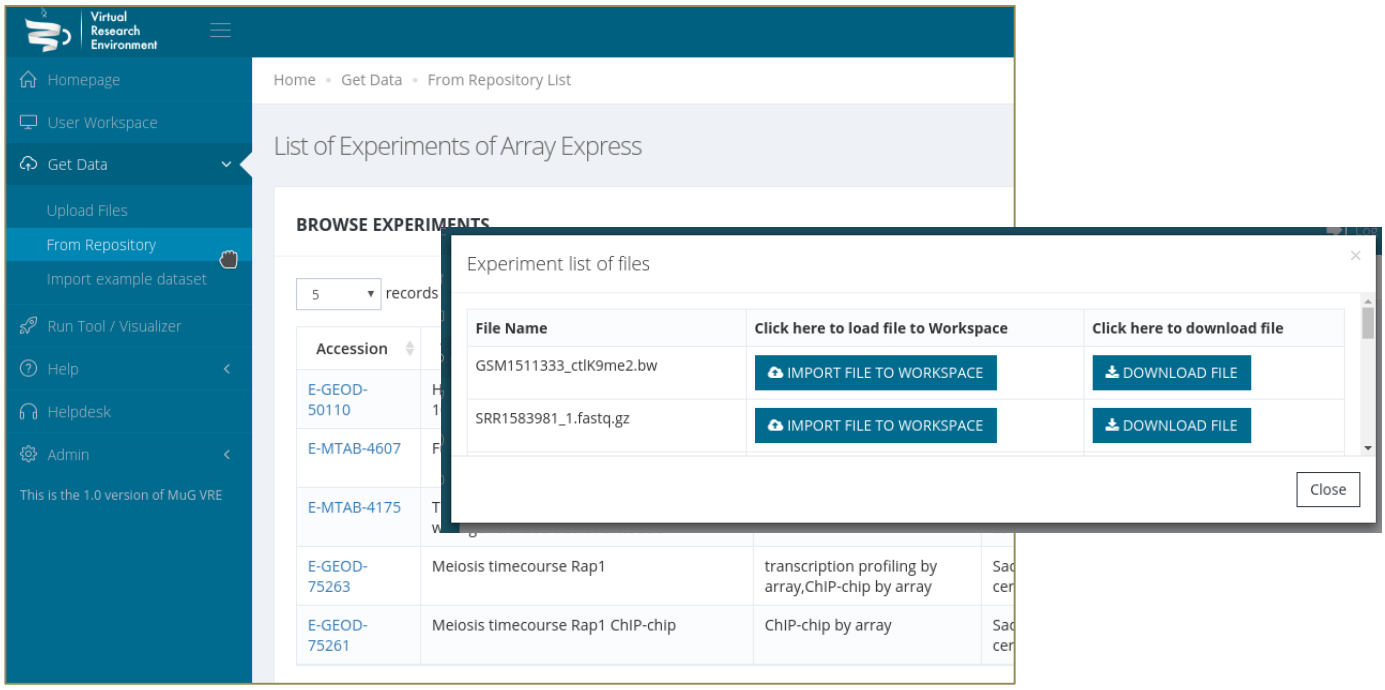


Figure 4.34: Browsing and selection of data from ArrayExpress.

4.3.3.3 Pilot for distributed data system

Both, public and user's data, are transparently accessible over the cloud VLAN, but not from outside. It limits the adoption of cloud computing federations, because even if PMES enables remote executions across multiple clouds, data dependencies would hamper the full implementation. Hence, available multi-institution storage solutions are explored with the aim to integrate the most convenient one into the current MuG DMP – preserving as much as possible the current model.

oneData system is chosen as the most suitable solution. It permits a totally transparent use of the remote data resources offering a virtual POSIX file system mounted by FUSE, son no major disruptions are expected in the DMP already in place. The data redundancy rule applied is "copy on read", convenient for unbalanced data clouds like MuG's, very diverse in terms of proximity to public repositories', capacity and accessibility. In this way, data will be transferred from one cloud to the other only under demand during job executions. Furthermore, the activation of user's virtual storage is a POSIX-based and non-interactive process, essential to be latter integrated into the dynamic and user-specific contextualization of the storage via OCCI at boot time. Additionally, oneData supports OpenID connect protocol, becoming trivial to encompass it with MuG authentication system. And finally, the system supports mechanisms for sharing datasets, a feature that could well complement MuGVRE.

Although oneData is not yet part of MuG platform, a **pilot installation** is built and configured at BSC premises in order to design a detailed integration plan. The installation is composed

of two VMs (oneProviderEBI and oneProviderBSC) emulating two geographically separated clouds each contributing with storage capacity. A third VM federates the providers under a unique view (oneZone), and a last VM represents a client (OneClient) accessing the storage via oneZone, for instance, MuGVRE or any of the VM tools. They all are emulating a scenario where a oneProvider server is on top of each MuG cloud, with its corresponding local NAS, while a Zone has access to them via TCP/IP ports. The system already interconnects with MuG users via MuG authentication server, and specific openID group's membership and scopes are set up, as oneData uses them to manage data access across clouds. According to these, MuG users have access to certain oneData Spaces, virtual views of data volumes supported by the distinct cloud NASs. MuG users are configured to have accessible by default two Spaces: a private Space, analog to current MuGVRE user's workspace, and a so called "public" Space, planned for caching public repositories' data. Upon user's login from the client (MuGVRE server), their spaces are transparently mounted by FUSE, making user's files accessible via POSIX regardless their physical cloud datastores. Thus, MuGVRE visualizers, running on the web server, could gain access to remote files, which are copied locally only when accessed. Similarly, if the system's login were set up during the contextualization of PMES-enabled VMs being deployed into the *Embassy Cloud*, the VM would access to user's uploaded input file, initially sitting on the cloud hosting MuGVRE and transparently transferred to Embassy's data block. A second execution there, would already find the file there. Scripts for user's registration and Space's configuration are prepared, all via REST oneData API. Next step is integrating them into MuGVRE backend.

4.3.3.4 Metadata

Metadata creation and handling has sharpened important MuGVRE features in terms of user experience, automatic application's administration, and reproducibility. Backend and frontend heavily rely on the metadata stored in Mongodb documents, whose collection entries are modeled after the PHP objects used in MuGVRE. The **data models** more relevant are the following:

Data model		**Definition**	**MuGVRE DB**	**Annex**
File	Descriptive and operational metadata defining files and directories.	MuG partners prepared REST APIs also based on this model. Its specification is annexed.	"Files" collection: operational metadata "Files Metadata": rest of metadata	*8.4.1* Data Model: "File"
Tool	Metadata describing the applications to be executed on the cloud	Defined using JSON schemas. One for tool developers willing to register his tool. A second, for internal use with some extra fields (*e.g.* identifiers)	"Tools" collection	*8.4.2* Data Model : "Tool"

Table 4.4: MuG data models

Such data models include descriptive and technical metadata that help MuGVRE to accomplish some basic functionalities like job accounting, data provenance, new tool and visualizer integration, tools interoperability, etc. “File” and “Tool” models structure the necessary data.

“File” data model

“File” defines a file or directory resource stored in the MuG infrastructure and is represented as exemplified in Snippet 4.7. “File” is the junction point between MuGVRE and the metadata management APIs implemented by MuG partners. The model corresponds to two synchronized collections in MuGVRE database, (i) “Files” collections, strictly storing operational metadata that cover the minimal functionalities of MuGVRE as a file server application, (ii), the rest of metadata required to build the VRE - semantics and descriptive metadata.

```
{
 "file_id": "MuGUSER59e5ead574743_5cf8c3d1b43156.06448028",
 "path_type": "file",
 "file_path": "MuGUSER59e5ead574743/__PROJ5c6c417267b522/uploads/G1.bam",
 "cloud":     "mug-irb",
 "user_id":   "MuGUSER59e5ead574743",
 "project":   "__PROJ5c6c417267b522.33832470",
 "size":      725988911,
 "parent_dir": "MuGUSER59e5ead574743_5c6c417280d635.07060077",
 "expiration_time": {"sec": 1559806931,"usec": 0},
 "creation_time":   {"sec": 1559806929,"usec": 0},
 "source_id": ["MuGUSER59e5ead574743_566041g28dd456.675598”],
 "tool_id":   "BAMindex",
 "arguments": {"sort": true },
 "data_type": "data_mnase_seq",
 "file_type": "BAM",
 "compressed": false,
 "metadata": {
    "refGenome": "R64-1-1",
    "taxon_id": 4932,
    "description": "MNase-seq for S. cerevisiae cells synchronized in G1",
    "paired": "paired",
    "sorted": "sorted",
    "associated_files": ["MuGUSER59e5ead574743_5cf8c3bf2151a3.62251628"],
    "validated": true,
    "visible": true
 }
}
```

Operational metadata

File provenance

Minimal description

Descriptive metadata

Snippet 4.7 : Example for “File” data model in MuGVRE

Operational metadata for user’s input and output files is collected, which provide the platform with a flexible data hierarchy, and a dynamic allocation system based on a unique “file_id” and “file_path” addresses relative to a cloud storage. Full data access is resolved at the application level, either by MuGVRE or MuG data APIs. Together with other metadata objects like “JobProcess” or “User”, MuGVRE provides job **accounting** and **data provenance system** by storing: file lineage at “source_id” that records job transformation operations; input files and arguments values for each run; file and job timestamps with a full registry;

tool's control versioning; operation logging; etc. Such metadata, together with the use of sandboxed executions on virtual environments are essentials for achieving reproducibility on the system.

Descriptive metadata accompanying files, as well as applications and visualizers, conforms the basis of **tool's interoperability** in the platform. "file_type" and "data_type" fields conform the minimal descriptive set of metadata required for MuGVRE to operate. They semantically define the content (*e.g.* "DNA sequence") and format (*e.g.* "FASTA") of a "File" record. In turn, they constrain suitable input files and specify expected output files when applied to "Tool" or "Visualizer" definitions. The metadata matching between both, "Tools" and" Files", permits to interoperate input and output file tools, as well as guide user experience, for instance, dynamically building toolkits of "Available Tools" responsive in front of user's workspace file selections.

"Tool" data model

"Tool" entity is the MuGVRE PaaS building block and it defines "what" and "how" a Tool Developer's application is to be executed. The following snippet represents a simplified example:

```
{
    "_id" : "naflex",
    "name" : "NAFlex analyses",
    "title" : "Nucleic Acids Flexibility Analysis",
    "short_description" : "Set of analyses to extract [...]",
    "long_description" : "NAFlex provides a [...]",
    "url" : "http://mmb.irbbarcelona.org/NAFlex/",
    "owner" : {
        "author" : " Adam Hospital",
        [...]
    },
    "status" : 1,
    "keywords" : ["dna", "rna", "dynamics"],
    "keywords_tool" : ["nucleic acid NA", "flexibility", "curves"],
    "infrastructure" : {
        "memory" : 16,
        "cpus" : 1,
        "executable" : "/home/MuG/NAFlex/NAFlex_Wrapper.py",
        "clouds" : {
            "mug-irb" : {
                "launcher" : "PMES",
                "minimumVMs" : 1,
                "initialVMs" : 1,
                "imageName" : "uuid_mugMD_99"
            }
        }
    },
    "input_files" : [
        {
            "name" : "pdb",
            "description" : "Input Structure, pdb format",
            "help" : "Input representative structure [...]",
            "file_type" : ["PDB" ],
```

Tool description

Deployment details

Input file requirements

```
            "data_type" : ["na_structure"],
            "required" : true,
            "allow_multiple" : false
        },
        {
            "name" : "top",
            "description" : "Input Topology, Amber Parmtop v7 format",
             [...]
        },
        {
            "name" : "crd",
            "description" : "Input Trajectory, Amber mdcrd format",
             [...]
    ],
    "input_files_combinations" : [
        {
            "description" : "Analyses from trajectory",
            "input_files" : ["pdb", "top", "crd"]
        },
        {
            "description" : "Analyses from structure",
            "input_files" : ["pdb"]
        }
    ],
    "arguments" : [
        {
            "name" : "operations",
            "description" : "Flexibility Analysis to be computed",
            "type" : "enum_multiple",
            "enum_items" : {
                "name" : ["Curves", "Nmr_NOEs", [...]],
                "description" : ["Curves", "NMR NOEs", [...]]
            },
            "required" : true,
            "allow_multiple" : false,
            "default" : ["Curves"]
        }
    ],
    "output_files" : [
        {
            "name" : "NAFlex_report",
            "required" : true,
            "allow_multiple" : false,
            "file" : {
                "file_type" : "TAR",
                "data_type" : "tool_report",
                "meta_data" : {
                    "description" : "NAFlex analyses [...]",
                    "compressed" : "gzip",
                    "visible" : true
                }
            }
        },
        {
            "name" : "CURVES_torsions",
             [...]
    ]
}
```

Input file requirements

Arguments

Expected output files

Snippet 4.8: Example for "Tool" data model in MuGVRE

The use of descriptive fields like "Title", "Description", or "keywords", allows MuGVRE to automatically create help and usage applets, tool discovery and browsing functionalities,

etc. The “infrastructure” nested object is focused on defining how to remotely invoke the application from MuGVRE. It includes the location of the application main “executable” in the VM, how to reach it (either via PMES or via SGE depending on the particular VM configuration), the computational resources that tool developer estimated necessary, and the elasticity bounding parameters.

On top of that, the enumeration and description of “input_files” and “arguments” per each tool permits MuGVRE to build **automatic launching tool forms** (Figure 4.35) based on PHP web templates - some JavaScript may be manually added, for instance, to control argument fields dependencies.

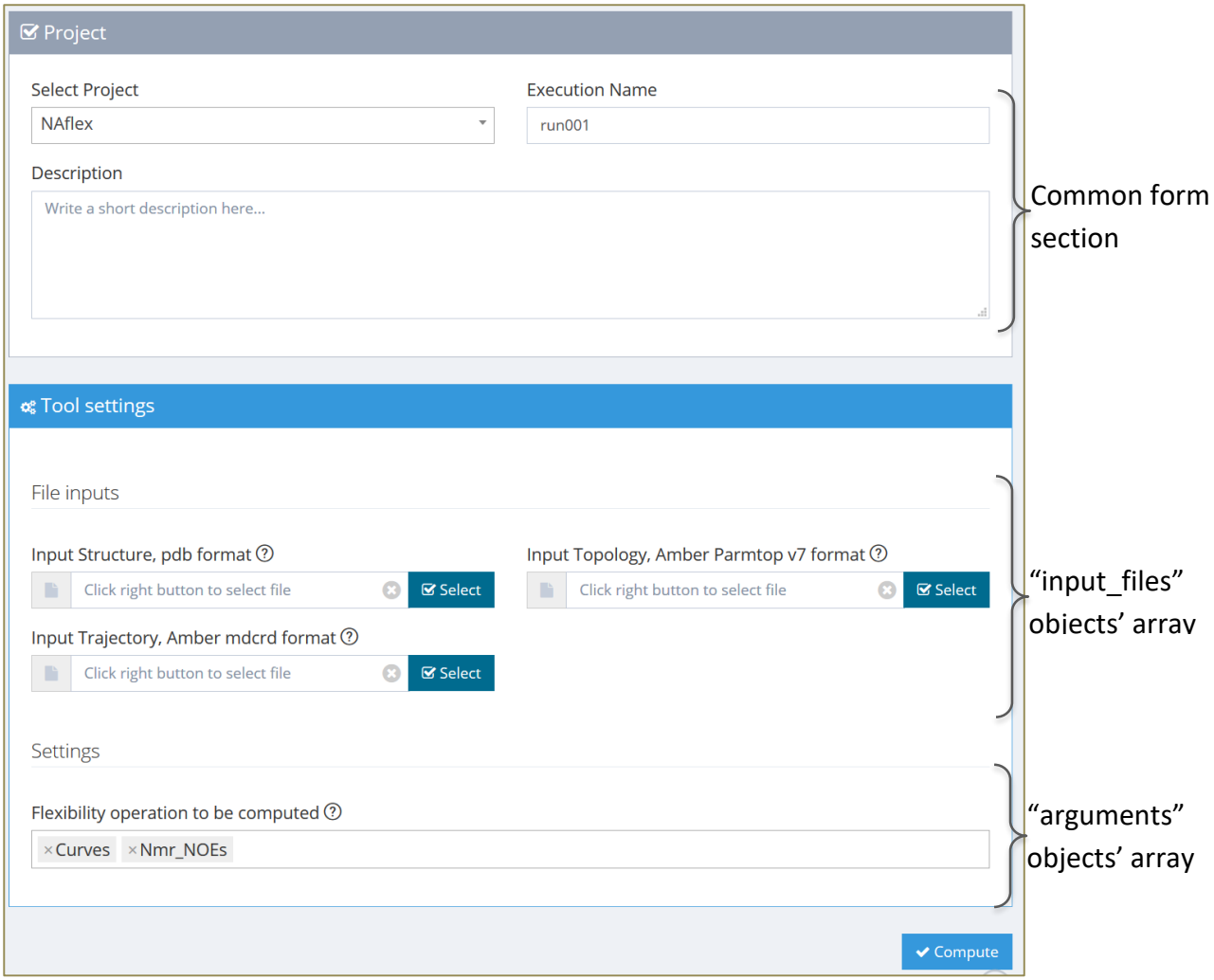


Figure 4.35: Tool web form in MuGVRE.
It is automatically build based on registered database “Tool” record. The example corresponds with the “Tool” instance in Snippet 4.8

Such metadata-driven automatism is the first stage of the complete **tool management lifecycle** (Figure 4.36), which is fully controlled by "Tool" and "File" metadata. Such data (i) defines MuGVRE submission into PaaS components, (ii) circulates to the deployed VMs, and (iii) validates application results. To achieve so,

I. "input_files", "arguments" and "input_files_public" as defined in the "Tool" object is used to build the web form, which the researcher fills in with the values for a particular run.
II. these are processed and written down in two auxiliary files. The two JSON files (*i.e.* "in_metadata.json" and "config.json") that are stored in the newly created "Run folder", which will become the tool working directory. Annex *8.4.3* Job Auxiliary Files contain examples of such files.
III. the "infrastructure" fields on the Tool object (*e.g.* "memory", "cores", "comps" enabling, SGE "queue", VM "image", application "executable", etc) are used to compose the job petition for the selected job processor (*i.e.* PMES or SGE), who triggers the execution on the underlying cloud.
IV. on the triggered command the two JSON files are passed as arguments to the virtual instance. MG-TOOL wrapper parses these files and composes to actual application command with all the necessary information regarding input files and arguments.

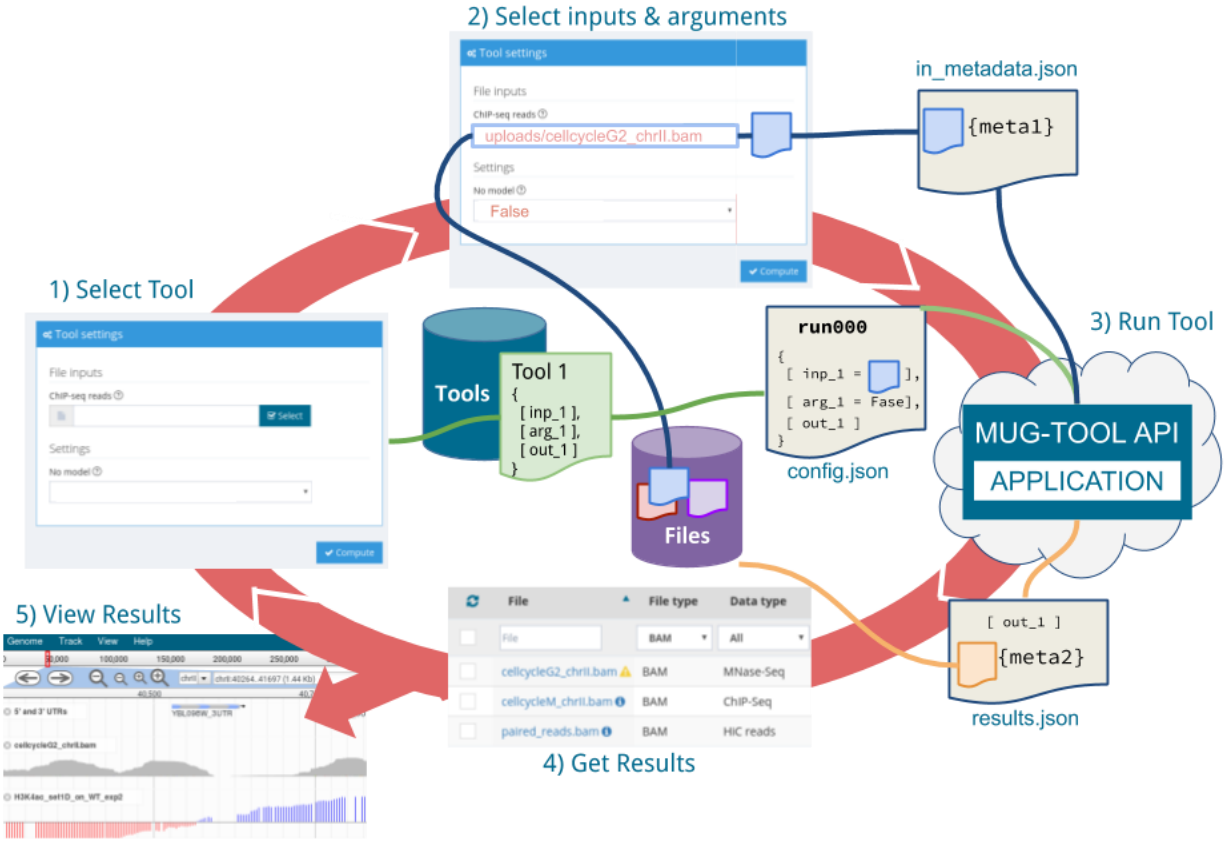


Figure 4.36: Metadata flow among MuG elements during a tool life cycle.
Illustration on how user's selection of input files and arguments are passed from the web to the virtual machine via two auxiliary files (config.json and in_metadata.json). "Tools" registry is the primary source.

V. once the application finishes, MG-TOOL wrapper writes down into "results.json" some relevant metadata regarding the output files just generated. Primarily, it includes the metadata that change on each run: "file_path" locations, file "source_id", or custom metadata attributes that tool developers want to inject into MuGVRE metadata ("*e.g.* docking_grid_size").

VI. after job completion, MuGVRE builds a File object for each output file by aggregating (a) the information stored in the "Tool" object under the "output_files" section (*e.g.* "file_type"), (b) dynamic fields read from "results.json" (*i.e.* "path"), and (c) operational data (*e.g.* "owner", "creation_time", etc). Once output files are registered in the "Files" collection, they are eligible from user's workspace.

4.3.4 Use case: Nucleosome Dynamics

The present section describes the deployment and usage of one of the analysis tools offered on the MuGVRE platform, the Nucleosome Dynamics suite. The use case illustrates how a scientific method development can benefit from the integration into community-driven computational platforms, like MuGVRE or the well-known Galaxy. The example covers the integration of the tool at the MuG infrastructure as well as other implementation models. The corresponding publication is annexed at *8.7* Publications.

Nucleosome Dynamics suite

Nucleosome Dynamics is a suite of programs to define nucleosome architecture and dynamics from noisy experimental data. Different studies demonstrated that nucleosome positioning is coupled to gene function [230] and that transcriptional activity and nucleosome architecture are tightly coupled. The package allows both the definition of nucleosome architectures and the detection of changes in nucleosome organization due to changes in cellular conditions from MNase-seq and ATAC-seq experimental data. Results are annotated sequence files (GFFs and BED) that can be displayed in the genomic context thanks to sequence browsers, allowing the user a holistic, multidimensional view of the genome/transcriptome. The package shows good performance for both locating equilibrium nucleosome architecture and nucleosome dynamics.

Two specific programs, nucleR and NucDyn, have been specifically developed to perform such studies:

- nucleR performs Fourier transform filtering and peak calling, in order to efficiently and accurately define and classify the location of nucleosomes.
- NucDyn is a method to detect changes in nucleosome architectures based on MNase-seq experiments. It identifies nucleosomes' insertions, evictions and shifts between two experiments at the read level.

Additionally, a list of other nucleosome-related analyses completes the suite:

- Location of nucleosome-free regions (NFRs)
- Classification of transcription start sites based on the surrounding nucleosomes
- Study of nucleosome periodicity at the gene level
- Stiffness of the nucleosomes derived from fitting a Gaussian function to nucleosome profiles

Implementation Models

Nucleosome Dynamics is implemented as a set for R packages and libraries provided under several distribution models to fulfill the needs of different users. Moreover, it is also offered as a service in two different research platforms. All available distributions are explained at Nucleosome Dynamics landing page[15], and summarized in Table 4.5:

Landing page http://mmb.irbbarcelona.org/NucleosomeDynamics/		
Code distribution		
Standalone installation	Nucleosome Dynamics CLI	https://github.com/nucleosome-dynamics/nucleosome_dynamics
	nucleR R package	https://github.com/nucleosome-dynamics/nucleR Bioconductor: https://www.bioconductor.org/packages/release/bioc/html/nucleR.html
	NucDyn R package	https://github.com/nucleosome-dynamics/NucDyn Bioconductor: (in review)
Containerized installation	Docker	https://github.com/nucleosome-dynamics/docker Docker-hub: mmbirb/nucleosome-dynamics
	Singularity	https://github.com/nucleosome-dynamics/nucleosome_dynamics_singularity Singularity-hub: https://singularity-hub.org/collections/2579
Platforms in use		
MuG Virtual Research Environment		https://vre.multiscalegenomics.eu/workspace/?from=nucldynwf
Galaxy Platform		https://dev.usegalaxy.es (in development) Galaxy Tool-Shed: https://toolshed.g2.bx.psu.edu/repository?repository_id=822e9c879cf92fd0

[15] http://mmb.irbbarcelona.org/NucleosomeDynamics/

Table 4.5: Implementation models for Nucleosome Dynamics

4.3.4.1 MuGVRE tool application

Nucleosome Dynamics (ND) is one of the analysis methods integrated at the platform that offers a set of analyses packed in a single modular MuGVRE tool, together with a custom report viewer. Besides, the suite can well **interact with other services** on the platform, (i) ArrayExpress repository includes MNase-seq experiments, that might be used either to import FASTQ reads, or bibliographically compare with other nucleosome annotated sequences, (ii) alignment tools (BWA, Bowtie2) can be used to produce the aligned reads (BAM format) that ND takes as input, and (iii) the resulting nucleosome architectures written as annotation files (GFF3, BW) might be visualized with two of the visualizers offered at MuGVRE: JBrowse and TADkit.

Nucleosome Dynamics' tool preparation

ND code (Nucleosome Dynamics CLI) is installed, together with its dependencies, in a MuG base **VMI** hosted at the MMB IRB cloud (see Method's section *3.1.2.1*), in this case, configured with SGE. According to MuGVRE Tool integration protocol, either (i) support team provides IaaS access to the virtual instance and the Tool Developer directly performs the installation there, or (ii) Tool Developer makes its code public, and support team installs it in the virtual instance. The chosen strategy mainly depends on the tool installation complexity. Here, we have requested direct access to the VM.

The **tool wrapping** is implemented on a thin python script dependent on MG-TOOL-API, that is able to respond at the MuG agreed CLI. The code parses MuG auxiliary files and extracts from them the input file locations and ND arguments. Depending on the received arguments, the script builds the ND command(s) of the requested analysis (*i.e.* nucleR, nuclDyn, NFR, etc) and calls a system subprocess on each (Figure 4.37). Pipeline analyses are switched ON/OFF by the user on the web, and the wrapper runs them accordingly (*i.e.* arguments.NUCLER: true/false). If multiple BAM instances are given, the pipeline is executed multiple times. MUG metadata is read to learn the library type of the given BAM (single, paired) and its reference genome (*e.g.* hg38). ND adaptor's code is publicly available at NucleosomeDynamics-MuGVRE repository[16], with some documentation and test data.

[16] https://github.com/nucleosome-dynamics/nucleosome_dynamics_MuGVRE

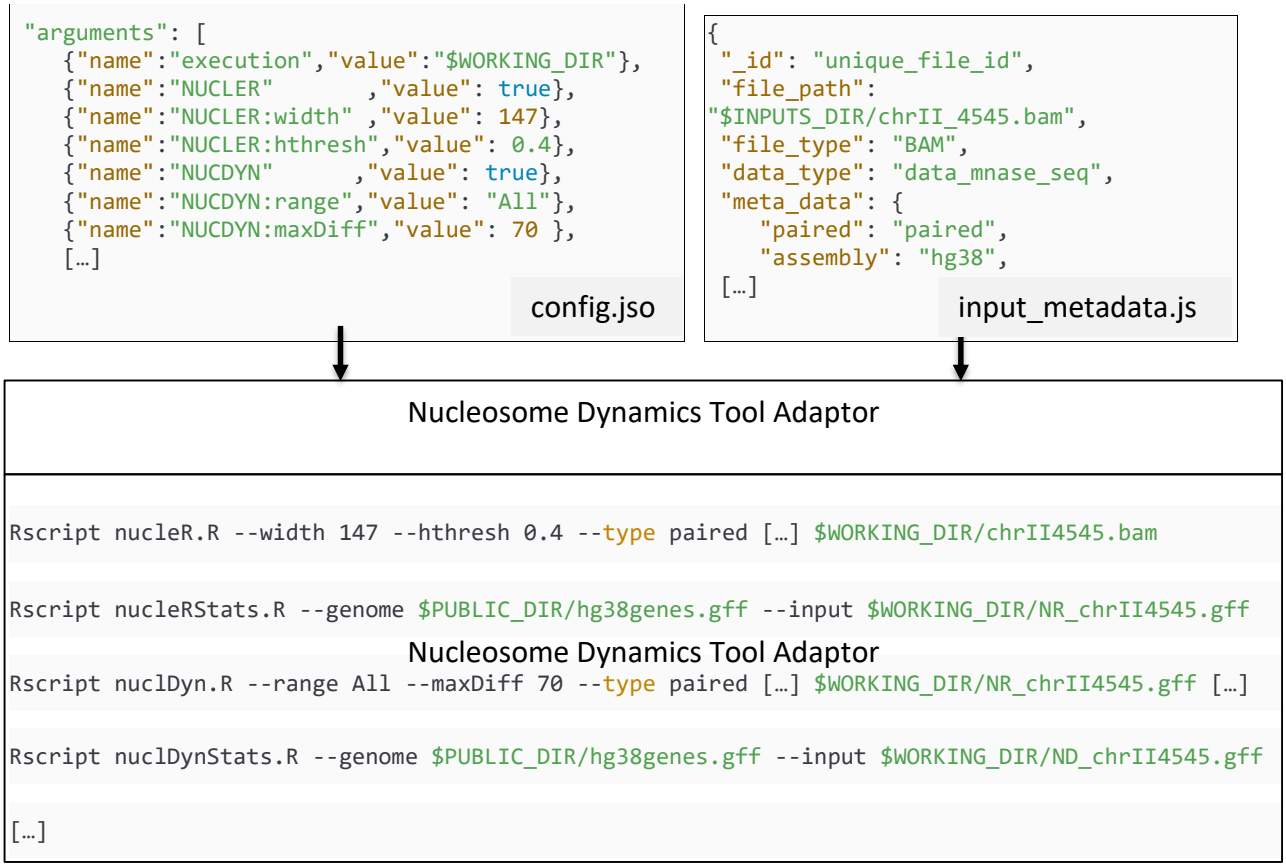


Figure 4.37: MuGVRE wrapper for Nucleosome Dynamics build Rscript calls

ND requires access to certain **reference data,** which is part of the MuG data repository accessible via NFS under the "public_dir" mounting point on all SGE-enabled VMs. It corresponds to assembly associated data (*e.g.* hg38 annotated genes) used to compute nucleosome-related statistics. MuGVRE sends the location of the public directory together with the rest of the arguments, and the script uses it to compose the path for the files it is interested in.

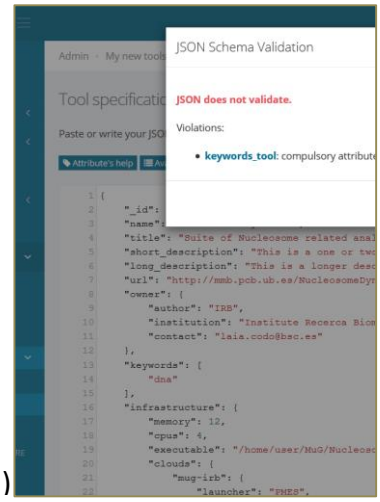
Nucleosome Dynamics' registration

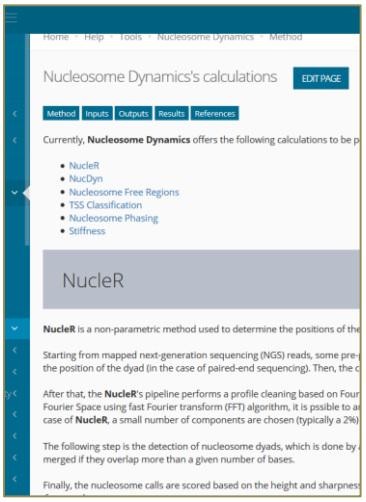
Once the code is tested and ready, it is time to register it at the platform using a Tool Developer user account. The new Tool petition is submitted online, and requires the acceptance by the MuGVRE support team. The following information is supplied to the platform:

The **Tool definition** is the first and essential requirement. All the attributes comprised in the Tool data model (*8.4* MuG data models) are to be filled in: list of input files, arguments, titles, keywords, etc. MuGVRE embeds on the developer's panel an online JSON schema validator that ensures data consistency of the submitted Tools (Figure 4.38 (a)).

Nucleosome Dynamic Tool definition is available at NucleosomeDynamics-MuGVRE repository.

- After ND is accepted, it appears on the web interface under “Testing” mode, so that only the Tool owner has access to it. Some **documentation** and how-to tutorials are then added on the help pages, which are online editable as Mark Down files (Figure 4.38 (b)). Extra descriptive information, logo images and ND snapshots are also submitted to be displayed on the home page.
- An auto-contained HTML/JavaScript single page (hereinafter, **the custom viewer**) is prepared in order to display the ND statistical information for each run. The viewer prints nicely some CSV files generated during the ND execution as histograms and HTML tables. Tool Developers submit it, and the support team integrates it at the platform. This is the way the MuGVRE offers the opportunity to provide a custom visualization for tool results. It is displayed in the side menu “View Results” for each Run folder at the workspace (Figure 4.38 (c)).
- **Example datasets** of ND input and output files are prepared and displayed at MuGVRE as sample data, enabling a one-click import of demo data at the workspace. Indeed, three different datasets are prepared ([220]–[222]) with the double propose of demonstrating ND usage to MuGVRE users, and show-casing ND method potential during a peer-to-peer journal review. Demo data is under “Get Data” section.

(a) 

(b) 

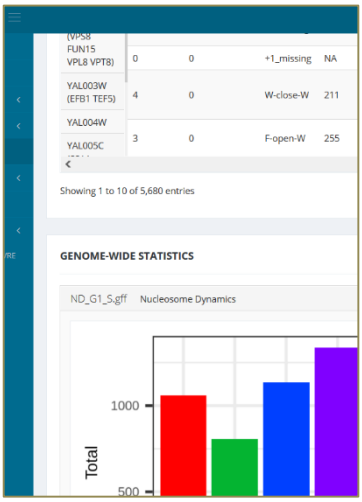
(c) 

Figure 4.38 Nucleosome Dynamics Tool submission on MuGVRE.
(a) Online validation of the Tool object definition. (b) Example of MD help page, online editable. (c) Nucleosome Dynamics Custom viewer

Running Nucleosome Dynamics (ND) at MuGVRE follows the **same basic flow** already commented on MuGVRE user’s perspective, which can be summarized as:

I. Get Data
II. Choose and Configure Tool
III. Display Results

Choosing “Import example dataset” and “Cell Cycle Data Set” results in a quick sample data loading sufficient for demonstration purposes. In our active workspace, two BAM files of type “MNaseq data” should appear under the “Uploads” folder.

We can reach ND Tool from either “Workspace” or “Launch Tool”. The first implies knowing in advance what are the input files taken for ND, selecting them at the Workspace, and choosing “Nucleosome Dynamics” from the “Available Tools” drop-down menu that appears below, next to the shopping card listing selected data. The latter is designed for a more exploratory use of MuGVRE. Launch Tool section lists available tools and visualizers, with some descriptive data to browse among them. Nucleosome Dynamics appears under “Analyse MNase-seq Data”. We should now set a name for the new job (by default run[000]), and configure ND pipeline. The webform lists the ND different analyses grouping them under a switch button to enable/disable them. Input file and argument fields are to be filled in. If files have already been selected in the “Workspace”, the form already shows them in place. If not, the selection of each input field opens a dialog with all the possible files suitable for such field, filtering out data with not appropriated “File Format” or “Data Type”.

After job submission, the Run folder appears at the “Workspace”, with all the job information (*e.g.* arguments in use, requested resources) and job status (*i.e.* PENDING, RUNNING, ERROR, FINISHED). Meanwhile, the execution progression also monitored checking the log file, either nicely formatted or in raw format.

On job completion (after 5 – 15’ depending on selected ND analyses), the list of GFF and BED files composing ND results appears grouped inside the Run folder. Under “View Results”, some run’s statistical data and reports are displayed. 2D sequence annotation files are eligible to be displayed by either JBrowse or TADkit, the second more focused on chromatin models, so that we choose the generic genome browser. JBrowse is available for a selection of reference genomes, including *S. Cerevisiae*, our dataset’s yeast. ND sequence tracks can be visualized and comparatively analyzed with other user’s data, from its workspace or imported from ArrayExpress, but also with bibliographic tracks integrated into JBrowse a public reference data.

4.3.4.2 Galaxy implementation of Nucleosome Dynamics

Galaxy platform is the second infrastructure offering Nucleosome Dynamics as an online service. It offers similar computational services than MuGVRE, yet Galaxy environment is not particularly focused on 3D/4D genomics. The software suite is implemented as a series of Galaxy Tools able to conform a complete analysis workflow.

Galaxy Tools preparation

The set of Galaxy Tools prepared for Nucleosome Dynamics (ND) is based on a **containerized implementation** of the ND suite on Docker. The software container encapsulates the R libraries of NucleosomeDynamics CLI, together with its dependencies, and a Perl script able to interact with them all and trigger their execution. The script is set up as the container's Entrypoint, and beside validating and building ND commands for individual analyses, it is able to sequentially evaluate a pipeline of ND commands. The corresponding build data and Dockerfile (installation and configuration recipe) is publicly available, as well as the Docker image, deposited at Docker Hub (see table Table 4.5). Interestingly identical installation workflow was followed to build VMIs for MuGVRE and Docker images

Galaxy Tools are enriched XML files that serve two basic functions, (i) composition of the tool web form layout (*e.g.* form fields, text, help, etc.), and (ii), specification to Galaxy Central on how to invoke the application. **Twelve Galaxy Tools** have been prepared correspondingly to the six different ND analyses included in the suite and their associated statistics module (*e.g.* NucleR.R and NucleR_stats.R). The invoked command calls BASH as job executor, who instantiate the ND Docker image with the appropriate arguments, but not before applying some renaming trick. ND software needs to preserve meaningful file extensions, and Galaxy erases all when internally storing the data. The temporal soft links applied before container instantiation manage to recover the original file extensions. A couple of other considerations are worth mentioning when dealing with containers in Galaxy. In order to make accessible Galaxy's data into the virtualized environment and enable persistence for generated output files, several Galaxy directories are mounted into the container, bypassing permission issues by impersonating Galaxy's UID (user identifier). Galaxy features a native runner for Docker-based tools, yet, it runs all the commands in the container, with no option to apply the renaming trick. Other Galaxy components like Planemo[231], for building Tool XML skeletons, and Galaxy Tool-Shed repository, for publishing resulting Galaxy Tools, have been plenty used.

Galaxy Tools Installation

ND tools have been installed in the development site of the **ELXIR-ES Galaxy server** (version 19.01) (see Table 4.5). The Galaxy Tool installation procedure is widely documented. Very essentially, it consists of placing Tool Configuration Files above described at the right Galaxy hierarchy directory structure. Some configuration files need to be edited in order to let Galaxy know about the new instances. The actual application software is expected to be installed where the Tool will run, here, the same development Galaxy server. For ND, the only requirement is Docker, who in turn would handle image pulling if need.

Regarding ND reference data and demo datasets, Galaxy permits to handle both. Reference data can be stored in any local location, while Galaxy configuration files were set accordingly. On the other hand, a Shared Galaxy Library is prepared with the "Cell cycle"

[220] sample data. Finally, the "**Nucleosome Dynamics" workflow** composed by the combination of the twelve Galaxy Tools is also published on the site.

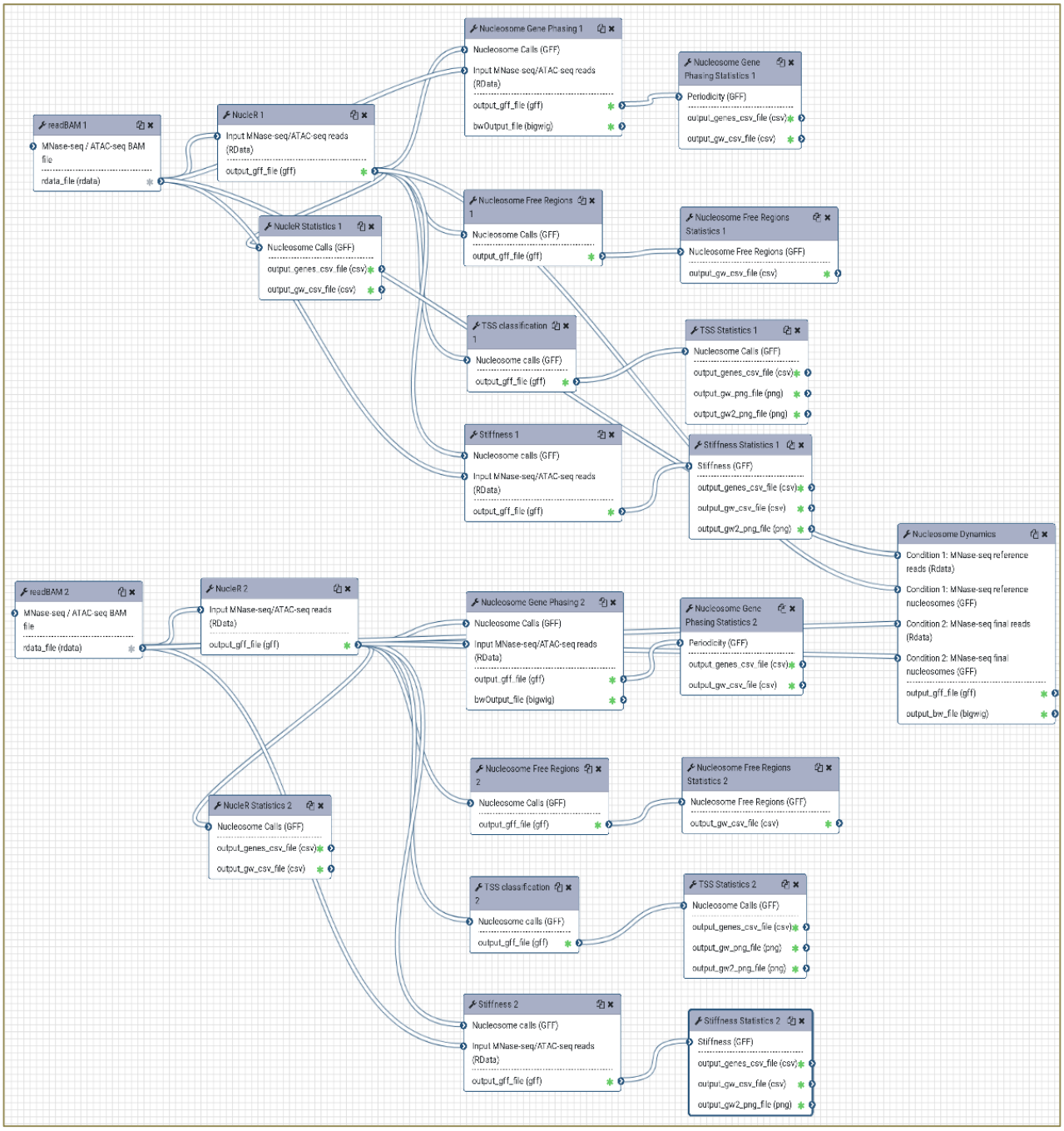


Figure 4.39 Nucleosome Dynamics Worflow in Galaxy

Running Galaxy Tools

Running Nucleosome Dynamics (ND) at Galaxy do not differ much from running any other analysis tool in the platform, which is extensively documented. Nonetheless, a Nucleosome Dynamics Galaxy wiki[17] page has been prepared with step by step information

.

[17] https://github.com/nucleosome-dynamics/nucleosome_dynamics_galaxy/wiki

4.4 Open Virtual Research Environment

The present chapter explains our efforts on building a neat PaaS orchestrator taking as a basis the virtual research environment built for the 3D/4D genomics community. To this end, MuGVRE is detached from all domain-specific features to become openVRE. To illustrate openVRE reusability, OpenEBench is presented, the ELIXIR's benchmarking platform.

The candidate has formalized openVRE, and participated in their adoption of it into the OpenEBench initiative.

4.4.1 Context

Given the increasing complexity of scientific challenges, multidisciplinary collaborations, research process harmonization and multiscale executions are becoming common practices. Infrastructures as those above presented, are natural vehicles to support such conducts, yet designing and implementing them represent important efforts. Our aim here is to lift cloud administrators part of this burden and offer a neat PaaS orchestrator ready to be adopted for other scientific communities.

4.4.2 openVRE

openVRE is a PaaS composer that collects job deployment requests from a web exposed virtual research environment, and coordinates the resource and service deployment over dynamic PMES virtual appliances or directly on a queueing batch system like SGE.

openVRE aims to be a white canvas with a set of operational services and protocols to handle the computational and data resources on an underlying OCCI-compliant cloud provider. As a result, a tailored computational infrastructure is rapidly assembled, enabling the build, run, and operation of applications in cloud-based infrastructures. After configuring openVRE for a particular project, the researcher access to a tailored VRE with a set of ready-to-use services (datasets, analysis tools and visualizations) fully adapted to their needs.

To ease the installation procedure, openVRE populate the new Mongodb with demonstration data, like a tool skeleton and sample data type and file type.

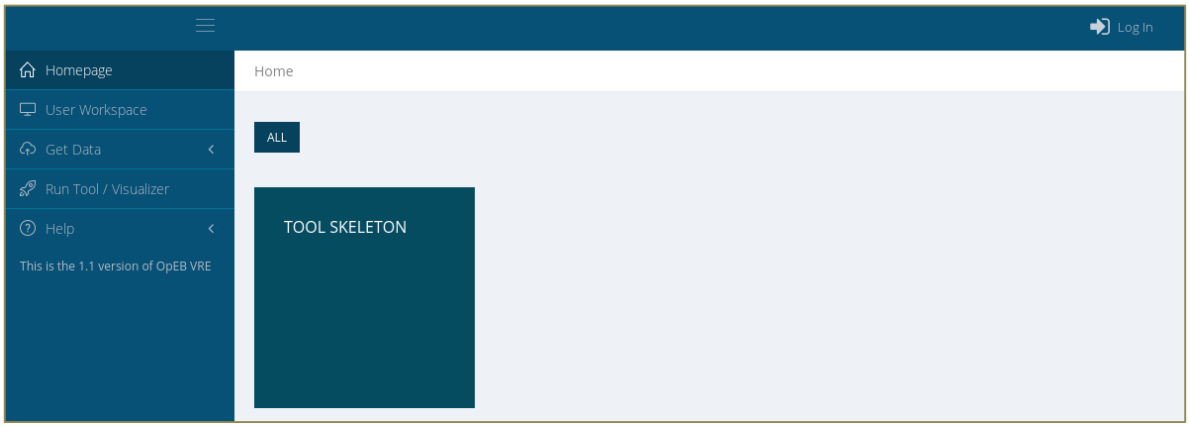


Figure 4.40: Home snapshot of a plain openVRE instance

Application's code, templates and administrative documentation is publicly available[18]. Following, the main configuration file set on bootstrap illustrates the configuration step required to set up the infrastructure.

```
# config/globals.inc.php.sample

// Main config
/************************
$GLOBALS['SERVER']    = "https://www.mydomain.com"; // host
$GLOBALS['BASEURL']   = "/"; // prefix
$GLOBALS['NAME']      = "Project Name"; // project name
$GLOBALS['SITETITLE'] = "Project Name | Virtual Research Environment"; // site title
// Email
$GLOBALS['mail_credentials'] = __DIR__."/mail.conf"; // SMTP credentials
$GLOBALS['ADMINMAIL'] = "admin@mail.es"; // BBC address for VRE ticket emails
// Mongo databases
$GLOBALS['db_credentials'] = __DIR__."/mongo.conf"; // Mongo access
$GLOBALS['dbname_VRE']     = "dbname"; // Database name
//VRE installation paths
$GLOBALS['root']       = dirname(__DIR__); // VRE root directory
$GLOBALS['shared']     = "/data/vre/"; // VRE data directory
$GLOBALS['dataDir']    = $GLOBALS['shared']."userdata/"; // User data directory
$GLOBALS['pubDir']     = $GLOBALS['shared']."public/"; // Public data directory
$GLOBALS['sampleData'] = $GLOBALS['shared']."sampleData/"; // Tool dataset directory
// File manager config
$GLOBALS['DISKLIMIT']       = 100 // Default user disk quota (GB)
$GLOBALS['MAXSIZEUPLOAD']   = 4000; // Maximum upload file size (MB)
$GLOBALS['caduca']          = "182"; // Expiration date for user files (days)
// Tool integration data models and templates
$GLOBALS['tool_json_schema']  = "https://github.com/projName/tool_schema.json";
$GLOBALS['tool_jsontemplate'] = "https://github.com/projName/tool_template.json";
// Oauth2 authentication
```

[18] https://github.com/inab/openVRE

```
$GLOBALS['auth_credentials']       = __DIR__."/oauth2.conf"; // oauth2 client credentials
$GLOBALS['authServer']             = 'https://auth.mydomain.come'; //external oauth2 server
$GLOBALS['authRealm']              = 'realmName'; // keycloak realm
$GLOBALS['urlAuthorize' ]          = $GLOBALS['authServer'].'/openid-connect/auth';
$GLOBALS['urlAccessToken']         = $GLOBALS['authServer'].'/openid-connect/token';
$GLOBALS['urlResourceOwnerDetails']= $GLOBALS['authServer'].'/openid-connect/userinfo';
/*******************************
// Project specific definitions
*******************************/
// Cloud infrastructures
$GLOBALS['cloud'] = "my_local_cloud"; // VRE central cloud. Options in GLOBALS['clouds']
$GLOBALS['clouds'] = Array(
       'my_local_cloud' => array(
              "http_host"         => "www.mydomain.com",
              "dataDir_fs"        => "/NFS/export/path/userdata/", // NFS server
              "pubDir_fs"         => "/NFS/export/path/public/",   // NFS server
              "dataDir_virtual"   => $GLOBALS['dataDir'],
              "pubDir_virtual"    => $GLOBALS['pubDir'],
              "PMESserver_domain" => "pmes.mydomain.com",
              "PMESserver_port"   => "80",
              "imageTypes"   =>  array(  // OCCI templates indexed by RAM (GB)
                  "2"  => array( // 2GB RAM with 1, 8 or 16 cores
                     "1"  => array("id" => "small", "name" => "small"),
                     "8"  => array("id" => "large-small", "name" => "large-small")
                     "16" => array("id" => "extra_large-small", "name" => "large-small")
                  ), // 4GB RAM with 4, 8 or 16 cores
                  "4"  => array(
                     "4"  => array("id" => "medium", "name" => "medium"),
                     "8"  => array("id" => "medium-medium", "name" => "medium-medium"),
                     "16"  => array("id" => "large-medium", "name" => "large-medium")
                  )
              "auth" => array("required" => False) // local access
       ),
       'my_remote_2' => array( //(*)
              "http_host"         => "www.mydomain2.com",
              "dataDir_fs"        => "/NFS/export/path2/userdata/", // NFS server
              "pubDir_fs"         => "/NFS/export/path2/public/",   // NFS server
              "dataDir_virtual"   => "/shared/path/for/userdata",
              "pubDir_virtual"    => "/shared/path/for/public",
              "PMESserver_domain" => "pmes.mydomain2.com",
              "PMESserver_port"   => "8080",
              "imageTypes"   =>  array([...]),
              "auth"  => array(        // openStack auth API (NOVA)
                     "OS_NO_CACHE"    => "True",
                     "OS_CLOUDNAME"   => "overcloud",
                     "OS_AUTH_URL"    => "https://extcloud05.ebi.ac.uk:13000/v2.0",
                     "NOVA_VERSION"   => "1.2",
                     "OS_USERNAME"    => "username@mail.es",
                     "OS_PASSWORD"    => "s3cr3t",
                     "OS_TENANT_NAME" => "tenancy_name"
              )
       [...]
```

Snippet 4.9: Sample configuration file for bootstrapping openVRE.

File truncated for the layout. (*) Registration of more than one cloud infrastructure is possible, as remote operations are enabled via PMES. However, data storage only reaches local cloud instances, so currently only on premises clouds are fully supported.

From MuGVRE to openVRE

The origins of openVRE lay on a deconvolution of the MuGVRE application. As a first step, the MuGVRE code has been decoupled from MuG domain-specific features, abstracting a neat and customizable platform. It includes also the Mongodb database, where the registration of analysis tools and metadata for user's input and output data is stored. Thanks to the modular design of MuGVRE, the abstraction updates are mainly limited to the front-end, and the data *per se* that populates the database.

Analysis and visualizer tools are set as separate entities that work as much as possible as pluggable appliances. They are sourced from both, a document in the Mongodb, and an auto-contained directory with the tool specific data (HTML templates, logo images, assets, etc). OpenVRE distribution includes no tools except for a skeleton tool intended to be used as a template for future tool integrations. In a similar way, MuG public repositories, browsable and ready to be loaded into the platform, have been clean.

Descriptive and operational metadata accompanying input and output files is an important element of the VRE. It is stored in the database, yet loaded through the frontend and validated in the backend. The code has been updated in order to reduce such metadata to a minimal set of generic attributes, not related anymore to the 3D/4D genomics community.

In order to improve the system's reuse, the number of customizable parameters and configuration files is not restricted to connectors or VRE settings, but includes project descriptions and associated data, so project references are not spread over the code anymore. All settings have been centralized in a configuration folder, providing template files for a rapid deployment of new openVRE instances, and skeleton data to populate new databases.

Source code is also more sustainable and easy to be installed, as third party dependencies have been compiled in a single Composer project file (a PHP package manager).

An effort has been put to improve the documentation, in particular, the administrator information, now included in the GIT repository under the install section. Moreover, the code has been reorganized to be more comprehensive.

A refactoring of several parts of the come has also been carried out, cleaning legacy code and fixing some bugs. They are annotated in repository's CHANGELOG file.

The result is an easy-to-install code, that centralizes the configuration of other platform's components and permits the modular integration of computational executions reachable even at remote virtual environments. Minimal platform's installation requires:

4.4.2.1 OpenEBench

openVRE seeks to be the basis to other computational infrastructures targeting specific communities. The VRE developed under OpenEBench's framework is an exemplification of that purpose.

Context

In the context of the EU H2020 ELIXIR-EXCELERATE project, OpenEBench is raised as the software **benchmarking platform**. It aims a transparent performance comparison across life sciences tools. OpenEBench supports communities by assisting in setting up emerging benchmarking efforts, fostering exchange between communities and, ultimately, making benchmarking not only more transparent, but also more efficient. The benchmarking encompasses technical performance of individual tools, servers and workflows, including software quality metrics, as well as scientific assessment in predefined community competitions - *e.g.* RMSD metric's comparison of predicted 3D structures in a DREAM challenge [232]. A fair and relevant comparison relies on the existence of scientific communities. They act as a proxy to identify relevant challenges, as well as can play an important role in defining reference datasets and metrics to assess those scientific challenges. OEB-VRE offers a platform to all these benchmarking communities

VRE for OpenEBench

The VRE for OpenEBench **(OEB-VRE)**[19] is a prototyping platform that integrates benchmarking workflows, benchmarking datasets and metric's visualizers on top of the OpenEBench cloud infrastructure. The VRE accomplishes a double purpose depending on user's role. For community managers, OEB-VRE acts like a platform as a service (PaaS), where community-developed benchmarking workflows are plugged into the framework as virtual appliances and offered as ready-to-use online applications. Furthermore, such workflows can be integrated with public benchmarking datasets to be used as golden data or reference. A second role accomplished by OEB-VRE targets those users willing to evaluate their methods. The platform offers a private workspace to load and test their results executing the community-given benchmarking workflows. The generated metrics (JSON files) can be fed into a metric's visualizer that generates dynamic plots comparing the performance of the submitted method against other community published methods.

OpenEBench **benchmarking workflows** are formalized as a series of three different types of Docker cont*a*iners in charge of (i) validating the file provided by the user, *i.e.* the method's result under evaluation, (ii) computing a set of assessment metrics endorsed by

[19] http://openebench.bsc.es/vre

communities' golden datasets, and (iii), producing aggregated data based on the individual metrics generated during the previous steps. In the current installation, the whole benchmarking workflow is encapsulated in a single virtual machine, which is internally orchestrated by the Nextflow manager. Each VRE tool (*i.e.* benchmarking workflow) has a main executable for triggering the workflow and collecting metrics files, which indeed corresponds to a python wrapper that make use of MG-TOOL functionalities to ensure compatibility with OEB-VRE. The job processor in use is SGE, as jobs are quick and not-demanding runs that can easily fit in the on-premises OpenEBench cloud, in the prototype installation, the BSC Life cluster (hardware details on table Table 3.1).

The infrastructure is up and running, with two engaged communities (The Cancer Genome Atlas (TCGA) [233] and Quest for Orthologs (QfO) [234] who already have set their own workflows, and others organizations like DREAMS or Global Microbial Identifiers (GMI) [235] that plan to follow suit.

5. Discussion

This chapter discusses from a global perspective the presented infrastructure solutions by addressing transversal dimensions of the same and showing the overall progress achieved in parallel with the progression in the field.

Designs and implementations presented in the current dissertation illustrate an evolution that well represents the advance of the research supporting technologies in the last years, during which e-science principles rooted down to all the investigative community.

Covering very diverse scenarios and divergent use cases, the first presented infrastructure is framed on the clinical data management field, and focuses on the data platforms build around two epidemiologic case studies on Immune-Mediated Inflammatory diseases (IMIDs), IMID-clinica and IMID-longitudinal. Making the leap to distributed infrastructures more oriented on analysis process support, the transPLANT infrastructure represents a first intrusion into the topical cloud computing model. It is focused on plant genomics and its design became the seed for a more integrative cloud-based solution, this time developed for members of the 3D/4GD genomics community. MuGVRE is the visible face of the resulting product. Becoming obvious the transversal potential of cloud-based computational infrastructures as virtual research environments, openVRE is implemented as an abstraction of MuGVRE. It offers a vanilla platform with computation, data and administration services ready to be adopted and customized by other scientific communities.

Uneven in architecture, motivation or context, all the proposed software platforms represent an opportunity to establish better research processes through enhanced collaboration, data management, analysis practices, and resources' optimization.

User's perspective

Considering the integrative approaches for decentralized or heterogeneous user's communities here presented, it is not surprising that a common trait between them is the **Web accessibly**. Web applications are the most popular strategy in life sciences to expose data and software tools to the researcher in a user-friendly way. Bioinformatics researchers were early adopters of web technologies, and online information and data sharing is part of their daily routine work. On the other hand, biomedical communities like those from IMIDs' project, are not that used.

Still today, right in the digital era, the human factor is one of the causes preventing the full establishment of IT health systems [236][237] in hospitals and health centres. However, large-scale multi-center initiatives like IMID-clinica or IMID-Longitudinal, would by unfeasible without online support. Increasingly, the **electronic data capture** (EDC) is being chosen by biomedical investigators over the traditional paper-based data collection in observational studies, as it is shown that EDC avoids mistakes and omissions, shortens

clinical study duration, reduces data collection costs, and facilitates post-analysis. Surprisingly enough, the implementation of electronic-based studies is still an open discussion because of patient's trustfulness or assistance work interference [238][239]. Overcoming this barrier is one of the challenges of IMIDs' projects, addressed to clinicians dedicated to providing high-quality patient care while conducting clinical and translational research. In order to enhance user's commitment, and offer full guidance to the clinicians undergoing the laborious task of entering the data into the system, IMIDs' applications provide complete user-friendly web interfaces, highly contextualized and with automatisms, data control toolkits and complete monitoring options. In order to produce effective research outcomes, critical values are data quality, consistency, and integrity, so that the application also integrates clinical data management components like monitoring, reviewing, accounting, statistics, etc. e

transPLANT community is diametrically opposed to IMIDs'. Plant genomics first concern is the actual computational capabilities of the infrastructure, more than its top-level user experience. **Programmatic access** via SOAP and RESTful APIs was the preferred way to gain access over computational services, while FTP was used to stage the data in and out of the cloud. Even so, the infrastructure features a dashboard to handle on the web the deployment and monitoring of virtual appliances as batch executions, and a web-based data manager to handle reference datasets, input files, and results. The two user interfaces accomplish with the propose of supporting the whole life cycle of genomics plant analyses on the web. Later, transPLANT "Data Manager" will become the seed of MuGVRE, extending and broadening the data-centric file manager until it grows into the workspace of a full research environment.

Virtual Research Environments (VREs) are a step forward in terms of user experience. They seek to support research processes on a particular domain while enhancing the coordination and cooperation of multidisciplinary, or even interdisciplinary, teams. Their objective is to make computational analyses accessible to non-programmers. Resource and service integration (*i.e.* data, tools, and visualizers) under a single framework is the chosen approach. The young 3D/4D genomics community is intrinsically multidisciplinary, integrated by several omics expertise and simulation methodologies. The users show highly diverse profiles, not only in terms of technical skills, which are presumably higher for Tool Developers than for researchers, but also when considering their unequal scientific background. With a clear inclusive focus, MuGVRE guides and assists the researcher along with online tasks, aided by a rich set of functional metadata that provides framework's customization, data discovery, automatic operation detection, etc. Interface's core is a simple to use and versatile workspace, and features also user's support tools like comprehensive help pages and tutorials or a ticketing system for addressing helpdesk issues to the actual tool developers. Together with the user-friendly one-stop interface, MuGVRE expects to contribute in shaping up the 3D/4D genomics community around the infrastructure. Recently, another relevant implementation has appeared in response to the

integration need of this community. HiC-Explorer [240] also offers a set of 3D chromatin modeling comprehensive tools with some specialized visualizers, in this case, based on a Galaxy server.

Other research domains might find applicability in implementing a modular and integrative platform upon a cloud approach. To this end, openVRE offers a **platform's abstraction** of MuGVRE where Tools, Visualizers, and datasets are decoupled from the core infrastructure to offer a vanilla version of the platform, ready to be adopted by other communities with similar needs. This is the case of OEB-VRE, the VRE for the OpenEBench ELIXIR benchmarking community. It aims to establish a standard methodology to objectively compare software performance. Tools encapsulate benchmarking workflows of scientific methods under evaluation. The advantage of the pluggable system is that the VRE core is unaware of the encapsulated code, which could be a standalone application, or a complex COMPSs workflow that in turn instantiates a transient virtual cluster at runtime. Actually, openVRE flavours (MuGVRE, OEB-VRE) are offering not only a bench of domain-specific tools on top of on-demand resources – *i.e.* a SaaS –, but also a platform where to develop and deploy new analysis methods, a PaaS. In these terms, Galaxy workbench follows a very similar approach, as well as Omics Pipe, both modular data analysis platforms offering comprehensible working environments and implemented as portable infrastructures (see *1.5.2.2* Workbenches). In fact, they even support workflow management, and particularly in Galaxy's case, they feature an important variety of tools. Yet, openVRE better exploits cloud elasticity, as it consumes virtual resources on-demand, as discussed below, while either of these other implementations are in fact static virtual clusters with no dynamic scalability mechanisms. Interestingly, Galaxy Europe has started a pilot project of distributed execution to partially address this issue, by extending the available resources in a community effort contributed by ELIXIR-ES (BSC).

Certainly, scientific communities tackled here are as diverse as present-day, but they exemplify how the proposed infrastructures might act as integrative tools and unifying factors for scientific data management and computational analysis, as well as articulators of new research practices.

Data access and management

The way research is conducted has evolved from simple experiments to computer-assisted computation, from individuals working in isolated laboratories to global networks of researchers collaborating on a single topic. Often, this new paradigm results in important amounts of data, and an extensive network of researchers, which focused the attention into the research data lifecycle, and how to properly manage it.

Efficient and adequate **management of data** is a crucial responsibility of computational infrastructures, especially challenging on distributed computational environments like the cloud. Their services are hosted in virtual environments where the services outlive the

computers they are hosted in - temporarily reside in virtual appliances that go up and down and migrate around different computers. Thus, networking is an intrinsic element of data management systems, and the basis of the proposed solutions. transPLANT stages user's data in and out on every deployment, while openVRE's systems are based on a centralized storage accessible on the network, with a pilot study of being federated.

PMES, as part of the natural tool execution lifecycle, was able to transfer data via FTP(S) into virtual appliances after booting and stage it out before the instance extinction. It does it transparently, and permits data persistence on the pre-emptible computational virtual environments as long as cloud connectivity capabilities allow so. Coupling **data staging with deployment** process provides transPLANT cloud data portability. Summed up to PMES remove invocations, transPLANT computational services achieved full portability, something notably advantageous as it opens the door to practices such as cloud brokering to off-load peak demands or move processing closer to data centres. Furthermore, transPLANT was based on standard protocols and privacy-preserved data sandboxing and transferences. Still and all, the solution entails high data transference rates, one of the major barriers recognized by researchers hindering cloud computing adoption at life sciences [241]. Even being transparent to the user, data exchange represents important overheads. Accordingly, the use of local repositories where public reference data is integrated, limiting the portability of those applications to those cloud installations that pre-allocating the required datasets. Another side effect of loading data into virtual appliances is the need for a dynamic dimensioning of the virtual local disk where files are loaded and output data created. Virtualization and standards like OCCI allow to dimension storage resources on each deployment, yet, the difficulty lays on automatically adjusting the size better fitting the individual run.

Alternatively, data can be centrally allocated in a cloud repository made accessible to virtual machines via contextualization. This is the strategy chosen within openVRE-based infrastructures (MuGVRE, OEB-VRE), where the web application loads **data into a network-attached storage** (NAS) via NFS, and virtual appliances access such data on the same way. Certainly, transferences are minimized, and privacy preserved by automatically contextualizing storage user-specific mounting points. The price to pay is usual NAS network limitations so that processes might become I/O bound unless cloud capabilities include a high-speed bandwidth [242]. And importantly, central data location constrains the deployment of applications to those IaaS where the data is being allocated, limiting elasticity to the resources of such specific infrastructure. The federation of several of these cloud datastores into a single virtual space would represent a step ahead and enable full portability and distribution.

Certainly, inter-cloud data mobilization continues being an open issue in the current cloud market. Security and privacy are important aspects when provisioning federated clouds, as well as performance or networking cost, which usually are not adequate for data-intensive

processing. That's why most federation solutions (see *1.4.4.2* Data-storage solutions) are associated with distributed caching systems. In the MuGVRE pilot installation, oneData [243] stack is investigated to implement a **single virtual data space** among MuG clouds. oneData provides a uniform layering of access on top of the federated cloud datastores. It features a POSIX file system interface that would enable a MuG distributed storage totally transparent to MuGVRE tools, so for virtual environments properly contextualized, user's data would be exposed as a standard file system. Moreover, oneData supports the "copy-on-read" replication strategy, which gives the opportunity to build a data-driven distributed computational infrastructure, where scheduling policies would consider not only job balancing on federated resources, but also data context. As such, data transferences would be reduced as much as possible, dynamically moving computation where data is located. A variety of data-aware schedulers are emerging with the need for running distributed workflows with data-intensive tasks [244]. Indeed, there is a clear need for research data distribution strategies. Galaxy is starting to use CVMFS to generate a virtual repository for access data from their regional servers. The limitations on the management of data ownership of this system restrict it to publicly available reference data.

At centralized platforms with the classical client-server architecture like IMID-clinica and IMID-longitudinal, challenges are more focused on implementing a flexible and extensible design's rationale that leverages data and metadata handling, indeed, the core of any clinical and translational research. Although using a traditional **relational database management system** (RDBMS) like MySQL, IMIDs' infrastructures implement an extensible data model of semi-structured clinical data very in pace with noSQL database engines widely popular in the present days. Based on an entity-value table, all the phenotypical data is coded and dumped in a single MySQL table, retaining data hierarchy based only on the coded identifier instead of the database schema. The non-relational use of the database allows untying the study-associated CRF data model from the data platform, which becomes a generic EDC system for either case-control or longitudinal cohort studies. Scalability and index cardinality are not well exploited under this approach, yet, we are not dealing with the terabyte-scale data. This approach, highly unusual at the time of IMID's design, is adopted in modern epidemiological data managers like Opal [245]. Being based on noSQL technologies, first, data model is imported (a complex dictionary), and afterwards the data is filled in, ready to be annotated and harmonized.

MuGVRE and OEB-VRE are web applications are already using **noSQL technologies**. Mongodb is used as a flexible storage for storing metadata about data and executions of openVREs, which would be difficult to map into relational tables because of its lack of a pre-defined data model – unstructured data. Operational metadata like logging and accounting information is a clear example. Moreover, the engine natively supports data sharding, a horizontal scalability strategy that truly exploits cluster-based installations with heterogeneous nodes as such presented here. The document-oriented model of Mongodb (as JSON) is also terabyte-scale when dealing with complex data models that might be

mapped as programming objects at the application layer. Indeed, openVRE bases its interoperability on Tools and Files data models, functional metadata enclosing prospective and retrospective provenance information, essential for interpreting data, determining ownership, providing reproducible results and fault-tolerance processes. Models are adhered to clear field specifications. Nowadays, is under question the quality of metadata accompanying research data, in spite of the essential role it plays on facilitating effective resource discovery, access, and sharing across distributed digital collections [32]. MuG REST data services make accessible – to authorized users – such rich metadata, so that datasets can be discovered and reused, according to the FAIR guidelines.

The nature of data handled across infrastructures is another factor highly influencing the research plan. **Human sensitive data** is entrusted under stringent legal restraints that restrict the purpose, ownership, security, accessibility, and distribution of data. In studies like GWAS, which use personally identifiable genetic markers of the participants as input, the privacy of the patients becomes of great importance. Such terms are considered when designing the IMID DMP and the specifications for the network of supporting infrastructures, as the biobank and the associated data infrastructures here presented, which store phenotypical data. IMID consortium, as most of the projects building cohort datasets, consume the data it produces and analyse it internally. Consequently, no open data distribution channels nor services are implemented. CRF data model is embodied into the locally accessed DBMS and remains proximal to the analysis unit. The centralized server perfectly serves the purpose of acquiring the data with SSL secured data transfers, patient's identification anonymized, and a central authentication and authorization database. With the emergence of data-centric research, the focus is brought into the uptake of open science practices data. Although such practices are being progressively adopted for large-scale studies, lack of credit and reward, legal, privacy issues or even financial aspects are hampering his adoption at medium/small studies or already structured institutions with legacy consents in place like IMID consortium [34][246]. Accordingly, funding agencies and publishers keenly encourage open access, and numerous long-term supported archives have emerged [247][4][9], along with a multitude of fine-grain sharing models that enable, for instance, closed data - open metadata [35][248]. Yet, in observational studies such as the present, the line between phenotypical data and metadata is blurred.

Infrastructure behind

The transition into the way research is being conducted would not be possible without the adoption of relevant IT technologies consolidated during the last years. Concepts such as virtualization, containerization, SOA or distributed computing are experiencing significant growth at the bioinformatics domain. While these technologies are not necessarily new, it is how they work as a single entity that ensures their effectiveness in front of today's data-driven world. This is happening alongside a tendency toward automating and managing infrastructures for availability, reproducibility, and scalability.

Such transition is well visible in the technologies in use at the proposed platforms. IMIDs' data platforms follow the most widely used and classical three-tier application architecture: a thick client server that includes the logic and presentation, and a data tier managed by a relational DBMS, MySQL. On the other hand, the developed computational infrastructures are based on the increasingly popular cloud computing model. Presentation layer corresponds to web applications or SOAP/REST APIs, who are also responsible for the logics tier together with other modules – *i.e.* remote job orchestrators and elastic workload managers, components of PaaS. Data-tier is composed of mass storage, a non-relational Mongodb database, and the virtual machines that conform the computing building blocks. Each platform is appropriately designed according to the system's needs, avoiding overreliance on novel technologies unless justified.

While using **traditional technologies**, IMIDs' applications accomplish with the requirements dictated by the DMP and compliant with ISO 9001 in a flexible and extensible manner. As data platforms, their purpose is to conveniently consolidate data and securely store it. Although offering a custom interface, the system is designed to be highly flexible by decoupling the clinical case report (CRF) design from the application itself. As such, rapid data prototyping and low maintenance tasks are achieved. It implies the use of a template engine, in-house designed, for rendering user's interface layout based on clinical variables in bulk – pioneer strategy at 2009. Prove of such adaptability is the fact that, albeit continuous renovations and extensions, IMID-clinica has been giving service to the active IMID consortium for nearly 10 years. IMID-longitudinal is actively collecting data for 2 years now.

transPLANT infrastructure represents a totally different paradigm. **Cloud computing** offers a compelling alternative for computational infrastructures in front of traditionally based HPC or grid systems, with the possibility to instantiate and configure in a multi-user environment on-demand resources such as virtual computers, storage, operating systems, and software tools. For scientists, this constitutes a shift from the model of owning computer hardware with the explicit need to then configure connectivity, necessary datastores software or system libraries, which in many cases constitutes a demanding and time-consuming task [122]. Infrastructures as a service (IaaS) might absorb such complexity. Along the present work, two different cloud philosophies are used, the first corresponds to the two OpenNebula cluster-based installations sitting at the Barcelona Supercomputing Center (BSC) and at the Institute of Biomedical Research (IRB), and the second to the *Embassy Cloud* tenancy hosted at the European Bioinformatics Institute (EMBL-EBI) and managed by OpenStack. Both models serve different needs, while OpenStack CMP is more suitable for "Infrastructure provisioning" clouds mostly designed for public use, OpenNebula is devised as a "datacenter virtualization" cloud better targeting on-premise clouds, like those at BSC and IRB, meant to cover institution's cloud needs. OpenStack offers a simplified view of life-cycle's virtual resources and its underlying infrastructure while providing a complete set of data and network services, integrated as independent modules

from several vendors – some, proprietary solutions. OpenNebula provides a more transparent view of physical resources, works more like a IT-as-a-Service inside the organization, it is highly compliant with open cloud standards, features some tool for orchestration and networking automation, but is more focused on server consolidation, which makes it convenient to support legacy infrastructure components. Furthermore, it is easy to be adapted atop existing cluster or HPC facilities, installing a classical cluster-like cloud architecture (*i.e.* cloud frontend and heterogeneous compute nodes behind). Bearing these considerations, MuG, OpenEbench, and other institution's projects, are managed by the cost-effective OpenNebula CMP, while *Embassy Cloud*, fully dedicated to provide IaaS to a large number of scientific projects across Europe, is based on OpenStack.

Hand in hand with cloud computing, the new computing model fostered the adoption of virtualization technologies on research environment, like **virtual machines** or software containers, as well as service architectures, which lead to two scientific major advantages: reproducibility and global access. Virtual machines are the building blocks of transPLANT and openVRE platforms, they are the deployment units encapsulating Tool Developer's code – as well as other infrastructure's software stack. VMs play a fundamental role in providing the custom and optimal hosting environment adapted to each users' application and, at the same time, isolate the execution from the rest of the infrastructure, thus preventing applications from harming the hosting environment or interfering with other executions. In fact, the same premises are behind the idea of virtual containers instead of full virtual machines, which entail some tweaks and additions to Linux kernel's "chroot" commands so that host and container share the same OS. In this way, they are more lightweight solutions with less boot/termination overheads and fewer resource consumption, simpler development processes and easily automatable with popular orchestration tools like Kubernetes or Docker Swarm. Still, there are a few limitations when using containers that constrain their use in multi-user infrastructures with shared facilities like BSC's or IRB's. Containers provide weak host isolation as compared to virtual machines. Sharing OS impacts kernel's security, complicate setting restricted network accesses, and importantly, it is risky to run them with privilege mode, unavoidable when installing most of the analysis software and libraries, as containers are daemon processes that might exploit such permissions to harm the host OS [249]. Actually, in order to mitigate security issues, it is quite common the use of containers nested in VMs instead of running on bare metal, off course, in performance exchange [250]. Popular PaaS like Cloud Foundry or OpenShift does so. transPLANT and MuG components, as well as current cloud standards, perfectly interact with transient stateless, compute VMs, and some of their limitations are partially addressed by the platform's design. For instance, MuGVRE includes not only PMES transient VMs but also permanent instances reached by a queuing system, SGE. These type of instances are conceived to encapsulate low-demanding applications by which a full VM deployment would represent an important overhead over a short and quick execution, while permanently allocated resources were only a minor handicap. Moreover, preparing

ready-to-use base images results in a rapid Tool prototyping, in particular when aided by PaaS programming models or adaptor like MG-TOOL-API. At the same time, the intrinsic portability of VMs is enhanced, using a standard image format (QCOW2) and a widely-used contextualization package as Cloud-init, supported by several CMPs. Finally, is worth mentioning that reproducibility is enabled at the VM level by design, indeed, VMs are blocked after the platform's integration. Yet, the ultimate responsibility on application reproducibility relies on the developer and how its tool or workflow are encapsulated.

Whatever is the job processor mechanism, PaaS components make sure to exploit at best the available cloud resources, as both, transPLANT and openVRE platforms, build private clusters of VM **instances on-demand,** without expecting administrators start, manage or resize them. Virtual appliances are built and destroyed dynamically thanks to automatic provisioning tools, and hence, a very efficient use of the existing hardware infrastructure is achieved, especially relevant on public clouds where you pay only on what you use. Elasticity is then enabled by PMES and/or COMPSs, capable of dynamically allocating resources only as they are needed. On the other hand, for tools launched using the traditional SGE queue system, an auto-scalability mechanism called OneFlow is coupled, so that the number of hosts backing the queues transparently increases and decreases based on system's workload. Not many platforms include this type of "just-in-time" provision. An example would be CloudVR (*1.4.3.5* Bioinformatics in the cloud), yet it is distributed as a stand-alone IaaS.

Cloud popularity growth in research environments is undeniable, but scalable resource outsourcing is not free of **downsides**. Funding agencies need to consider resource tenancy pricings as applicable, although as discussed, open-source academic cloud providers are growing and achieving maturity. At the same time, they also contribute to dilute the concern on vendor lock-in, while diversifying providers and adopting open standards like those adopted here. It enables interoperability and portability, enabling brokering strategies like hybrid and multi-cloud strategies. EOSC development partially addresses these concerns, but a clear approach is not yet defined. Yet, major concerns are regarding security and privacy issues when dealing with sensitive data, and data transference and archiving when handling large amounts of data. The overhead represented by data exchange might be very relevant, and as discussed above, seeking proximity between computation and storage is a usual strategy to minimize them. Meanwhile, transparency, confidentiality, and control are central concerns for data protection. Currently, there are emerging protocols that make possible to craft cloud-based setups compliant with data protection laws and regulations [251], [252] while GÉANT is starting to provide dedicated private networks under request [253].

Actually, **security** is a common denominator across infrastructure services. ELSI (Ethical Legal Social Implications) and GDPR (General Data Protection Regulation) are aspects every day more relevant on research for those platforms handling and processing user's data. For

addressing them, platforms include measures like the establishment of a sign-in process where the user has to read and understand the ELSI constraints and best practices, data registry which contains information on the test datasets, or ownership visibility of the openVRE integrated tools. Technically, virtualization layer provides isolation, web accesses are secured (SSL, firewalls), and **authentication** services are put in place: IMIDs' platforms have a local authentication based on their DBMSs; transPLANT cloud is based on an LDAP centralized authentication server enabling standard POSIX access; and openVREs' clouds implement a single sign-on (SSO) based on OpenID protocol with OAuth2 authentication tokens that integrate third-party identity providers like Google, ORCID or ELIXIR. The use of standard and widely used protocols favor the integration with other infrastructures, *i.e.* ELIXIR identity, that permits most users sign-on using their home institution credentials. It opens the door to exploit openID scopes for authorization flows, not only at the application level (*i.e.* mapping oneData datastores into VRE workspaces), but also across other European applications like EGI virtual organizations (VOs) or PRACE resources.

Platform model

transPLANT and openVRE computational platforms here presented follows a **PaaS model**, more specifically, a "PaaS for SaaS", as the ultimate goal of the developed applications are to be offered as SaaS, either through PMES dashboard/API or from the corresponding VRE.

PaaS is characterized to offer a resourceful approach to develop, operate and deploy applications at the expense of low cost and time. Certainly, we offer Tool Developers an offline development process based on a ready-to-use VMI contextualized to be interoperable with our PaaS components. Developers are not responsible nor aware, of the infrastructure layer, data staging, service connectivity, etc. Such layers are controlled by the platform.

Here lays the difference with other PaaSs that facilitate the development of fully standalone applications, including the definition of a web server, IP address, ports, etc. Developers require much more knowledge about the underlying infrastructure, so configuration is more complex, and the resulting application is not meant to be part of any framework [254]. Indigo cloud and Phenomenal, differences aside, belong to this PaaS sub-model, characterized to provide a **Desktop-as-a-Service**, a full research environment on each deployment. Indigo developed their own PaaS orchestrator, Phenomenal uses Terraform, and in both cases, the developer configures Indigo/Phenomenal developed pieces using IaC technologies to provide a custom virtual framework for the researcher. Indigo focuses more on easing PaaS-IaaS interaction, not particularly offering VREs as the final deployed application, but any type of custom virtual framework. On the contrary, Phenomenal focuses on offering VRE instances tailored for metabolomics research.

In our case, deployed applications correspond to bioinformatics tools intended to be integrated into a VRE – **PaaS for SaaS**. UI, web server, logging, accounting, or error handling

are some of the transversal services covered by openVRE that permit users to concentrate on developing the actual tool. Importantly, VREs add extra value to each individual application derived from bringing together interrelated analysis tools which might interoperate, take profit of similar data sources or make use of specialized visualizers. Interoperability, visibility, reusability, and collaboration are the appealing elements that are expected to attract and recruit developers to become part of community-driven computational platforms. Galaxy is the most mature platform showing a similar deployment model. The flip side of the coin is the adoption of restrictions derived from being part of an overall framework flow. Presented computational infrastructures support applications in batch mode, with no user interactivity at runtime, and file-based input and output data. Nucleosome Dynamics tool suite [219] (attached at *8.7* Publications), is a good representative of this software releasing tendency by which the developer concentrates on the actual application, instead of developing and maintaining their own web application to make the new software available. For instance, Nucleosome Dynamics is a set of R packages implemented as both, MuGVRE and Galaxy tools. Such practice aims to find synergies, economize time, and integrate and standardize methods.

PaaS orchestrators, or simply the systems centralizing application's control, are aware of the applications only after an installation procedure. In openVRE derived platforms, the **platform integration** is facilitated by MG-tool-API, that provides a Tool skeleton with some auxiliary methods to facilitate the testing and development process, *i.e.* parsing and validation of arguments and inputs, logging, output metadata annotation, etc. Still, the code is pure python, and the application would be totally functional and autonomous outside openVRE environment. The use of adaptors is a practice increasingly extended in front of the plethora of novel task managers and orchestrators. They permit the creation of building blocks (here VRE Tools) uniformly accessed that easily can be translated into other frameworks. BioExcel Building Blocks [255] (annexed in *8.7.3* Thesis not related papers) is an example that follows this practice. Galaxy integration is someway similar. The developer prepares a Tool wrapper to interact with Galaxy API. Yet, the wrapper uses an XML-based domain-specific language (DSL), so that the resulting application cannot operate outside Galaxy environment. The reason is that the Galaxy Tool wrapper not only acts as CLI adaptor, but also as tool registry and UI template, functions that at openVRE or transPLANT are separately accomplished by storing tool metadata as database entries. In this way, openVRE achieves greater flexibility as Tool configuration is centralized in the database entry, while Tool implementation can live outside the ecosystem.

Sustainability

Sustainability represents a challenge within the existing funding structures. Long-term preservation of data and services after a project and its funding have ended is one of the major infrastructure dilemmas. Open Science implies, among other things, the optimal accessibility and sharing of research data, but without sustained services, these activities

are hardly feasible. The main approach is the roadmap synchronization with other national or European research infrastructures.

BSC is in close relationship with European and global tendencies for **cloud computing in life sciences.** In particular, we are active in the European bioinformatics infrastructure ELIXIR, the European Open Science Cloud Initiative (EOSC), and the Global Alliance for Genomics and Health (GA4GH). Although a federated cross-border cloud ecosystem is an ambitious target, these initiatives are in the way of coordinating national infrastructures and setting up standard specifications [154]. Presented designs align with these driver projects' interests, in particular, the cloud-based VRE. The global layout of openVRE infrastructures, including distributed data and execution, are fully aligned with the combined plans of the ELIXIR compute platform and the GA4GH, who is settling specifications and frameworks for tools' registry (TRS), task executors (TES), and data services (DRS). This combined work is the basis of cloud computing management for life science at EOSC. One-to-one, the presented cloud components align such standards - *i.e.* Tool DB corresponds to a TRS implementation; PMES is a TES, and MuG data services might map DRS specifications. Actually, we are working on wrapping such functionalities under REST APIs following GA4GH specifications. Decomposing openVRE into such functionalities would allow the integration of the cloud-based backend into other platforms, like Galaxy. Some particular transPLANT applications were already accessible using the Galaxy UI, yet doing so via standard GA4GH mechanisms would ensure a more stable and sustainable integration.

In a clear next step, seeking interactions with complementary infrastructures helps to build a cohesive network of research infrastructures. Therefore, future plans include supporting the full **research data life cycle**, including data sharing and publication. Onedata is the base technology used in well-stablished data infrastructures like EGI-DataHub or INDIGO data cloud. This reinforces the decision of employing it for data sharing across geographically distributed openVRE clouds, as an attempt to proximate data and compute resources. In turn, OneData, opposite to other approaches like Galaxy's CVMFS, provides a fine-grain access control based on a OpenID custom method, fundamental to share dataset private sand full workspaces with other researchers on the platform. Moreover, the data cycle would be complete if generated datasets could be published to dedicated long-term archives, like EGA, EUDAT or DOI-issuing platforms like Zenodo. Thus, we are exploring the possibility to include submission portals based on REST APIs to these relevant data e-infrastructures.

Security is another design principle that permits to encompass technical and software services and processes. Committing MuGVRE to ELIXIR AAI integrates the platform into a common access management system, while helping to build and maintain trust in the infrastructure.

Regarding interoperability, cloud computing is a relatively new IT industry and has not yet been fully unified and standardized. Even so, proposed PaaS components are **compliant**

with well stablished specification widely supported. OCCI interacts with the cloud broker and handles deployment and provisioning of infrastructure resources. Cloud-init has evolved as *de facto* standard for virtual instances customization, and enables the integration with configuration management solutions like Ansible, Puppet of Chef, if needed. Remote job submission uses grid-inherited specifications, like BES flow and JSDL templates, supported by popular workload managers like LSF or UNICORE. At the application level, portability is ensured by VM's sandboxing, yet, tool description is enclosed in an *ad hoc* domain-specific data model – *i.e.* our Tool data model. There are some specifications available covering similar processes, yet, none did fully meet our requirements. For instance, INDIGO data cloud uses TOSCA, but it provides a language to describe the application deployment as an orchestration of service components (*e.g.* Tomcat server, database instance), disregarding the application CLI per se as is INDIGO does not undergo batch executions. The command line tool description of CWL would better fit the purpose, yet it does not cover deployment-related details, only partially included for applications with container-based dependencies, and do not consider retrospective file metadata like "Data Type" – only format is considered. As such, and in spite of the undeniable benefits of adopting community standards, we are right now preserving our data model for representing our PaaS applications, while using, however, widely spread technologies like JSON and versioned JSON Schemas. Other platforms, like Galaxy, introduced their own DSL to cover their infrastructure needs.

6. Conclusions

Following are enumerated the conclusions of the present dissertation

1. We have designed and implemented *IMID-clinica* and *IMID-longitudinal*, web-based clinical data management systems for the recollection of phenotypical data on Immune Mediated Inflammatory diseases (IMIDs) across more than 90 health centers. The system offers a user-friendly and familiar interface and reinforce research quality data measures, easing the management of large scale data collections.

2. The use of a non-relational data models (even on top of a traditional SQL database manager) allows to decouple clinical record schemas from the database design, and contributes to gain flexibility in accommodating different epidemiological case studies. Clinical data, stored as simple key-value pairs, can be easily used in multi-cohort complex analysis.

3. We have assembled the compute platform for the transPLANT project as a flexible and portable cloud-based infrastructure where plant genomics tools and pipelines were packed and offered either programmatically or via web-based applications

4. TransPLANT cloud was the initial setup including all the necessary components for an integrated user experience, like data manager, software schedulers, and a unified procedure to host analysis applications in a portable structure

5. The adoption of virtual machines as compute units, together with a set of middleware able to provision them on-demand offer fully personalized, portable, and secure environments where platform's components conveniently scale both, vertically and horizontally. As such, an elastic and cost-efficient is enabled.

6. We have designed and developed the Virtual Research Environment for the MuG project (*MuGVRE*), a web-based framework seamlessly integrating cloud-based resources for a selection of data, visualization, and analysis services relevant for the 3D genomics community.

7. The use of a Platform-as-a-Service (PaaS) model, with pluggable tools contributed by the research software developers, facilitates that such developers get abstracted from the administrative and deployment process and concentrate on the scientific implementation. The application *Nucleosome Dynamics* has benefited from this model, being offered through the MuGVRE platform.

8. The VRE concept has evolved toward the building of *openVRE*, a neat backbone for a rapid prototyping of Virtual Research Environments on the cloud. As a first adopter, we have implemented *OEB-VRE*, an integrative platform for scientific benchmarking.

7. References

[1] M. O. Dayhoff and R. Ledley, “COMPROTEIN, a Computer Program to Aid Primary Protein Structure Determination,” in *Managing Requirements Knowledge, International Workshop*, 1962, p. 262.

[2] N. M. Luscombe, D. Greenbaum, and M. Gerstein, “What is bioinformatics? A proposed definition and overview of the field.,” *Methods Inf. Med.*, vol. 40, no. 4, pp. 346–58, 2001.

[3] C. Kanz *et al.*, “The EMBL Nucleotide Sequence Database,” *Nucleic Acids Res.*, vol. 33, no. Database issue, pp. D29–D33, Dec. 2004.

[4] K. Clark, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers, “GenBank.,” *Nucleic Acids Res.*, vol. 44, no. D1, pp. D67-72, Jan. 2016.

[5] H. M. Berman *et al.*, “The Protein Data Bank,” *Nucleic Acids Res.*, vol. 28, no. 1, pp. 235–242, Jan. 2000.

[6] J. B. Hagen, “The origins of bioinformatics,” *Nat. Rev. Genet.*, vol. 1, no. 3, pp. 231–236, Dec. 2000.

[7] J. Gauthier, A. T. Vincent, S. J. Charette, and N. Derome, “A brief history of bioinformatics,” *Brief. Bioinform.*, Aug. 2018.

[8] K. D. Pruitt, T. Tatusova, W. Klimke, and D. R. Maglott, “NCBI Reference Sequences: current status, policy and new initiatives,” *Nucleic Acids Res.*, vol. 37, no. Database, pp. D32–D36, Jan. 2009.

[9] H. Parkinson *et al.*, “ArrayExpress update--an archive of microarray and high-throughput sequencing-based functional genomics experiments,” *Nucleic Acids Res.*, vol. 39, no. Database, pp. D1002–D1004, Jan. 2011.

[10] T. Barrett *et al.*, “NCBI GEO: archive for functional genomics data sets--10 years on,” *Nucleic Acids Res.*, vol. 39, no. Database, pp. D1005–D1010, Jan. 2011.

[11] G. D. Bader, I. Donaldson, C. Wolting, B. F. Ouellette, T. Pawson, and C. W. Hogue, “BIND--The Biomolecular Interaction Network Database,” *Nucleic Acids Res.*, vol. 29, no. 1, pp. 242–245, Jan. 2001.

[12] A. Ceol *et al.*, “MINT, the molecular interaction database: 2009 update,” *Nucleic Acids Res.*, vol. 38, no. suppl_1, pp. D532–D539, Jan. 2010.

[13] M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, and M. Hirakawa, “KEGG for representation and analysis of molecular networks involving diseases and drugs,” *Nucleic Acids Res.*, vol. 38, no. suppl_1, pp. D355–D360, Jan. 2010.

[14] L. H. Greene *et al.*, “The CATH domain structure database: new protocols and classification levels give a more comprehensive resource for exploring evolution,” *Nucleic Acids Res.*, vol. 35, no. Database, pp. D291–D297, Jan. 2007.

[15] M. Ashburner *et al.*, “Gene Ontology: tool for the unification of biology,” *Nat. Genet.*, vol. 25, no. 1, pp. 25–29, May 2000.

[16] E. W. Sayers *et al.*, “Database resources of the National Center for Biotechnology Information,” *Nucleic Acids Res.*, vol. 39, no. Database, pp. D38–D51, Jan. 2011.

[17] D. J. Rigden and X. M. Fernández, “The 2018 Nucleic Acids Research database issue and the online molecular biology database collection,” *Nucleic Acids Res.*, vol. 46, no. D1, pp. D1–D7, Jan. 2018.

[18] H. J. Imker, “25 Years of Molecular Biology Databases: A Study of Proliferation, Impact, and Maintenance,” *Front. Res. Metrics Anal.*, vol. 3, p. 18, May 2018.

[19] J. Ison *et al.*, “Tools and data services registry: a community effort to document bioinformatics resources,” *Nucleic Acids Res.*, vol. 44, no. D1, pp. D38–D47, Jan. 2016.

[20] Z. D. Stephens *et al.*, “Big Data: Astronomical or Genomical?,” *PLOS Biol.*, vol. 13, no. 7, p. e1002195, Jul. 2015.

[21] L. Stein, “Creating a bioinformatics nation.,” *Nature*, vol. 417, no. 6885, pp. 119–20, May 2002.

[22] A. Trefethen, “e-Science and its implications,” *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*

[23] P. R. Titilade and E. I. Olalekan, “The Importance of Marine Genomics to Life,” *J. Ocean Res.*, vol. 3, no. 1, pp. 1–13, 2015.

[24] D. Tzovaras, “Overview of the European strategy in research infrastructures,” *CEUR Workshop Proc.*, vol. 1752, pp. 187–194, 2016.

[25] IRG, “Guide to e-Infrastructure Requirements for European Research Infrastructures,” e-IRG, 2017.

[26] ELIXIR, “A distributed infrastructure for life-science information.” [Online]. Available: https://elixir-europe.org/. [Accessed: 12-Sep-2019].

[27] INB, “National Institute of Bioinformatics.” [Online]. Available: https://inb-elixir.es/. [Accessed: 12-Sep-2019].

[28] ELIXIR, “Core Data Resources,” 2019. [Online]. Available: https://elixir-europe.org/platforms/data/core-data-resources. [Accessed: 15-Aug-2019].

[29] S. O. M. Dyke *et al.*, “Registered access: authorizing data access,” *Eur. J. Hum. Genet.*, vol. 26, no. 12, pp. 1721–1731, Dec. 2018.

[30] GA4GGH, “Golbal Alliance for Genomics and Heath.” [Online]. Available: https://www.ga4gh.org/. [Accessed: 12-Sep-2019].

[31] X. Yang *et al.*, “Cloud computing in e-Science: research challenges and opportunities,” *J. Supercomput.*, vol. 70, no. 1, pp. 408–464, Oct. 2014.

[32] R. S. Gonçalves and M. A. Musen, “The variable quality of metadata about biological samples used in biomedical experiments,” *Sci. Data*, vol. 6, no. 1, p. 190021, Mar. 2019.

[33] G. Popkin, “Data sharing and how it can benefit your scientific career,” *Nature*, vol. 569, no. 7756, pp. 445–447, May 2019.

[34] C. L. Borgman, “The conundrum of sharing research data,” *J. Am. Soc. Inf. Sci. Technol.*, vol. 63, no. 6, pp. 1059–1078, Jun. 2012.

[35] M. Fiume *et al.*, “Federated discovery and sharing of genomic data using Beacons,” *Nat. Biotechnol.*, vol. 37, no. 3, pp. 220–224, Mar. 2019.

[36] FAIRsharing Databases, “Catalogue data repositories.” [Online]. Available: https://fairsharing.org/database. [Accessed: 12-Sep-2019].

[37] re3data.org, “Registry of Research Data Repositories.” [Online]. Available: https://www.re3data.org/. [Accessed: 12-Sep-2019].

[38] M. D. Wilkinson *et al.*, “The FAIR Guiding Principles for scientific data management and stewardship,” *Sci. Data*, vol. 3, no. 1, p. 160018, Dec. 2016.

[39] FORCE11, “The FAIR Data Principles.” [Online]. Available: https://www.force11.org/group/fairgroup/fairprinciples. [Accessed: 13-Sep-2019].

[40] The Software Sustainability Institute, “Research Software Healthcheck.” [Online]. Available: https://www.software.ac.uk/. [Accessed: 27-Sep-2019].

[41] R. Gupta, H. Gupta, and M. Mohania, “Cloud Computing and Big Data Analytics: What Is New from Databases Perspective?,” 2012, pp. 42–61.

[42] M. Baker, “1,500 scientists lift the lid on reproducibility,” *Nature*, vol. 533, no. 7604, pp. 452–454, May 2016.

[43] R. Das *et al.*, “Structure prediction for CASP7 targets using extensive all-atom refinement with Rosetta@home,” *Proteins Struct. Funct. Bioinforma.*, vol. 69, no. S8, pp. 118–128, Jan. 2007.

[44] N. J. Marianayagam, N. L. Fawzi, and T. Head-Gordon, “Protein folding by distributed computing and the denatured state ensemble.,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 102, no. 46, pp. 16684–9, Nov. 2005.

[45] Curecoin, “curecoin.net.” [Online]. Available: https://curecoin.net/.

[46] Nebula, “nebula.org.” [Online]. Available: https://nebula.org/.

[47] H. I. Ozercan, A. M. Ileri, E. Ayday, and C. Alkan, “Realizing the potential of blockchain technologies in genomics.,”

Genome Res., vol. 28, no. 9, pp. 1255–1263, Aug. 2018.

[48] J. P. Ziebarth and S. Carruth, “The role of high performance computing in precollege education,” pp. 38–41, 2002.

[49] F. Marozzo and P. Trunfio, “Infrastructures for High-Performance Computing: Cloud Computing Development Environments,” *Encycl. Bioinforma. Comput. Biol.*, pp. 247–251, Jan. 2019.

[50] T. G. Peter Mell, “The NIST Definition of Cloud Computing.”

[51] M. L. Bote-Lorenzo, Y. A. Dimitriadis, and E. Gómez-Sánchez, “Grid Characteristics and Uses: A Grid Definition,” Springer, Berlin, Heidelberg, 2004, pp. 291–298.

[52] I. Foster, “Globus Toolkit Version 4: Software for Service-Oriented Systems,” *J. Comput. Sci. Technol.*, vol. 21, no. 4, pp. 513–520, Jul. 2006.

[53] UNICORE, “Distributed computing and data resources,” 2019. [Online]. Available: https://www.unicore.eu/. [Accessed: 15-Aug-2019].

[54] TORQUE, “Resource Manager.” [Online]. Available: http://www.adaptivecomputing.com/products/torque/. [Accessed: 13-Aug-2019].

[55] SLURM, “Workload Manager.” [Online]. Available: https://slurm.schedmd.com/documentation.html. [Accessed: 13-Aug-2019].

[56] T. Tannenbaum, D. Wright, K. Miller, and M. Livny, “Condor—A Distributed Job Scheduler,” *Beowulf Clust. Comput. with Linux*, 2001.

[57] OGS, “Open Grid Scheduler. Former Sun Grid Engine.” [Online]. Available: https://sourceforge.net/projects/gridscheduler/. [Accessed: 13-Aug-2019].

[58] A. Anjomshoaa *et al.*, “Job Submission Description Language (JSDL) Specification, Version 1.0,” *GFD-R.056*, pp. 1–72, 2006.

[59] M. Morgan, U. Virginia, S. Newhouse, U. Southampton, D. Pulsipher, and M. Theimer, “OGSA Basic Execution Service,” *Response*, 2008.

[60] QEMU, “QEMU,” 2019. [Online]. Available: https://www.qemu.org/. [Accessed: 03-Aug-2019].

[61] Xenserver, “Citrix Hypervisor | Open Source Server Virtualization,” 2019. [Online]. Available: https://xenserver.org/. [Accessed: 03-Aug-2019].

[62] VMware, “VMware – Official Site,” 2019. [Online]. Available: https://www.vmware.com/. [Accessed: 03-Aug-2019].

[63] Virtual PC, “Download Windows Virtual PC from Official Microsoft Download Center,” 2019. [Online]. Available: https://www.microsoft.com/es-es/download/details.aspx?id=3702. [Accessed: 03-Aug-2019].

[64] VirtualBox, “Oracle VM VirtualBox,” 2019. [Online]. Available: https://www.virtualbox.org/. [Accessed: 03-Aug-2019].

[65] OpenVz, “Open source container-based virtualization for Linux.,” 2019. [Online]. Available: https://openvz.org/. [Accessed: 03-Aug-2019].

[66] Linux Containers, “Linux Containers,” 2019. [Online]. Available: https://linuxcontainers.org/. [Accessed: 03-Aug-2019].

[67] Docker, “Enterprise Container Platform | Docker,” 2019. [Online]. Available: https://www.docker.com/. [Accessed: 03-Aug-2019].

[68] Winehq, “WineHQ - Ejecuta aplicaciones de Windows en Linux, BSD, Solaris y macOS,” 2019. [Online]. Available: https://www.winehq.org/. [Accessed: 03-Aug-2019].

[69] vCuda, “GPU Virtualization using vCuda / Wiki / Home,” 2019. [Online]. Available: https://sourceforge.net/p/gpuvirtualize/wiki/Home/. [Accessed: 03-Aug-2019].

[70] KVM, “KVM,” 2019. [Online]. Available: https://www.linux-kvm.org/page/Main_Page. [Accessed: 03-Aug-2019].

[71] ESXi, “ESXi | Hipervisor sin sistema operativo | VMware,” 2019. [Online]. Available: https://www.vmware.com/es/products/esxi-and-esx.html. [Accessed: 03-Aug-2019].

[72] Microsoft Hyper-V, “Hyper-V en Windows 10 | Microsoft Docs,” 2019. [Online]. Available: https://docs.microsoft.com/es-es/virtualization/hyper-v-on-windows/. [Accessed: 03-Aug-2019].

[73] D. Merkel, “Docker: Lightweight Linux Containers for Consistent Development and Deployment,” *Linux J.*, vol. 2014, no. 239, 2014.

[74] M. Helsley, “LXC: Linux container tools,” *IBM devloperWorks Tech. Libr.*, vol. 11, 2009.

[75] G. M. Kurtzer, V. Sochat, and M. W. Bauer, “Singularity: Scientific containers for mobility of compute,” *PLoS One*, vol. 12, no. 5, p. e0177459, May 2017.

[76] Flannel, “Documentation,” 2019. [Online]. Available: https://coreos.com/flannel/docs/latest/. [Accessed: 03-Aug-2019].

[77] Calico, “About Calico,” 2019. [Online]. Available: https://docs.projectcalico.org/v3.8/introduction/. [Accessed: 03-Aug-2019].

[78] Kube-Router, “Kube-Router : Turnkey Kubernetes networking solution,” 2019. [Online]. Available: https://www.kube-router.io/. [Accessed: 03-Aug-2019].

[79] Container Network Interface, “GitHub - containernetworking/cni: Container Network Interface - networking for Linux containers,” 2019. [Online]. Available: https://github.com/containernetworking/cni. [Accessed: 03-Aug-2019].

[80] Cisco Open SDN controller, “Cisco Open SDN Controller - Cisco,” 2019. [Online]. Available: https://www.cisco.com/c/en/us/products/cloud-systems-management/open-sdn-controller/index.html. [Accessed: 03-Aug-2019].

[81] Opendaylight, “Home - OpenDaylight,” 2019. [Online]. Available: https://www.opendaylight.org/. [Accessed: 03-Aug-2019].

[82] A. Joshua and F. Ogwueleka, “Cloud Computing with Related Enabling Technologies,” *Int. J. Cloud Comput. Serv. Sci.*, vol. 2, no. 1, Oct. 2012.

[83] Ironic, “Ironic - OpenStack,” 2019. [Online]. Available: https://wiki.openstack.org/wiki/Ironic. [Accessed: 03-Aug-2019].

[84] D. M. Smith, “Hype Cycle for Cloud Computing , 2011,” *Gart. Inc., Stamford*, vol. 71, no. July, pp. 1–74, 2011.

[85] I. Gartner, “Market Share Analysis: IaaS and IUS, Worldwide, 2018.” [Online]. Available: https://www.businesswire.com/news/home/20190729005169/en/Gartner-Worldwide-IaaS-Public-Cloud-Services-Market. [Accessed: 28-Sep-2019].

[86] A. Matsunaga, M. Tsugawa, and J. Fortes, “CloudBLAST: Combining MapReduce and Virtualization on Distributed Resources for Bioinformatics Applications,” in *2008 IEEE Fourth International Conference on eScience*, 2008, pp. 222–229.

[87] Heroku, “Cloud Application Platform | Heroku,” 2019. [Online]. Available: https://www.heroku.com/. [Accessed: 03-Aug-2019].

[88] OpenShift, “Container Application Platform by Red Hat, Built on Docker and Kubernetes,” 2019. [Online]. Available: https://www.openshift.com/. [Accessed: 15-Aug-2019].

[89] Nimbus, “Nimbus is cloud computing for science.” [Online]. Available: http://www.nimbusproject.org/. [Accessed: 12-Sep-2019].

[90] Eucalyptus, “Eucalyptus open-source cloud.” [Online]. Available: https://www.eucalyptus.cloud/. [Accessed: 12-Sep-2019].

[91] OpenNebula, “Cloud Management Platform.” [Online]. Available: https://opennebula.org/. [Accessed: 11-Sep-

2019].

[92] OpenStack, “Build the future of Open Infrastructure.” [Online]. Available: https://www.openstack.org/. [Accessed: 11-Sep-2019].

[93] Apache CloudStack:, “Open Source Cloud Computing.” [Online]. Available: https://cloudstack.apache.org/. [Accessed: 12-Sep-2019].

[94] Xen Cloud Platform, “Open source software to build private and public clouds.” [Online]. Available: http://www-archive.xenproject.org/products/cloudxen.html. [Accessed: 12-Sep-2019].

[95] S. Ismaeel, A. Miri, D. Chourishi, and S. M. R. Dibaj, “Open Source Cloud Management Platforms: A Review,” in *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*, 2015, pp. 470–475.

[96] S. Shahzadi, M. Iqbal, Z. U. Qayyum, and T. Dagiuklas, “Infrastructure as a service (IaaS): A comparative performance analysis of open-source cloud platforms,” in *2017 IEEE 22nd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2017, pp. 1–6.

[97] M. Mahjoub, A. Mdhaffar, R. Ben Halima, and M. Jmaiel, “A Comparative Study of the Current Cloud Computing Technologies and Offers,” in *2011 First International Symposium on Network Cloud Computing and Applications*, 2011, pp. 131–134.

[98] V. Atanasovski and A. Leon-Garcia, *Future Access Enablers for Ubiquitous and Intelligent Infrastructures. Chapter: Cloud portability*, vol. 159. Cham: Springer International Publishing, 2015.

[99] Open Grid Forum, “Open Grid Forum: OCCI, Open Cloud Computing Interface.” [Online]. Available: http://occi-wg.org/. [Accessed: 15-Aug-2019].

[100] OASIS TOSCA, “Topology and Orchestration Specification for Cloud Applications Version 1.0.” [Online]. Available: http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.html. [Accessed: 12-Sep-2019].

[101] SNIA CDMI, “Cloud Data Management Interface.” [Online]. Available: https://www.snia.org/cdmi. [Accessed: 12-Sep-2019].

[102] CNCF Cloud Native Computing Fundation, “Cloud Native Interactive Landscape.” [Online]. Available: https://landscape.cncf.io/. [Accessed: 25-Sep-2019].

[103] OpenStack Nova, “OpenStack Compute Nova.” [Online]. Available: https://docs.openstack.org/nova/latest/. [Accessed: 15-Sep-2019].

[104] OneFlow, “Services Management — OpenNebula 5.6.2 documentation.” [Online]. Available: https://docs.opennebula.org/5.6/advanced_components/application_flow_and_auto-scaling/appflow_use_cli.html. [Accessed: 11-Sep-2019].

[105] F. Lordan *et al.*, “ServiceSs: An Interoperable Programming Framework for the Cloud,” *J. Grid Comput.*, vol. 12, no. 1, pp. 67–91, Mar. 2014.

[106] cloud-init, “The standard for customising cloud instances.” [Online]. Available: https://cloud-init.io/. [Accessed: 15-Sep-2019].

[107] one-context, “Linux VM Contextualization for OpenNebula.” [Online]. Available: https://github.com/OpenNebula/addon-context-linux. [Accessed: 15-Sep-2019].

[108] amiconfig, “Contextualization client for Amazon EC2.” [Online]. Available: https://github.com/sassoftware/amiconfig. [Accessed: 15-Sep-2019].

[109] Ansible, “Simple IT Automation.” [Online]. Available: https://www.ansible.com/. [Accessed: 15-Sep-2019].

[110] Chef, “Chef Automate.” [Online]. Available: https://www.chef.io/products/automate/. [Accessed: 15-Sep-2019].

[111] SaltStack, “Intelligent IT Automation.” [Online]. Available: https://www.saltstack.com/. [Accessed: 15-Sep-2019].

[112] Puppet, “Cloud management.” [Online]. Available: https://puppet.com/solutions/cloud-management. [Accessed: 15-Sep-2019].

[113] S. V. Zykov, L. D. Shumsky, A. V. Tykushin, and A. G. Tormasov, “Applicative-based automatic configuration management for virtual machines,” *Procedia Comput. Sci.*, vol. 126, pp. 1771–1778, 2018.

[114] Apache Mesos, “Home page.” [Online]. Available: http://mesos.apache.org/. [Accessed: 15-Sep-2019].

[115] Nomad, “Workload orchestrator.” [Online]. Available: https://www.nomadproject.io/. [Accessed: 15-Sep-2019].

[116] Docker Swarm, “Cluster of Dockers.” [Online]. Available: https://docs.docker.com/engine/swarm/. [Accessed: 15-Sep-2019].

[117] Kubernetes, “Production-Grade Container Orchestration.” [Online]. Available: https://kubernetes.io/. [Accessed: 15-Sep-2019].

[118] B. Calabrese and M. Cannataro, “Cloud Computing in Healthcare and Biomedicine,” *Scalable Comput. Pract. Exp.*, vol. 16, no. 1, Feb. 2015.

[119] Genesis Cloud, “Unlimited GPU Power.” [Online]. Available: https://www.genesiscloud.com/. [Accessed: 16-Sep-2019].

[120] Google Cloud HPC, “Hight performance computing | Google Cloud.” [Online]. Available: https://cloud.google.com/solutions/hpc/?hl=es. [Accessed: 23-Sep-2019].

[121] Embassy Cloud, “Analyse your life-science data.” [Online]. Available: https://www.embassycloud.org/. [Accessed: 11-Sep-2019].

[122] B. Langmead and A. Nellore, “Cloud computing for genomic data analysis and collaboration,” *Nat. Rev. Genet.*, vol. 19, no. 4, pp. 208–219, Apr. 2018.

[123] S. V Angiuoli *et al.*, “CloVR: A virtual machine for automated and portable sequence analysis from the desktop using cloud computing,” *BMC Bioinformatics*, vol. 12, no. 1, p. 356, Dec. 2011.

[124] K. Krampis *et al.*, “Cloud BioLinux: pre-configured and on-demand bioinformatics computing for the genomics community,” *BMC Bioinformatics*, vol. 13, no. 1, p. 42, Dec. 2012.

[125] H. Lee *et al.*, “BioVLAB-MMIA: A Reconfigurable Cloud Computing Environment for microRNA and mRNA Integrated Analysis,” in *2011 IEEE International Conference on Bioinformatics and Biomedicine*, 2011, pp. 494–499.

[126] T. Muth, J. Peters, J. Blackburn, E. Rapp, and L. Martens, “ProteoCloud: A full-featured open source proteomics cloud computing pipeline,” *J. Proteomics*, vol. 88, pp. 104–108, Aug. 2013.

[127] Hyungro Lee *et al.*, “BioVLAB-MMIA: A Cloud Environment for microRNA and mRNA Integrated Analysis (MMIA) on Amazon EC2,” *IEEE Trans. Nanobioscience*, vol. 11, no. 3, pp. 266–272, Sep. 2012.

[128] J. Goecks, A. Nekrutenko, J. Taylor, and T. Galaxy Team, “Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences,” *Genome Biol.*, vol. 11, no. 8, p. R86, 2010.

[129] E. Afgan, D. Baker, N. Coraor, B. Chapman, A. Nekrutenko, and J. Taylor, “Galaxy CloudMan: delivering cloud compute clusters,” *BMC Bioinformatics*, vol. 11, no. Suppl 12, p. S4, 2010.

[130] P. Moreno *et al.*, “Galaxy-Kubernetes integration: scaling bioinformatics workflows in the cloud,” *bioRxiv*, p. 488643, Dec. 2018.

[131] L. Jourdren, M. Bernard, M.-A. Dillies, and S. Le Crom, “Eoulsan: a cloud computing-based framework facilitating high throughput sequencing analyses,” *Bioinformatics*, vol. 28, no. 11, pp. 1542–1543, Jun. 2012.

[132] K. Peters *et al.*, “PhenoMeNal: processing and analysis of metabolomics data in the cloud,” *Gigascience*, vol. 8, no. 2, Feb. 2019.

[133] Project Jupyter, “Notebook interface.” [Online]. Available: https://jupyter.org/. [Accessed: 16-Sep-2019].

[134] D. Salomoni *et al.*, “INDIGO-DataCloud: a Platform to Facilitate Seamless Access to E-Infrastructures,” *J. Grid Comput.*, vol. 16, no. 3, pp. 381–408, Sep. 2018.

[135] T. Nguyen, W. Shi, and D. Ruden, "CloudAligner: A fast and full-featured MapReduce based tool for sequence mapping.," *BMC Res. Notes*, vol. 4, p. 171, Jun. 2011.

[136] V. Popic and S. Batzoglou, "A hybrid cloud read aligner based on MinHash and kmer voting that preserves privacy," *Nat. Commun.*, vol. 8, no. 1, p. 15311, Aug. 2017.

[137] B. Langmead, M. C. Schatz, J. Lin, M. Pop, and S. L. Salzberg, "Searching for SNPs with cloud computing," *Genome Biol.*, vol. 10, no. 11, p. R134, 2009.

[138] X. Feng, R. Grossman, and L. Stein, "PeakRanger: A cloud-enabled peak caller for ChIP-seq data," *BMC Bioinformatics*, vol. 12, no. 1, p. 139, Dec. 2011.

[139] B. Langmead, K. D. Hansen, and J. T. Leek, "Cloud-scale RNA-sequencing differential expression analysis with Myrna," *Genome Biol.*, vol. 11, no. 8, p. R83, 2010.

[140] A. Yang, M. Troup, P. Lin, and J. W. K. Ho, "Falco: a quick and flexible single-cell RNA-seq processing framework on the cloud," *Bioinformatics*, vol. 33, no. 5, p. btw732, Dec. 2016.

[141] A. McKenna *et al.*, "The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data," *Genome Res.*, vol. 20, no. 9, pp. 1297–1303, Sep. 2010.

[142] M. Niemenmaa, A. Kallio, A. Schumacher, P. Klemelä, E. Korpelainen, and K. Heljanko, "Hadoop-BAM: directly manipulating next generation sequencing data in the cloud," *Bioinformatics*, vol. 28, no. 6, pp. 876–877, Mar. 2012.

[143] S. R. Ellingson and J. Baudry, "High-throughput virtual molecular docking with AutoDockCloud," *Concurr. Comput. Pract. Exp.*, vol. 26, no. 4, pp. 907–916, Mar. 2014.

[144] J. J. Alnasir and H. P. Shanahan, "The application of Hadoop in structural bioinformatics," *Brief. Bioinform.*, Nov. 2018.

[145] S. Lewis *et al.*, "Hydra: a scalable proteomic search engine which utilizes the Hadoop distributed computing framework," *BMC Bioinformatics*, vol. 13, no. 1, p. 324, Dec. 2012.

[146] DNAnexus, "Enabling Progress." [Online]. Available: https://www.dnanexus.com/. [Accessed: 16-Sep-2019].

[147] Globus Genomics, "Cutting-edge reseach on cloud." [Online]. Available: https://www.globusgenomics.org/. [Accessed: 23-Sep-2019].

[148] Azure, "Microsoft Azure CycleCloud." [Online]. Available: https://azure.microsoft.com/es-es/features/azure-cyclecloud/. [Accessed: 28-Sep-2019].

[149] C. Birger *et al.*, "FireCloud, a scalable cloud-based platform for collaborative genome analysis: Strategies for reducing and controlling costs," *bioRxiv*, p. 209494, Nov. 2017.

[150] A. Golberg *et al.*, "Cloud-Enabled Microscopy and Droplet Microfluidic Platform for Specific Detection of Escherichia coli in Water," *PLoS One*, vol. 9, no. 1, p. e86341, Jan. 2014.

[151] V. Navale and P. E. Bourne, "Cloud computing applications for biomedical science: A perspective," *PLOS Comput. Biol.*, vol. 14, no. 6, p. e1006144, Jun. 2018.

[152] AWS OpenData, "Registry of Open Data on Amazon Web services," 2014. [Online]. Available: https://registry.opendata.aws/. [Accessed: 17-Sep-2019].

[153] Local EGA, "Local European Genome-phenome Archive." [Online]. Available: https://localega.readthedocs.io/en/latest/. [Accessed: 17-Sep-2019].

[154] G. Saunders *et al.*, "Leveraging European infrastructures to access 1 million human genomes by 2022," *Nat. Rev. Genet.*, pp. 1–9, Aug. 2019.

[155] B. Satzger, W. Hummer, C. Inzinger, P. Leitner, and S. Dustdar, "Winds of Change: From Vendor Lock-In to the Meta Cloud," *IEEE Internet Comput.*, vol. 17, no. 1, pp. 69–73, Jan. 2013.

[156] Ł. Dutka *et al.*, "Onedata – A Step Forward towards Globalization of Data Access for Computing Infrastructures," *Procedia Comput. Sci.*, vol. 51, pp. 2843–2847, Jan. 2015.

[157] C. A. Sanchez, J. Bloomer, … P. B.-P. of X., and undefined 2008, "CVMFS-a file system for the CernVM virtual appliance," *adsabs.harvard.edu*.

[158] G. Seth and L. Nancy, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services," *ACM SIGACT News*, 2002.

[159] D. De Roure, "e-Science and the Web," *Computer (Long. Beach. Calif).*, vol. 43, no. 5, pp. 90–93, May 2010.

[160] T. Berners-Lee, "COMPUTER SCIENCE: Enhanced: Creating a Science of the Web," *Science (80-.).*, vol. 313, no. 5788, pp. 769–771, Aug. 2006.

[161] M. P. Papazoglou, "Service-oriented computing: concepts, characteristics and directions," in *Proceedings of the 7th International Conference on Properties and Applications of Dielectric Materials (Cat. No.03CH37417)*, 2019, pp. 3–12.

[162] V. Curcin, M. Ghanem, and Y. Guo, "Web services in the life sciences," *Drug Discov. Today*, vol. 10, no. 12, pp. 865–871, 2005.

[163] W3C1, "Standards - W3C," 2019. [Online]. Available: https://www.w3.org/standards/. [Accessed: 03-Aug-2019].

[164] OASIS, "OASIS | Advancing open standards for the information society," 2019. [Online]. Available: https://www.oasis-open.org/. [Accessed: 03-Aug-2019].

[165] M. D. Wilkinson and M. Links, "BioMOBY: An open source biological web services proposal," *Brief. Bioinform.*, vol. 3, no. 4, pp. 331–341, Jan. 2002.

[166] S. Pettifer *et al.*, "The EMBRACE web service collection," *Nucleic Acids Res.*, vol. 38, no. Web Server, pp. W683–W688, Jul. 2010.

[167] D. Repchevsky and J. L. Gelpi, "BioSWR – Semantic Web Services Registry for Bioinformatics," *PLoS One*, vol. 9, no. 9, p. e107889, Sep. 2014.

[168] J. Bhagat *et al.*, "BioCatalogue: a universal catalogue of web services for the life sciences," *Nucleic Acids Res.*, vol. 38, no. Web Server, pp. W689–W694, Jul. 2010.

[169] Rails on Rails, "Ruby on Rails | A web-application framework that includes everything needed to create database-backed web applications according to the Model-View-Controller (MVC) pattern.," 2019. [Online]. Available: https://rubyonrails.org/. [Accessed: 03-Aug-2019].

[170] Django, "Hire Top Django Developers | Netguru," 2019. [Online]. Available: https://www.netguru.com/services/django-development. [Accessed: 03-Aug-2019].

[171] Flask, "Flask | The Pallets Projects," 2019. [Online]. Available: https://palletsprojects.com/p/flask/. [Accessed: 03-Aug-2019].

[172] Web2py, "Weblet Importer," 2019. [Online]. Available: http://www.web2py.com/. [Accessed: 03-Aug-2019].

[173] Laravel, "Laravel - The PHP Framework For Web Artisans," 2019. [Online]. Available: https://laravel.com/. [Accessed: 03-Aug-2019].

[174] Symfony, "Symfony, High Performance PHP Framework for Web Development," 2019. [Online]. Available: https://symfony.com/. [Accessed: 03-Aug-2019].

[175] Slim, "Slim Framework - Slim Framework," 2019. [Online]. Available: https://www.slimframework.com/. [Accessed: 03-Aug-2019].

[176] Bootstrap, "Bootstrap · The most popular HTML, CSS, and JS library in the world.," 2019. [Online]. Available: https://getbootstrap.com/. [Accessed: 03-Aug-2019].

[177] Angular, "Angular," 2019. [Online]. Available: https://angular.io/. [Accessed: 03-Aug-2019].

[178] jQuery, "jQuery," 2019. [Online]. Available: https://jquery.com/. [Accessed: 03-Aug-2019].

[179] React, "React – A JavaScript library for building user interfaces," 2019. [Online]. Available: https://reactjs.org/. [Accessed: 03-Aug-2019].

[180] Sass, “Sass: Syntactically Awesome Style Sheets,” 2019. [Online]. Available: https://sass-lang.com/. [Accessed: 03-Aug-2019].

[181] less, “Getting started | Less.js,” 2019. [Online]. Available: http://lesscss.org/. [Accessed: 03-Aug-2019].

[182] WebSocket API, “The WebSocket API (WebSockets) - Web APIs | MDN,” 2019. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API. [Accessed: 03-Aug-2019].

[183] WebTorrent, “WebTorrent - Streaming browser torrent client,” 2019. [Online]. Available: https://webtorrent.io/. [Accessed: 03-Aug-2019].

[184] Gun, “Decentralized Database,” 2019. [Online]. Available: https://gun.eco/. [Accessed: 03-Aug-2019].

[185] ExPASy, “SIB Bioinformatics Resource Portal.” [Online]. Available: https://www.expasy.org/. [Accessed: 18-Sep-2019].

[186] G. D. Schuler, J. A. Epstein, H. Ohkawa, and J. A. Kans, “Entrez: Molecular biology database and retrieval system,” *Methods Enzymol.*, vol. 266, pp. 141–162, Jan. 1996.

[187] T. Etzold and P. Argos, “SRS--an indexing and retrieval tool for flat file data libraries.,” *Comput. Appl. Biosci.*, vol. 9, no. 1, pp. 49–57, Feb. 1993.

[188] M. A. Kallio *et al.*, “Chipster: user-friendly analysis software for microarray and other high-throughput data,” *BMC Genomics*, vol. 12, no. 1, p. 507, Dec. 2011.

[189] K. M. Fisch *et al.*, “Omics Pipe: a community-based framework for reproducible multi-omics data analysis,” *Bioinformatics*, vol. 31, no. 11, pp. 1724–1728, Jun. 2015.

[190] F. Halbritter, H. J. Vaidya, and S. R. Tomlinson, “GeneProf: analysis of high-throughput sequencing experiments,” *Nat. Methods*, vol. 9, no. 1, pp. 7–8, Jan. 2012.

[191] M. Reich, T. Liefeld, J. Gould, J. Lerner, P. Tamayo, and J. P. Mesirov, “GenePattern 2.0,” *Nat. Genet.*, vol. 38, no. 5, pp. 500–501, May 2006.

[192] E. Afgan *et al.*, “Genomics Virtual Laboratory: A Practical Bioinformatics Workbench for the Cloud,” *PLoS One*, vol. 10, no. 10, p. e0140829, Oct. 2015.

[193] J. Koster and S. Rahmann, “Snakemake--a scalable bioinformatics workflow engine,” *Bioinformatics*, vol. 28, no. 19, pp. 2520–2522, Oct. 2012.

[194] P. Di Tommaso, M. Chatzou, E. W. Floden, P. P. Barja, E. Palumbo, and C. Notredame, “Nextflow enables reproducible computational workflows,” *Nat. Biotechnol.*, vol. 35, no. 4, pp. 316–319, Apr. 2017.

[195] A. Carusi and T. Reimer, “Virtual Research Environment Collaborative Landscape Study,” 2010.

[196] LifeWatch-ERIC, “ERIC infrastructure: LifeWatch.” [Online]. Available: https://www.lifewatch.eu/. [Accessed: 28-Sep-2019].

[197] LifeWatch-ERIC, “Catalogue of Virtual Labs.” [Online]. Available: https://www.lifewatch.eu/web/guest/catalogue-of-virtual-labs?p_p_id=101_INSTANCE_bfMpaSj5Bwkq&p_p_lifecycle=0&p_p_state=normal&p_p_mode=view&p_p_col_id=column-1&p_p_col_count=1&_101_INSTANCE_bfMpaSj5Bwkq_delta=4&_101_INSTANCE_bfMpaSj5Bwkq_keywords=&_101_I. [Accessed: 28-Sep-2019].

[198] F. Giacomoni *et al.*, “Workflow4Metabolomics: a collaborative research infrastructure for computational metabolomics,” *Bioinformatics*, vol. 31, no. 9, pp. 1493–1495, May 2015.

[199] MySQL, “The most popular open source database.” [Online]. Available: https://www.mysql.com/. [Accessed: 12-Sep-2019].

[200] MySQL Cluster CGE, “MySQL distributed database.” [Online]. Available: https://www.mysql.com/products/cluster/. [Accessed: 12-Sep-2019].

[201] MongoDB, “Open Source Document Database.” [Online]. Available: https://www.mongodb.com/es. [Accessed:

11-Sep-2019].

[202] Cinder, “OpenStack Docs: OpenStack Block Storage documentation,” 2019. [Online]. Available: https://docs.openstack.org/cinder/latest/. [Accessed: 05-Sep-2019].

[203] Tomcat 7, “Apache Tomcat 7 (7.0.96).” [Online]. Available: https://tomcat.apache.org/tomcat-7.0-doc/setup.html. [Accessed: 24-Sep-2019].

[204] Node.js, “JavaScript runtime.” [Online]. Available: https://nodejs.org/en/. [Accessed: 24-Sep-2019].

[205] pm2 runtime, “Process Manager 2 for NodeJS applications.” [Online]. Available: https://www.npmjs.com/package/pm2. [Accessed: 24-Sep-2019].

[206] PMESClient, “PMES API 2.3.2 | Documentation,” 2016. [Online]. Available: http://compss.bsc.es/releases/pmes/pmes-framework/2.3.2/docs/api/com/bsc/pmes/client/PMESClient.html. [Accessed: 16-Aug-2019].

[207] Keycloak, “Keycloak,” 2019. [Online]. Available: https://www.keycloak.org/. [Accessed: 03-Aug-2019].

[208] OpenLDAP, “OpenLDAP Lightweight Directory Access Protocol.” [Online]. Available: https://www.openldap.org/. [Accessed: 21-Sep-2019].

[209] M. Thijs and A. Edmonds, “Open Cloud Computing Interface – RESTful HTTP Rendering,” 2011. [Online]. Available: http://www.ogf.org/documents/GFD.185.pdf.

[210] B. Parák, Z. Sustr, F. Feldhaus, … P. K.-… S. on G. and, and undefined 2014, “The rOCCI project: providing cloud interoperability with OCCI 1.1,” *pos.sissa.it*.

[211] S. Andreozzi and M. Marzolla, “A RESTful Approach to the OGSA Basic Execution Service Specification,” in *2009 Fourth International Conference on Internet and Web Applications and Services*, 2009, pp. 131–136.

[212] B. L. Cantarel *et al.*, “MAKER: an easy-to-use annotation pipeline designed for emerging model organism genomes.,” *Genome Res.*, vol. 18, no. 1, pp. 188–96, Jan. 2008.

[213] G. Slater and E. Birney, “Automated generation of heuristics for biological sequence comparison.,” *BMC Bioinformatics*, vol. 6, no. 1, p. 31, Feb. 2005.

[214] I. Korf, “Gene finding in novel genomes.,” *BMC Bioinformatics*, vol. 5, no. 1, p. 59, May 2004.

[215] M. Stanke, A. Tzvetkova, and B. Morgenstern, “AUGUSTUS at EGASP: using EST, protein and genomic alignments for improved gene prediction in the human genome.,” *Genome Biol.*, vol. 7, no. Suppl 1, p. S11, 2006.

[216] S. Altschul *et al.*, “Gapped BLAST and PSI-BLAST: a new generation of protein database search programs,” *Nucleic Acids Res.*, vol. 25, no. 17, pp. 3389–3402, Sep. 1997.

[217] Institute for Systems Biology, “RepeatMasker.” [Online]. Available: http://repeatmasker.org/. [Accessed: 27-Sep-2019].

[218] G. Benson, “Tandem repeats finder: a program to analyze DNA sequences,” *Nucleic Acids Res.*, vol. 27, no. 2, pp. 573–580, Jan. 1999.

[219] L. Codó & *et al.*, “Nucleosome Dynamics: a new tool for the dynamic analysis of nucleosome positioning,” *Nucleic Acids Res.*, Aug. 2019.

[220] Ö. Deniz, O. Flores, M. Aldea, M. Soler-López, and M. Orozco, “Nucleosome architecture throughout the cell cycle,” *Sci. Rep.*, vol. 6, no. 1, p. 19729, Apr. 2016.

[221] N. Nocetti and I. Whitehouse, “Nucleosome repositioning underlies dynamic gene expression,” *Genes Dev.*, vol. 30, no. 6, pp. 660–672, Mar. 2016.

[222] N. Kaplan *et al.*, “The DNA-encoded nucleosome organization of a eukaryotic genome,” *Nature*, vol. 458, no. 7236, pp. 362–366, Mar. 2009.

[223] EGroupware, “Collaborative Software,” 2019. [Online]. Available: https://github.com/EGroupware/egroupware. [Accessed: 07-Sep-2019].

[224] Apache Taverna, "| Apache Taverna (incubating)." [Online]. Available: https://taverna.incubator.apache.org/. [Accessed: 22-Aug-2019].

[225] vsFTPd, "Secure, fast FTP server for UNIX-like systems." [Online]. Available: https://security.appspot.com/vsftpd.html. [Accessed: 16-Aug-2019].

[226] MakerAnnotation, "Warris, Sven / makerAnnotation · GitLab," 2019. [Online]. Available: https://git.wageningenur.nl/warri004/makerAnnotation. [Accessed: 03-Aug-2019].

[227] L. Codó *et al.*, "MuGVRE. A virtual research environment for 3D/4D genomics," *bioRxiv*, p. 602474, Apr. 2019.

[228] A. Hospital *et al.*, "BIGNASim: a NoSQL database structure and analysis portal for nucleic acids simulation data," *Nucleic Acids Res.*, vol. 44, no. D1, pp. D272–D278, Jan. 2016.

[229] MuG DM-API, "MuG Metadata Management REST API." [Online]. Available: https://github.com/Multiscale-Genomics/mg-dm-api. [Accessed: 26-Sep-2019].

[230] W. K. M. Lai and B. F. Pugh, "Understanding nucleosome dynamics and their links to gene expression and DNA replication," *Nat. Rev. Mol. Cell Biol.*, vol. 18, no. 9, pp. 548–562, Sep. 2017.

[231] Planemo, "Documentation 0.61.0.dev0." [Online]. Available: https://planemo.readthedocs.io/en/latest/. [Accessed: 20-Aug-2019].

[232] Dream Challenges, "by Sage Bionetworks." [Online]. Available: http://dreamchallenges.org/challenges/. [Accessed: 20-Aug-2019].

[233] TCGA, "| Mutation Calling Benchmark 4." [Online]. Available: https://gdc.cancer.gov/resources-tcga-users. [Accessed: 20-Aug-2019].

[234] Quest For Orthologs, "| Home." [Online]. Available: https://questfororthologs.org/. [Accessed: 20-Aug-2019].

[235] Global Microbial Identifier, "Home page." [Online]. Available: https://www.globalmicrobialidentifier.org/. [Accessed: 20-Aug-2019].

[236] P. Turner, A. Kushniruk, and C. Nohr, "Are We There Yet? Human Factors Knowledge and Health Information Technology – the Challenges of Implementation and Impact," *Yearb. Med. Inform.*, vol. 26, no. 01, pp. 84–91, Sep. 2017.

[237] A. Xie and P. Carayon, "A systematic review of human factors and ergonomics (HFE)-based healthcare system redesign for quality of care and patient safety," *Ergonomics*, vol. 58, no. 1, pp. 33–49, Jan. 2015.

[238] D. Kingston *et al.*, "Pregnant Women's Views on the Feasibility and Acceptability of Web-Based Mental Health E-Screening Versus Paper-Based Screening: A Randomized Controlled Trial.," *J. Med. Internet Res.*, vol. 19, no. 4, p. e88, 2017.

[239] E. Braekman *et al.*, "Measurement agreement of the self-administered questionnaire of the Belgian Health Interview Survey: Paper-and-pencil versus web-based mode.," *PLoS One*, vol. 13, no. 5, p. e0197434, 2018.

[240] J. Wolff *et al.*, "Galaxy HiCExplorer: a web server for reproducible Hi-C data analysis, quality control and visualization," *Nucleic Acids Res.*, vol. 46, no. Web Server issue, p. W11, 2018.

[241] B. Langmead and A. Nellore, "Cloud computing for genomic data analysis and collaboration.," *Nat. Rev. Genet.*, vol. 19, no. 4, pp. 208–219, 2018.

[242] L. Han, H. Huang, and C. Xie, "Performance Analysis of NAND Flash Based Cache for Network Storage System," in *2013 IEEE Eighth International Conference on Networking, Architecture and Storage*, 2013, pp. 68–75.

[243] Ł. Dutka *et al.*, "Onedata – A Step Forward towards Globalization of Data Access for Computing Infrastructures," *Procedia Comput. Sci.*, vol. 51, pp. 2843–2847, Jan. 2015.

[244] A. Pasdar, K. Almi'ani, and Y. C. Lee, "Data-Aware Scheduling of Scientific Workflows in Hybrid Clouds," 2018, pp. 708–714.

[245] D. Doiron, Y. Marcon, I. Fortier, P. Burton, and V. Ferretti, "Software Application Profile: Opal and Mica: open-source software solutions for epidemiological data management, harmonization and dissemination," *Int. J.*

Epidemiol., vol. 46, no. 5, pp. 1372–1378, Oct. 2017.

[246] J. C. Wallis, E. Rolando, and C. L. Borgman, "If We Share Data, Will Anyone Use Them? Data Sharing and Reuse in the Long Tail of Science and Technology," *PLoS One*, vol. 8, no. 7, p. e67332, 2013.

[247] EGA, "EGA European Genome-Phenome Archive." [Online]. Available: https://ega-archive.org/. [Accessed: 28-Aug-2019].

[248] C. Bonte, E. Makri, A. Ardeshirdavani, J. Simm, Y. Moreau, and F. Vercauteren, "Towards practical privacy-preserving genome-wide association study," *BMC Bioinformatics*, vol. 19, no. 1, p. 537, Dec. 2018.

[249] A. K. Yadav, M. L. Garg, and Ritika, "Docker Containers Versus Virtual Machine-Based Virtualization," 2019, pp. 141–150.

[250] P. Sharma, L. Chaufournier, P. Shenoy, and Y. C. Tay, "Containers and Virtual Machines at Scale," in *Proceedings of the 17th International Middleware Conference on - Middleware '16*, 2016, pp. 1–13.

[251] X. Shi and X. Wu, "An overview of human genetic privacy," *Ann. N. Y. Acad. Sci.*, vol. 1387, no. 1, pp. 61–72, Jan. 2017.

[252] E. Ayday, J. L. Raisaro, U. Hengartner, A. Molyneaux, and J.-P. Hubaux, "Privacy-Preserving Processing of Raw Genomic Data," Springer, Berlin, Heidelberg, 2014, pp. 133–147.

[253] GÉANT, "VPN Services." [Online]. Available: https://www.geant.org/Services/Connectivity_and_network/Pages/VPN_Services.aspx. [Accessed: 28-Sep-2019].

[254] H. P. Shanahan, A. M. Owen, and A. P. Harrison, "Bioinformatics on the Cloud Computing Platform Azure," *PLoS One*, vol. 9, no. 7, p. e102642, Jul. 2014.

[255] P. Andrio *et al.*, "BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows," *Sci. Data*, vol. 6, no. 1, p. 169, Dec. 2019.

[256] K. G. Srinivasa and A. K. Muppalla, *Guide to High Performance Distributed Computing*. Cham: Springer International Publishing, 2015.

[257] D. Huang and H. Wu, "Virtualization," in *Mobile Cloud Computing*, Elsevier, 2018, pp. 31–64.

[258] Gartner, "Gartner," 2019. [Online]. Available: https://www.gartner.com/en. [Accessed: 03-Aug-2019].

[259] P. Kulkarni and P. Frommolt, "Challenges in the Setup of Large-scale Next-Generation Sequencing Analysis Workflows.," *Comput. Struct. Biotechnol. J.*, vol. 15, pp. 471–477, 2017.

8. Annexes

8.1 Participant centers of IMID's clinical studies

Following, the complete list of health centers participating to one of the two IMID's clinical studies here presented, IMID-Clinica or IMID-Longitudinal.

Center	Patology	IMID Clinica	IMID Longitudinal
Centro de Salud Virgen de los Reyes (Sevilla)	AR	x	x
Complejo Hospitalario de León (León)	EC	x	x
	CU	x	x
Complejo Hospitalario Juan Canalejo (A Coruña)	PS	x	x
Complexo Hospitalario de Ourense (Ourense)	AP	x	x
Fundació Clínic per a la Recerca Biomédica (Barcelona)	AR	x	x
Fundación Hospital Alcorcón (Madrid)	PS		x
Hosiptal Universitario A Coruña (A Coruña)	LE	x	x
Hospital 12 de Octubre	AP	x	x
Hospital Clínic de Barcelona (Barcelona)	AP	x	x
Hospital Clínico San Carlos (Madrid)	EC	x	x
	CU	x	x
	AR	x	x
Hospital Clínico Universitario (Santiago de Compostela)	EC	x	x
	CU	x	x
Hospital Clínico Universitario (Zaragoza)	EC	x	x
	CU	x	x
Hospital Complejo Asistencial Universitario de León (León)	LE		x
Hospital de la Santa Creu i Sant Pau (Barcelona)	EC	x	x
	CU	x	x
	PS	x	x
	AR	x	x
Hospital del Mar (Barcelona)	AR	x	x
	AP	x	x
	LE	x	x
Hospital del SAS de Jerez de la Frontera (Cádiz)	AP	x	x
	LE	x	x
Hospital del Valme (Sevilla)	LE	x	x
Hospital do Meixoeiro (Vigo)	LE	x	x
Hospital Donostia (San Sebastián)	EC	x	x
	CU	x	x
Hospital Dr Negrin (Las Palmas de Gran Canaria)	LE	x	x
Hospital Fuenlabrada (Madrid)	EC	x	x
	CU	x	x

Hospital Galdakao (Bilbao)	EC	x	x
	CU	x	x
Hospital General de Alicante (Alicante)	EC	x	x
	CU	x	x
Hospital General La Mancha Centro (Ciudad Real)	AR	x	x
	AP	x	x
Hospital General Universitario de Alicante (Alicante)	PS	x	x
	LE	x	x
Hospital General Universitario de Valencia (Valencia)	PS		x
Hospital Gregorio Marañón (Madrid)	EC	x	x
	CU	x	x
	AP	x	x
Hospital Infanta Sofía /Hospital Universitario La Paz (Madrid)	AP	x	x
Hospital Juan Canalejo (A Coruña)	AR	x	x
Hospital la Fe (Valencia)	EC	x	x
	CU	x	x
Hospital La Paz (Madrid)	EC	x	x
	CU	x	x
Hospital Manises (Valencia)	EC	x	x
	CU	x	x
Hospital Moises Broggi (Hospitalet de Llobregat)	AR	x	x
Hospital Monte Naranco (Oviedo)	AP	x	x
Hospital Mútua de Terrasa	CU	x	x
	EC	x	x
	AP	x	x
Hospital Parc Taulí (Sabadell)	AP	x	x
Hospital Ramón y Cajal (Madrid)	EC	x	x
	LE	x	x
	CU	x	x
Hospital Regional Universitario Carlos Haya (Málaga)	AR	x	x
	LE	x	x
Hospital Sant Rafael (Barcelona)	AR	x	x
Hospital Son Espases (Mallorca)	AR	x	x
Hospital Universitari de Bellvitge (Barcelona)	LE	x	x
	CU	x	x
	EC	x	x
	AP	x	x
Hospital Universitari German Trias i Pujol (Badalona)	LE	x	x
	EC	x	x
	CU	x	x
	AR	x	x
	AP	x	x

	PS	x	x
Hospital Universitari Vall d'Hebron (Barcelona)	AP	x	x
	LE	x	x
	PS	x	x
	AR	x	x
	EA	x	x
Hospital Universitario A Coruña	AP	x	x
Hospital Universitario Central de Asturias (Asturias)	AR	x	x
Hospital Universitario Central de Asturias (Oviedo)	AP	x	x
Hospital Universitario de Basurto (Bilbao)	AR	x	x
Hospital Universitario de Canarias (Tenerife)	AR	x	x
Hospital Universitario de Gran Canaria Dr. Negrín (Gran Canarias)	AR	x	x
Hospital Universitario de la Princesa (Madrid)	EC	x	x
	CU	x	x
Hospital Universitario de Salamanca (Salamanca)	PS	x	x
Hospital Universitario 12 de Octubre (Madrid)	PS	x	x
Hospital Universitario Doce de Octubre (Madrid)	LE	x	x
Hospital Universitario Guadalajara (Guadalajara)	AR	x	x
Hospital Universitario Infanta Leonor (Madrid)	PS	x	x
Hospital Universitario La Fe (Valencia)	PS	x	x
Hospital Universitario La Princesa (Madrid)	PS	x	x
	AR	x	x
	AP	x	x
Hospital Universitario Marqués de Valdecilla (Santader)	LE	x	x
Hospital Universitario Puerta de Hierro (Madrid)	EC	x	x
	CU	x	x
	LE	x	x
Hospital Universitario Reina Sofía (Córdoba)	EC	x	x
	CU	x	x
Hospital Virgen de la Macarena (Sevilla)	PS	x	x
Hospital Virgen de la Vega (Salamanca)	AP	x	x
Parc de Salut Mar (Barcelona)	PS	x	x
Centro de Atención especializada (CAE) (Cornellà)	AP	x	x
Hospital clínico Virgen de la Victoria (Málaga)	PS	x	x
Hospital Comarcal Alt Penedés (Vilafranca)	AP	x	x
Hospital Sant Rafael	AP	x	x
Hospital Universitario Virgen de la Arrixaca (Murcia)	AR	x	x
	AP	x	x
Servicio de Reumatologí Hospital Universitario Araba (Alava)	LE	x	x

Table 8.: IMID participant health centers.
LES: Systemic lupus erythematosus, PS: psoriasis, AP: psoriatic arthritis, AR: arthritis rheumatoid, CU: ulcerative cholitis, EC: Chron's disease

8.2 transPLANT tools

List of the virtual machine images (VMI) packing the transPLANT tooling for plant genomics

Virtual Machine Image	Distribution	Software	Description
COMPSs	Debian (6.0.5)	COMPSs (1.1.2)	Generic VM to run COMPSs based applications
gene-detection	Debian (6.0.5)	COMPSs (1.1.2), Blast (2.2.27+), Blast2gene (2.2), GeneWise (2.0)	Detects genes in a genome using a reference protein from a closely related species using Genewise.
serial_maker+	Debian (6.0.5)	Maker (2.28), Exonerate (2.2.0), Snap (2006-07-28), Augustus (2.5.5), Blast (2.2.28+), RepeatMasker (1.295), TRF (4.07b)	Genome annotation pipeline
bwapipeline	Ubuntu (14.04 LTS)	BWA (0.1.17), bcftools (0.1.18), samtools (0.7.5a)	Aligns paired-end reads against a reference genome using BWA2
bowtie+	Debian (6.0.5)	Bowtie (2-2.2.3), tophat (2.0.12), boost (1_55), samtools boost (1_55), samtools (0.1.80.1.8)	Aligns paired-end reads against a reference genome using fast Bowtie
Abyss	Ubuntu (14.04 LTS)	Abyss (1.3.6), samtools (0.7.5a)	*De novo*, parallel, paired-end sequence assembler, designed for short reads.
Repet (URGI)	Ubuntu (14.04 LTS)	TEDeNovo, Blast (2.2.20 / 2.2.29+), Censor (4.2), genometools (1.5.2), Hmmer (3.1b1), MCL (12-068), Mreps (2.5), Piler (1.0), Recon (1.07), RepeatMasker (4), RepeatScout (1.0.5), Samtools (0.1.19), TRF (1.0), tRNAscan (1.23)	Pipeline dedicated to detect, annotate and analyse repeats in genomic sequences, in particular, transposable elements (TEs)

Table 8.1: TransPLANT tools

8.3 MuG tools & visualizers

Data visualizers

The current list of tools integrated, or in the way of being integrated (status: submitted) is the following:

Visualizer	Description	Author	Tool Status (*)
Jbrowse	A next-generation genome browser built with JavaScript and HTML5.	IRB	Active
NGL	NGL Viewer is a web application for molecular visualization. WebGL is employed to display molecules like proteins and DNA/RNA with a variety of representations.	IRB	Active
TADkit	3D genome browser and TADbit front-end	CNAG	Active

CytoScape	Software platform for visualizing molecular interaction networks and integrate them state data like gene expression profiles, chromatin assortativity, etc.	Centre de Recherches en Cancérologie de Toulouse	Submitted

Table 8.2: MuGVRE visualizers

(*) Tool/visualizer are: (1) active: the tool is eligible to be run for all users (2) Inactive: the tool is not eligible to be run. (3) Testing: the tool is eligible only by the tool developer owing the tool. (4) Submitted: the tool has started the integration process but is not yet integrated.

Analysis Tools

The current list of tools integrated, or in the way of being integrated (status: submitted) is the following:

Tool	**Description**	**Author**	**Tool status (*)**
3DConsensus	Analyse a protein-DNA complex 3D structure to identify interactions	UNOT	Active
Bowtie2	Align FASTQ data using Bowtie2	EMBL-EBI	Testing
BWA MEM	Align FASTQ data using BWA mem	EMBL-EBI	Testing
Chromatin Dynamics	With the Chromatin Dynamics tool you can create your own 'beads-on-a-string' representation of a chromatin fiber.	IRB	Active
MACS2	MACS identifies statistically significantly enriched genomic regions in ChIP- and DNase-seq data	EMBL-EBI	Active
MC-DNA	MC-DNA is a coarse-grained DNA model in which the internal dynamics are described with helical parameters.	IRB	Active
MD Energy Refinement	Molecular Dynamics workflow to energetically minimize a structure. MD Setup + 100ps free MD. Output last structure from the 100ps.	IRB	Active
NAFlex analyses	Set of analyses to extract Nucleic Acids flexibility properties from Molecular Dynamics trajectories	IRB	Active
Nucleosome Dynamics	Nucleosome Dynamics Tools for performing nucleosome-related analysis based on MNase-seq experimental data	IRB	Active
Process ChIP-seq	Align ChIP-seq data, filtering with BioBamBam and Peak Calling using MACS2.	EMBL-EBI	Testing
Process Genomes	Generates BWA, Bowtie2 and GEM indexes for a given genome	EMBL-EBI	Active
Process RNA-seq	Align RNA-seq data, gene expression calling with Kallisto	EMBL-EBI	Testing
Process WGBS	Align WGBS data, uses BS Seeker2 and Bowtie2	EMBL-EBI	Testing
PyDockDNA	Docking Protein-DNA	BSC	Active
PyDock	Protein-Protein Docking	BSC	Active
TADbit bin	TADbit Hi-C binning.	CNAG	Active
TADbit map, parse and filter	TADbit Hi-C mapping, parsing mapped reads and filtering of artifactual reads.	CNAG	Active
TADbit model	TADbit 3D modeling.	CNAG	Active

TADbit normalize	TADbit Hi-C normalize.	CNAG	Active
TADbit segment	TADbit Hi-C segment (TADs and compartments).	CNAG	Active
ChICAGO	CHiCAGO pipeline for calling significant interactions in Capture HiC data, such as Promoter Capture HiC	EMBL-EBI	Submitted
Chromatin Assortativity	Calculation of chromatin assortativity to integrate the epigenomic landscape of a specific cell type with its chromatin interaction network	Centre de Recherches en Cancérologie de Toulouse	Testing
Process DamID-seq[29]	Align DamID-seq data, filtering with BioBamBam and Peak Calling using iDEAR	EMBL-EBI	Submitted
BioBamBam2 Filtering	Mark technical duplicates using BioBamBam2 and then remove them with samtools	EMBL-EBI	Submitted
FASTQ Trimming	Trimming of single and paired end FASTQ reads using TrimGalore	EMBL-EBI	Submitted
BWA ALN	Aligns single and paired end data using the BWA ALN method	EMBL-EBI	Submitted
Analyse FASTQ[29]	Analyse the quality of reads within a fastq file and provide relevant statistics	EMBL-EBI	Submitted
BS Seeker 2 Peak Caller	WGBS BS Seeker2 Methylation Peak Caller	EMBL-EBI	Submitted
Process BS Seeker2 Aligner	Align WGBS data using BS Seeker2 and Bowtie2	EMBL-EBI	Submitted
Process BS Seeker2 Filter	Filter WGBS data, uses BS Seeker2	EMBL-EBI	Submitted
Process WGBS BS Seeker 2 Indexer	Create the custom Bowtie2 index required by BS Seeker2	EMBL-EBI	Submitted

Table 8.3: MuGVRE analysis tools list
(*) Tool/visualizer are: (1) active: the tool is eligible to be run for all users (2) Inactive: the tool is not eligible to be run. (3) Testing: the tool is eligible only by the tool developer owing the tool. (4) Submitted: the tool has started the integration process but is not yet integrated.

8.4 MuG data models

8.4.1 Data Model: "File"

Following table describe the fields and their dependencies of the minimal metadata set of a File object as defined for MuG project.

Parameter	Required	Description
file_id	YES	This is an auto-generated ID that is created when the data is entered. The ID is unique to the entire system
user_id	YES	The unique user ID for whom the file is associated with

file_path	YES	Location of the file either within the file system or a URL to an archive or repository.
path_type	YES	Defines if the file_path is a file (type= “file”) or directory (type=”dir”).
source_id	YES	List of the file IDs that were used during the creation of this file.
parent_dir	YES	Defines the file_id of the directory containing it. Personal user’s root directory is set as “0”
files_dir	DEPENDENT	If the path_type = “dir”, then this parameter defines the file_ids contained by the directory
creation_time	YES	This is the time inserted by the API and is not required from the user.
file_type	YES	Format file. Several formats may support a same file_type. For example FASTQ
data_type	YES	Semantic description of file content. MuG accepted file types listed below. For example, WGBS reads
size	YES	Size in bytes
taxon_id	YES	The taxonomic ID of the species from which the data was taken. Whether the file has been compressed. Type of compression used depends on the format in question.
compressed	NO	Whether the file has been compressed. Type of compression used depends on the format in question.
meta_data	DEPENDENT	There are cases where additional data is required for some files that is not relevant to other file types. It corresponds a catch-all section. Examples: - ‘assembly’ - Files that have been generated and are dependent on alignments require that the meta_data has as the key with the assembly for which the alignment was made against - ‘paired’ - If the file is a BAM, the type of library is required. Options: “paired”, “unpaired” - ‘sorted’ - If the file is a BAM, whether the reads are sorted or not is required- Boolean

Table 8.4: File attributes

(*) Described in this annex under “File Type” and “Data type” sections

Data Model: “File Type” & “Data Type”

“File” data model includes attributes “*data type”* and “*file type”,* among other fields. Here, the complete collection of *data types* supported by MuGVRE, and their associated *file types*.

Data Type (identifier)	**Data Type (Name)**	**Associated File Types**
chromatin_3dmodel	Chromatin 3D structure	PDB
chromatin_3dmodel_ensemble	Ensemble of chromatin 3D structures	JSON
chromatin_compartments	Chromatin compartments data	TXT
chromatin_tads	Chromatin TADs	BED, TXT,
chromatin_traj	Chromatin trajectory	DCD
configuration_file	Tool configuration file	JSON, TXT, TSV
data_atac_seq	ATAC-Seq	FASTQ, BAM, BED, WIG

data_chip_seq	ChIP-Seq	BED, FASTQ, BAM, TSV
data_dna_methylation	DNA methylation	FASTQ, WIG, TSV
data_fish	FISH data	LIF, TIFF, PNG
data_mnase_seq	MNase-Seq	FASTQ, BAM, BED
data_rna_seq	RNA-Seq	FASTQ, TSV, HDF5, JSON
data_wgbs	Whole Genome Bisulfite Sequencing	FASTQ, BAM, BAI, WIG, TSV, TXT
docking_ranking	Docking ranking score	CSV, TXT, TSV
hic_biases	HiC Biases	PICKLE
hic_contacts_coverage	HiC contacts coverage	WIG, BW, TXT
hic_contacts_differential	HiC differential contacts	TSV
hic_contacts_matrix	HiC contact matrix	TXT, HDF5
hic_contacts_peaks	HiC contact peaks	TSV
hic_directionality	HiC directionality index	TXT
hic_reads	HiC sequencing reads	FASTQ
hic_sequences	HiC aligned reads	BAM
hic_tads_scale	HiC TADs scaling factor	WIG
md_restart	MD restart file	RST, CPT
na_md_atom_traj_coords	Nucleic acid MD trajectory coordinates	XTC, NETCDF, MDCRD
na_md_atom_traj_top	Nucleic acid MD trajectory topology	TOP, TPR, PARMTOP, PDB
na_md_cg_traj	Nucleic acid MD CG trajectory	MDCRD
na_structure	Nucleic acid 3D structure	PDB
na_traj	Nucleic acid trajectory	DCD, MDCRD
na_traj_coords	Nucleic acid trajectory coordinates	XTC, NETCDF, MDCRD
na_traj_top	Nucleic acid topology	TOP, TPR, PARMTOP, PDB
nucleosome_dynamics	Nucleosome dynamics	BW, GFF3, BED, WIG, RDATA
nucleosome_free_regions	Nucleosome free regions	BW, GFF3, BED, WIG
nucleosome_gene_phasing	Nucleosome phasing	BW, GFF3, BED, WIG
nucleosome_positioning	Nucleosome positioning	BW, GFF3, BED, WIG, TXT
nucleosome_stiffness	Nucleosome stiffness	BW, GFF3, BED, WIG
prot_dna_specificity	Protein-DNA specificity	TSV
prot_dna_structure	Protein-DNA complex structure	PDB
prot_structure	Protein 3D structure	PDB
sequence_annotation	Sequence Annotation	BED, BB, BEDGRAPH, WIG, BW, GFF, GFF3, GTF, VCF, TBI
sequence_dna	DNA sequence	FASTA, TXT
sequence_genomic	Genomic sequence	FASTA
sequence_mapping_index_bowtie	Bowtie2 index files	BT2, TXT
sequence_mapping_index_bwa	BWA index files	AMB, ANN, BWT, PAC, SA
sequence_mapping_index_gem	Sequence mapping index	GEM
sequence_mapping_index_kallisto	Kallisto index file	IDX
sequence_prot	Protein sequence	FASTA
sequence_rna	RNA sequence	FASTA

structure	3D structure	PDB
tool_intermediate_file	Tool Intermediate file	TAR
tool_statistics	Tool summary file	TAR
tss_classification_by_nucleosomes	Nucleosome TSS	BW, GFF3, BED, WIG

Table 8.5: MuGVRE file types

8.4.2 Data Model : "Tool"

"Tool" data model is specified using a JSON schema (draft 4) we prepared in order to (i) validate data structures received from tool developers, and (ii) validate MuGVRE internal operations before database insertion.

1. tool definition JSON - schema https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_specification/tool_schema.json
2. tool definition JSON - schema (internal) https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_specification/tool_schema_internal.json
3. tool definition JSON – example (another example annexed below) https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_specification/examples/pydockdna.json

Following, an example of a MuGVRE tool entry as registered in the database


```
{
    "_id": "pydockdna",
    "name": "pyDockDNA",
    "title": "Protein-DNA docking",
    "short_description": "PyDockDNA is a tool for the structural prediction of [..]",
    "owner": {
        "institution": "BSC",
        "author": "Brian Jimenez-Garcia",
        "contact": "bjimenez@bsc.es",
        "url": "https://life.bsc.es/pid/pidweb/default/tools"},
    "external": true,
    "has_custom_viewer": true,
    "keywords": [ "protein","dna"],
    "infrastructure": {
        "memory": 4.0,
        "cpus": 4,
        "executable": "/home/user/bin/mug/pydockdna",
        "clouds": {
            "mug-bsc": {
                "launcher": "PMES",
                "workflowType": "Single",
                "default_cloud": true,
                "minimumVMs":1,
                "initialVMs":1,
                "imageName":"uuid_pydockdna_51"},
            "mug-irb": {
```

```
            "launcher": "SGE",
            "queue":"pydock.q"}}},
    "input_files": [
        {
            "name": "receptor",
            "description": "Receptor PDB structure",
            "help": "Select the input receptor",
            "file_type": [ "PDB" ],
            "data_type": [ "prot_structure" ],
            "required": true,
            "allow_multiple": false},
        {
            "name": "ligand",
            "description": "Ligand PDB structure",
            "help": "Select the input ligand",
            "file_type": ["PDB"],
            "data_type": ["na_structure","prot_structure"],
            "required": true,
            "allow_multiple": false },
        {
            "name": "complex",
            "description": "Complex PDB structure",
            "help": "Select the complex",
            "file_type": [ "PDB"],
            "data_type": ["prot_dna_structure"],
            "required": false,
            "allow_multiple": false
        }
    ],
    "input_files_combinations": [
        ["receptor","ligand"],
        ["complex"]
    ],
    "arguments": [
        {
            "name": "scoring",
            "description": "Available energetic scoring functions",
            "help": "pyDockDNA is able to use different scoring […]",
            "type": "enum",
            "enum_items": {
                "description": [ "pyDock-DNA" , “VdW” ],
                "name": ["pydockdna", “VdW”]},
            "required": true,
            "default": "PyDock DNA"},
        {
            "name": "models",
            "description": "Number of structures to model",
            "help": "The number of generated models by this tool",
            "type": "enum",
            "enum_items": {
                "name": ["1", "5","10", "50"]},
            "minimum": 1,
            "maximum": 50,
            "required": true,
            "default": "10"}
    ],
    "output_files": [
        {
            "name": "energy_table",
            "required": true,
            "allow_multiple": false,
            "file": {
                "file_type": "CSV",
                "data_type": "docking_ranking",
                "compressed": "null",
```

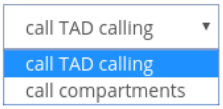
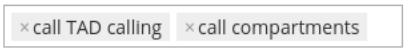
```
                "meta_data": {
                    "description": "Ranking of docking poses",
                    "visible": true}}},
        {
            "name": "top10",
            "required": true,
            "allow_multiple": true,
            "file": {
                "file_type": "PDB",
                "data_type": "prot_dna_structure",
                "compressed": "null",
                "meta_data": {
                    "description": "Top10 predicted structures",
                    "visible": true}}},
         {
            "name": "results",
            "required": true,
            "allow_multiple": false,
            "file": {
                "file_type": "TAR",
                "data_type": "tool_statistics",
                "compressed": "gzip",
                "meta_data": {
                    "description": "Compressed results of the tool",
                    "visible": false}}}
    ]
}
```

Snippet 8.1 Example of MuGVRE tool: pyDockDNA

A description of the individual attributes appearing in “Tool” schema is listed in the following table:

Field name	**Field type**	**Field description**
_id	string	Internal tool identifier. It cannot have spaces nor special characters.
name	string	Short tool name, which appears in the MuGVRE tool selectors. Maximum characters: 25.
title	string	Long tool name, which appears in the specific tool MuGVRE pages. Maximum characters: 50
short_description		Short description of the tool. It is going to appear on the “Launch Tool” table that list all integrated tools. Maximum characters: 150.
short_description		Longer description of the tool. It is going to appear on the “Launch Tool” table, after selecting the tool, as well as at the home page. Maximum characters: 500.
owner	object	Defines tool’s author
owner.institution	string	Research institute affiliation
owner.author	string	Tool’s author’s name
owner.contact	string	Contact mail of tool author.
owner.url	string	Tool reference URL (web page, code repository, etc)
owner.licence	string	Type of code licence. Check accepted options in the schema.
publication	string	DOI reference of the tool publication, if any.

Field name	Field type	Field description
keywords	array	Keywords that will be used in the main home page to classify the tool. Available options are: “Chromatin”, “Dynamics”, “Visualization”, “Hi-C”, “Epigenetics”, “Nucleosome”, “Modelling”, “Next Gen Sequencing”, “Interaction”, “RNA”, “DNA”, “Protein”, “Sequence”, “Structure”, “other”. Check them also at Check accepted options in the schema.
keywords_specific	array	Keywords that will be used in the “Launch Tool” table for discovering tools
infrastructure	object	Defines how the tool VM is to be deployed
infrastructure. memory	string	Size of the tool virtual machine. RAM memory in Gbs. By default “1.0” (1Gb).
infrastructure.cpus	integer	Size of the tool virtual machine. Cores. By default, 1.
infrastructure. executable	string	Path to the main executable of the tool. This file will be called by the MuGVRE core when the end user selects your tool from the web portal. It should be the absolute path, and be local to the tool virtual machine with execution permissions (+x).
infrastructure.clouds	object	List of ‘cloud’ objects. Define in which MuG cloud your tool VM is loaded. Formally, available options are in the schema, yet currently, the only cloud in production is “mug-irb”.
infrastructure.cloud	string	Defines the deployment requirements in each cloud installation
Infrastructure.cloud .workflowType	string	Type of workflow orchestration implemented in the tool. Check accepted options in the schema.
Infrastructure.cloud .default_cloud	boolean	Whether the cloud is the default one or not. If only one production cloud available, by default it is set to “true”.
input_files	array	List of ‘input files’ objects. Defines the input files accepted by the tool.
input_files.name	string	Unique name that will identify the input file. Used in the execution configuration files to refer to this input_files object. No spaces or special characters allowed.
input_files.description	string	Full input file name as it will be printed in the tool web form.
input_files.help	string	Long description of the input file. It will appear in the tool web for under the ‘question mark’ toolkit.
input_files.file_type	array	List of the allowed file formats for the input file. A comprehensive list of the accepted file types is in the file types section, yet, check accepted options in the schema.
input_files.data_type	array	List of allowed data types for the input file. A comprehensive list of the accepted file types is in the data types section, yet, check the identifiers for the accepted options in the schema.
Input_files.required	boolean	true if not optional
input_files. allow_multiple	boolean	true if more than one instance of this input file is allowed.
arguments	array	List of ‘arguments’ objects. Defines all the arguments accepted by the tool
arguments.name	string	Unique name that will identify the argument. Used in the execution configuration files to refer to this argument object. No spaces or special characters allowed.

Field name	**Field type**	**Field description**
arguments.description	string	Full argument name as it will be printed in the tool web form.
arguments.help	string	Long description of the input file. It will appear in the tool web for under the 'question mark' toolkit.
arguments.type	string	Type of the argument. Accepted options are "integer", "number","string", "enum","enum_multiple", "boolean", yet you can check them in the schema. There is correspondence between this type and the way the argument is displayed in the web form. For instance: "type: integer" → visualized as a input text field  "type:boolean" → visualized as a ON/OFF button.  "type: enum" → visualized as a drop down selector  "type: boolean multiple" → visualized as a multiple drop down selector 
arguments.required	boolean	true if not optional
arguments. allow_multiple	boolean	true if more than one instance of this input file is allowed.
arguments.default	string	Default value given for the argument
output_files	array	List of outputfiles' objects. Defines the output files generated by the tool.
output_files.name	string	Unique name that will identify the output file. Used in supplementary file that defines the tool output medatada in some special cases. No spaces or special characters allowed.
output_files.required	boolean	true if not optional. A tool execution will be considered as failed if a output file set as required is not generated.
output_files.allow_multiple	boolean	true if more than one instance of this output file is generated.
output_files.file	object	Defines the metadata associated with the output file
output_files.file.file_type	string	File format for the output file that the tool will generate. A comprehensive list of the accepted file types is in the supported data section, yet, check accepted options in the schema.
output_files.file. data_type	string	Data types for the output file the tool will generate. A comprehensive list of the accepted file types is in the supported data section, yet, check the identifiers for the accepted options in the schema.
output_files.file. file_path	string	File name for the expected output name. If the tool names the output file after the input file, or the file name is dynamically set during the execution, this field does not need to be defined here, but when wrapping your tool through a second tool output medatada file.
output_files.file. compressed	string	Defines whether the output file is compressed. And if so, which algorithm is used. Check accepted options in the schema.
output_files.file.	object	Defines all extra metadata attributes of the output file.

Field name	Field type	Field description
metadata		
output_files.file. description	string	Short description for the outfile generated. End use will have it accessible at his workspace
output_files.file. visible	boolean	If set to false, the output file will be saved and registered by MuGVRE, but not listed in the user workspace. A hidden output file is useful when the tool wants to send some data to MuGVRE transparently. Default: true

Table 8.6: Attributes description of Tool data model

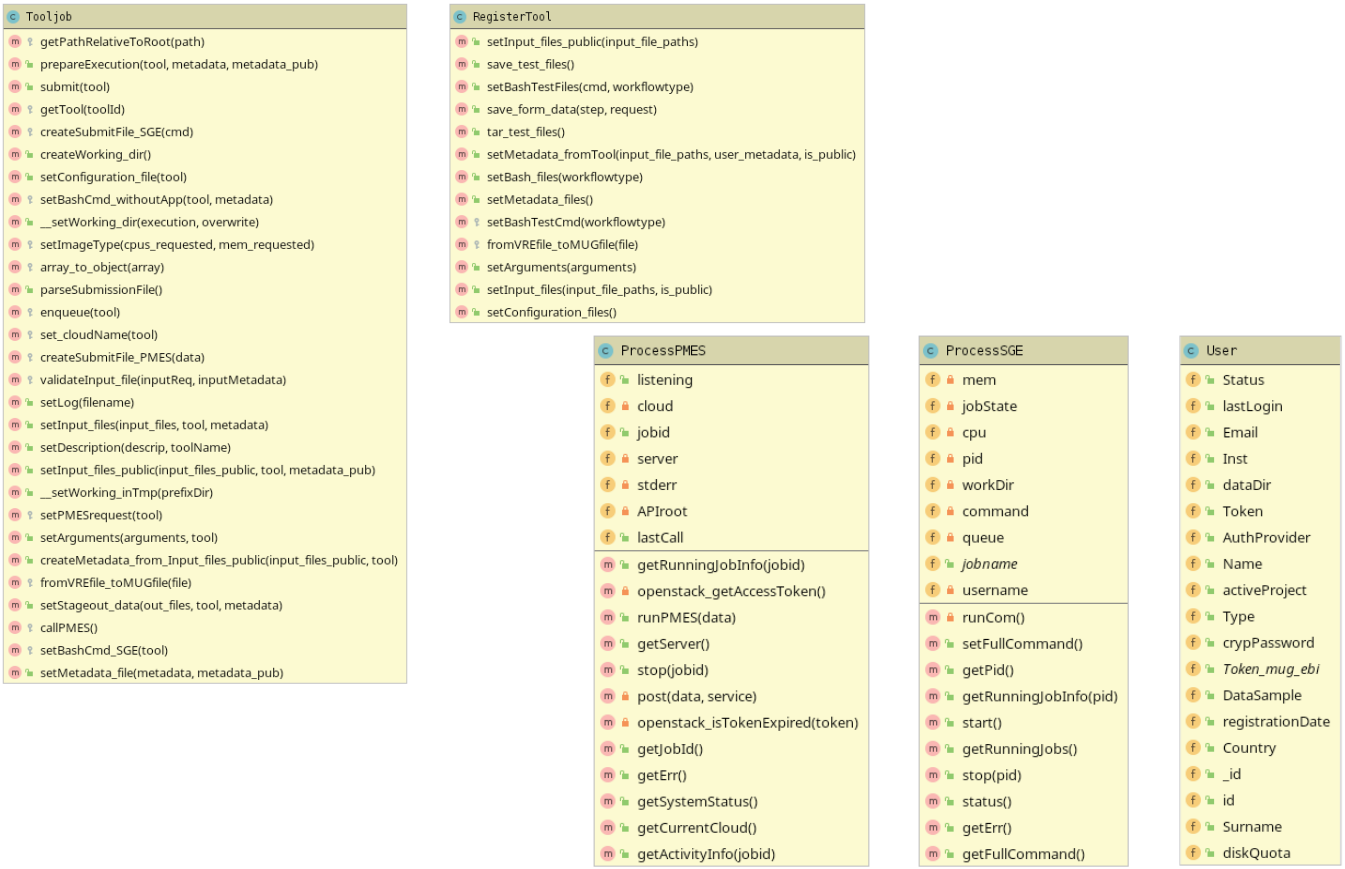
8.4.3 Job Auxiliary Files

These are illustrative examples of the files MuGVRE builds to submit a job (1, 2, 3). The last example corresponds to how MG-tool-API annotates the metadata for the output files generated during the execution.

1. Input metadata File (example) https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_execution/sample_project/myPydockProject/.input_metadata.json
2. Configuration Job File (example) https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_execution/sample_project/myPydockProject/.config.json
3. PMES JSDL File (example) https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_execution/sample_project/myPydockProject/.submit
4. Output metadata File (example) https://github.com/Multiscale-Genomics/VRE_tool_jsons/blob/dev/tool_execution/sample_project/myPydockProject_out/.results.json

8.5 OpenVRE classes

Following, a schema of the main PHP programming classes used at OpenVRE.



8.6 PMES documentation

8.6.1 PMES server and dashboard

1 Configuration

1.1 User configuration

PMES vm should have the user pmes.

1.2 Folder structure

At pmes home directory (/home/pmes/) should be the following folders:

- pmes
  - Dashboard
  - logs
  - config
  - jobs

tomcat7 user should have permission to write on folders logs and jobs. Give permissions to the tomcat7 user:

```
sudo usermod -a -G tomcat7 pmes
sudo chmod g+w myfolder
```

1.3 config file

The config file should be at `/home/pmes/pmes/config/config.xml`. The structure is as follows:

```
<pmes>
   <workspace>/home/pmes/pmes</workspace>
   <connector className="rOCCIHelper">
       <property>
           <key>providerName</key>
           <value>ONE</value>
       </property>
       ...
   </connector>
   <hosts>
       <host>
           <name>host12</name>
          <MAX_CPU>2400</MAX_CPU>
          <MAX_MEM>99195808</MAX_MEM>
       </host>
       ...
   </hosts>
   <logPath>/home/pmes/pmes/logs</logPath>
   <logLevel>DEBUG</logLevel>
   <timeout>60</timeout>
   <pollingInterval>5</pollingInterval>
   <runCmd>
       <cmd>echo "config script"</cmd>
   </runCmd>
   <auth-keys>
       <key>...</key>
   </auth-keys>
</pmes>
```

1.4 ssh

Disable known_hosts

```
Add "StrictHostKeyChecking no" to /etc/ssh/ssh_config
cd ~/.ssh
rm known_hosts
ln -s /dev/null known_hosts
```

1.5 logs

Logs will be at `/home/pmes/pmes/logs/`. Tomcat logs are at `$CATALINA_HOME/logs/catalina.out`

2 Deploy PMES

2.1 PMES Service

The PMES service is deployed using tomcat7. To deploy PMES service copy `pmes.war` to webapps folder (usually at `/var/lib/tomcat7/webapps`) and restart tomcat.

```
sudo service tomcat7 stop
sudo cp -r pmes.war /var/lib/tomcat7/webapps/
sudo service tomcat7 start
```

2.2 Dashboard

The Dashboard service is deployed using pm2.

```
cd /home/pmes/pmes/Dashboard/PMES2Dash/
# Install Dependencies
npm install --save
# Init pm2 Dashboard service
pm2 init /home/pmes/pmes/Dashboard/PMES2Dash/pm.yaml
```

Data is stored at mongodb database `pmes2`.

```
# start mongo service
sudo service mongodb start
# Open mongo console
mongo
# Inside mongo console use pmes2 database
> use pmes2
# show pmes collections
> show collections
# show pmes users
db.users.find()
```

3 Usage

3.1 Dashboard

PMES Dashboard is deployed using pm2. The endpoint is `http://localhost:3000`.
There is an initial user created `user: pmes@pmes.com, password: pmes`

3.2 PMES Service

PMES service is deployed using tomcat7. The endpoint is `http://localhost:8080/pmes/pmes/`.
You can call the service using curl (see APIDefinition document) or you can call the service using the python script `/home/pmes/pmes/scripts/curlPmesApi.py`.

```
# api call getSystemStatus
python3 curlPmesApi.py getSystemStatus
# api call getActivityReport
python3 curlPmesApi.py getActivityReport job_id
# api call terminateActivity
python3 curlPmesApi.py terminateActivity job_id
# api call createActivity
python3 curlPmesApi.py createActivity createVM.json
```

`createVM.json` is a json file with a job definition. For example:

```
[{ "jobName": "HelloTest2_584817558cb7550b5e9970b0",
          "wallTime": "5",
          "minimumVMs": "1",
          "maximumVMs": "1",
          "limitVMs": "1",
          "initialVMs": "1",
          "memory": "1.0",
          "cores": "1",
          "disk" : "1.0",
          "inputPaths": ["/home/"],
          "outputPaths": ["/home/"],
          "mountPath":"",
          "numNodes": "1",
           "user":
             { "username": "lcodo",
               "credentials":
               { "pem": "/home/pmes/certs/pmes.pem",
                 "key": "/home/pmes/certs/pmes.key"}
             },
           "img":
             { "imageName": "os_tpl#4f916ede-218b-47e4-93aa-b795a5acf813",
               "imageType": "resource_tpl#721112dd-2f33-40eb-8975-7bd34dbabfc8"
               "cores": "2"
               "memory": "2.0"
               "disk": "4.0"
             },
           "app":
             { "name": "HelloTest2",
               "target": "/home/pmes/testSimple",
               "source": "launch.sh",
               "args": { "val1": "Hola", "val2": "Mundo" } ,
               "type": "COMPSs"
             },
           "compss_flags": {}
       }]
```

4 Dependencies

The image and the template should have the following permissions: Use and Manage for user, group and other.

4.1 PMES VM

Dependencies:

- Rocci Client - {https://github.com/gwdg/rOCCI-cli; https://rvm.io/rvm/install#explained}

  ```
  # Install occi client
  curl -L http://go.egi.eu/fedcloud.ui | /bin/bash -
  ```

  If occi uses certificates move grid-security certificates to /etc/

- tomcat7: install tomcat7 `sudo apt-get install tomcat7`

  be sure that tomcat7 is using java8. (Export java home)

  ```
  sudo nano /usr/share/tomcat7/bin/setenv.sh
  export JAVA_HOME=/usr/lib/jvm/java-8-oracle/
  ```

  if default tomcat7 user is used: no extra configuration is needed.

  if the tomcat user is changed to pmes: the following configuration is needed.

```
sudo nano /etc/default/tomcat7 # change TOMCAT7_USER=pmes, TOMCAT7_GROUP=pmes

sudo nano /etc/init.d/tomcat7 # change TOMCAT7_USER=pmes, TOMCAT7_GROUP=pmes
```

- mongodb: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu/
- pm2
- node (version >= 0.8)

4.2 APP VM

Dependencies:

1. COMPSs
2. cloud-init: http://cloudinit.readthedocs.io/en/latest/topics/examples.html
3. package nis or cifs to mount shared folders. (see document mountFolders)

8.6.2 PMES REST API

PMES API Definition

Sandra Corella - Workflows and distributed computing

Draft - December 2016

1 API Definition

1. **createActivity**: Submits a list of jobs to the PMES service.
2. **getActivityStatus**: Retrieves the JobStatus object of a set of submitted jobs.
3. **getActivityReport**: Gets the activity documents of a set of jobs giving the: JSDLs, jobs status, execution progress, elapsed time and error messages.
4. **terminateActivity**: Terminates a set of submitted jobs.
5. **getSystemStatus**: Provides information about the resources consuption of the system.

Method	Name	Input	Return
POST	createActivity	`ArrayList<JobDefinition> jobDef`	`ArrayList<String> jobIds`
POST	getActivityStatus	`ArrayList<String> jobids`	`ArrayList<JobStatus> jobStatus`
POST	getActivityReport	`ArrayList<String> jobids`	`ArrayList<JobReport> jobReports`
POST	terminateActivity	`ArrayList<String> jobIds`	`ArrayList<String> terminateMessages`
GET	getSystemStatus	-	`SystemStatus`

Table 1: PMES API specification

2 Types

2.1 Main Types

- JobDefinition:

JobDefinition	
Type	name
String	id
String	jobName
App	app
Image	image
User	user
[String]	inputPaths
[String]	outputPaths
String	mountPath
Integer	wallTime
Integer	numNodes
Integer	cores
Float	memory
Float	disk
`HashMap<String, String>`	compss_flags
Integer	initialVMs
Integer	minimumVMs
Integer	maximumVMs
Integer	limitVMs

- JobStatus

JobStatus
PENDING
RUNNING
FINISHED
CANCELLED
FAILED
ALL

- JobReport

JobReport	
Type	name
JobDefinition	jobDefinition
String	jobOutputMessage
String	jobErrorMessage
JobStatus	jobStatus
String	elapsedTime

- SystemStatus

SystemStatus	
Type	name
`ArrayList<Host>`	cluster

2.2 Secondary Types

- App:

App	
Type	name
String	id
String	name
String	target
String	source
String	type
`HashMap<String, String>`	args

- Image:

Image	
Type	name
String	id
String	imageName
String	imageType

- User:

User	
Type	name
String	username
`HashMap<String, String>`	credentials

The credentials should have: uid, gid and token or key and pem.

- Host:

Host	
Type	name
String	name
Integer	usedCores
Integer	totalCores
Float	usedMemory
Float	totalMemory

3 Usage example

```
> curl http://localhost:8080/pmes/pmes/getSystemStatus
{"cluster":[
    {"name":"bsccv14",
     "usedCores":1,
     "totalCores":2400,
     "usedMemory":1.0,
     "totalMemory":9.9195808E7},
    {"name":"bsccv15",
     "usedCores":0,
     "totalCores":2400,
     "usedMemory":0.0,
     "totalMemory":9.9195808E7}
    ]
}
```

Listing 1: getSystemStatus

```
> curl -H 'Content-Type: application/json'
       -X POST
       --data '["18045e7f-a670-46fe-a067-3b1a19870bcf"]'
       http://localhost:8080/pmes/pmes/getActivityStatus
```

Listing 2: getActivityStatus

```
> curl -H 'Content-Type: application/json'
       -X POST
       --data '["18045e7f-a670-46fe-a067-3b1a19870bcf"]'
       http://localhost:8080/pmes/pmes/terminateActivity

["Job with id 18045e7f-a670-46fe-a067-3b1a19870bcf cannot be
    cancelled,
  the job has been finished."]
```

Listing 3: terminateActivity

```
> curl -H 'Content-Type: application/json'
       -X POST
       --data
       '[{ "jobName": "HelloTest2_584817558cb7550b5e9970b0",
           "wallTime": "5",
           "minimumVMs": "1",
           "maximumVMs": "1",
           "limitVMs": "1",
           "initialVMs": "1",
           "memory": "1",
           "cores": "1",
           "inputPaths": ["/home/data.txt"],
           "outputPaths": ["/home/result.txt"],
           "mountPath": "/data2/test/usr1"
           "numNodes": "1",
           "user":
              { "username": "usr1",
                "credentials":
                { "pem": "/home/pmes/certs/usr1.pem",
                  "key": "/home/pmes/certs/usr1.key",
                  "uid": "306",
                  "gid": "306",
                  "token": "12345"}
              },
           "img": { "imageName": "uuid_pmescompss_83", "imageType":
    "small" },
           "app":
              { "name": "HelloTest2",
                "target": "/home/pmes/testSimple",
                "source": "launch.sh",
                "args": { "val1": "Hola", "val2": "Mundo" }
              }
        }]'
    http://localhost:8080/pmes/pmes/createActivity

["31eb1268-b6bc-4be2-9fa9-f8a046b752db"]
```

Listing 4: createActivity

8.7 Publications

Following, the list of papers related to each of the sections present dissertation:

8.7.1 Data management infrastructure for IMIDs' research

1) Aterido, Adrià, Cañete, J. D., Tornero, J., Ferrándiz, C., Pinto, J. A., Gratacós, J., … Codó, L., Gelpí, J. L. …, Julià, A. (2019). Genetic variation at the glycosaminoglycan metabolism pathway contributes to the risk of psoriatic arthritis but not psoriasis. Annals of the Rheumatic Diseases, 78(3), 355–364. https://doi.org/10.1136/annrheumdis-2018-214158

2) Aterido, Adrià, Palau, N., Domènech, E., Nos Mateu, P., Gutiérrez, A., Gomollón, F., … Codó, L., Gelpí, J. L. …, Julià, A. (2019). Genetic association between CD96 locus and immunogenicity to anti-TNF therapy in Crohn's disease. The Pharmacogenomics Journal. https://doi.org/10.1038/s41397-019-0090-4

3) Aterido, Adrià, Julià, A., Ferrándiz, C., Puig, L., Fonseca, E., Fernández-López, E., … Codó, L., Gelpí, J. L. …, Marsal, S. (2016). Genome-Wide Pathway Analysis Identifies Genetic Pathways Associated with Psoriasis. Journal of Investigative Dermatology, 136(3), 593–602. https://doi.org/10.1016/j.jid.2015.11.026

4) Julià, A., Blanco, F., Fernández-Gutierrez, B., González, A., Cañete, J. D., Maymó, J., … Codó, L., Gelpí, J. L. …, Marsal, S. (2016). Identification of IRX1 as a Risk Locus for Rheumatoid Factor Positivity in Rheumatoid Arthritis in a Genome-Wide Association Study. Arthritis & Rheumatology, 68(6), 1384–1391. https://doi.org/10.1002/art.39591

5) Julià, A., González, I., Fernández-Nebro, A., Blanco, F., Rodriguez, L., González, A., … Codó, L., Gelpí, J. L. …, Marsal, S. (2016). A genome-wide association study identifies SLC8A3 as a susceptibility locus for ACPA-positive rheumatoid arthritis. Rheumatology, 55(6), 1106–1111. https://doi.org/10.1093/rheumatology/kew035

6) Julià, A., Pinto, J. A., Gratacós, J., Queiró, R., Ferrándiz, C., Fonseca, E., … Codó, L., Gelpí, J. L. …, Marsal, S. (2015a). A deletion at ADAMTS9-MAGI1 locus is associated with psoriatic arthritis risk. Annals of the Rheumatic Diseases, 74(10), 1875–1881. https://doi.org/10.1136/annrheumdis-2014-207190

7) Julià, A., Domènech, E., Ricart, E., Tortosa, R., García-Sánchez, V., Gisbert, J. P., … Codó, L., Gelpí, J. L. …, Marsal, S. (2013). A genome-wide association study on a southern European population identifies a new Crohn's disease susceptibility locus at RBX1-EP300. Gut, 62(10), 1440–1445. https://doi.org/10.1136/gutjnl-2012-302865

8.7.2 MuG: Multiscale Complex Genomics VRE

8) Buitrago, D†, . Codó, L.†, Illa, R., de Jorge, P., Battistini, F., Flores, O., … Orozco, M. (2019). Nucleosome Dynamics: a new tool for the dynamic analysis of nucleosome positioning. Nucleic Acids Research. https://doi.org/10.1093/nar/gkz759

† *The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors*

9) Codó, L., Bayarri, G., Cid-Fuentes, J. A., Conejero, J., Hospital, A., Royo, R., … Gelpí, J. L. (2019). MuGVRE. A virtual research environment for 3D/4D genomics. BioRxiv, 602474. https://doi.org/10.1101/602474

10) Hospital, A., Andrio, P., Cugnasco, C., Codó, L., Becerra, Y., Dans, P. D., … Gelpí, J. L. (2016). BIGNASim: a NoSQL database structure and analysis portal for nucleic acids simulation data. Nucleic Acids Research, 44(D1), D272–D278. https://doi.org/10.1093/nar/gkv1301

8.7.3 Thesis not related papers

11) Andrio, P., Hospital, A., Conejero, J., Jordà, L., del Pino, M., Codó, L., … Gelpí, J. L. (2019). BioBB: BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows. Nature Scientific Data, 6(1), 169. https://doi.org/10.1038/s41597-019-0177-4

12) Georgieva, M. V, Yahya, G., Codó, L., Ortiz, R., Teixidó, L., Claros, J., … Aldea, M. (2015). Inntags: small self-structured epitopes for innocuous protein tagging. Nature Methods, 12(10), 955–958. https://doi.org/10.1038/nmeth.3556

1) Aterido, Adrià, Cañete, J. D., Tornero, J., Ferrándiz, C., Pinto, J. A., Gratacós, J., … Codó, L., Gelpí, J. L. …, Julià, A. (2019). Genetic variation at the glycosaminoglycan metabolism pathway contributes to the risk of psoriatic arthritis but not psoriasis. Annals of the Rheumatic Diseases, 78(3), 355–364. https://doi.org/10.1136/annrheumdis-2018-214158

CLINICAL SCIENCE

Genetic variation at the glycosaminoglycan metabolism pathway contributes to the risk of psoriatic arthritis but not psoriasis

Adrià Aterido,[1,2] Juan D Cañete,[3] Jesús Tornero,[4] Carlos Ferrándiz,[5] José Antonio Pinto,[6] Jordi Gratacós,[7] Rubén Queiró,[8] Carlos Montilla,[9] Juan Carlos Torre-Alonso,[10] José J Pérez-Venegas,[11] Antonio Fernández Nebro,[12] Santiago Muñoz-Fernández,[13] Carlos M González,[14] Daniel Roig,[15] Pedro Zarco,[16] Alba Erra,[17] Jesús Rodríguez,[18] Santos Castañeda,[19] Esteban Rubio,[20] Georgina Salvador,[21] Cesar Díaz-Torné,[22] Ricardo Blanco,[23] Alfredo Willisch Domínguez,[24] José Antonio Mosquera,[25] Paloma Vela,[26] Simon Angel Sánchez-Fernández,[27] Héctor Corominas,[22,28] Julio Ramírez,[3] Pablo de la Cueva,[29] Eduardo Fonseca,[30] Emilia Fernández,[31] Lluis Puig,[32] Esteban Dauden,[33] José Luís Sánchez-Carazo,[34] José Luís López-Estebaranz,[35] David Moreno,[36] Francisco Vanaclocha,[37] Enrique Herrera,[38] Francisco Blanco,[39] Benjamín Fernández-Gutiérrez,[40] Antonio González,[41] Carolina Pérez-García,[42] Mercedes Alperi-López,[8] Alejandro Olivé Marques,[43] Víctor Martínez-Taboada,[23] Isidoro González-Álvaro,[19] Raimon Sanmartí,[3] Carlos Tomás Roura,[44] Andrés C García-Montero,[45] Sílvia Bonàs-Guarch,[46] Josep Maria Mercader,[46] David Torrents,[46,47] Laia Codó,[48] Josep Lluís Gelpí,[48] Mireia López-Corbeto,[1] Andrea Pluma,[1] Maria López-Lasanta,[1] Raül Tortosa,[1] Nuria Palau,[1] Devin Absher,[49] Richard Myers,[49] Sara Marsal,[1] Antonio Julià[1]

Handling editor Josef S Smolen

► Additional material is published online only. To view please visit the journal online (http://dx.doi.org/10.1136/annrheumdis-2018-213588).

For numbered affiliations see end of article.

Correspondence to
Antonio Julià and Sara Marsal, Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona 08035, Spain; toni.julia@vhir.org, sara.marsal@vhir.org, Juan D Cañete, Rheumatology Department, Hospital Clínic de Barcelona and IDIBAPS, Barcelona 08036, Spain; jcanete@clinic.ub.es

Received 24 July 2018
Revised 16 November 2018
Accepted 16 November 2018
Published Online First 14 December 2018

To cite: Aterido A, Cañete JD, Tornero J, *et al*. *Ann Rheum Dis* 2019;**78**:355–364.

ABSTRACT

Objective Psoriatic arthritis (PsA) is a chronic inflammatory arthritis affecting up to 30% of patients with psoriasis (Ps). To date, most of the known risk loci for PsA are shared with Ps, and identifying disease-specific variation has proven very challenging. The objective of the present study was to identify genetic variation specific for PsA.

Methods We performed a genome-wide association study in a cohort of 835 patients with PsA and 1558 controls from Spain. Genetic association was tested at the single marker level and at the pathway level. Meta-analysis was performed with a case–control cohort of 2847 individuals from North America. To confirm the specificity of the genetic associations with PsA, we tested the associated variation using a purely cutaneous psoriasis cohort (PsC, n=614) and a rheumatoid arthritis cohort (RA, n=1191). Using network and drug-repurposing analyses, we further investigated the potential of the PsA-specific associations to guide the development of new drugs in PsA.

Results We identified a new PsA risk single-nucleotide polymorphism at *B3GNT2* locus (p=1.10e-08). At the pathway level, we found 14 genetic pathways significantly associated with PsA (p_{FDR}<0.05). From these, the glycosaminoglycan (GAG) metabolism pathway was confirmed to be disease-specific after comparing the PsA cohort with the cohorts of patients with PsC and RA. Finally, we identified candidate drug targets in the

Key messages

What is already known about this subject?
- Psoriatic arthritis (PsA) has a higher heritability than psoriasis, indicating the presence of a specific genetic risk component.
- So far, very little genetic variation has been specifically associated with the development of PsA.

What does this study add?
- Using a pathway-based genome-wide association study on two case–control cohorts, we have identified multiple pathways associated with PsA risk.
- The glycosaminoglycan (GAG) metabolism pathway is specifically associated with PsA risk and is not associated with purely cutaneous psoriasis or rheumatoid arthritis.
- Network-based analysis of the GAG pathway suggests two new drug candidates for PsA.

How might this impact on clinical practice or future developments?
- The integration of biological knowledge on to genetic analysis can improve the ability to identify more effective therapies for PsA.

GAG metabolism pathway as well as new PsA indications for approved drugs.

Conclusion These findings provide insights into the biological mechanisms that are specific for PsA and could contribute to develop more effective therapies.

INTRODUCTION

Psoriatic arthritis (PsA) is an inflammatory arthritis affecting up to 0.5% of the population and ~30% of patients with psoriasis (Ps).[1 2] Compared with patients having skin-only affectation (ie, purely cutaneous psoriasis, PsC), patients with PsA have a substantially worse quality of life.[3] Most of the drugs currently used to treat Ps are also indicated for PsA,[4] but the efficacy can differ significantly. Therefore, there is a need to better understand the biological mechanisms underlying PsA in order to develop more effective therapies.

Familiar aggregation studies have demonstrated that PsA has a larger sibling recurrence rate (λ_s) than Ps (PsA λ_s~37 vs Ps λ_s~7),[5–8] indicating the presence of a specific genetic risk. To date, more than 15 genome-wide association studies (GWAS) have been performed in Ps,[9–14] identifying more than 50 susceptibility loci. Conversely, only a few GWAS have been performed exclusively in PsA. These studies have allowed the identification of 15 PsA risk loci.[15–22] However, most of these risk variants are shared with Ps, indicating that the biological mechanisms that cause autoimmunity to the skin are also central for PsA. Identifying disease-specific loci has proven elusive, and to date only *PTPN22*, *CSF2-P4HA2* and *ADAMTS9-MAGI1* have shown a significant association with PsA. Taking into account the effects of all known risk loci, less than 50% of the heritability for PsA is currently explained.[11 13 23] Therefore, new biological mechanisms could still be discovered that are relevant for disease aetiology, resulting in more effective therapies than the present ones.

A major challenge in the genetics of complex diseases is the identification of genes with small effects.[24] To overcome this problem, the predominant approach has been to recruit increasingly larger patient cohorts.[25] While this can help to identify new risk variation, this is extremely costly and time-consuming. Most of these GWAS have been performed at the single marker level, and therefore the statistical power to detect new risk variation soon becomes insufficient. To address this issue, different strategies have been developed. One of the most successful approaches has been to leverage the biological information underlying DNA variation like biological pathway annotation.[26] Genome-wide pathway analysis (GWPA) efficiently integrates the risk variation from multiple, functionally related genes into a unique statistic.[27] Additionally, using well-curated biological information in GWPA significantly accelerates the translation of the genetic association results.[28] With this strategy, new genetic variation has been identified in different common diseases, including autoimmune diseases like Ps.[29 30]

To identify new genetic variation specifically associated with PsA, we have performed a GWAS at the single marker and pathway levels. We genotyped 835 patients with PsA and 1558 controls from Spain and performed a meta-analysis with a previous GWAS of PsA consisting of 1430 cases and 1417 controls from North America. Using this approach, we identified a new association at *B3GNT2* locus and 14 genetic pathways associated with PsA risk. We next tested these genetic associations in GWAS cohorts of patients with PsC and rheumatoid arthritis (RA), and we found the glycosaminoglycan (GAG) metabolism pathway to be specific for PsA. Based on this evidence, we propose the GAG metabolism as a new source for drug discovery in PsA. Using network analysis and knowledge on drug action, we find evidence that the GAG pathway could be a useful target to treat PsA. These findings confirm the utility of GWAS to identify specific biological mechanisms and suggest repositioning of existing drugs for PsA.

METHODS

Study population

Patients with PsA were selected from the rheumatology departments of 16 Spanish hospitals belonging to the Immune-Mediated Inflammatory Disease Consortium.[31] All patients with PsA were diagnosed according to the Classification Criteria for Psoriatic Arthritis.[32] Controls were recruited from healthy blood donors from Spanish hospitals in collaboration with the Spanish DNA Bank. A case–control cohort of 835 patients with PsA and 1588 controls were finally recruited and used for GWAS.

Meta-analysis was performed with a previous GWAS performed on 1430 patients with PsA and 1417 controls collected from USA and Canada.[15] To identify PsA-specific variation, we used GWAS data from a cohort of 614 patients with PsC and 1191 patients with RA from Spain. Patients with PsC were defined as patients with plaque-type Ps for >10 years and free of any inflammatory disease in the joints. The main features of these cohorts are shown in online supplementary material 1 and online supplementary table S1.

Genome-wide genotyping and imputation

GWAS genotyping of the 2393 individuals from Spain was performed using the Illumina Quad610 array (Illumina, USA). Genotype calling and quality control (QC) were performed using GenomeStudio V.2011.1 (Illumina) and PLINK software, respectively (online supplementary material 1 and online supplementary figure S1). After QC, 506 926 single-nucleotide polymorphisms (SNPs) from 744 patients with PsA and 1454 controls were available for analysis.

We conducted genotype imputation to facilitate meta-analysis with the North America GWAS data. Only high-quality and directly genotyped SNPs (n=506 926 SNPs) were used for this analysis. After prephasing the haplotypes of each loci using SHAPEIT V.2-644 software, imputation was conducted with the IMPUTE V.2 software.[33] We used the phase 1 release of the 1000 Genomes Project V.3 as reference panel.[34] Only SNPs showing an minor allele frequency (MAF) >0.05 and an imputation quality >0.8 were selected for the GWPA. After filtering, 1 387 382 variants were available for analysis.

GWAS genotyping of the independent PsA case–control cohort was performed using the Illumina HumanOmni1-Quad array (Illumina) as previously described.[15] After QC, 791 217 SNPs from 1430 patients with PsA and 1417 controls were used for the single marker and pathway meta-analyses.

The disease specificity of the validated loci and pathways was tested using GWAS data from two cohorts of patients with PsC and RA. These data sets were generated using the Illumina Quad610 array (Illumina) as previously described.[29 35]

In order to investigate the existence of PsA-specific variation across the human leukocyte antigen (HLA) region, we performed imputation of the classical alleles and amino acid polymorphisms from the HLA class I (*HLA-A*, *HLA-B*, *HLA-C*) and class II (*HLA-DPA1*, *HLA-DPB1*, *HLA-DQA1*, *HLA-DQB1* and *HLA-DRB1*) loci in the PsA and PsC cohorts from Spain. HLA imputation was conducted using the SNP2HLA V.1.0.3 software.[36]

Association analysis of single genetic markers

The SNPs previously associated with Ps risk (p<5e-08; online supplementary table S2) were tested for association using a logistic regression model. The same analytical procedure was followed to analyse the association between whole genome variation and PsA risk in the Spain cohort using the SNPTEST V.2 software.[37]

The imputed HLA alleles and amino acid polymorphisms from the HLA class I and class II loci were tested for association using a stepwise logistic regression model. In this analysis, the most strongly associated marker was included as a covariate by addition to the null model until no markers reached the significance threshold determined by the false discovery rate (FDR) method. Since HLA haplotypes *HLA-C*06:02* and *HLA-B-27* have been previously shown to be differentially associated in PsA compared with PsC,[38 39] the stepwise association analysis was started conditioning on these two established disease risk markers.

Genome-wide meta-analysis in PsA

The two independent PsA case–control cohorts were genotyped using different Illumina arrays ($n_{Quad610\text{-}QC}$=1 387 382 SNPs; $n_{HumanOmni1\text{-}QC}$=791 217 SNPs). Before GWAS meta-analysis, we identified the genetic variants that were commonly genotyped in both cohorts (n=720 582 SNPs; online supplementary table S3). We subsequently performed the GWAS meta-analysis using the METAL software.[40] The association statistics were weighted by the sample size of the two cohorts and adjusted for the genomic inflation factor (λ_{Spain} Spain 1.08, $\lambda_{NorthAmerica}$ North America 1.26).

Genome-wide pathway analysis

A total of 1077 reference pathways from the Molecular Signatures Database were included in the study. The SNP-gene mapping was performed using proximity-based criteria, which is the predominant approach in GWPA.[26 41] According to reference studies in GWPA,[26 28 42] we used an SNP-gene distance window of 20 Kb. The National Center for Biotechnology Information (NCBI) RefSeq Database Release 63 was used for SNP annotation (online supplementary table S4). Given the high linkage disequilibrium and gene density of the *HLA* region, the proximity-based criteria could yield false-positive results for pathways including genes within this locus. Similar to previous studies, the *HLA* region genes were excluded from the GWPA (online supplementary material 1).[29 43] The statistical association between genetic pathways and disease risk was analysed using the set-based method implemented in PLINK, as described in online supplementary material 1.[29 30]

Gene expression analysis of the GAG metabolism pathway

To investigate the specificity of the GAG metabolism pathway in PsA at the functional level, we used whole blood transcriptomic data obtained from a previous study on patients with PsA, patients with PsC and healthy controls (Gene Expression Omnibus data set: GSE61281).[44] In this previous study, the whole genome expression profile was evaluated aiming to identify differentially expressed genes. Here, based on our previous evidence at the genetic level, we hypothesised that the GAG pathway as a whole would be differentially expressed in PsA. To do this, after QC and quantile normalisation of the gene expression data, we tested for differential expression of the GAG metabolism pathway genes (t-test, nominal significance p=0.05) between patients with PsA (n=20) and PsC (n=20), and between each disease and healthy controls (n=12). We then used the binomial test to assess whether the observed number of differentially expressed genes in the pathway is greater than expected by chance. Furthermore, we also studied the changes in the coexpression of the GAG metabolism pathway between diseases. For this objective, we calculated the intramodular connectivity (IMC) measure. IMC is a network measure that efficiently captures gene coexpression information and is computed as the average of the gene connectivity within the pathway genes. The IMC is implemented in the WGCNA software.[45] Student's t-test was used to compare the IMC values of the GAG metabolism pathway between diseases.

Exploratory drug-repurposing analysis of the GAG pathway

To investigate the GAG metabolism as a new source for drug discovery in PsA, we combined network and drug-repurposing analyses. First, we performed a network analysis on the GAG metabolism pathway to identify those genes that are central to the network and, therefore, more likely to be key for the pathway functionality. Second, we screened the drugs approved by the Food and Drug Administration (FDA) to identify drugs that target central genes in the GAG metabolism pathway. Third, we defined a topology-based measure to evaluate the functional impact of these drugs on the GAG metabolism. Fourth, we compared the topology-based measure between the GAG metabolism and the rest of human biological processes for each of the candidate drugs. The details of this exploratory analysis are described in online supplementary material 1 and online supplementary tables S5, S6, S7.

RESULTS

Replication of established Ps risk variation

We found that 17 out of the 77 SNPs previously associated with Ps risk were also associated with PsA susceptibility in the Spain cohort (p<0.05; table 1). The 'DNA repair' pathway, a gene set previously associated with Ps risk, was also associated with PsA in our cohort (p=0.01; online supplementary table S8).

Identification of new genetic loci associated with PsA

In the GWAS meta-analysis, we identified five loci associated with PsA at the genome-wide scale (p<5e-08; table 2). From these, the *B3GNT2* locus (rs10865331, p=1.10e-08) has not been previously associated with PsA risk (figure 1). The complete list of associated markers is shown in online supplementary table S9.

In the association analysis of the HLA markers comparing patients with PsA and PsC, we confirmed the genome-wide significant association between the *HLA-C*06:02* allele and PsA risk (p=6.96e-11). The amino acid residues *HLA-B* Leu95 and *HLA-C* Ala305 were found to be also strongly associated with the risk of developing PsA (p<5e-8). In the stepwise conditional analysis, only the amino acid residue *HLA-A Ala77* remained significant (p_{FDR}<0.05). The complete list of associated HLA markers is shown in online supplementary table S10 (p<0.05).

Identification of genetic pathways associated with PsA

In the Spain cohort, we identified 76 genetic pathways significantly associated with PsA risk (p_{FDR}<0.05; online supplementary table S11). Fifty out of these pathways (65.8%) were found to include genes from the *HLA* region and/or the *IL12B* gene (online supplementary table S12), which are the strongest genetic risk loci for both Ps and PsA. To confirm that the observed pathway associations are not only due to the strong signals present at these loci, we retested the pathways after removing these regions (*HLA* region: 25.6–33.3 Mb in chromosome 6, n=4021 SNPs; *IL12B:* 158 741 791–158 757 481 bp in

Table 1 Established Ps risk variants associated with PsA susceptibility in the Spanish cohort

SNP	Chr	Pos	Gene	RA	Phenotype	P value	OR (95% CI)
rs12044149	1	67600686	*IL23R*	T	PsA-controls	1.73e-05	1.35 (1.18 to 1.56)
rs1990760	2	163124051	*IFIH1*	T	PsA-controls	1.32e-03	1.24 (1.09 to 1.41)
rs4921482	5	158764478	*IL12B, ADRA1B*	T	PsA-controls	6.73e-03	0.83 (0.72 to 0.95)
rs918520	5	158826310	*IL12B*	G	PsA-controls	4.26e-02	0.85 (0.72 to 0.99)
rs33980500	6	111913262	*TRAF3IP2*	T	PsA-controls	1.01e-07	1.84 (1.47 to 2.29)
rs4795067	17	26106675	*NOS2*	G	PsA-controls	1.29e-03	1.24 (1.09 to 1.41)
rs146571698	5	150471878	*TNIP1*	T	PsC-controls	8.10e-03	1.38 (1.09 to 1.74)
rs918520	5	158826310	*IL12B*	G	PsC-controls	4.26e-02	0.85 (0.72 to 0.99)
rs9481169	6	111929862	*TRAF3IP2*	T	PsC-controls	1.29e-05	1.55 (1.27 to 1.89)
rs7536201	1	25293084	*RUNX3*	C	Ps-controls	1.76e-03	1.22 (1.08 to 1.39)
rs9988642	1	67726104	*IL23R*	T	Ps-controls	1.90e-02	0.75 (0.59 to 0.96)
rs10865331	2	62551472	*B3GNT2*	A	Ps-controls	3.62e-07	0.72 (0.63 to 0.81)
rs17715343	2	163167746	*IFIH1*	C	Ps-controls	4.23e-03	0.73 (0.58 to 0.92)
rs27432	5	96119273	*ERAP1*	A	Ps-controls	3.00e-02	0.87 (0.76 to 0.99)
rs2233278	5	150467189	*TNIP1*	C	Ps-controls	7.67e-03	1.38 (1.09 to 1.75)
rs918520	5	158826310	*IL12B*	G	Ps-controls	4.26e-02	0.85 (0.72 to 0.99)
rs33980500	6	111913262	*TRAF3IP2*	T	Ps-controls	1.01e-07	1.84 (1.47 to 2.29)
rs610037	11	65546857	*AP5B1*	C	Ps-controls	4.02e-03	1.20 (1.06 to 1.37)
rs34394770	13	40333369	*COG6*	T	Ps-controls	4.11e-03	0.82 (0.71 to 0.94)
rs367569	16	11365500	*PRM3, SOCS1*	C	Ps-controls	1.72e-02	0.84 (0.73 to 0.97)
rs545979	18	51819750	*POLI*	T	Ps-controls	6.93e-03	1.21 (1.05 to 1.38)

P values <0.05 are shown.
Chr, chromosome; Phenotype, phenotype comparison; Pos, base pair in GRCh37/hg19; Ps, psoriasis; PsA, psoriatic arthritis; PsC, purely cutaneous psoriasis; RA, risk allele (genetic variants whose reference allele was found to confer risk for the indicated phenotype are shown); SNP, single-nucleotide polymorphism.

chromosome 5, n=81 SNPs). After excluding these regions, we found that 9 genetic pathways were still significantly associated with PsA, giving a total of 35 pathways for validation in the independent cohort. Using the North America case–control cohort, we replicated the association of 16 genetic pathways with PsA risk (45.7%, $p_{FDR}<0.05$; table 3).

Biological pathways can share a varying number of genes, and therefore redundancy in pathway association can occur. To filter out highly redundant results, we computed the gene overlap between the PsA-associated pathways. We found a marked gene overlap (>95% shared genes) between the 'Costimulation by the CD28 family' (n=63 genes) and 'CD28 costimulation' pathways (n=32 genes), as well as between the 'Metabolism of carbohydrates' (n=247 genes) and 'GAG metabolism' pathways (n=111 genes; online supplementary figure S2). In these two cases, we selected the pathway most significantly associated with PsA for downstream analyses (ie, 'Costimulation by the CD28 family' and 'GAG metabolism' pathways).

GAG metabolism pathway is specifically associated with PsA

In order to test for disease-specific risk, we tested the new PsA locus and pathways in the PsC and RA cohorts. We found that *B3GNT2* risk allele is significantly associated with PsA in both comparisons ($p_{[PsAvsPsC]}=0.029$, $OR_{[95\%\ CI]}=1.16$ [1.02 to 1.36]; $p_{[PsAvsRA]}=2.41e\text{-}04$, $OR_{[95\%\ CI]}=1.26$ [1.08 to 1.44]). When comparing each disease with the control cohort, we found a significant association between PsC and *B3GNT2* (p=6.11e-3, $OR_{[95\%\ CI]}=1.21$ [1.05 to 1.38]) but not with RA (p=0.07, $OR_{[95\%\ CI]}=1.11$ [0.99 to 1.24]).

In the pathway analysis we found that the GAG metabolism pathway was significantly associated with PsA when compared with PsC (p=0.018; online supplementary table S13) and RA (p=0.0018; online supplementary table S13). Subsequent testing of these two autoimmune diseases against the control cohort showed no evidence of association (p=0.71 and p=0.58 for PsC and RA pathway analyses, respectively).

Table 2 Genetic variants associated with PsA risk at the genome-wide scale

SNP*	Chr	Pos	Gene	A1	A2	OR^S	P^S	OR^N	P^N	Z^M	P^M
rs458017†	6	111696091	*REV3L*	T	C	1.85 (1.45 to 2.35)	9.29e-07	1.65 (1.37 to 1.99)	1.16e-07	−6.66	2.68e-11
rs4921483	5	158768365	*IL12B*	A	G	1.40 (1.19 to 1.65)	4.31e-05	1.43 (1.25 to 1.63)	1.34e-07	6.13	8.94e-10
rs10865331	**2**	**62551472**	***B3GNT2***	**A**	**G**	**1.39 (1.23 to 1.58)**	**3.62e-07**	**1.22 (1.10 to 1.36)**	**2.06e-04**	**5.72**	**1.10e-08**
rs74817271	5	150469973	*TNIP1*	A	G	1.38 (1.09 to 1.74)	8.10e-03	1.79 (1.47 to 2.17)	3.20e-09	5.64	1.66e-08
rs2546890	5	158759900	*IL12B*	A	G	0.81 (0.71 to 0.91)	8.77e-04	0.75 (0.68 to 0.84)	1.37e-07	5.64	1.70e-08

P values ($p^M<5.00e\text{-}08$) are shown.
*The new PsA risk loci are shown in bold. Independent (linkage disequilibrium $r^2<0.45$) non-HLA SNPs are shown.
†The SNP rs458017 is in high linkage disequilibrium with the SNP rs33980500 at the known PsA risk gene *TRAF3IP2* ($r^2=0.74$).
Z, z-statistic that summarises the magnitude and direction effects relative to the reference allele)A1, minor allele; A2, major allele; CI, confidence interval; Chr, chromosome; M, meta-analysis; N, PsA case–control cohort from North America; OR, odds ratio and 95% confidence interval (the odds ratios shown are relative to the minor allele); OR, odds ratio; Pos, base pair in build GRCh37/hg19; PsA, psoriatic arthritis; S, PsA case–control cohort from Spain; SNP, single-nucleotide polymorphism.

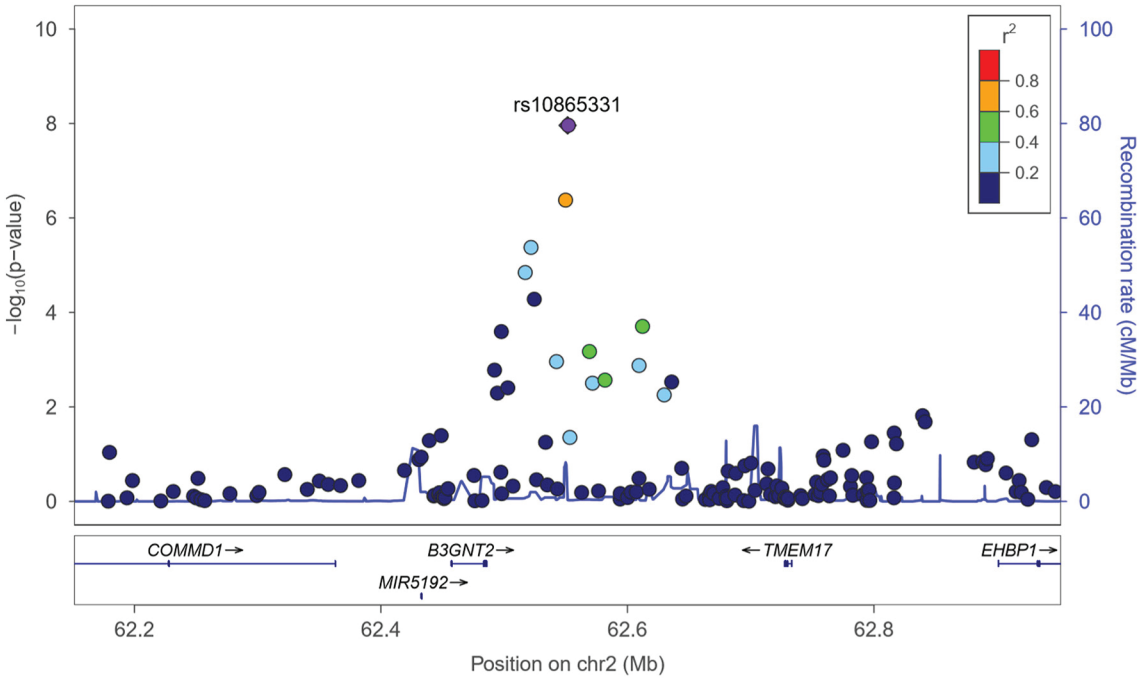


Figure 1 Regional association plot of the new genetic variant rs10865331 associated with psoriatic arthritis (PsA) risk. Each circle represents a genetic variant that is plotted according to its association with PsA risk in the negative logarithmic scale. Circles are coloured according to the linkage disequilibrium with the SNP rs10865331 (ie, violet circle). The blue line shows the recombination rate across the plotted region (data source: 1000 Genomes Project; build GRCh37/hg19). In the bottom line, the genes mapping to the PsA-associated locus are shown.

GAG metabolism is associated with PsA at the transcriptomic level

Using whole blood transcriptomic data from patients with PsA and PsC,[46] we found that 14 out of the 111 genes included in the GAG metabolism pathway were differentially expressed between the two diseases (online supplementary table S14). This difference is higher than expected by chance ($p_{Binomial} \leq 0.005$; figure 2). When comparing each disease with the control group (online supplementary table S14), the number of differentially expressed genes was also found to be significant in PsA ($p_{Binomial} \leq 0.005$), but not in PsC ($p_{Binomial} = 0.27$).

In the coexpression analysis of the GAG metabolism pathway, we detected that the pathway genes have a higher coexpression in PsA (IMC=1.43) than in PsC (IMC=1.38). Of relevance, when comparing the coexpression of the pathway between the PsA cohort and the mixed cohort of patients with PsA and PsC (IMC=0.54), the coexpression of the pathway was found to

Table 3 Genetic pathways associated with PsA risk and validated in the replication stage

Genetic pathway	Database	Gene (n)	SNPs^{S}*	P^{S}	FDR^{S}	P^{SE}	FDR^{SE}	SNPs^{N1}	P^{N}	FDR^{N}	P^{NE}	FDR^{NE}	P^{C}
Type I diabetes mellitus†‡	KEGG	44	1791	<1.00e-08	<6.50e-07	4.00e-04	7.50e-03	1436	<1.00e-08	<9.00e-08	1.32e-04	2.11e-03	<3.78e-15
JAK-STAT signalling †‡	KEGG	155	4193	8.10e-04	1.98e-02	5.00e-03	3.12e-02	2594	<1.00e-08	<9.00e-08	5.00e-04	4.00e-03	<2.15e-10
Costimulation by CD28 family‡	Reactome	63	1410	8.00e-06	4.43e-04	7.00e-04	8.00e-03	964	1.50e-06	7.80e-06	1.73e-02	4.61e-02	3.14e-10
Th1/Th2 differentiation‡	Biocarta	19	523	2.00e-05	9.16e-04	1.34e-02	8.00e-03	355	1.50e-06	7.80e-06	1.21e-02	3.87e-02	7.57e-10
Purine metabolism	KEGG	159	5360	2.90e-04	8.72e-03	4.20e-03	3.07e-02	3295	4.00e-05	9.45e-05	8.30e-03	3.32e-02	2.24e-07
G alpha signalling	Reactome	195	6271	1.84e-03	3.28e-02	1.05e-02	4.38e-02	3890	2.00e-05	5.78e-05	4.20e-03	2.24e-02	6.67e-07
CD28 costimulation	Reactome	32	786	1.40e-04	5.08e-03	–	–	488	2.95e-03	4.79e-03	–	–	6.48e-06
Extracellular matrix	Biocarta	24	743	6.00e-05	2.34e-03	–	–	439	8.16e-03	1.06e-02	–	–	7.60e-06
mTOR signalling	KEGG	52	1280	5.80e-04	1.57e-02	–	–	780	1.04e-03	1.80e-03	–	–	9.24e-06
Rac-1 cell motility signalling	Biocarta	23	751	6.50e-04	1.71e-02	–	–	459	4.83e-03	6.98e-03	–	–	4.29e-05
Glycosaminoglycan metabolism	Reactome	111	6035	1.04e-03	2.38e-02	–	–	3705	3.41e-03	5.22e-03	–	–	4.81e-05
ErbB signalling‡	KEGG	87	4099	1.70e-03	3.14e-02	–	–	2459	7.36e-03	1.01e-02	–	–	1.54e-04
Met signalling	Biocarta	37	1402	8.80e-04	2.11e-02	–	–	890	1.74e-02	1.74e-02	–	–	1.85e-04
Metabolism of carbohydrates	Reactome	247	8964	1.52e-03	3.01e-02	–	–	5469	1.14e-02	1.35e-02	–	–	2.07e-04
Interleukin-7 signal transduction	Biocarta	17	689	2.76e-03	4.21e-02	–	–	432	1.49e-02	1.55e-02	–	–	4.56e-04
Glycosaminoglycan biosynthesis of keratan sulfate	KEGG	15	651	3.35e-03	4.83e-02	–	–	428	1.32e-02	1.43e-02	–	–	4.88e-04

*Number of SNP mapping to the indicated pathway.
†Increased permutations to refine the p value (n=1.00e-08).
‡Pathways previously associated with the overall Ps risk.
SNP, single-nucleotide polymorphism.C, combined; E, exclusion of *IL12B* and/or HLA genes; FDR, false discovery rate; KEGG, Kyoto Encyclopedia of Genes and Genomes; N, PsA case–control cohort from North America; Ps, psoriasis; PsA, psoriatic arthritis; S, PsA case–control cohort from Spain.

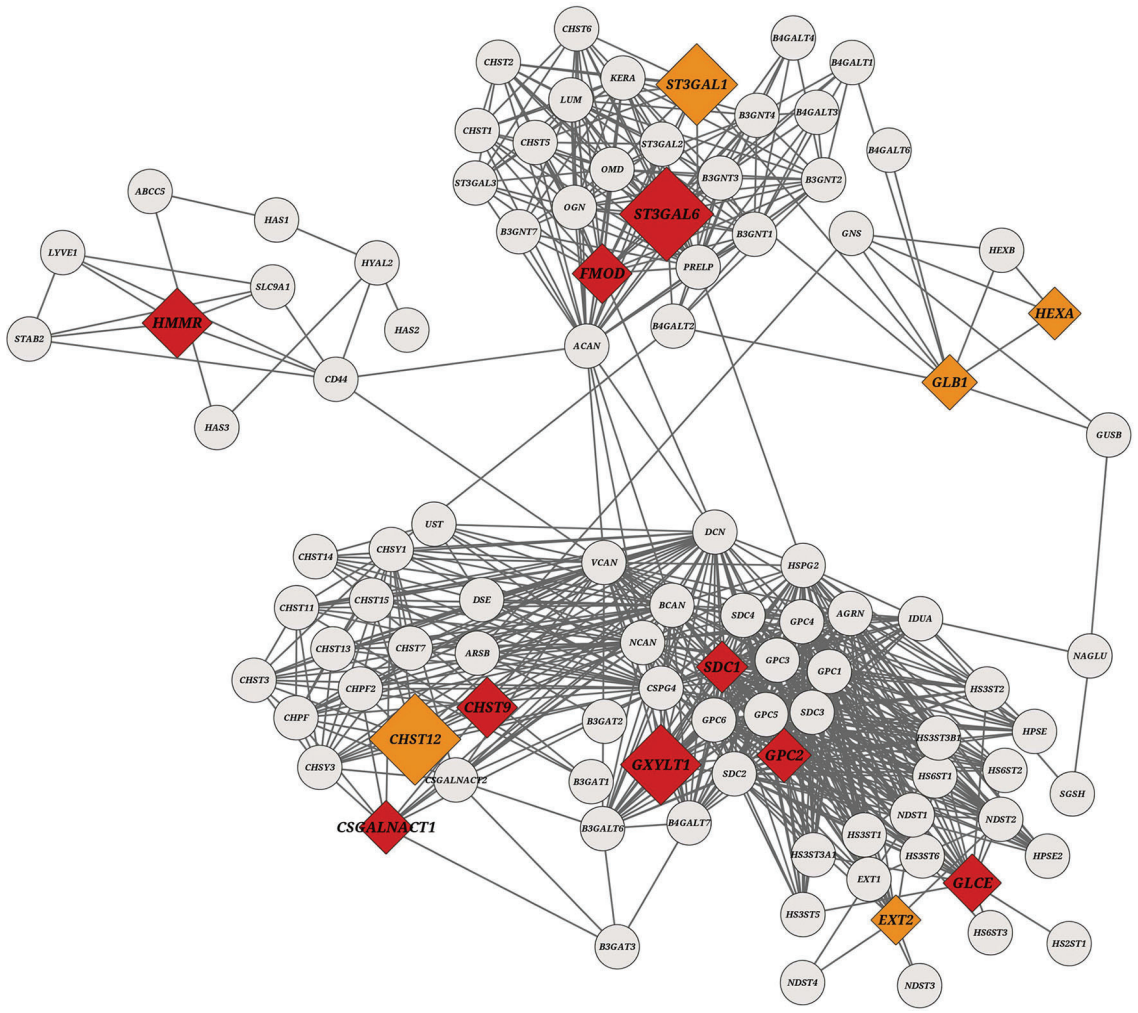


Figure 2 GAG metabolism pathway genes are differentially expressed in patients with PsA and PsC. Network representation of the GAG metabolism pathway. Genes, represented as nodes, are connected by edges according to the evidence of functional association between their encoded proteins. Differentially ($p<0.05$) and non-differentially ($p\geq0.05$) expressed genes are represented by rhombus and grey circles, respectively. Differentially expressed genes that are upregulated and downregulated in PsA are coloured in red and orange, respectively. Gene diameter is proportional to the significance of differential expression in the negative logarithmic scale. GAG, glycosaminoglycan; PsA, psoriatic arthritis.

significantly drop when both diseases are analysed together as a single disease entity (p=8.37e-10).

GAG metabolism is a new source for drug discovery in PsA

In this exploratory analysis, we identified six FDA-approved drugs that target proteins encoded by GAG pathway genes (figure 3A). The analysis of the network topology of the GAG pathway showed that *NCAN* and *VCAN* genes, both targeted by the hyaluronic acid drug (HA_d), and the *DCN* gene, targeted by the tromethamine drug (TM_d), have the highest centrality properties (figure 3B). Since a higher network centrality is indicative of a predominant regulatory role in the pathway,[47] we next evaluated the impact of HA_d and TM_d in the regulation of the GAG pathway. We found that both HA_d and TM_d significantly modulate the GAG pathway functionality ($p_{HAd}=0.0097$ and $p_{TMd}=0.017$), and therefore these two drugs could be repurposed as candidates for PsA treatment.

DISCUSSION

The identification of genetic variation that is associated with PsA and not with PsC has proven very challenging. Here, we have analysed two large PsA case–control cohorts from independent populations to identify disease-specific variation both at the single marker and at the pathway levels. Using this approach, we have identified a significant association of *B3GNT2* locus as well as 14 genetic pathways with PsA risk. From these, we have found that genetic variation at the GAG metabolism pathway is specifically associated with PsA. Investigating the GAG metabolism pathway with drug-repurposing and network analyses, we have identified potentially new PsA indications for common drugs as well as new candidate drug targets for PsA.

The SNP rs10865331 associated with PsA risk is located on chromosome 2p15 at 99.6 Kb downstream of the *B3GNT2* gene, which encodes for a transmembrane enzyme that synthesises the carbohydrate structure of polylactosamine onto glycoproteins.[48] This polymorphism maps to a genomic region enriched in promoter and enhancer histone marks from blood, which has a marked immune cell burden.[49] According to the Genotype-Tissue Expression, there is evidence of strong *cis*-regulation between this SNP and *B3GNT2* expression in whole blood (p=5.1e-13; online supplementary figure S3).[50] Previous studies in other arthritic diseases have shown that the SNP rs10865331 is associated with ankylosing spondylitis and the *B3GNT2* locus with RA.[51 52] Consistently, *B3gnt2* knockout mice have shown

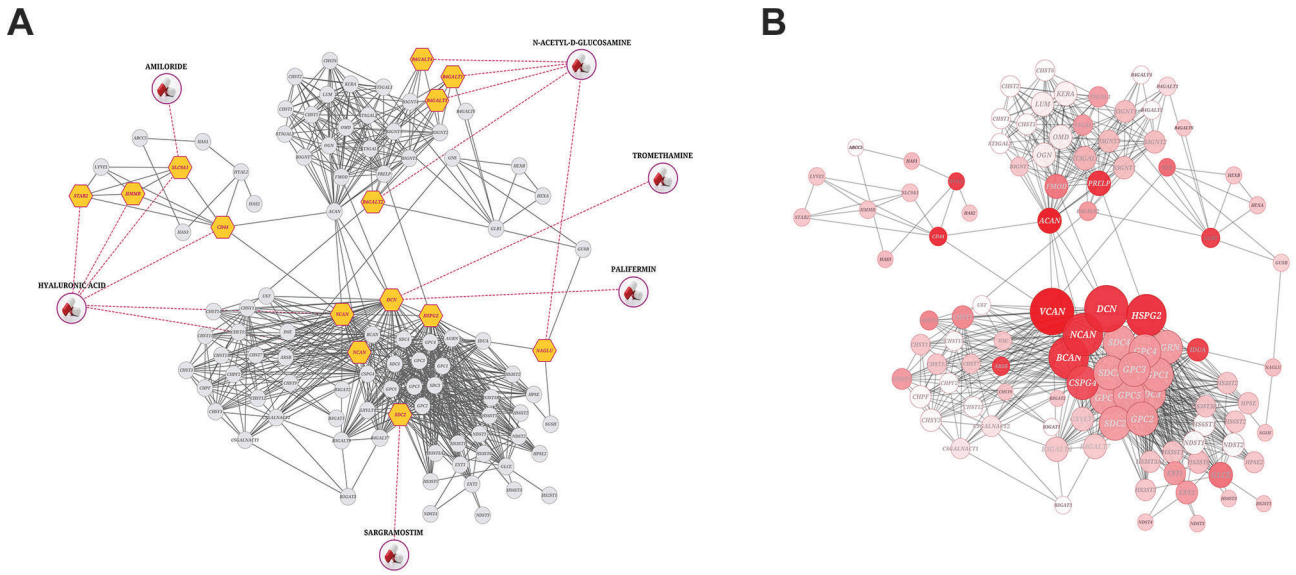


Figure 3 Drug-repurposing and network analyses in the GAG metabolism pathway. (A) Six FDA-approved drugs (amiloride, hyaluronic acid, N-acetyl-D-glucosamine, palifermin, sargramostim and tromethamine) have target genes in the GAG metabolism pathway. (B) Identification of the most central genes in the functional-based network of the GAG metabolism pathway. Gene diameter is proportional to its degree centrality value and is coloured according to its betweenness centrality value, ranging from white (lowest) to red (highest). FDA, Food and Drug Administration; GAG, glycosaminoglycan.

hyperactivation of T and B lymphocytes as well as enhanced macrophage activation.[48 53] In a GWAS meta-analysis, *B3GNT2* has also been associated with Ps risk.[11] In this previous study, however, a stratified analysis with patients affected with PsA was not performed, and consequently it remains unclear the contribution of each disease to the observed association. Here we show, for the first time, that the SNP rs10865331 at *B3GNT2* locus is associated with PsA at the genome-wide level and that the frequency of the risk allele is significantly higher in PsA than in PsC.

The identification of disease-specific genetic variation is highly useful to discover relevant pathogenic mechanisms in complex diseases.[54] In PsA, the existence of disease-specific variation is supported by the evidence of a larger familial aggregation compared with Ps.[5–8] Using the GWAS data from this study, we find that, while PsA and PsC show a significant genetic correlation (r^2=0.73, SE=0.12, p=3.76e-9), the SNP-based heritability of PsA (46%, SE=12%) is significantly higher than PsC (34%, SE=2%, p<0.05) (online supplementary material 1). In line with familial aggregation studies, our findings also support the existence of PsA-specific genetic variation. To this regard, when directly comparing the PsA and PsC cohorts, we have not only replicated the association of the *HLA-C*06:02* and *HLA-B*27* haplotypes,[38 39] but also we have identified a new association between the *HLA-A* Asp77 and PsA risk. At the pathway level, we have identified a specific association between PsA and the GAG metabolism pathway. GAGs are linear, negatively charged oligosaccharides that include hyaluronic acid (HA), chondroitin sulfate and keratan sulfate (KS).[55] Of relevance, GAGs are crucial components of proteoglycans and the major component of the cartilage, which is the main target tissue of PsA inflammatory destruction.[56 57] Our results suggest that cartilage degradation in PsA could derive from an altered GAG metabolism that is not perturbed in other arthritis like RA.

GAG metabolism has been shown to be altered in complex diseases including autoimmune diseases.[58 59] In in vitro models, uncontrolled proteolysis of aggrecan (ie, cartilage-specific proteoglycan) in response to proinflammatory cytokines promotes cartilage damage in the articular joint.[60] After aggrecan destruction, GAGs are released from the extracellular matrix (ECM) to the synovial fluid.[61] Accordingly, the levels of GAGs like HA have been found increased in patients with PsA compared with control subjects, both in serum and synovial fluid.[62 63] Previous experimental studies have demonstrated the existence of a GAG-mediated mechanism for cartilage destruction that is driven by the degradation of HA by chondrocytes and that is independent from aggrecanases.[64] Consistent with this evidence, genetic variation at the GAG metabolism pathway could diminish the cartilage-specific biosynthesis of HA, and consequently reduce its availability for both aggrecan and cartilage formation in patients with PsA.

In this study we also validated the association between 13 genetic pathways and PsA risk. From these, JAK-STAT signalling, type 1 diabetes mellitus, costimulation by CD28 family, Th1/Th2 differentiation and ErbB signalling pathways had been previously associated with Ps risk.[29] Crucial proinflammatory cytokines for the development of Ps like interleukin (IL)-12 and IL-23 rely on the JAK-STAT signalling pathway. Importantly, small molecule inhibitors of JAK proteins (eg, ruxolitinib and tofacitinib) have been recently proven successful for the treatment of the disease.[65] Our results provide additional genetic evidence supporting the functional role of this group of biological pathways in the aetiology of Ps. The eight remaining pathways (ie, G alpha signalling, purine metabolism, KS biosynthesis, extracellular matrix, mTOR signalling, IL-7 signal transduction, Rac-1 cell motility signalling and Met signalling) were found to be only significantly associated with PsA. Recent studies have shown that the mTOR signalling pathway is responsible for inducing the proliferation of a synovial T cell subpopulation (ie, Th9 cells) that enhances the immune response in PsA.[66] There is also previous evidence supporting the implication of the IL-7 signal transduction pathway in the development of PsA. In in vitro models, lymphocytes and synovial fluid fibroblasts have shown to produce IL-7 cytokine and promote the formation of osteoclasts,[67] which are the main mediators of bone matrix degradation in PsA. Additional studies on independent PsA and

PsC cohorts will be needed to confirm the PsA-specific nature of these additional pathway associations.

Current drug discovery research is shifting from targeting single genes towards the modulation of specific biological pathways.[68] Here, we show that the GAG metabolism could be a druggable pathway for PsA treatment. Our analyses suggest that FDA-approved drugs HA_d and TM_d are good candidates for repurposing for PsA, since they target central genes in the GAG metabolism network and have a significant impact on its functionality. We, like others, show the power of genetics to identify potential new drug targets and opportunities for drug repurposing in autoimmune diseases.[69] In all these studies, however, downstream validation of the in silico findings in adequate in vitro and in vivo studies is still an indispensable step. Therefore, future experimental and clinical studies will be necessary to corroborate the utility of these two new drug targets to treat PsA.

Compared with previous GWAS, the use of PsC and RA cohorts to differentiate the genetic pathways that are PsA-specific from those that are not is a distinctive strength of the present study. The pathway-based analysis methodology used here has limitations, nonetheless. One limitation is the SNP annotation to the genes within each pathway. SNPs that are located far from the genes or in other chromosomes and that could regulate gene expression through *cis*-expression Quantitative Trait Loci (eQTL) and *trans*-eQTL mechanisms were not included in the present GWPA. To our knowledge, there is yet no pathway-based method that integrates this information and that has been able to identify disease risk variation. One of the major problems for this approach is the context-dependent nature of many eQTLs. There is growing evidence that many eQTLs are cell type-dependent and also vary in relation to many contextual aspects like the level and type of stimulation.[70–72] The integration of this regulatory information is therefore still a challenge for GWPA analysis methods. With the increasing regulatory information that is currently being derived from single-cell expression studies,[73] a more profound ascertainment of the impact of SNP variation on gene expression levels will be obtained, and eventually more comprehensive GWPA methods will be developed.

In conclusion, we have identified variation at *B3GNT2* locus and 14 pathways significantly associated with the risk of PsA. From these, the GAG metabolism pathway showed a specific association with PsA when contrasted to PsC and RA. Using network and drug-repurposing analyses, we provide evidence that the GAG pathway could be a new source for drug discovery in PsA. This study represents an important step towards the characterisation of biological mechanisms that are specific for PsA and the finding of more effective drugs in PsA treatment.

Author affiliations
[1]Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, Spain
[2]Department of Experimental and Health Sciences, Universitat Pompeu Fabra, Barcelona, Spain
[3]Rheumatology Department, Hospital Clínic de Barcelona and IDIBAPS, Barcelona, Spain
[4]Rheumatology Department, Hospital Universitario Guadalajara, Guadalajara, Spain
[5]Dermatology Department, Hospital Universitari Germans Trias i Pujol, Badalona, Spain
[6]Rheumatology Department, Complejo Hospitalario Juan Canalejo, A Coruña, Spain
[7]Rheumatology Department, Hospital Parc Taulí, Sabadell, Spain
[8]Rheumatology Department, Hospital Universitario Central de Asturias, Oviedo, Spain
[9]Rheumatology Department, Hospital Virgen de la Vega, Salamanca, Spain
[10]Rheumatology Department, Hospital Monte Naranco, Oviedo, Spain
[11]Rheumatology Department, Hospital de Jerez de la Frontera, Cádiz, Spain
[12]Rheumatology Department, Instituto de Investigación Biomédica de Málaga, Hospital Regional Universitario de Málaga, Málaga, Spain
[13]Rheumatology Department, Hospital Universitario Infanta Sofía, Universidad Europea, Madrid, Spain
[14]Rheumatology Department, Hospital Universitario Gregorio Marañón, Madrid, Spain
[15]Rheumatology Department, Hospital Moisès Broggi, Barcelona, Spain
[16]Rheumatology Department, Hospital Universitario Fundación Alcorcón, Madrid, Spain
[17]Rheumatology Department, Hospital Sant Rafael, Barcelona, Spain
[18]Rheumatology Department, Hospital Universitari de Bellvitge, Barcelona, Spain
[19]Rheumatology Department, Hospital Universitario La Princesa, IIS La Princesa, Madrid, Spain
[20]Rheumatology Department, Centro de Salud Virgen de los Reyes, Sevilla, Spain
[21]Rheumatology Department, Hospital Universitario Mútua de Terrassa, Terrassa, Spain
[22]Rheumatology Department, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain
[23]Rheumatology Department, Hospital Universitario Marqués de Valdecilla, Santander, Spain
[24]Rheumatology Department, Complexo Hospitalario de Ourense, Ourense, Spain
[25]Rheumatology Department, Complejo Hospitalario Hospital Provincial de Pontevedra, Pontevedra, Spain
[26]Rheumatology Department, Hospital General Universitario de Alicante, Alicante, Spain
[27]Rheumatology Department, Hospital La Mancha Centro, Alcázar de San Juan, Spain
[28]Rheumatology Department, Hospital Dos de Maig, Barcelona, Spain
[29]Dermatology Department, Hospital Universitario Infanta Leonor, Madrid, Spain
[30]Dermatology Department, Complejo Hospitalario Universitario de A Coruña, A Coruña, Spain
[31]Dermatology Department, Hospital Universitario de Salamanca, Salamanca, Spain
[32]Dermatology Department, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain
[33]Dermatology Department, Hospital Universitario La Princesa, IIS La Princesa, Madrid, Spain
[34]Dermatology Department, Hospital General Universitario de Valencia, Valencia, Spain
[35]Dermatology Department, Hospital Universitario Fundación Alcorcón, Madrid, Spain
[36]Dermatology Department, Hospital Virgen Macarena, Sevilla, Spain
[37]Dermatology Department, Hospital Universitario 12 de Octubre, Madrid, Spain
[38]Dermatology Department, Hospital Universitario Virgen de la Victoria, Málaga, Spain
[39]Rheumatology Department, INIBIC-Hospital Universitario A Coruña, A Coruña, Spain
[40]Rheumatology Department, Hospital Clínico San Carlos, IDISSC, Madrid, Spain
[41]Instituto de Investigación Sanitaria Hospital Clínico Universitario de Santiago, Santiago de Compostela, Spain
[42]Rheumatology Department, Parc de Salut Mar Barcelona, Barcelona, Spain
[43]Rheumatology Department, Hospital Universitari Germans Trias i Pujol, Barcelona, Spain
[44]Rheumatology Department, Hospital Comarcal d'Amposta, Tarragona, Spain
[45]Banco Nacional de ADN Carlos III, University of Salamanca, Salamanca, Spain
[46]Barcelona Supercomputing Centre (BSC), Joint BSC-CRG-IRB Research Program in Computational Biology, Barcelona, Spain
[47]Institució Catalana de Recerca i Estudis Avançats (ICREA), Barcelona, Spain
[48]Life Sciences Department, Barcelona Supercomputing Centre, Barcelona, Spain
[49]HudsonAlpha Institute for Biotechnology, Huntsville, Alabama, USA

Acknowledgements We thank the patients and clinical specialists collaborating in the IMID Consortium for participation. Genotype and phenotype data from the North American PsA case–control cohort were provided by Dr JT Elder, University of Michigan, with collaborators Dr D Gladman, University of Toronto, and Dr P Rahman, Memorial University of Newfoundland, providing samples.

Contributors AA, JDC, SM and AJ conducted the study design and data interpretation. JT, CF, JAP, JG, RQ, CM, JCT-A, JJP-V, AFN, SM-F, CMG, DR, PZ, AE, JR, SC, ER, GS, CD-T, RB, AWD, JAM, PV, SAS-F, HC, JR, PdlC, EdF, EmF, LP, ED, JLS-C, JLL-E, DM, FV, EH, FB, BF-G, AG, CP-G, MA-L, AOM, VM-T, IG-A, RS, CTR, ML-C, AP, ML-L, RT, NP and SM were involved in sample collection. AA, AJ, SB-G, JMM, DT, LC, HLG, DA and RM performed genetic analyses. AA and AJ performed functional and drug-repurposing analyses. AA, JDC, SM and AJ wrote the manuscript. All authors revised the manuscript and gave final approval for its submission.

Funding This study was funded by the Spanish Ministry of Economy and Competitiveness (grant numbers: PSE-010000-2006-6 and IPT-010000-2010-36, cofunded by the European Regional Development Fund). This work was also sponsored by the 'Agència de Gestió d'Ajuts Universitaris i de Recerca' (AGAUR, FI-DGR2016, grant number: 00587), which is supported by the 'Secretaria d'Universitats i Recerca' (Economy and Knowledge Department, Generalitat de Catalunya) and cofunded by the European Social Fund. The obtention of the GWAS data from the PsA case-control cohort from North American population was supported by grants from the NIH, the Canadian Institutes of Health Research, the Krembil Foundation, the Babcock Memorial Trust, the Barbara and Neal Henschel

Charitable Foundation and the Ann Arbor Veterans Affairs Hospital. The study sponsors had no role in the collection, analysis or interpretation of the data

Competing interests None declared.

Patient consent Obtained.

Ethics approval The study was approved by Hospital Universitari Vall d'Hebron Clinical Research Ethics Committee. This study was conducted according to the principles of the Declaration of Helsinki. Protocols were reviewed and approved by the local institutional review board of each participating centre.

Provenance and peer review Not commissioned; externally peer reviewed.

REFERENCES

1. Ibrahim G, Waxman R, Helliwell PS. The prevalence of psoriatic arthritis in people with psoriasis. *Arthritis Rheum* 2009;61:1373–8.
2. Chandran V, Raychaudhuri SP. Geoepidemiology and environmental factors of psoriasis and psoriatic arthritis. *J Autoimmun* 2010;34:J314–J321.
3. Rosen CF, Mussani F, Chandran V, *et al*. Patients with psoriatic arthritis have worse quality of life than those with psoriasis alone. *Rheumatology* 2012;51:571–6.
4. Ritchlin CT, Colbert RA, Gladman DD. Psoriatic Arthritis. *N Engl J Med* 2017;376:957–70.
5. Bhalerao J, Bowcock AM. The genetics of psoriasis: a complex disorder of the skin and immune system. *Hum Mol Genet* 1998;7:1537–45.
6. Chandran V, Schentag CT, Brockbank JE, *et al*. Familial aggregation of psoriatic arthritis. *Ann Rheum Dis* 2009;68:664–7.
7. Elder JT, Nair RP, Guo SW, *et al*. The genetics of psoriasis. *Arch Dermatol* 1994;130:216–24.
8. Gladman DD, Farewell VT, Pellett F, *et al*. HLA is a candidate region for psoriatic arthritis. evidence for excessive HLA sharing in sibling pairs. *Hum Immunol* 2003;64:887–9.
9. Genetic Analysis of Psoriasis Consortium & the Wellcome Trust Case Control Consortium 2, Strange A, Capon F, *et al*. A genome-wide association study identifies new psoriasis susceptibility loci and an interaction between HLA-C and ERAP1. *Nat Genet* 2010;42:985–90.
10. Tsoi LC, Spain SL, Ellinghaus E, *et al*. Enhanced meta-analysis and replication studies identify five new psoriasis susceptibility loci. *Nat Commun* 2015;6:7001.
11. Tsoi LC, Spain SL, Knight J, *et al*. Identification of 15 new psoriasis susceptibility loci highlights the role of innate immunity. *Nat Genet* 2012;44:1341–8.
12. Yin X, Low HQ, Wang L, *et al*. Genome-wide meta-analysis identifies multiple novel associations and ethnic heterogeneity of psoriasis susceptibility. *Nat Commun* 2015;6:6916.
13. Yin X, Wineinger NE, Cheng H, *et al*. Common variants explain a large fraction of the variability in the liability to psoriasis in a Han Chinese population. *BMC Genomics* 2014;15:87.
14. Zuo X, Sun L, Yin X, *et al*. Whole-exome SNP array identifies 15 new susceptibility loci for psoriasis. *Nat Commun* 2015;6:6793.
15. Stuart PE, Nair RP, Tsoi LC, *et al*. Genome-wide association analysis of psoriatic arthritis and cutaneous psoriasis reveals differences in their genetic architecture. *Am J Hum Genet* 2015;97:816–36.
16. Bowes J, Budu-Aggrey A, Huffmeier U, *et al*. Dense genotyping of immune-related susceptibility loci reveals new insights into the genetics of psoriatic arthritis. *Nat Commun* 2015;6:6046.
17. Liu Y, Helms C, Liao W, *et al*. A genome-wide association study of psoriasis and psoriatic arthritis identifies new disease loci. *PLoS Genet* 2008;4:e1000041.
18. Hüffmeier U, Uebe S, Ekici AB, *et al*. Common variants at TRAF3IP2 are associated with susceptibility to psoriatic arthritis and psoriasis. *Nat Genet* 2010;42:996–9.
19. Ellinghaus E, Stuart PE, Ellinghaus D, *et al*. Genome-wide meta-analysis of psoriatic arthritis identifies susceptibility locus at REL. *J Invest Dermatol* 2012;132:1133–40.
20. Julià A, Pinto JA, Gratacós J, *et al*. A deletion at ADAMTS9-MAGI1 locus is associated with psoriatic arthritis risk. *Ann Rheum Dis* 2015;74:1875–81.
21. Apel M, Uebe S, Bowes J, *et al*. Variants in RUNX3 contribute to susceptibility to psoriatic arthritis, exhibiting further common ground with ankylosing spondylitis. *Arthritis Rheum* 2013;65:1224–31.
22. Bowes J, Loehr S, Budu-Aggrey A, *et al*. PTPN22 is associated with susceptibility to psoriatic arthritis but not psoriasis: evidence for a further PsA-specific risk locus. *Ann Rheum Dis* 2015;74:1882–5.
23. Chen H, Poon A, Yeung C, *et al*. A genetic risk score combining ten psoriasis risk loci improves disease prediction. *PLoS One* 2011;6:e19454.
24. Manolio TA, Collins FS, Cox NJ, *et al*. Finding the missing heritability of complex diseases. *Nature* 2009;461:747–53.
25. Du Y, Xie J, Chang W, *et al*. Genome-wide association studies: inherent limitations and future challenges. *Front Med* 2012;6:444–50.
26. Wang K, Li M, Hakonarson H. Analysing biological pathways in genome-wide association studies. *Nat Rev Genet* 2010;11:843–54.
27. de Leeuw CA, Neale BM, Heskes T, *et al*. The statistical properties of gene-set analysis. *Nat Rev Genet* 2016;17:353–64.
28. Wu MC, Kraft P, Epstein MP, *et al*. Powerful SNP-set analysis for case-control genome-wide association studies. *Am J Hum Genet* 2010;86:929–42.
29. Aterido A, Julià A, Ferrándiz C, *et al*. Genome-wide pathway analysis identifies genetic pathways associated with psoriasis. *J Invest Dermatol* 2016;136:593–602.
30. Aterido A, Julià A, Carreira P, *et al*. Genome-wide pathway analysis identifies VEGF pathway association with oral ulceration in systemic lupus erythematosus. *Arthritis Res Ther* 2017;19:138.
31. Julià A, Tortosa R, Hernanz JM, *et al*. Risk variants for psoriasis vulgaris in a large case-control collection and association with clinical subphenotypes. *Hum Mol Genet* 2012;21:4549–57.
32. Taylor W, Gladman D, Helliwell P, *et al*. Classification criteria for psoriatic arthritis: development of new criteria from a large international study. *Arthritis Rheum* 2006;54:2665–73.
33. Howie B, Marchini J, Stephens M. Genotype imputation with thousands of genomes. *G3* 2011;1:457–70.
34. 1000 Genomes Project Consortium, Abecasis GR, Auton A, *et al*. An integrated map of genetic variation from 1,092 human genomes. *Nature* 2012;491:56–65.
35. Julià A, Blanco F, Fernández-Gutierrez B, *et al*. Identification of IRX1 as a risk locus for rheumatoid factor positivity in rheumatoid arthritis in a genome-wide association study. *Arthritis Rheumatol* 2016;68:1384–91.
36. Jia X, Han B, Onengut-Gumuscu S, *et al*. Imputing amino acid polymorphisms in human leukocyte antigens. *PLoS One* 2013;8:e64683.
37. Marchini J, Howie B, Myers S, *et al*. A new multipoint method for genome-wide association studies by imputation of genotypes. *Nat Genet* 2007;39:906–13.
38. Bowes J, Ashcroft J, Dand N, *et al*. Cross-phenotype association mapping of the MHC identifies genetic variants that differentiate psoriatic arthritis from psoriasis. *Ann Rheum Dis* 2017;76:1774–9.
39. Okada Y, Han B, Tsoi LC, *et al*. Fine mapping major histocompatibility complex associations in psoriasis and its clinical subtypes. *Am J Hum Genet* 2014;95:162–72.
40. Willer CJ, Li Y, Abecasis GR. METAL: fast and efficient meta-analysis of genomewide association scans. *Bioinformatics* 2010;26:2190–1.
41. Ramanan VK, Shen L, Moore JH, *et al*. Pathway analysis of genomic data: concepts, methods, and prospects for future development. *Trends Genet* 2012;28:323–32.
42. Edwards YJ, Beecham GW, Scott WK, *et al*. Identifying consensus disease pathways in Parkinson's disease using an integrative systems biology approach. *PLoS One* 2011;6:e16917.
43. Chen D, Enroth S, Ivansson E, *et al*. Pathway analysis of cervical cancer genome-wide association study highlights the MHC region and pathways involved in response to infection. *Hum Mol Genet* 2014;23:6047–60.
44. Barrett T, Wilhite SE, Ledoux P, *et al*. NCBI GEO: archive for functional genomics data sets--update. *Nucleic Acids Res* 2013;41(Database issue):D991–D995.
45. Langfelder P, Horvath S. WGCNA: an R package for weighted correlation network analysis. *BMC Bioinformatics* 2008;9:559.
46. Pollock RA, Abji F, Liang K, *et al*. Gene expression differences between psoriasis patients with and without inflammatory arthritis. *J Invest Dermatol* 2015;135:620–3.
47. Hahn MW, Kern AD. Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks. *Mol Biol Evol* 2005;22:803–6.
48. Togayachi A, Kozono Y, Ishida H, *et al*. Polylactosamine on glycoproteins influences basal levels of lymphocyte and macrophage activation. *Proc Natl Acad Sci U S A* 2007;104:15829–34.
49. Ward LD, Kellis M. HaploReg: a resource for exploring chromatin states, conservation, and regulatory motif alterations within sets of genetically linked variants. *Nucleic Acids Res* 2012;40(Database issue):D930–D934.
50. GTEx Consortium. The Genotype-Tissue Expression (GTEx) project. *Nat Genet* 2013;45:580–5.
51. Okada Y, Terao C, Ikari K, *et al*. Meta-analysis identifies nine new loci associated with rheumatoid arthritis in the Japanese population. *Nat Genet* 2012;44:511–6.
52. Australo-Anglo-American Spondyloarthritis Consortium (TASC), Reveille JD, Sims AM, *et al*. Genome-wide association study of ankylosing spondylitis identifies non-MHC susceptibility loci. *Nat Genet* 2010;42:123–7.
53. Togayachi A, Kozono Y, Kuno A, *et al*. Beta3GnT2 (B3GNT2), a major polylactosamine synthase: analysis of B3GNT2-deficient mice. *Methods Enzymol* 2010;479:185–204.
54. Wang K, Li M, Bucan M. Pathway-based approaches for analysis of genomewide association studies. *Am J Hum Genet* 2007;81:1278–83.
55. Gulati K, Poluri KM. Mechanistic and therapeutic overview of glycosaminoglycans: the unsung heroes of biomolecular signaling. *Glycoconj J* 2016;33:1–17.
56. Schett G, Coates LC, Ash ZR, *et al*. Structural damage in rheumatoid arthritis, psoriatic arthritis, and ankylosing spondylitis: traditional views, novel insights gained from TNF blockade, and concepts for the future. *Arthritis Res Ther* 2011;13(Suppl 1):S4.
57. Caterson B, Flannery CR, Hughes CE, *et al*. Mechanisms involved in cartilage proteoglycan catabolism. *Matrix Biol* 2000;19:333–44.
58. Afratis N, Gialeli C, Nikitovic D, *et al*. Glycosaminoglycans: key players in cancer cell biology and treatment. *Febs J* 2012;279:1177–97.
59. Hansen C, Otto E, Kuhlemann K, *et al*. Glycosaminoglycans in autoimmunity.. *Clin Exp Rheumatol* 1996;14(Suppl 15):S59–67.
60. Arner EC, Hughes CE, Decicco CP, *et al*. Cytokine-induced cartilage proteoglycan degradation is mediated by aggrecanase. *Osteoarthritis Cartilage* 1998;6:214–28.

61. van den Boom R, van der Harst MR, Brommer H, *et al*. Relationship between synovial fluid levels of glycosaminoglycans, hydroxyproline and general MMP activity and the presence and severity of articular cartilage change on the proximal articular surface of P1. *Equine Vet J* 2005;37:19–25.
62. Patterson AM, Cartwright A, David G, *et al*. Differential expression of syndecans and glypicans in chronically inflamed synovium. *Ann Rheum Dis* 2008;67:592–601.
63. Elkayam O, Yaron I, Shirazi I, *et al*. Serum levels of hyaluronic acid in patients with psoriatic arthritis. *Clin Rheumatol* 2000;19:455–7.
64. Sugimoto K, Iizawa T, Harada H, *et al*. Cartilage degradation independent of MMP/aggrecanases. *Osteoarthritis Cartilage* 2004;12:1006–14.
65. Baker KF, Isaacs JD. Novel therapies for immune-mediated inflammatory diseases: What can we learn from their use in rheumatoid arthritis, spondyloarthritis, systemic lupus erythematosus, psoriasis, Crohn's disease and ulcerative colitis? *Ann Rheum Dis* 2018;77:175–87.
66. Kundu-Raychaudhuri S, Abria C, Raychaudhuri SP. IL-9, a local growth factor for synovial T cells in inflammatory arthritis. *Cytokine* 2016;79:45–51.
67. Colucci S, Brunetti G, Cantatore FP, *et al*. Lymphocytes and synovial fluid fibroblasts support osteoclastogenesis through RANKL, TNFalpha, and IL-7 in an in vitro model derived from human psoriatic arthritis. *J Pathol* 2007;212:47–55.
68. Gibbs JB. Mechanism-based target identification and drug discovery in cancer research. *Science* 2000;287:1969–73.
69. Okada Y, Wu D, Trynka G, *et al*. Genetics of rheumatoid arthritis contributes to biology and drug discovery. *Nature* 2014;506:376–81.
70. Francesconi M, Lehner B. The effects of genetic variation on gene expression dynamics during development. *Nature* 2014;505:208–11.
71. van der Wijst MGP, Brugge H, de Vries DH, *et al*. Single-cell RNA sequencing identifies celltype-specific cis-eQTLs and co-expression QTLs. *Nat Genet* 2018;50:493–7.
72. Fairfax BP, Humburg P, Makino S, *et al*. Innate immune activity conditions the effect of regulatory variants upon monocyte gene expression. *Science* 2014;343:1246949.
73. Bernstein BE, Stamatoyannopoulos JA, Costello JF, *et al*. The NIH Roadmap Epigenomics Mapping Consortium. *Nat Biotechnol* 2010;28:1045–8.

2) Aterido, Adrià, Palau, N., Domènech, E., Nos Mateu, P., Gutiérrez, A., Gomollón, F., … Codó, L., Gelpí, J. L. …, Julià, A. (2019). Genetic association between CD96 locus and immunogenicity to anti-TNF therapy in Crohn's disease. The Pharmacogenomics Journal. https://doi.org/10.1038/s41397-019-0090-4

The Pharmacogenomics Journal
https://doi.org/10.1038/s41397-019-0090-4

ARTICLE

Genetic association between *CD96* locus and immunogenicity to anti-TNF therapy in Crohn's disease

Adrià Aterido[1,2] · Núria Palau[1] · Eugeni Domènech[3,4] · Pilar Nos Mateu[4,5] · Ana Gutiérrez[4,6] · Fernando Gomollón[4,7] · Juan L. Mendoza[8] · Esther Garcia-Planella[9] · Manuel Barreiro-de Acosta[10] · Fernando Muñoz[11] · Maribel Vera[12] · Cristina Saro[13] · Maria Esteve[4,14] · Montserrat Andreu[15] · María Chaparro[4,16] · Julián Panés[4,17] · Valle García-Sánchez[18] · María López-Lasanta[1] · Andrea Pluma[1] · Laia Codó[19] · Andrés García-Montero[20] · Josep Manyé[3] · Javier P. Gisbert[4,16] · Sara Marsal[1] · Antonio Julià[1]

Received: 15 June 2018 / Revised: 28 March 2019 / Accepted: 2 April 2019

Abstract

The production of antibodies to anti-tumor necrosis factor alpha (TNF) agents is one of the main causes of treatment failure in Crohn's disease (CD). To date, however, the contribution of genetics to anti-TNF immunogenicity in CD is still unknown. The objective of the present study was to identify genetic variation associated with anti-TNF immunogenicity in CD. We performed a two-stage genome-wide association study in a cohort of 96 and 123 adalimumab-treated patients, respectively. In the discovery stage, we identified a genome-wide significant association between the *CD96* locus and the production of antibodies to anti-TNF treatment ($P = 1.88e{-}09$). This association was validated in the replication stage ($P < 0.05$). The risk allele for anti-TNF immunogenicity was found to be also associated with a lack of response to anti-TNF therapy ($P = 0.019$). These findings represent an important step toward the understanding of the immunogenicity-based mechanisms that underlie anti-TNF response in CD.

Introduction

Crohn's disease (CD) is an immune-mediated gastrointestinal disorder that affects ~0.5% of the worldwide population [1]. This chronic disease has a multifactorial etiology where genetic and environmental factors contribute to trigger inflammatory processes in the intestinal mucosa [2]. To date, there is no curative medical treatment for CD and therefore current therapeutic strategies are aimed at inducing remission and minimizing adverse reactions to improve the patients' quality of life.

There is compelling evidence that the proinflammatory cytokine tumor necrosis factor alpha (TNF) plays a key role in the pathogenesis of CD [3]. In the last years, the introduction of anti-TNF drugs has greatly improved the disease course and quality of life in many patients with severe and moderate CD [4]. However, up to 50% of the anti-TNF-treated patients experience a loss of clinical response over time that will require either dose escalation or treatment discontinuation [5].

One of the main causes of anti-TNF treatment failure is the production of antidrug antibodies [6]. These antibodies can neutralize the biological activity of anti-TNF agents and increase their clearance [7]. Accordingly, the presence of antidrug antibodies has been associated with low clinical response in CD [8]. Adalimumab is one of the most common anti-TNF agents used to treat CD [9]. Despite being a fully humanized monoclonal antibody [10], it has shown to be immunogenic. Recently, multiple studies have demonstrated the presence of anti-adalimumab antibodies (AAAs), although there is substantial heterogeneity on the percentage of AAA-positive patients, ranging from 9 to 35% adalimumab-treated patients [8, 11–13]. This heterogeneity could be due to drug properties but also to host-related

These authors contributed equally: Adrià Aterido, Núria Palau

Supplementary information The online version of this article (https://doi.org/10.1038/s41397-019-0090-4) contains supplementary material, which is available to authorized users.

✉ Javier P. Gisbert
javier.p.gisbert@gmail.com

✉ Sara Marsal
sara.marsal@vhir.org

Extended author information available on the last page of the article.

Published online: 02 May 2019

factors like genetic variation [14]. To date, however, the contribution of genetics to anti-TNF immunogenicity in CD is still unknown.

The association between genetic variation and the production of antibodies to anti-TNF drugs has been yet investigated in CD. So far, only a few candidate studies in rheumatoid arthritis (RA) have tested genetic variants with this objective. *IGHG1* immunoglobulin heavy chain-coding gene was tested for association with antibodies to infliximab, but no association was found [15]. In adalimumab-treated RA patients, the production of AAA has been associated with genetic variation at *IL-10* gene, a key participant in B-cell differentiation [16]. Although this genetic association has not been yet tested for replication, it suggests the existence of a genetic basis for anti-TNF immunogenicity in autoimmune diseases.

In order to identify new genetic variation associated with the production of AAAs, we have performed a genome-wide association study (GWAS) for immunogenicity to anti-TNF therapy in CD. For this analysis, we have used a discovery cohort of 96 CD patients and an independent validation cohort of 123 CD patients who had received anti-TNF therapy. Furthermore, we have investigated whether this genetic variation is also associated with the clinical response to anti-TNF therapy. The results of this study provide new insights into the genetic basis of anti-TNF immunogenicity in CD.

Materials and methods

Patients and samples

Discovery cohort

A total of 96 CD patients were recruited for the discovery stage of present study. All patients were collected between July 2007 and June 2012 from the outpatient's clinics of the gastroenterology departments from 15 Spanish University Hospitals belonging to the Immune-Mediated Inflammatory Disease Consortium (IMIDC). The IMIDC is a Spanish network of researchers investigating the genomic basis of immune-mediated inflammatory diseases [17]. All patients were diagnosed as having CD according to the Lennard-Jones diagnostic criteria [18], were Caucasian European, and with the four grandparents born in Spain.

Replication cohort

An independent cohort of 123 CD patients was used to validate the significant genetic associations identified in the discovery stage. As described for the discovery cohort, the patient collection was also conducted by the IMIDC between July 2007 and June 2012, and the disease diagnosis was performed according to the Lennard-Jones diagnostic criteria. All patients included in the replication cohort were also Caucasian European and with all four grandparents born in Spain.

Patient treatment

The CD patients included in the present study had received adalimumab therapy. The protocol used for the adalimumab treatment included an initial dose of 160 mg at baseline followed by 80 mg at week 2, as well as a maintenance dose of 40 mg/week. The treatment protocol allowed concomitant treatment with corticosteroids. Only the patients treated with adalimumab to induce or maintain remission in the inflamed gastrointestinal tract were selected for the GWAS. Consequently, those patients who received adalimumab to promote the fistula closure or to follow a postoperative prophylaxis therapy were not included. The patients who experienced delayed hypersensitivity, anaphylactic, or infusion reactions were also excluded from the analysis. After this stringent clinical criteria filtering, a total of 62 and 88 CD patients were used for the discovery and replication stages, respectively.

Patient characteristics before treatment initiation

All patients from both the discovery and replication cohorts were naive to anti-TNF therapy at the moment of treatment initiation. In addition, to increase the homogeneity of the cohorts, only patients with at least 3 years of follow-up since diagnosis were included in the present study.

Identification of AAA-positive CD patients

In order to identify those CD patients who are positive for the production of AAA, we first assessed the AAA concentration in each CD patient. For this objective, we measured the AAA concentration in plasma samples using a commercial enzyme-linked immunosorbent assay (ELISA, Promonitor® Anti-Adalimumab, Progenika Biopharma S.A., Spain) and following the instructions provided by the manufacturer (Progenika Biopharma, Spain). Importantly, to increase the sensitivity of the test and avoid the occurrence of false-negative results, we conducted a previous acidification of the samples to dissociate potential adalimumab–AAA complexes, as previously described [19]. A total of 20 μL of serum was mixed with 100 mL of 300 mM acetic acid to a final pH of 3.0. After incubation at room temperature during 15 min, samples were neutralized by adding 31 μL of 1 M Tris base solution. The samples were then incubated during 5 min with gentle shaking and, subsequently, 49 μL of the kit dilution buffer were added to

obtain a final dilution factor 1:10. All samples were analyzed in duplicate for AAA using the Promonitor kits with acid pre-treatment. The CD patients were finally reported as AAA positive if the concentration of AAA was higher than the cut-point value of 10 AU/mL established by the manufacturer (Supplementary Table S1).

DNA extraction and genome-wide genotyping

Whole blood samples (5 mL) were collected from all CD patients of the discovery cohort. Genomic DNA was then isolated using the Chemagic Magnetic Separation Module I (PerkinElmer, USA).

Genome-wide genotyping of the 62 individuals from the discovery group was performed using Illumina Quad610 Beadchip array (Illumina, USA) at HudsonAlpha Institute for Biotechnology (Huntsville, Alabama, USA). This genotyping array scans 618,150 polymorphisms, including 598,258 Single Nucleotide Polymorphisms (SNPs) and 19,892 Copy Number Variants (CNV) probes. The genotype calling and the quality control (QC) analysis were performed using the GenomeStudio (v2011.1, Illumina, USA) and PLINK softwares, respectively [20]. From the 582,539 autosomal SNPs that were selected for the QC analysis, we excluded those SNPs showing >5% of missing data (1.61% SNPs) and a minor allele frequency (MAF) <0.05 (5.62%). Using an additional cohort of 1454 healthy controls from the same population [21], we tested the deviation of the SNPs from the Hardy–Weinberg equilibrium (HWE). Those SNPs that were not in HWE (0.03% SNPs, $P < 1e{-}4$) were subsequently removed. After the QC analysis, a final data set of 540,221 SNPs were available for the association analysis.

Population stratification (i.e., allele frequency differences between patients due to systemic ancestry differences) is one of the most important confounding factors that can yield biased results in large-scale GWAS if not properly addressed [22, 23]. To evaluate the presence of stratification in the CD discovery cohort, we used the additional cohort of 1454 controls and the principal component analysis implemented in EIGENSOFT (v4.2) software [24]. Using the first 10 Principal Components (PCs) of variation over 10 iterations, no samples showing an outlier genetic background were detected (Supplementary Fig. S1). After population stratification analysis, a total 62 CD patients remain available for the association analysis.

The genotyping of the associated SNP rs9828223 in the validation group of CD patients was performed using the Taqman Real-Time PCR genotyping platform (Thermo Fisher Scientific, USA). The TaqMan assay used for the SNP genotyping was C__29625246_10 and the thermal cycle conditions were as follows: 50 °C for 2 min and 95 °C for 10 min, followed by 40 cycles of 92 °C for 15 s and 60 °C for 1 min. All PCR and end point fluorescent readings were performed using an ABI PRISM® 7900 HT detection system (Thermo Fisher Scientific, USA).

Treatment response definition

The clinical response to adalimumab treatment was assessed using the Harvey–Bradshaw index (HBI) at 4 weeks after the first infusion of adalimumab in CD patients [25]. Based on the HBI after adalimumab therapy, remission was defined as a HBI equal or lower than four at the end of the induction period. Partial response was defined as a reduction of more than three points on the HBI. Non-response was defined as a decrease of at least three points on the HBI. Accordingly, patients with CD that were in remission or showed a partial response were combined into a single responder group, whereas patients who failed to achieve the previously described decrease on the HBI were defined as non-responders. Those patients whose treatment response was not determined were excluded from the association analysis between genetic variation and the clinical response to adalimumab therapy ($N = 16$ CD patients).

Statistical analysis

In order to compare the distribution of each clinical and epidemiological variable between the discovery and replication groups, between AAA-positive and AAA-negative patients, as well as between clinical responders and non-responders, we used the Student's *t*-test and Fisher's exact test for quantitative and qualitative variables, respectively (Table 1). According to reference GWAS [26, 27], the Cochran–Armitage trend test with one degree of freedom was used to perform the genome-wide association analysis between genetic variation and adalimumab immunogenicity under the additive genetic model [28]. The odds ratios (OR) and their 95% confidence intervals (CIs) were computed using the logistic regression model. In order to obtain the global statistical significance of the validated associations, we combined the *P*-values resulting from the discovery and replication stages using the Fisher's method [29].

The statistical association analysis between variation at *CD96* locus and clinical response to adalimumab therapy was performed using the χ^2 statistical test in R statistical software [30], and following a retrospective study design. In order to increase the statistical power of the analysis, we combined the discovery and replication groups of patients. Combining both patient cohorts, data on clinical response to adalimumab was available for a total of 134 CD patients, from which 121 (81%) had a positive response and 13 (9%) were non-responders.

Table 1 Main epidemiological and clinical features of Crohn's disease patients

Characteristics[a]	Discovery group	Replication group	*P*-value
Individuals (*N*)	62	88	–
Genetic population	Caucasian European	Caucasian European	–
Males	27/35 (43.54)	44/44 (50.00)	0.51
Age	39.02 ± 12.22	37.12 ± 12.54	0.33
Age disease onset	29.24 ± 12.07	30.27 ± 12.52	0.77
Body mass index	23.74 ± 3.49	23.82 ± 3.78	0.25
Smoking			
Current smoker	20/36 (35.71)	29/56 (34.11)	0.99
Ex-smoker	14/42 (25.00)	22/63 (25.88)	1.00
Never smoker	22/34 (39.29)	33/52 (38.83)	1.00
Disease location			
Ileal (L1)	9/35 (20.45)	19/36 (34.55)	0.18
Colonic (L2)	11/33 (25.00)	9/46 (16.36)	0.32
Ileocolonic (L3)	24/20 (54.55)	27/28 (49.09)	0.69
Upper gastrointestinal tract (L4)	9/34 (20.93)	8/41 (16.33)	0.60
Disease behavior			
Inflammatory	33/22 (60.00)	46/42 (52.27)	0.39
Stenosing	22/34 (40.00)	25/63 (28.41)	0.20
Fistulizing	19/39 (32.76)	22/66 (25.00)	0.35
Extraintestinal manifestations	19/36 (34.54)	20/68 (22.73)	0.13

The comparison between the discovery and replication cohorts was conducted using the Fisher's exact test for categoric variables and Student's *t*-test for quantitative variables

M mean, *N* sample size, *SD* standard deviation

[a]For categoric variables, we show the number of patients displaying and non-displaying the indicated characteristic as well as the percentage of positive patients for this variable. For quantitative variables, we show the mean (i.e., years for ages and kg/m^2 for body mass index) together with the standard deviation

Results

Phenotypic characterization of the study cohorts

The main epidemiological and clinical characteristics of the discovery and replication cohorts of CD patients treated with anti-TNF therapy are shown in Table 1. No significant differences in the distribution of the different epidemiological and clinical variables were found between the discovery and replication cohorts (Table 1).

Genetic variation associated with anti-TNF immunogenicity in the discovery stage

In the discovery stage, we found that the 14.5% of the 62 CD patients treated with adalimumab were positive for AAA production (Supplementary Table S1). The clinical characterization of both the AAA-positive and AAA-negative CD patients is shown in Table 2. The body mass index was found to be significantly higher in AAA-positive compared with AAA-negative CD patients ($P = 0.019$).

When analyzing the association between genetic variation and adalimumab immunogenicity, we identified a single genome-wide significant association between genetic variation at *CD96* locus ($P = 1.88e{-}09$; OR = 20.2; 95% CI, 5.57–73.27; MAF = 13.1%) and the production of AAA in CD (Fig. 1). No association was detected between the previously reported variation at *IL-10* and adalimumab immunogenicity. The complete list of markers showing nominal significance is shown in the Supplementary Table S2.

Replication of the association between the *CD96* locus and anti-TNF immunogenicity

In the replication stage, we detected that 3.4% of the 88 CD patients treated with adalimumab were positive for AAA production (Supplementary Table S1). No significant differences in the distribution of the clinical variables were found between AAA-positive and AAA-negative CD patients (Table 2). In the validation analysis, we replicated the association between variation at *CD96* locus and adalimumab immunogenicity ($P = 0.044$; OR = 1.16; 95% CI, 1.09–1.23; MAF = 10.0%; $P_{combined} = 1.83e{-}09$, Table 3).

Table 2 Clinical characterization of Crohn's disease patients according to their capacity to produce antidrug antibodies

Clinical characteristics[a]	AAA-positive discovery ($N = 9$)	AAA-negative discovery ($N = 53$)	*P*-value discovery	AAA-positive replication ($N = 3$)	AAA-negative replication ($N = 85$)	*P*-value replication
Males	4/5 (44.44)	23/30 (43.39)	1.00	2/1 (66.67)	42/43 (49.41)	1.00
Age	41.49 ± 18.08	38.36 ± 11.12	0.48	29.06 ± 5.33	37.29 ± 12.59	0.26
Age disease onset	31.16 ± 18.54	28.92 ± 10.83	0.61	22.33 ± 5.05	30.64 ± 12.57	0.25
Body mass index	25.96 ± 2.75	23.37 ± 3.01	0.019	24.69 ± 2.85	23.38 ± 3.99	0.58
Smoking						
Current smoker	4/5 (44.44)	16/31 (34.04)	0.71	2/1 (66.67)	27/55 (32.93)	0.26
Ex-smoker	3/6 (50.00)	11/36 (23.40)	0.68	0/3 (0.00)	22/60 (26.83)	0.57
Never smoker	2/7 (22.22)	20/27 (42.55)	0.46	0/3 (0.00)	33/49 (40.24)	0.28
Disease location						
Ileal (L1)	1/5 (16.67)	8/30 (21.05)	1.00	1/2 (33.33)	18/34 (34.62)	1.00
Colonic (L2)	1/5 (16.67)	10/28 (26.32)	0.98	0/3 (0.0)	9/43 (17.31)	1.00
Ileocolonic (L3)	4/2 (66.67)	20/18 (52.63)	0.67	2/1 (66.67)	25/27 (48.08)	0.61
Upper gastrointestinal tract (L4)	2/5 (40.00)	7/29 (25.93)	0.62	1/2 (33.33)	7/39 (15.22)	0.42
Disease behavior						
Inflammatory	5/3 (62.5)	28/19 (59.57)	1.00	3/0 (100.00)	43/42 (50.59)	0.24
Stenosing	5/4 (55.56)	17/29 (36.96)	0.46	1/2 (33.33)	24/61 (28.24)	1.00
Fistulizing	2/7 (22.22)	17/32 (34.69)	0.71	0/3 (0.00)	22/63 (25.88)	0.57
Extraintestinal manifestations	3/5 (37.5)	16/31 (34.04)	1.00	0/3 (0.0)	20/65 (23.52)	1.00

The comparison between AAA-positive and AAA-negative CD patients was conducted using the Fisher's exact test for categoric variables and Student's *t*-test for quantitative variables

AAA production of antidrug antibodies, *M* mean, *N* sample size, *SD* standard deviation

[a]For categoric variables, we show the number of patients displaying and non-displaying the indicated characteristic as well as the percentage of positive patients for this variable. For quantitative variables, we show the mean (i.e., years for ages and kg/m^2 for body mass index) together with the standard deviation

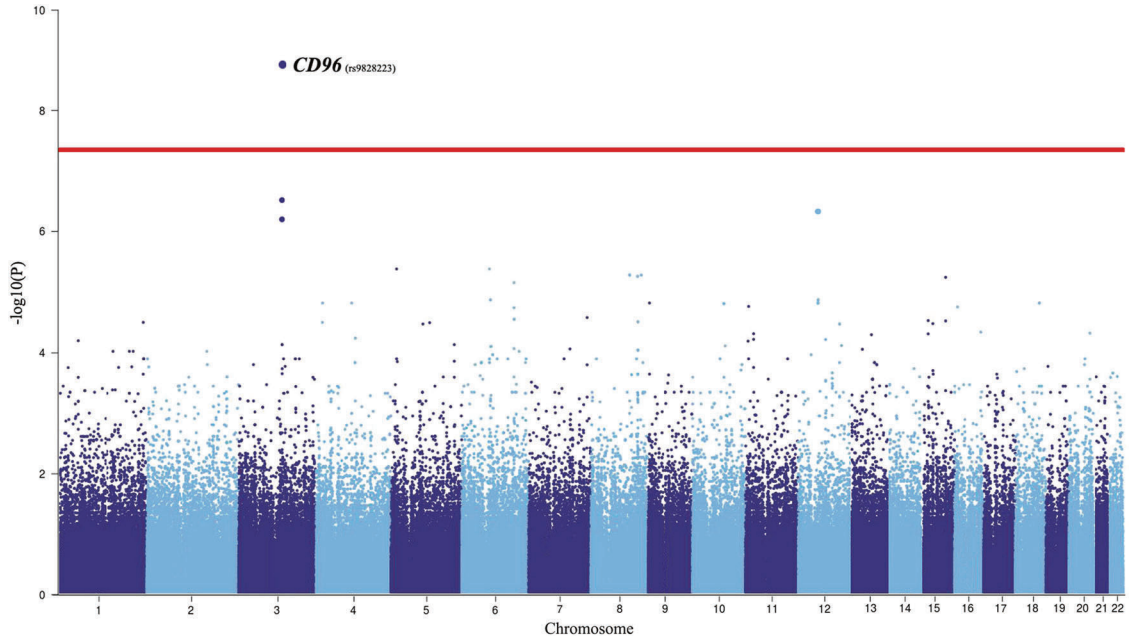


Fig. 1 Manhattan plot for immunogenicity to adalimumab therapy in the discovery cohort. This figure represents the genome-wide *P*-values of association between genetic variation and the formation of antidrug antibodies in Crohn's disease. The *y* axis shows the $-\log_{10}$ *P*-values of 540,221 SNPs, and the *x* axis shows their chromosomal positions (SNP base pair in build GRCh37/hg19). The horizontal red line represents the genome-wide significance threshold ($P = 5.00e{-}8$)

Table 3 Genetic association between the *CD96* locus and both the production of antidrug antibodies and anti-TNF response

SNP	Chr	Pos	Locus	Clinical phenotype[a]	Risk allele	*P*-value	OR
rs9828223	3	111224925	*CD96*	Antidrug antibodies	*A*	1.83e–09	15.87 (4.38–57.49)
				Anti-TNF response	*A*	0.019	1.77 (1.09–5.02)

Chr chromosome, *Pos* base pair in build GRCh37/hg19, *OR* odds ratio and 95% confidence interval (the odds ratios shown are relative to the minor allele)

[a]For antidrug antibody production, the association statistics that are shown in this table result from combining the discovery and replication groups

Association between the *CD96* locus and the clinical response to anti-TNF therapy

Given that antidrug immunogenicity is one of the main factors influencing the clinical response to anti-TNF therapy, we further investigated the association between the validated *CD96* polymorphism and the clinical response to adalimumab. Combining both patient cohorts to increase the statistical power of the analysis ($N_{responders} = 121$; $N_{non\text{-}responders} = 13$; Table 4), we found a significant association between the clinical response to adalimumab and genetic variation at *CD96* locus ($P = 0.019$; OR = 1.77, 95% CI, 1.09–5.02). As expected, the risk allele for the development of adalimumab immunogenicity (*A*) was also the allele associated with a lack of clinical response to adalimumab therapy.

Table 4 Clinical characterization of Crohn's disease patients according to their response to anti-TNF therapy

Clinical characteristics[a]	Responders ($N = 121$)	Non-responders ($N = 13$)	*P*-value
Males	60/61 (49.59)	5/8 (38.46)	0.56
Age	37.82 ± 11.56	41.99 ± 11.32	0.22
Age disease onset	29.80 ± 11.91	33.12 ± 14.66	0.35
Body mass index	23.76 ± 3.82	23.81 ± 2.68	0.96
Smoking			
Current smoker	40/73 (35.40)	4/9 (30.77)	1.00
Ex-smoker	32/81 (28.32)	3/10 (23.10)	1.00
Never smoker	41/72 (36.28)	6/7 (46.15)	0.56
Disease location			
Ileal (L1)	54/25 (68.35)	2/8 (20.00)	0.0046
Colonic (L2)	16/63 (20.25)	1/9 (10.00)	0.68
Ileocolonic (L3)	38/41 (48.10)	7/3 (70.00)	0.31
Upper gastrointestinal tract (L4)	12/61 (16.44)	2/5 (28.57)	0.59
Disease behavior			
Inflammatory	62/52 (54.39)	8/5 (61.54)	0.77
Stenosing	44/75 (36.97)	2/10 (16.67)	0.21
Fistulizing	27/87 (23.68)	4/8 (33.33)	0.48
Extraintestinal manifestations	34/81 (29.56)	1/11 (8.33)	0.18

The comparison between clinical responders and non-responders to anti-TNF was conducted using the Fisher's exact test for categoric variables and Student's *t*-test for quantitative variables

M mean, *N* sample size, *SD* standard deviation, *anti-TNF* tumor necrosis alpha inhibitory drugs

[a]For categoric variables, we show the number of patients displaying and non-displaying the indicated characteristic as well as the percentage of positive patients for this variable. For quantitative variables, we show the mean (i.e., years for ages and kg/m^2 for body mass index) together with the standard deviation

Discussion

One of the major challenges in the treatment of CD is to understand the biological mechanisms responsible for anti-TNF failure. Although the genetic basis for anti-TNF immunogenicity has been investigated in autoimmune diseases, the genetic component underlying the production of antibodies to anti-TNF has not been so far analyzed in CD. In order to identify new genetic variation, we have performed, for the first time, a GWAS on anti-TNF immunogenicity in CD patients. Using a discovery cohort from European ancestry and a validation cohort from the same ancestry, we have identified and validated the association between *CD96* locus and the production of antibodies to anti-TNF therapy.

The SNP associated with AAA—rs9828223—is located 36-kb upstream from the *CD96* gene. CD96 is a member of the immunoglobulin superfamily, and is mainly expressed in the cell membrane of natural killer (NK), CD8+T and CD4+T cells as well as some subsets of B cells [31]. Also, *CD96* expression has been found to be higher in the terminal ileum compared with other human tissues (Supplementary Fig. S2) [32]. CD96 is a membrane receptor that shares a common ligand, CD155, with membrane receptors CD226 and TIGIT [33]. Recent studies in *Cd96* knockout mice have demonstrated that CD96 negatively regulates the cytokine response of NK cells competing with CD226 to bind to their ligand CD155 expressed on antigen presenting cells [34]. Importantly, genetic variation at CD155 has been previously associated with an increased production of neutralizing antibodies to rubella vaccine in humans [35, 36]. Consistent with this finding, we have also detected evidence of association between *CD155* and adalimumab immunogenicity in our CD cohorts, although not in the previously reported polymorphism (Supplementary Table S2).

In *Cd155* knockout mice, this CD96 ligand has been shown to be implicated in the development of the humoral response. *Cd155–/–* mice show a less efficient humoral response to orally administered antigens due to a decreased production of IgG and IgA compared with wild-type mice [37]. Also, significantly higher titers of Th1-associated IgG isotypes IgG2a/c are detected after subcutaneous immunization in *Cd155–/–* mice compared with wild-type littermates [38]. This finding suggests that CD155 promotes polarization of naive CD4+T cells to the Th2 phenotype. Taken together, these experimental observations are consistent with our results. Genetic variation at CD96–CD155 signal transduction pathway could therefore predispose to produce antidrug antibodies by promoting CD155 upregulation and subsequent polarization to the Th2 phenotype that leads to the activation of B cells.

The present study has limitations, which are mainly due to the retrospective nature of the study and the modest sample size of the patient cohorts. Therefore, further studies following a prospective design and using larger cohorts of adalimumab-treated CD patients are warranted. Despite these limitations, we have been able to successfully identify and replicate the first genetic variant associated with adalimumab immunogenicity in CD. Studies involving other anti-TNF therapies approved for CD and even other biological therapies for which antidrug antibodies are a relevant cause for the lack of efficacy should be carried out to determine the implication of *CD96* variation.

In conclusion, we have identified and validated an association between *CD96* locus and immunogenicity to adalimumab in CD. We have also found that variation at this locus is associated with the clinical response to adalimumab therapy in CD. Taken together, these results provide new insights into the genetic basis of immunogenicity and clinical response to anti-TNF therapy in CD patients.

Acknowledgements We thank the patients and clinical specialists collaborating in the IMID Consortium for participation. We also thank Francisca Llinares-Tello (Marina Baixa Hospital, Spain) for her technical recommendations to implement the AAA-detection protocol. This study was funded by the Spanish Ministry of Economic Affairs and Competitiveness (RETOS COLABORACIÓN 2014, grant number: RTC-2014–2920–1), by the Spanish Ministry of Economy and Competitiveness (grant numbers: PSE-010000–2006–6 and IPT-010000–2010–36), and by the "Agència de Gestió d'Ajuts Universitaris i de Recerca" (AGAUR, FI-DGR 2016, grant number: 00587), which is supported by the "Secretaria d'Universitats i Recerca" (Economy and Knowledge Department, Generalitat de Catalunya) and co-funded by the European Social Fund. The study sponsor had no role in the collection, analysis or interpretation of the data.

Compliance with ethical standards

Conflict of interest Dr. Panés has received consulting fees from Abbvie, Boehringer-Ingelheim, Celgene, Ferring, Genentech, GSK, Janssen, MSD, Oppilan, Pfizer, Second Genome, Roche, Takeda, Theravance and TiGenix. Speaker fees from Abbvie, Ferring, Janssen, MSD, and Takeda. Dr. Gisbert has served as a speaker, a consultant and advisory member for or has received research funding from MSD, Abbvie, Hospira, Pfizer, Kern Pharma, Biogen, Takeda, Janssen, Roche, Ferring, Faes Farma, Shire Pharmaceuticals, Dr. Falk Pharma, Tillotts Pharma, Chiesi, Casen Fleet, Gebro Pharma, Otsuka Pharmaceutical, and Vifor Pharma. Dr. Barreiro-de Acosta has served as a speaker, a consultant and advisory member for or has received research funding from MSD, Abbvie, Pfizer, Kern Pharma, Biogen, Takeda, Janssen, Ferring, Faes Farma, Shire Pharmaceuticals, Dr. Falk Pharma, Gebro Pharma, Otsuka Pharmaceutical, and Vifor Pharma. The remaining authors declare that they have no conflict of interest.

Ethics statement Informed consent was obtained from all participants, and protocols were reviewed and approved by local institutional review boards. The present study was conducted according to the Declaration of Helsinki principles.

References

1. Kaplan GG. The global burden of IBD: from 2015 to 2025. Nat Rev Gastroenterol Hepatol. 2015;12:720–7.
2. Xavier RJ, Podolsky DK. Unravelling the pathogenesis of inflammatory bowel disease. Nature. 2007;448:427–34.
3. Van Deventer SJ. Tumour necrosis factor and Crohn's disease. Gut. 1997;40:443–8.
4. Berns M, Hommes DW. Anti-TNF-alpha therapies for the treatment of Crohn's disease: the past, present and future. Expert Opin Invest Drugs. 2016;25:129–43.
5. Gisbert JP, Panes J. Loss of response and requirement of infliximab dose intensification in Crohn's disease: a review. Am J Gastroenterol. 2009;104:760–7.
6. Atzeni F, Talotta R, Salaffi F, Cassinotti A, Varisco V, Battellino M, et al. Immunogenicity and autoimmunity during anti-TNF therapy. Autoimmun Rev. 2013;12:703–8. https://doi.org/10.1016/j.autrev.2012.10.021.
7. Chaparro M, Guerra I, Munoz-Linares P, Gisbert JP. Systematic review: antibodies and anti-TNF-alpha levels in inflammatory bowel disease. Aliment Pharm Ther. 2012;35:971–86.
8. Paul S, Moreau AC, Del Tedesco E, Rinaudo M, Phelip JM, Genin C., et al. Pharmacokinetics of adalimumab in inflammatory bowel diseases: a systematic review and meta-analysis. Inflamm Bowel Dis. 2014;20:1288–95. 0.097/MIB.0000000000000037.
9. Stidham RW, Lee TC, Higgins PD, Deshpande AR, Sussman DA, Singal AG, et al. Systematic review with network meta-analysis: the efficacy of anti-TNF agents for the treatment of Crohn's disease. Aliment Pharm Ther. 2014;39:1349–62.
10. Hyams JS, Griffiths A, Markowitz J, Baldassano RN, Faubion WA Jr., Colletti RB, et al. Safety and efficacy of adalimumab for moderate to severe Crohn's disease in children. Gastroenterology. 2012;143:365–74.e2. https://doi.org/10.1053/j.gastro.2012.04.046.
11. Roblin X, Marotte H, Rinaudo M, Del Tedesco E, Moreau A, Phelip JM, et al. Association between pharmacokinetics of adalimumab and mucosal healing in patients with inflammatory bowel diseases. Clin Gastroenterol Hepatol. 2014;12:80–4.e2. https://doi.org/10.1016/j.cgh.2013.07.010.
12. Mazor Y, Almog R, Kopylov U, Ben Hur D, Blatt A, Dahan A., et al. Adalimumab drug and antibody levels as predictors of clinical and laboratory response in patients with Crohn's disease.

Aliment Pharm Ther. 2014;40:620–8. https://doi.org/10.1111/apt.12869.
13. West RL, Zelinkova Z, Wolbink GJ, Kuipers EJ, Stokkers PC, van der Woude CJ. Immunogenicity negatively influences the outcome of adalimumab treatment in Crohn's disease. Aliment Pharm Ther. 2008;28:1122–6. 10.11/j.365-2036.08.03828.x.
14. Hemmer B, Stuve O, Kieseier B, Schellekens H, Hartung HP. Immune response to immunotherapy: the role of neutralising antibodies to interferon beta in the treatment of multiple sclerosis. Lancet Neurol. 2005;4:403–12.
15. Magdelaine-Beuzelin C, Vermeire S, Goodall M, Baert F, Noman M, Assche GV, et al. IgG1 heavy chain-coding gene polymorphism (G1m allotypes) and development of antibodies-to-infliximab. Pharm Genom. 2009;19:383–7.
16. Bartelds GM, Wijbrandts CA, Nurmohamed MT, Wolbink GJ, de Vries N, Tak PP, et al. Anti-adalimumab antibodies in rheumatoid arthritis patients are associated with interleukin-10 gene polymorphisms. Arthritis Rheum. 2009;60:2541–2.
17. Julia A, Tortosa R, Hernanz JM, Canete JD, Fonseca E, Ferrandiz C, et al. Risk variants for psoriasis vulgaris in a large case-control collection and association with clinical subphenotypes. Hum Mol Genet. 2012;21:4549–57.
18. Lennard-Jones JE. Classification of inflammatory bowel disease. Scand J Gastroenterol Suppl. 1989;170:2–6. discussion16-9
19. Llinares-Tello F, Rosas-Gomez de Salazar J, Senabre-Gallego JM, Santos-Soler G, Santos-Ramirez C, Salas-Heredia E, et al. Practical application of acid dissociation in monitoring patients treated with adalimumab. Rheuma Int. 2014;34:1701–8.
20. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MA, Bender D, et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet. 2007;81:559–75.
21. Aterido A, Julia A, Ferrandiz C, Puig L, Fonseca E, Fernandez-Lopez E, et al. Genome-wide pathway analysis identifies genetic pathways associated with psoriasis. J Invest Dermatol. 2016;136:593–602.
22. Tian C, Gregersen PK, Seldin MF. Accounting for ancestry: population substructure and genome-wide association studies. Hum Mol Genet. 2008;17:R143–50.
23. Price AL, Zaitlen NA, Reich D, Patterson N. New approaches to population stratification in genome-wide association studies. Nat Rev Genet. 2010;11:459–63.
24. Price AL, Patterson NJ, Plenge RM, Weinblatt ME, Shadick NA, Reich D. Principal components analysis corrects for stratification in genome-wide association studies. Nat Genet. 2006;38:904–9.
25. Harvey RF, Bradshaw JM. A simple index of Crohn's-disease activity. Lancet. 1980;1:514.
26. Hirota T, Takahashi A, Kubo M, Tsunoda T, Tomita K, Doi S, et al. Genome-wide association study identifies three new susceptibility loci for adult asthma in the Japanese population. Nat Genet. 2011;43:893–6.
27. Satake W, Nakabayashi Y, Mizuta I, Hirota Y, Ito C, Kubo M, et al. Genome-wide association study identifies common variants at four loci as genetic risk factors for Parkinson's disease. Nat Genet. 2009;41:1303–7.
28. Clarke GM, Anderson CA, Pettersson FH, Cardon LR, Morris AP, Zondervan KT. Basic statistical analysis in genetic case-control studies. Nat Protoc. 2011;6:121–33.
29. Kugler KG, Mueller LA, Graber A. MADAM - an open source meta-analysis toolbox for R and Bioconductor. Source Code Biol Med. 2010;5:3.
30. Ihaka R, Gentleman GR. R: a language for data analysis and graphics. J Comput Graph Stat. 1996;5:299–314.
31. Wang PL, O'Farrell S, Clayberger C, Krensky AM. Identification and molecular cloning of tactile. A novel human T cell activation antigen that is a member of the Ig gene superfamily. J Immunol. 1992;148:2600–8.
32. Lonsdale J, Thomas J, Salvatore M, Phillips R, Lo E, Shad S, et al. The Genotype-Tissue Expression (GTEx) project. Nat Genet. 2013;45:580-5.
33. Martinet L, Smyth MJ. Balancing natural killer cell activation through paired receptors. Nat Rev Immunol. 2015;15:243–54.
34. Chan CJ, Martinet L, Gilfillan S, Souza-Fonseca-Guimaraes F, Chow MT, Town L, et al. The receptors CD96 and CD226 oppose each other in the regulation of natural killer cell functions. Nat Immunol. 2014;15:431–8.
35. Haralambieva IH, Lambert ND, Ovsyannikova IG, Kennedy RB, Larrabee BR, Pankratz VS, et al. Associations between single nucleotide polymorphisms in cellular viral receptors and attachment factor-related genes and humoral immunity to rubella vaccination. PLoS ONE. 2014;9:e99997.
36. Ovsyannikova IG, Salk HM, Larrabee BR, Pankratz VS, Poland GA. Single nucleotide polymorphisms/haplotypes associated with multiple rubella-specific immune response outcomes post-MMR immunization in healthy children. Immunogenetics. 2015;67:547–61.
37. Maier MK, Seth S, Czeloth N, Qiu Q, Ravens I, Kremmer E, et al. The adhesion receptor CD155 determines the magnitude of humoral immune responses against orally ingested antigens. Eur J Immunol. 2007;37:2214–25.
38. Kamran N, Takai Y, Miyoshi J, Biswas SK, Wong JS, Gasser S. Toll-like receptor ligands induce expression of the costimulatory molecule CD155 on antigen-presenting cells. PLoS ONE. 2013;8:e54406.

Affiliations

Adrià Aterido[1,2] · Núria Palau[1] · Eugeni Domènech[3,4] · Pilar Nos Mateu[4,5] · Ana Gutiérrez[4,6] · Fernando Gomollón[4,7] · Juan L. Mendoza[8] · Esther Garcia-Planella[9] · Manuel Barreiro-de Acosta[10] · Fernando Muñoz[11] · Maribel Vera[12] · Cristina Saro[13] · Maria Esteve[4,14] · Montserrat Andreu[15] · María Chaparro[4,16] · Julián Panés[4,17] · Valle García-Sánchez[18] · María López-Lasanta[1] · Andrea Pluma[1] · Laia Codó[19] · Andrés García-Montero[20] · Josep Manyé[3] · Javier P. Gisbert[4,16] · Sara Marsal[1] · Antonio Julià[1]

[1] Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, Spain

[2] Department of Experimental and Health Sciences, Universitat Pompeu Fabra, Barcelona, Spain

[3] Gastroenterology and Hepatology Department, Hospital Universitari Germans Trias i Pujol, Badalona, Spain

[4] Centro de Investigación Biomédica en Red de Enfermedades Hepáticas y Digestivas (CIBEREHD), Madrid, Spain

[5] Digestive Medicine Service, Hospital la Fe, Valencia, Spain

[6] Gastroenterology Service, Hospital General de Alicante, Alicante, Spain

[7] Digestive System Service, Hospital Clínico Universitario, Zaragoza, Spain

[8] Gastroenterology Service, Hospital Clínico San Carlos, Madrid, Spain

[9] Gastroenterology Department, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain

[10] Gastroenterology Service, Hospital Clínico Universitario, Santiago de Compostela, Spain

[11] Gastroenterology Service, Complejo Hospitalario de León, León, Spain

[12] Gastroenterology Service, Hospital Universitario Puerta de Hierro, Madrid, Spain

[13] Internal Medicine Service, Hospital de Cabueñes, Gijón, Spain

[14] Gastroenterology Service, Hospital Universitari Mutua de Terrassa, Barcelona, Spain

[15] Gastroenterology Department, Institut Hospital del Mar d'Investigacions Mèdiques, Institute of Research Hospital del Mar, Parc de Salut Mar, Pompeu Fabra University, Barcelona, Spain

[16] Gastroenterology Service, Hospital Universitario de La Princesa and Instituto de Investigación Sanitaria Princesa (IIS-IP), Madrid, Spain

[17] Gastroenterology Department, Hospital Clínic de Barcelona, Institut d'Investigacions Biomèdiques August Pi i Sunyer, Barcelona, Spain

[18] Digestive System Service, Universidad de Córdoba/Instituto Maimónides de Investigación Biomédica de Córdoba/Hospital Universitario Reina Sofía, Córdoba, Spain

[19] Life Sciences Department, Barcelona Supercomputing Centre, Barcelona, Spain

[20] Banco Nacional de ADN Carlos III, University of Salamanca, Salamanca, Spain

3) Aterido, Adrià, Julià, A., Ferrándiz, C., Puig, L., Fonseca, E., Fernández-López, E., … Codó, L., Gelpí, J. L. …, Marsal, S. (2016). Genome-Wide Pathway Analysis Identifies Genetic Pathways Associated with Psoriasis. Journal of Investigative Dermatology, 136(3), 593–602. https://doi.org/10.1016/j.jid.2015.11.026

Genome-Wide Pathway Analysis Identifies Genetic Pathways Associated with Psoriasis

Adrià Aterido[1], Antonio Julià[1], Carlos Ferrándiz[2], Lluís Puig[3], Eduardo Fonseca[4], Emilia Fernández-López[5], Esteban Dauden[6], José Luís Sánchez-Carazo[7], José Luís López-Estebaranz[8], David Moreno-Ramírez[9], Francisco Vanaclocha[10], Enrique Herrera[11], Pablo de la Cueva[12], Nick Dand[13], Núria Palau[1], Arnald Alonso[1], María López-Lasanta[1], Raül Tortosa[1], Andrés García-Montero[14], Laia Codó[15], Josep Lluís Gelpí[15], Jaume Bertranpetit[16], Devin Absher[17], Francesca Capon[13], Richard M. Myers[17], Jonathan N. Barker[13,18] and Sara Marsal[1]

Psoriasis is a chronic inflammatory disease with a complex genetic architecture. To date, the psoriasis heritability is only partially explained. However, there is increasing evidence that the missing heritability in psoriasis could be explained by multiple genetic variants of low effect size from common genetic pathways. The objective of this study was to identify new genetic variation associated with psoriasis risk at the pathway level. We genotyped 598,258 single nucleotide polymorphisms in a discovery cohort of 2,281 case-control individuals from Spain. We performed a genome-wide pathway analysis using 1,053 reference biological pathways. A total of 14 genetic pathways ($P_{FDR} \leq 2.55 \times 10^{-2}$) were found to be significantly associated with psoriasis risk. Using an independent validation cohort of 7,353 individuals from the UK, a total of 6 genetic pathways were significantly replicated ($P_{FDR} \leq 3.46 \times 10^{-2}$). We found genetic pathways that had not been previously associated with psoriasis risk such as retinol metabolism ($P_{combined} = 1.84 \times 10^{-4}$), the transport of inorganic ions and amino acids ($P_{combined} = 1.57 \times 10^{-7}$), and post-translational protein modification ($P_{combined} = 1.57 \times 10^{-7}$). In the latter pathway, *MGAT5* showed a strong network centrality, and its association with psoriasis risk was further validated in an additional case-control cohort of 3,429 individuals ($P < 0.05$). These findings provide insights into the biological mechanisms associated with psoriasis susceptibility.

Journal of Investigative Dermatology (2016) **136,** 593–602; doi:10.1016/j.jid.2015.11.026

[1]*Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, Spain;* [2]*Dermatology Department, Hospital Universitari Germans Trias i Pujol, Badalona, Barcelona, Spain;* [3]*Dermatology Department, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain;* [4]*Dermatology Department, Complejo Hospitalario Universitario de A Coruña, A Coruña, Spain;* [5]*Dermatology Department, Hospital Universitario de Salamanca, Salamanca, Spain;* [6]*Dermatology Department, Hospital Universitario La Princesa, Madrid, Spain;* [7]*Dermatology Department, Hospital General Universitario de Valencia, Valencia, Spain;* [8]*Dermatology Department, Hospital Universitario Fundación Alcorcón, Madrid, Spain;* [9]*Dermatology Department, Hospital Universitario Virgen Macarena, Sevilla, Spain;* [10]*Dermatology Department, Hospital Universitario 12 de Octubre, Madrid, Spain;* [11]*Dermatology Department, Hospital Universitario Virgen de la Victoria, Málaga, Spain;* [12]*Dermatology Department, Hospital Universitario Infanta Leonor, Madrid, Spain;* [13]*Division of Genetics and Molecular Medicine, King's College London School of Medicine, Guy's Hospital, London, UK;* [14]*Spanish National DNA Bank, Universidad de Salamanca, Salamanca, Spain;* [15]*Life Sciences, Barcelona Supercomputing Centre, Barcelona, Spain;* [16]*Spanish National Genotyping Centre (CeGen), Universitat Pompeu Fabra, Barcelona, Spain;* [17]*HudsonAlpha Institute for Biotechnology, Huntsville, Alabama, USA; and* [18]*St John's Institute of Dermatology, King's College London, London, UK*

Correspondence: Antonio Julià, Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, 08035, Spain. E-mail: toni.julia@vhir.org or Sara Marsal, Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, 08035, Spain. E-mail: sara.marsal@vhir.org

Abbreviations: BC, betweenness centrality; DC, degree centrality; FDR, false discovery rate; GWAS, genome-wide association studies; SNP, single nucleotide polymorphism

Received 13 July 2015; revised 27 October 2015; accepted 12 November 2015; accepted manuscript published online 29 December 2015; corrected proof published online 23 January 2016

INTRODUCTION

Psoriasis is a common chronic inflammatory disease of the skin that affects approximately 2% of the worldwide population (Nestle et al., 2009). In psoriasis, immune cells infiltrate the skin leading to an increased proliferation of keratinocytes (Ferenczi et al., 2000; Gudjonsson and Elder, 2007). It is a genetically complex disease with a complex mode of inheritance (Vyse and Todd, 1996). HLA class I gene *HLA-C*0602* haplotype association explains the largest part of the known heritability of psoriasis (Nair et al., 2006; Strange et al., 2010).

Genome-wide association studies (GWAS) have been successful in the characterization of the genetic architecture of many complex human diseases (Manolio, 2010). To date, more than 15 GWAS have been performed using large psoriasis cohorts from Caucasian and Asian populations and have collectively identified more than 50 susceptibility loci for psoriasis (Bowes et al., 2015; Tsoi et al., 2015b; Yin et al., 2015; Zuo et al., 2015). Despite progress in characterizing psoriasis genetic etiology, loci outside the HLA region only explain less than 25% of the estimated psoriasis heritability (Tsoi et al., 2012; Yin et al., 2014).

Recent research has shown that the missing heritability of complex human diseases can be explained by common genetic variants, rare variants or a combination of genetic, epigenetic, and environmental interactions (Gibson, 2012). From these, common genetic variants could explain more

than 60% of the heritability of the most prevalent autoimmune diseases (Golan et al., 2014). Importantly, most of these common genetic variants are characterized by having low effect sizes (Park et al., 2010).

Although GWAS based on single markers have successfully identified disease-susceptibility variants, this strategy is not adequate to identify genetic variants with low effect sizes that are genuinely associated with disease risk (Du et al., 2012). In single-marker GWAS, a large number of genetic variants are tested for association with a complex trait. To avoid false positive results, a stringent genome-wide significant threshold must be used (Johnson et al., 2010). This conservative threshold, however, does not allow the identification of modest effect risk loci, unless extremely large samples sizes of cases and controls are used (Wang et al., 2010). Importantly, single-marker GWAS consider only the individual effect of each single nucleotide polymorphism and ignore the joint effect of multiple causal genetic variants as well as the biological context where disease genes operate (Zhang et al., 2010).

Functionally related genes have been shown to collectively contribute to disease susceptibility, including those loci that do not reach individually the genome-wide significant threshold (Zhong et al., 2010). Recently, new methods that are able to analyze genetic associations at the pathway level have been developed (Gui et al., 2011). Pathway-based approaches are robust statistical methodologies that integrate genetic and biological knowledge to test whether sets of functionally related genes are jointly associated with a complex trait (Ramanan et al., 2012). Therefore, pathway-based methods increase the statistical power of the association analysis by reducing the number of association tests that must be performed and allow a functional interpretation of the results (Wu et al., 2010).

Pathway-based analyses have been recently performed to study the genetic basis of cancer subtypes using either selected candidate pathways, but also at a genome-wide scale (Chen et al., 2014; Koster et al., 2014). Although the genome-wide pathway analysis can have a high computational cost, this approach is able to identify novel genetic pathways associated with disease risk. The identification of new pathways associated with disease risk could increase the probability to develop new therapeutic strategies in complex diseases such as psoriasis. To date, however, the genome-wide pathway analysis approach has not been performed in psoriasis.

To gain a better understanding of the genetic risk basis of psoriasis, we performed a genome-wide pathway analysis on a large multicenter cohort of patients with psoriasis. In this study, we analyzed the association of 1,053 reference biological pathways using 1,263 patients with psoriasis and 1,558 controls from Spain. Using an independent cohort of 2,178 cases and 5,175 controls from the UK, we then performed a validation study of the significantly associated pathways in the discovery cohort. With this approach, we identified genetic pathways that had not been previously associated with psoriasis risk such as retinol metabolism, transport of inorganic ions and amino acids, and post-translational protein modification. These results provide important insights into the genetic etiology of psoriasis.

Table 1. Pathways associated with psoriasis risk and validated in the replication stage

Pathway	Database	Genes	$SNPs^{D}$[1]	P^{D}	FDR^{D}	P^{DE}	FDR^{DE}	$SNPs^{R}$[1]	P^{R}	FDR^{R}	P^{RE}	FDR^{RE}	P^{C}
Inflammatory response[2]	Biocarta	29	628	$<9.99 \times 10^{-8}$	5.25×10^{-5}	1.35×10^{-2}	4.71×10^{-2}	606	$<3.33 \times 10^{-7}$	5.77×10^{-7}	1.53×10^{-2}	1.58×10^{-2}	1.06×10^{-12}
Natural killer T cell[2]	Biocarta	29	638	$<9.99 \times 10^{-8}$	5.25×10^{-5}	1.17×10^{-2}	4.71×10^{-2}	603	$<3.33 \times 10^{-7}$	5.77×10^{-7}	1.59×10^{-2}	1.59×10^{-2}	1.06×10^{-12}
DNA repair[2]	Reactome	112	2,050	1.33×10^{-4}	9.36×10^{-3}	–	–	1,962	$<3.33 \times 10^{-7}$	5.77×10^{-7}	–	–	1.10×10^{-9}
Amino acid transport across the plasma membrane	Reactome	31	1,025	2.00×10^{-4}	1.24×10^{-2}	–	–	993	4.00×10^{-5}	5.63×10^{-5}	–	–	1.57×10^{-7}
Post-translational protein modification	Reactome	188	5,965	2.00×10^{-4}	1.24×10^{-2}	–	–	5,725	4.00×10^{-5}	5.63×10^{-5}	–	–	1.57×10^{-7}
Transport to the Golgi and subsequent modification	Reactome	33	1,557	3.33×10^{-4}	1.95×10^{-2}	–	–	1,516	3.20×10^{-3}	4.38×10^{-3}	–	–	1.57×10^{-5}
Asparagine N-linked glycosylation	Reactome	81	2,760	4.00×10^{-4}	2.11×10^{-2}	–	–	2,639	8.67×10^{-3}	1.13×10^{-2}	–	–	4.72×10^{-5}
Transport of inorganic ions and amino acids	Reactome	94	4,010	4.00×10^{-4}	2.11×10^{-2}	–	–	3,872	2.00×10^{-5}	3.25×10^{-5}	–	–	1.57×10^{-7}
Retinol metabolism	KEGG	64	1,512	5.33×10^{-4}	2.55×10^{-2}	–	–	1,406	2.79×10^{-2}	3.46×10^{-2}	–	–	1.84×10^{-4}

Abbreviations: C, combined; D, discovery cohort; E, exclusion *IL12B* gene; FDR, false discovery rate; KEGG, Kyoto Encyclopedia of Genes and Genomes; P, empirical set-based *P*-value; R, replication cohort.
[1]Number of single nucleotide polymorphisms mapping to a particular pathway.
[2]Increased permutations to refine the *P*-value (n = 10,000,000).

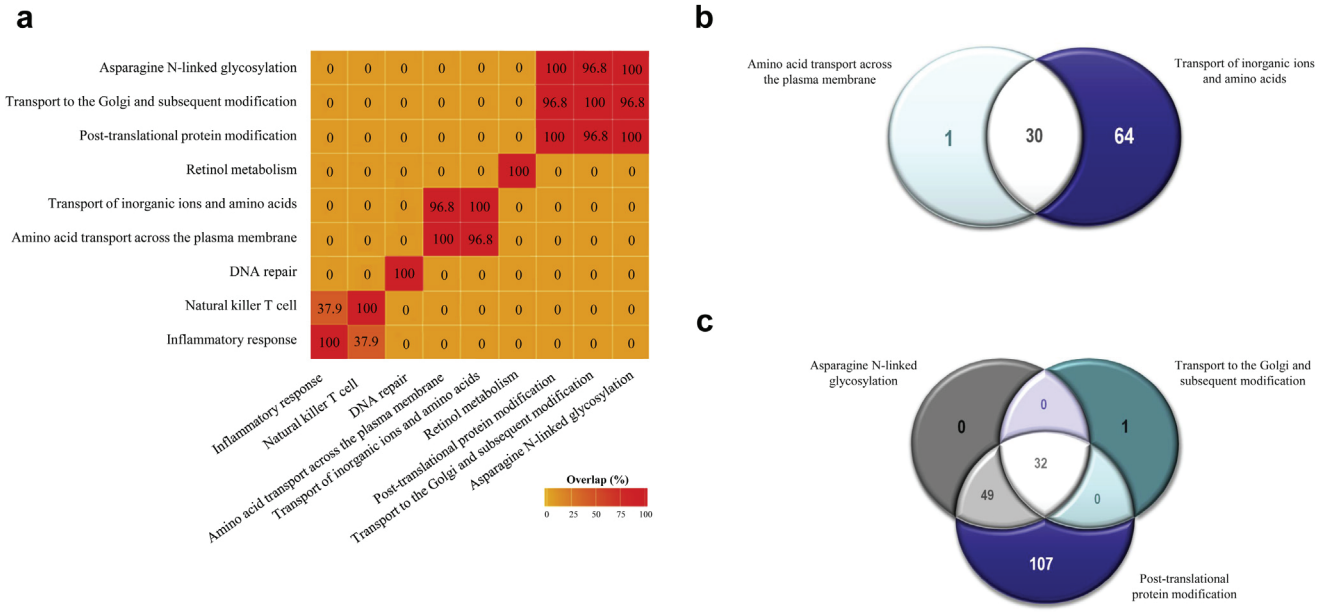


Figure 1. Gene overlap of genetic pathways associated with psoriasis risk. (**a**) Heat map representing the percentage of genes that are shared between each pathway pair. (**b**) Venn diagram of the overlapping pathways representing the transport of inorganic ions and amino acids process as well as the number of genes shared between them. (**c**) Venn diagram of the overlapping pathways representing the post-translational protein modification process as well as the number of genes shared between them.

RESULTS

Identification of genetic pathways associated with psoriasis risk

In the discovery stage, the genome-wide pathway analysis identified a total of 26 genetic pathways significantly associated with psoriasis risk after multiple test correction ($P_{FDR} < 0.05$, Supplementary Table S1 online). The complete results of the genome-wide pathway analysis performed in the discovery study are shown in Supplementary Table S2 online.

From the 26 significantly associated pathways, we found that 14 pathways included *IL12B* gene. After *HLA-C*0602*, *IL12B* is one of the strongest known genetic risk factors for psoriasis. To confirm that the observed pathway associations were the result of the joint effect of multiple genes and not the result of a single risk locus strongly associated with the disease, we removed *IL12B* from these genetic pathways and tested again for association. After extracting *IL12B*, two genetic pathways—"Inflammatory response" and "Natural killer T cell"—remained significantly associated with psoriasis risk ($P_{FDR} < 0.05$). Consequently, only these two pathways from the group containing *IL12B* gene were selected for replication. Together with the other 12 pathways, a total of 14 different genetic pathways were finally tested for validation in the UK population. Using this independent case-control cohort we significantly validated the association of 9 genetic pathways with psoriasis risk ($P_{FDR} < 0.05$, Table 1).

Characterization of the genetic pathways associated with psoriasis risk

To discard the presence of redundant pathways, we evaluated the level of gene overlap between all associated pathways. From the nine validated genetic pathways, we found that the "Amino acid transport across the plasma membrane" and "Transport of inorganic ions and amino acids" pathways, as well as the "Asparagine N-linked glycosylation," "Transport to the Golgi and subsequent modification," and "Post-translational protein modification" pathways had a high degree of overlap between them (>95% of shared genes, Figure 1a). Consequently, and to avoid redundancy, only the pathway showing the highest level of significance was selected to represent each biological process. The "Transport of inorganic ions and amino acids" ($P_{combined} = 1.57 \times 10^{-7}$, Figure 1b) and "Post-translational protein modification" ($P_{combined} = 1.57 \times 10^{-7}$, Figure 1c) pathways were therefore selected from each overlapping pathway group. The "Inflammatory response" ($P_{combined} = 1.06 \times 10^{-12}$), "Natural killer T cell" ($P_{combined} = 1.06 \times 10^{-12}$), "DNA repair" ($P_{combined} = 1.10 \times 10^{-9}$), and "Retinol metabolism" ($P_{combined} = 1.84 \times 10^{-4}$) pathways did not show a significant degree of overlap and were therefore considered as independent biological processes.

Within the final group of six genetic pathways associated with disease risk and representing independent biological processes, we analyzed the association between each particular gene and psoriasis risk (Table 2). We found 37 small-effect genes that were nominally associated with psoriasis risk both in the discovery and replication cohorts ($P \leq 1.29 \times 10^{-2}$, Table 3). The complete list of genetic associations obtained from each genetic pathway is shown in Supplementary Table S3 online. The linkage disequilibrium pattern between the SNPs mapping to each genetic pathway associated with psoriasis risk is shown in Supplementary Figure S1 online.

Functional-based networks associated with psoriasis risk

To understand the relevance of each particular gene within the genetic pathway associated with psoriasis risk, we used biological knowledge to build the associated functional-based network (Figure 2). Using known or predicted functional

Table 2. Association results of the top five genes involved in each pathway associated with psoriasis risk

Pathway[1]	Database	SNP^D	COORD	A1	A2	OR^D	P^D	Gene^D	SNP^R	COORD	A1	A2	OR^R	P^R	Gene^R
Inflammatory response	Biocarta	rs20541	5:131995964	A	G	0.72	4.18×10^{-5}	*IL13,IL4*	rs2965012	1:218786549	A	C	0.83	7.56×10^{-4}	*TGFB2*
		rs11739623	5:131864152	A	G	1.21	1.79×10^{-3}	*IL5*	rs2243123	3:159709651	G	A	1.14	1.06×10^{-3}	*IL12A*
		rs2799083	1:218581617	G	A	1.22	2.82×10^{-3}	*TGFB2*	rs25890	5:131437562	G	A	0.88	1.09×10^{-3}	*CSF2*
		rs2366408	3:159696099	A	C	1.19	3.35×10^{-3}	*IL12A*	rs20541	5:131995964	A	G	0.86	2.41×10^{-3}	*IL13,IL4*
		rs2069837	7:22768027	G	A	1.33	3.93×10^{-3}	*IL6*	rs4963517	12:6947800	A	G	0.90	2.94×10^{-3}	*CD4*
Natural killer T cell	Biocarta	rs20541	5:131995964	A	G	0.72	4.18×10^{-5}	*IL4*	rs4297265	1:67852335	G	A	0.83	4.01×10^{-7}	*IL12RB2*
		rs11739623	5:131864152	A	G	1.21	1.79×10^{-3}	*IL5*	rs749873	2:136817088	G	A	0.84	2.61×10^{-5}	*CXCR4*
		rs2799083	1:218581617	G	A	1.22	2.82×10^{-3}	*TGFB2*	rs2965012	1:218786549	A	C	0.83	7.56×10^{-4}	*TGFB2*
		rs2114808	2:137249556	G	A	0.81	3.09×10^{-3}	*CXCR4*	rs2243123	3:159709651	G	A	1.14	1.06×10^{-3}	*IL12A*
		rs2366408	3:159696099	A	C	1.19	3.35×10^{-3}	*IL12A*	rs25890	5:131437562	G	A	0.88	1.09×10^{-3}	*CSF2*
Retinol metabolism	KEGG	rs2173201	4:100250970	A	C	0.77	5.82×10^{-5}	*ADH1C,ADH1B*	rs7188923	16:81336356	A	G	0.89	1.81×10^{-3}	*BCMO1*
		rs4148295	4:70475866	C	A	1.23	3.41×10^{-4}	*UGT2A1*	rs10882144	10:94852448	A	G	0.87	2.55×10^{-3}	*CYP26A1*
		rs17614939	4:70360229	G	A	0.78	5.21×10^{-4}	*UGT2B4*	rs4319546	12:57346828	A	G	0.89	4.96×10^{-3}	*RDH16*
		rs2279345	19:41515702	A	G	0.84	2.44×10^{-3}	*CYP2B6*	rs4405788	2:72235688	A	G	0.90	5.48×10^{-3}	*CYP26B1*
		rs17864686	2:234591339	A	G	1.25	3.37×10^{-3}	*UGT1A8*	rs11670760	19:41336795	G	A	1.12	5.73×10^{-3}	*CYP2A6*
DNA repair	Reactome	rs240956	6:111616051	A	C	1.46	3.16×10^{-6}	*REV3L*	rs458017	6:111696091	G	A	1.65	1.40×10^{-13}	*REV3L*
		rs20541	5:131995964	A	G	0.72	4.18×10^{-5}	*RAD50*	rs2240116	9:35094373	A	G	1.36	5.22×10^{-4}	*FANCG*
		rs2213178	8:48816716	A	G	1.29	6.11×10^{-5}	*PRKDC*	rs7099120	10:131015367	A	G	1.15	9.51×10^{-4}	*MGMT*
		rs2985689	14:50098031	C	A	1.28	1.66×10^{-3}	*POLE2*	rs3783819	14:61316264	A	G	0.89	1.13×10^{-3}	*MNAT1*
		rs1887181	10:131594850	G	A	1.46	1.86×10^{-3}	*MGMT*	rs11693731	2:58887650	A	G	0.89	1.13×10^{-3}	*FANCL*
Post-translational protein modification	Reactome	rs1007108	1:26104973	A	G	1.43	2.74×10^{-6}	*MAN1C1*	rs9886302	7:70751484	A	G	0.81	7.29×10^{-6}	*WBSCR17*
		rs10865331	2:62551472	A	G	1.25	7.88×10^{-5}	*B3GNT2*	rs7220464	17:7210836	A	C	0.85	2.09×10^{-5}	*EIF5A*
		rs3791312	2:135183045	G	A	0.71	8.04×10^{-5}	*MGAT5*	rs4528932	3:118941441	A	G	1.17	5.28×10^{-5}	*B4GALT4*
		rs1495086	8:15378013	A	G	0.78	1.02×10^{-4}	*TUSC3*	rs7780461	7:151641016	A	G	1.24	8.26×10^{-5}	*GALNTL5*
		rs977905	3:5882683	G	A	1.24	1.69×10^{-4}	*EDEM1*	rs12262718	10:17343706	A	G	1.31	8.68×10^{-5}	*ST8SIA6*
Transport of inorganic ions and amino acids	Reactome	rs12661704	6:111560890	A	G	1.60	2.34×10^{-7}	*SLC16A10*	rs12661704	6:111560890	A	G	1.42	2.38×10^{-9}	*SLC16A10*
		rs10205402	2:40710953	A	G	0.77	8.78×10^{-6}	*SLC8A1*	rs2385844	2:220839453	G	A	0.85	9.23×10^{-6}	*SLC4A3*
		rs532237	20:48467560	G	C	1.31	1.09×10^{-4}	*SLC9A8*	rs6012750	20:48430680	A	G	0.86	6.81×10^{-5}	*SLC9A8*
		rs538385	13:30229665	G	A	0.82	6.91×10^{-4}	*SLC7A1*	rs1874361	1:205908186	A	C	1.15	1.63×10^{-4}	*SLC26A9*
		rs17050441	4:139402774	G	A	1.27	1.14×10^{-3}	*SLC7A11*	rs11668878	19:47268373	A	C	1.27	1.65×10^{-4}	*SLC1A5*

Abbreviations: A1, minor allele; A2, major allele; COORD, SNP coordinates in build GRCh37/hg19; D, discovery cohort; KEGG, Kyoto Encyclopedia of Genes and Genomes; OR, odds ratio; P, *P*-value; R, replication cohort.

[1]The detailed description of the "Inflammatory response" and "Natural killer T cell" pathways corresponds to the association results after excluding the *IL12B* gene from the genome-wide pathway analysis.

Table 3. Genes associated with psoriasis risk in the discovery and replication stages for each validated pathway

Pathway[1]	Database	Gene[2]	P^D	P^R
Inflammatory response	Biocarta	*IL12A*	3.35×10^{-3}	1.06×10^{-3}
		IL12B	3.02×10^{-10}	1.69×10^{-18}
		IL13	4.18×10^{-5}	2.41×10^{-3}
		IL4	4.18×10^{-5}	2.41×10^{-3}
		TGFB2	2.82×10^{-3}	7.56×10^{-4}
Natural killer T cell	Biocarta	*CXCR4*	3.09×10^{-3}	2.61×10^{-5}
		IL12A	3.35×10^{-3}	1.06×10^{-3}
		IL12B	3.02×10^{-10}	1.69×10^{-18}
		IL4	4.18×10^{-5}	2.41×10^{-3}
		IL4R	7.35×10^{-3}	1.37×10^{-3}
		TGFB2	2.82×10^{-3}	7.56×10^{-4}
Retinol metabolism	KEGG	*ADH1B*	5.82×10^{-5}	1.21×10^{-2}
		UGT2B4	5.21×10^{-4}	6.13×10^{-3}
		RPE65	5.10×10^{-3}	7.34×10^{-3}
DNA repair	Reactome	*FANCL*	3.22×10^{-3}	1.13×10^{-3}
		MGMT	1.86×10^{-3}	9.51×10^{-4}
		RAD50	4.18×10^{-5}	2.41×10^{-3}
		REV3L	3.16×10^{-6}	1.40×10^{-13}
		RFC3	2.14×10^{-3}	1.94×10^{-3}
Transport of inorganic ions and amino acids	Reactome	*SLC16A10*	2.34×10^{-7}	2.38×10^{-9}
		SLC1A4	2.04×10^{-3}	7.44×10^{-3}
		SLC38A1	1.34×10^{-3}	9.44×10^{-3}
		SLC43A2	1.29×10^{-2}	1.15×10^{-2}
		SLC7A1	6.91×10^{-4}	6.62×10^{-3}
		SLC7A11	1.14×10^{-3}	6.22×10^{-3}
		SLC7A7	5.09×10^{-3}	2.77×10^{-3}
		SLC8A1	8.75×10^{-6}	1.30×10^{-3}
		SLC9A8	1.09×10^{-4}	6.81×10^{-5}
		SLC9A9	4.37×10^{-3}	3.22×10^{-3}
Post-translational protein modification	Reactome	*ALG10*	3.89×10^{-3}	4.15×10^{-3}
		B3GNT2	7.88×10^{-5}	1.01×10^{-3}
		EDEM1	1.69×10^{-4}	5.81×10^{-4}
		EIF5A	3.42×10^{-4}	2.09×10^{-5}
		FUT8	5.30×10^{-3}	1.87×10^{-4}
		GALNT1	5.22×10^{-4}	9.72×10^{-4}
		MAN1A1	2.82×10^{-4}	1.04×10^{-2}
		MAN2A1	7.29×10^{-3}	3.53×10^{-3}
		MGAT5	8.04×10^{-5}	9.34×10^{-3}
		SEMA6D	2.06×10^{-3}	1.46×10^{-3}
		ST8SIA6	8.17×10^{-3}	8.68×10^{-5}
		TUSC3	1.02×10^{-4}	6.07×10^{-3}

Abbreviations: D, discovery cohort; KEGG, Kyoto Encyclopedia of Genes and Genomes; OR, odds ratio; P, *P*-value; R, replication cohort.

[1]The detailed description of the "Inflammatory response" and "Natural killer T cell" pathways corresponds to the association results before excluding the *IL12B* gene from the genome-wide pathway analysis.

[2]Genes contained in the genetic pathways that were nominally associated with psoriasis risk in the discovery and replication stages.

associations between the pathway genes, functional-based networks are a powerful approach to represent and analyze the topological structure of a biologic pathway.

To characterize the network properties of the resulting functional-based networks, we determined the betweenness centrality (BC) and degree centrality (DC) statistics (Supplementary Table S4 online). These two measures are useful to identify those network elements (genes in this case) that are likely to be more influential in the structure of the network. BC and DC have been widely used to identify the genes that are more likely to be essential for pathway functionality (Hahn and Kern, 2005; Joy et al., 2005; Vallabhajosyula et al., 2009). We found that *SLC7A11* from the "Transport of inorganic ions and amino acids" pathway and *MGAT5* from the "Post-translational protein modification" pathway had markedly high BC values (BC $\geq$ 0.1). From these, *MGAT5* gene also showed a much stronger DC value than *SLC7A11* ($DC_{MGAT5} = 19$, $DC_{SLC7A11} = 3$).

Given the strong network centrality properties found for *MGAT5* gene in the "Post-translational protein modification" pathway, we decided to further test the association of this key gene with psoriasis risk in an independent case-control cohort. Using this additional replication cohort, we significantly validated the association of *MGAT5* gene with psoriasis risk ($P = 1.3 \times 10^{-2}$; odds ratio [95% confidence interval] = 0.85 [0.74–0.96]).

Functional analysis of *MGAT5* variation

MGAT5 encodes for a key enzyme in the N-glycosylation pathway, a post-translational process that is directly implicated in T-cell activation and differentiation (Demetriou et al., 2001). To assess the functional role of *MGAT5* in psoriasis pathogenesis, we evaluated the association between genetic variation at *MGAT5* gene and the levels of T-cell surface glycosylation. Flow cytometry analysis of in vitro activated $CD4^+$ and $CD8^+$ T cells obtained from 27 patients with psoriasis showed an increase in N-glycosylation levels in patients carrying one or two copies of the protective allele (G) compared with homozygous individuals for the risk allele (A) (Figure 3). The increased glycosylation levels in individuals carrying at least one copy of (G) allele was observed both in activated $CD8^+$ and $CD4^+$ T cells. In $CD4^+$ T lymphocytes, the glycosylation level was significantly higher in GG homozygotes compared with AA homozygotes ($P = 0.01$, Figure 3).

DISCUSSION

Genome-wide association analyses have successfully identified more than 50 loci associated with psoriasis susceptibility. To date, however, the genetic basis of psoriasis is still not completely understood. In this study, we have performed a genome-wide pathway analysis of psoriasis genetic risk. Using a discovery cohort from Spain and an independent cohort from the UK, we have identified and validated the association of six genetic pathways with psoriasis susceptibility. Importantly, these validated pathways include biological processes such as retinol metabolism, transport of inorganic ions and amino acids, and post-translational protein modification that had not been previously associated with psoriasis risk at the genetic level. In addition, analyzing the network properties of these validated pathways we have found that *MGAT5* gene has a strong centrality in the post-translational protein modification pathway. Using an additional independent case-control cohort from Spain, we have further replicated the association of *MGAT5* with psoriasis risk. Taken together,

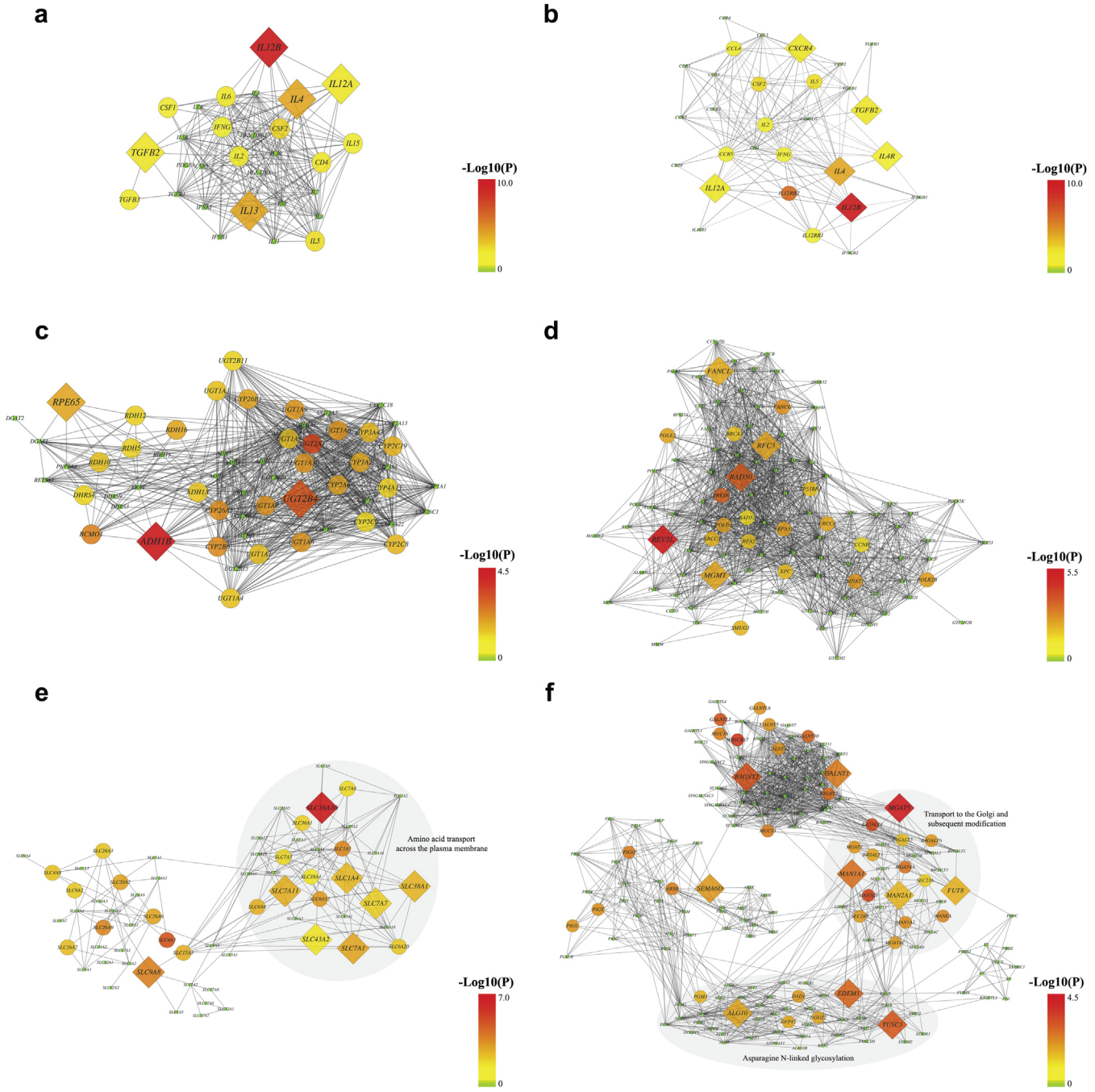


Figure 2. Functional-based network of each genetic pathway associated with psoriasis risk. (**a**) "Inflammatory response." (**b**) "Natural killer T cell." (**c**) "Retinol metabolism." (**d**) "DNA repair." (**e**) **"**Transport of inorganic ions and amino acids." (**f**) **"**Post-translational protein modification." The color of each gene represents the *P*-value of its association with psoriasis in the negative logarithmic scale, ranging from the lowest significance (green) to the strongest (red). The gene shape represents the association with the disease in neither the discovery nor the replication study (square), only in either the discovery or the replication study (circle) and the association found in both discovery and replication studies (rhombus). The edge width is proportional to the confidence of the functional association between two genes. Disconnected genes are hidden.

these findings contribute to a better understanding of the genetic risk basis of psoriasis and provide important insights into the biological mechanisms associated with the disease pathogenesis.

Retinol has been demonstrated to inhibit inflammatory processes in dermatological diseases (Balato et al., 2013). In particular, retinol inhibits the regulatory activity of the nuclear factor kappa B (NFKB) in the skin (Austenaa et al., 2004). NFKB is an established transcriptional factor that regulates multiple proinflammatory genes that are key in psoriasis pathogenesis like tumor necrosis factor and interleukin-17 (Goldminz et al., 2013). The NFKB signaling pathway has also been associated with the regulation of the proliferation of epidermal keratinocytes (Tsuruta, 2009). These findings are consistent with the elevated levels of NFKB that have been found in lesional and non-lesional psoriatic

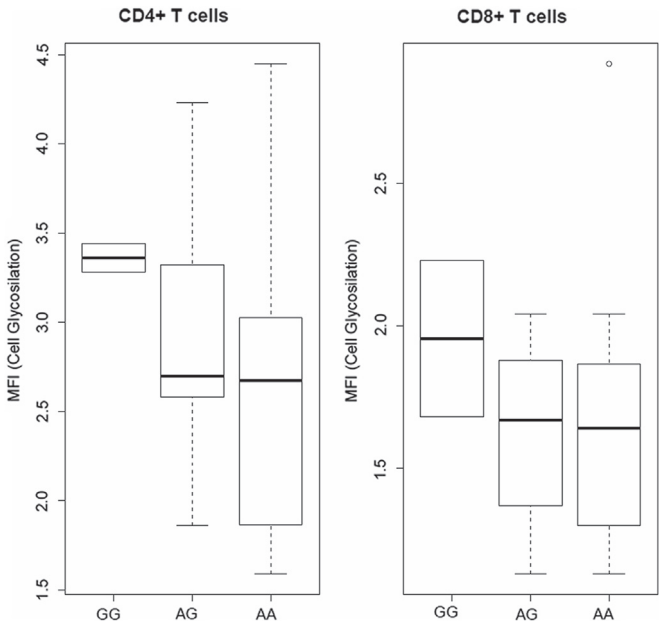


Figure 3. N-Glycosylation on activated T lymphocytes according to *MGAT5* genotype. Boxplots of mean fluorescence intensity (MFI) of cell membrane glycosylation of in vitro activated $CD4^+$ (left) and $CD8^+$ (right) T cells from patients with psoriasis. Patients with one and two copies of the protective (G) allele of *MGAT5* SNP rs3791318 tend to have higher glycosylation levels, thus increasing the threshold for T-cell receptor-mediated response as well as lowering the threshold for cytotoxic T-lymphocyte-associated antigen-4-mediated arrest of T-cell proliferation.

skin samples compared with non-psoriatic skin (Lizzul et al., 2005). Therefore, genetic variation in the retinol metabolism pathway could reduce the retinol production leading to a weakened NFKB signaling and, consequently, promoting both inflammatory and proliferative hallmarks of psoriasis.

Psoriasis risk was also associated with the genetic pathway implicated in the transport of both inorganic ions and amino acids. An increased transport of inorganic ions in $CD4^+$ helper T cells has been shown to contribute to autoimmune and inflammatory diseases (Lang et al., 2014). In particular, the intracellular transport of calcium is crucial for controlling the expression of proinflammatory genes in immune cells (Khananshvili, 2013; Vig and Kinet, 2009). Accordingly, the transport of inorganic ions and amino acids pathway associated with psoriasis risk includes the *SLC8A1* gene, which modulates the cytoplasmic calcium concentration (Clapham, 2007). The transport of amino acids into T cells is essential to maintain the increased production of proinflammatory cytokines in activated human T cells (Hayashi et al., 2013). Importantly, the expression of amino acid transporters has been found to be differentially regulated in psoriatic inflammatory processes (Jaeger et al., 2008). These results therefore suggest that genetic variation in the transport of amino acids and inorganic ions pathway could increase the risk to develop psoriasis by modulating T-cell functionality.

The post-translational protein modification pathway is responsible for the N-linked glycosylation of the asparagine residues in the HLA molecules (Rudd et al., 2001). This post-translational modification pathway has been found to be necessary for the immune system tolerance to self-antigens (Ryan and Cobb, 2012). Previous studies have found that a deficient or aberrant asparagine glycosylation can induce autoimmune diseases (Green et al., 2007). Also, post-translationally modified autoantigens have been associated with psoriasis (Iversen et al., 2011). In patients with psoriasis, the peptide glycosylation activity has been found to be markedly increased in comparison with healthy controls (Damasiewicz-Bodzek and Wielkoszynski, 2012). Furthermore, specific post-translational modifications on glycoproteins expressed on the surface of T lymphocytes have been shown to target these cells to the inflamed skin (Fuhlbrigge et al., 1997). Therefore, genetic variation in the post-translational protein modification pathway could perturb the glycosylation processes that are crucial to maintain the immune system tolerance.

MGAT5 encodes for a key enzyme in the N-glycosylation pathway. This pathway has been directly implicated in T-cell activation and autoimmunity (Demetriou et al., 2001). Recent research has found an association between MGAT5 glycosylation activity and multiple sclerosis etiology both in experimental models and in humans (Grigorian and Demetriou, 2011; Mkhikian et al., 2011). In this study, we have found that the *MGAT5* is a key gene in the post-translational protein modification pathway associated with psoriasis. Subsequently, we found that genetic variation at *MGAT5* is associated with the level of glycosylation of in vitro activated T cells. This result is consistent with previous findings showing that deficiency of MGAT5 glycosylation activity reduces the T-cell activation threshold and, consequently, promotes the triggering of autoimmune diseases (Demetriou et al., 2001). Further studies evaluating the implication of the T-cell surface glycosylation in clinically relevant outcomes in psoriasis such as skin severity are warranted.

The association of psoriasis risk with the inflammatory response and the natural killer T-cell pathways involves more than 10 immune-related genes, including *IL12B*. In a recent pathway analysis study using association results of a meta-analysis for psoriasis risk (Tsoi et al., 2015a), these two pathways were also found to be associated. These findings, however, were not validated using an independent cohort. Our study, therefore, provides strong confirmation of the implication of these two genetic pathways in the risk of psoriasis. Also, the permutation-based approach used in our study allowed to control for the potential bias associated with the presence of strong linkage disequilibrium patterns within genes. Our results indicate that the association of these pathways is not only driven by *IL12B* gene, but it is the result of the joint contribution of other small-effect genes in these pathways. One of these genes is *CXCR4,* which encodes for a chemokine receptor from the natural killer T-cell pathway (Colantonio et al., 2002). Although *CXCR4* gene has not been previously associated with psoriasis risk in single-marker GWAS, CXCR4 chemokine has been shown to reduce keratinocyte proliferation and, consequently, the expansion of psoriatic plaques by regulating the proliferative cytokine signals that are activated in psoriatic lesions (Takekoshi et al., 2013). In addition, the inflammatory angiogenesis of psoriatic skin that leads to vascular remodeling has been recently shown to be modulated by CXCR4 chemokine (Zgraggen et al., 2014). Using the pathway analysis, we can therefore identify small-effect genes like *CXCR4* that cannot be

detected by single-marker GWAS but that are biologically implicated in key processes of the disease pathophysiology.

In this study, we have also found a significant association between the DNA repair genetic pathway and psoriasis risk. Together with the dysregulation of immune system processes, the epidermal hyperproliferation is another well-known biological process implicated in the psoriasis pathophysiology (Wolf et al., 2012). The application of ultraviolet radiation in psoriasis skin lesions to induce apoptosis in aberrantly proliferating keratinocytes has proved to be a successful treatment for the clearance of plaque psoriasis in approximately 70% of patients (Weatherhead et al., 2011). The ultraviolet radiation induces DNA damage that promotes the transcription of the DNA repair pathway genes (Roos and Kaina, 2006). Consequently, the enzymatic machinery of the pathway repairs the DNA damage and also triggers the cell death by activating the p53 apoptotic signaling (Lavin et al., 2005). Therefore, these results suggest that genetic variation in the DNA repair pathway promotes an inefficient activation of the p53 apoptotic signaling that leads to an increased keratinocyte proliferation, as well as an inefficient response to ultraviolet therapy in patients with psoriasis.

Although the pathway-based analysis is a powerful approach to identify small-effect genetic variants associated with disease risk, this methodology is not exempt of limitations. Intergenic SNPs across the whole genome that map physically far away from genes were not included in this study. These genetic variants could be known risk loci (e.g., rs12188300 is associated with psoriasis risk and is located at >20Kb from *IL12B* gene) or may regulate the expression of genes through *cis*- and *trans*-expression quantitative trait loci mechanisms (Gilad et al., 2008). Also, some SNPs might not be functionally related to the closest genes. With the increasing regulatory information derived from expression quantitative trait loci and epigenomic data (Bernstein et al., 2010; Martens and Stunnenberg, 2013; Raney et al., 2011), intergenic SNPs could be integrated in the pathway-based analysis in the next few years.

The complex linkage disequilibrium structure of the *HLA* region together with the strong association with the susceptibility to multiple common diseases has been shown to generate false positive results in pathway-based methods (Wang et al., 2010). Following recent studies, in this study we removed the SNPs mapping to this locus to perform the present pathway analysis (Chen et al., 2014). As a result, known pathways associated with psoriasis risk that include genes from the *HLA* region, like the NFKB pathway, were not analyzed in this study. Importantly, however, in this study we have found and validated the association between genetic pathways related to IL12 signaling, an established genetic risk pathway for psoriasis and psoriasis risk. Also, within the associated pathways there are known risk genes for psoriasis (e.g., *REV3L* and *IL4* within the DNA repair and inflammatory response pathways, respectively). Together, these results confirm the accuracy of the present pathway-based approach to identify relevant genetic variation associated with psoriasis risk.

The present genome-wide pathway analysis has two important strengths. First, we used PLINK software (Boston, MA) to identify genetic pathways associated with psoriasis risk. This pathway analysis method uses genotype data in contrast to the methodologies that are only based on association statistics. An important limitation of these latter methodologies is that they do not account for the linkage disequilibrium between SNPs. This can result in highly biased results and a significant increase in false positive results (Wang et al., 2010). Instead, the pathway analysis approach that we used, although can be computationally costly, efficiently overcomes these biases by maintaining the correct linkage disequilibrium patterns between SNPs. Finally, compared with previous pathway-based studies in other complex diseases, we have performed a two-stage pathway analysis in two large cohorts from different populations. Using an independent population, we have validated genetic pathways associated with psoriasis risk.

In conclusion, using a genome-wide pathway analysis approach we have identified to our knowledge previously unreported genetic pathways associated with psoriasis risk. These biological pathways include retinol metabolism, transport of inorganic ions and amino acids, and post-translational protein modification. The results of this study represent an important contribution to the characterization of the genetic risk basis of psoriasis.

MATERIALS AND METHODS

Study population

A total of 1,263 patients with psoriasis and 1,558 controls were recruited for the discovery stage (Supplementary Table S5 online). An independent case-control cohort of 7,353 individuals from the UK was used to validate the significantly associated pathways in the discovery cohort. An independent cohort of 1,381 patients with psoriasis and 2,048 controls from Spain was used to replicate the association between *MGAT5* gene and psoriasis risk (Supplementary Materials, Supplementary Table S6 online).

All the procedures were followed in compliance with the principles of the Declaration of Helsinki and all patients provided written informed consent to participate in this study. The study and the consent procedure were approved by the local Institutional Review Board of each participating center.

DNA extraction and genome-wide genotyping

GWAS genotyping of the 2,821 individuals from the discovery cohort was performed using Illumina Quad610 Beadchips (Illumina, San Diego, CA) (Supplementary Materials). After the quality control analysis, a final data set of 541,926 SNPs from 1,172 patients with psoriasis was available for the pathway-based analysis. The genome-wide genotyping of the patients with psoriasis from the validation stage was performed using the Illumina Human660W-Quad (Illumina) and the healthy controls were genotyped using the Illumina custom Human1.2M-Duo (Illumina) as has been previously described (Strange et al., 2010). The final data set used for the replication study included 515,703 SNPs from 2,178 patients with psoriasis. The genotyping of the *MGAT5* replication cohort was performed using the Taqman real-time PCR platform (Applied Biosystems, Foster City, CA) (Supplementary Materials).

Pathway-based analysis

Gene set definition. Reference biological pathway annotation databases BioCarta (www.biocarta.com), Kyoto Encyclopedia of Genes and Genomes (Kanehisa and Goto, 2000), and Reactome (Croft et al., 2014) were used to determine the global pathways

(Supplementary Materials, Supplementary Tables S7 and S8 online). The final gene set included in this study was composed of 215,948 SNPs mapping to 1,053 pathways.

Gene-set association analysis. The statistical association analysis was performed using the PLINK set-based test (Purcell et al., 2007) (Supplementary Materials). To obtain the global statistical significance of each validated pathway, we combined the empirical *P*-values resulting from the discovery and replication stages using Fisher's method (Kugler et al., 2010). We tested the association of 1,053 pathways with psoriasis risk. The false discovery rate (FDR) method (Hochberg and Benjamini, 1990) was used to account for multiple testing.

Sensitivity analysis by removing the HLA and IL12B loci. In pathway-based analysis, the presence of a single marker with very strong effects can lead to false positive associations. In these cases, the joint contribution of the pathway genes to disease risk is masked and not adequately evaluated (Wang et al., 2010). Similar to previous studies, to avoid this type of spurious associations, we removed all SNPs mapping to the *HLA* region (Megabases 25.6 to 33.3 in chromosome 6) (Chen et al., 2014). In the discovery stage, we found genetic pathways in which the *IL12B* gene was significantly associated with disease risk at a genome-wide scale. *IL12B* is a well-known psoriasis risk gene that shows a large effect on disease susceptibility and, like the *HLA* region, could generate false positive results (Cargill et al., 2007; Nair et al., 2008; Zhu et al., 2013). Accordingly, we removed this psoriasis susceptibility locus (from 158,741,791 to 158,757,481 base pairs in chromosome 5) from the significant pathways and we repeated the analysis. We excluded 73 and 58 SNPs from the discovery and replication studies, respectively.

Characterization of the genetic pathways associated with psoriasis risk

Genetic pathways involved in similar biological processes may share genes. To identify pathways representing different and independent biological processes, we computed the gene overlap between each pair of genetic pathways associated with psoriasis risk (Supplementary Materials).

The statistical significance of the association between pathway genes and psoriasis risk was determined according to the most significant SNP mapping to each particular gene.

Analysis of the functional-based networks associated with psoriasis risk

The biological knowledge representing the functional association between gene pairs was used to build the functional-based network of each genetic pathway associated with psoriasis risk. To identify those genes that are more likely to play a central role in the genetic pathways associated with psoriasis risk, we analyzed the network statistical properties of each functional-based network (Supplementary Materials). Using the genes that were nominally associated with psoriasis risk in both discovery and replication stages, we identified the most influential gene according to the highest values of these network statistics.

Functional analysis of *MGAT5* variation

Following the methodology previously described (Chen et al., 2009), we evaluated the association of *MGAT5* psoriasis risk variant with the level of cell surface glycosylation of in vitro activated $CD4^+$ and $CD8^+$ T cells isolated from n = 27 patients with psoriasis (Supplementary Materials).

CONFLICT OF INTEREST

The authors state no conflict of interest.

ACKNOWLEDGMENTS

This study was funded by of the Spanish Ministry of Economy and Competitiveness, grant numbers: PSE-010000-2006-6 and IPT-010000-2010-36.

SUPPLEMENTARY MATERIAL

Supplementary material is linked to the online version of the paper at www.jidonline.org, and at http://dx.doi.org/10.1016/j.jid.2015.11.026.

REFERENCES

BIOCARTA Pathways. [http://www.biocarta.com].

Austenaa LM, Carlsen H, Ertesvag A, et al. Vitamin A status significantly alters nuclear factor-kappaB activity assessed by in vivo imaging. Faseb J 2004;18:1255–7.

Balato A, Schiattarella M, Lembo S, et al. Interleukin-1 family members are enhanced in psoriasis and suppressed by vitamin D and retinoic acid. Arch Dermatol Res 2013;305:255–62.

Bernstein BE, Stamatoyannopoulos JA, Costello JF, et al. The NIH Roadmap Epigenomics Mapping Consortium. Nat Biotechnol 2010;28:1045–8.

Bowes J, Budu-Aggrey A, Huffmeier U, et al. Dense genotyping of immune-related susceptibility loci reveals new insights into the genetics of psoriatic arthritis. Nat Commun 2015;6:6046.

Cargill M, Schrodi SJ, Chang M, et al. A large-scale genetic association study confirms IL12B and leads to the identification of IL23R as psoriasis-risk genes. Am J Hum Genet 2007;80:273–90.

Clapham DE. Calcium signaling. Cell 2007;131:1047–58.

Colantonio L, Recalde H, Sinigaglia F, et al. Modulation of chemokine receptor expression and chemotactic responsiveness during differentiation of human naive T cells into Th1 or Th2 cells. Eur J Immunol 2002;32:1264–73.

Croft D, Mundo AF, Haw R, et al. The Reactome pathway knowledgebase. Nucleic Acids Res 2014;42:D472–7.

Chen D, Enroth S, Ivansson E, et al. Pathway analysis of cervical cancer genome-wide association study highlights the MHC region and pathways involved in response to infection. Hum Mol Genet 2014;23:6047–60.

Chen HL, Li CF, Grigorian A, et al. T cell receptor signaling co-regulates multiple Golgi genes to enhance N-glycan branching. J Biol Chem 2009;284:32454–61.

Damasiewicz-Bodzek A, Wielkoszynski T. Advanced protein glycation in psoriasis. J Eur Acad Dermatol Venereol 2012;26:172–9.

Demetriou M, Granovsky M, Quaggin S, et al. Negative regulation of T-cell activation and autoimmunity by Mgat5 N-glycosylation. Nature 2001;409:733–9.

Du Y, Xie J, Chang W, et al. Genome-wide association studies: inherent limitations and future challenges. Front Med 2012;6:444–50.

Ferenczi K, Burack L, Pope M, et al. CD69, HLA-DR and the IL-2R identify persistently activated T cells in psoriasis vulgaris lesional skin: blood and skin comparisons by flow cytometry. J Autoimmun 2000;14:63–78.

Fuhlbrigge RC, Kieffer JD, Armerding D, et al. Cutaneous lymphocyte antigen is a specialized form of PSGL-1 expressed on skin-homing T cells. Nature 1997;389:978–81.

Gibson G. Rare and common variants: twenty arguments. Nat Rev Genet 2012;13:135–45.

Gilad Y, Rifkin SA, Pritchard JK. Revealing the architecture of gene regulation: the promise of eQTL studies. Trends Genet 2008;24:408–15.

Golan D, Lander ES, Rosset S. Measuring missing heritability: inferring the contribution of common variants. Proc Natl Acad Sci USA 2014;111:E5272–81.

Goldminz AM, Au SC, Kim N, et al. NF-kappaB: an essential transcription factor in psoriasis. J Dermatol Sci 2013;69:89–94.

Green RS, Stone EL, Tenno M, et al. Mammalian N-glycan branching protects against innate immune self-recognition and inflammation in autoimmune disease pathogenesis. Immunity 2007;27:308–20.

Grigorian A, Demetriou M. Mgat5 deficiency in T cells and experimental autoimmune encephalomyelitis. ISRN Neurol 2011:374314.

Gudjonsson JE, Elder JT. Psoriasis: epidemiology. Clin Dermatol 2007;25:535–46.

Gui H, Li M, Sham PC, et al. Comparisons of seven algorithms for pathway analysis using the WTCCC Crohn's Disease dataset. BMC Res Notes 2011;4:386.

Hahn MW, Kern AD. Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks. Mol Biol Evol 2005;22:803–6.

Hayashi K, Jutabha P, Endou H, et al. LAT1 is a critical transporter of essential amino acids for immune reactions in activated human T cells. J Immunol 2013;191:4080–5.

Hochberg Y, Benjamini Y. More powerful procedures for multiple significance testing. Stat Med 1990;9:811–8.

Iversen OJ, Lysvand H, Hagen L. The autoantigen Pso p27: a post-translational modification of SCCA molecules. Autoimmunity 2011;44:229–34.

Jaeger K, Paulsen F, Wohlrab J. Characterization of cationic amino acid transporters (hCATs) 1 and 2 in human skin. Histochem Cell Biol 2008;129:321–9.

Johnson RC, Nelson GW, Troyer JL, et al. Accounting for multiple comparisons in a genome-wide association study (GWAS). BMC Genomics 2010;11:724.

Joy MP, Brock A, Ingber DE, et al. High-betweenness proteins in the yeast protein interaction network. J Biomed Biotechnol 2005;2005:96–103.

Kanehisa M, Goto S. KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Res 2000;28:27–30.

Khananshvili D. The SLC8 gene family of sodium-calcium exchangers (NCX)—structure, function, and regulation in health and disease. Mol Aspects Med 2013;34:220–35.

Koster R, Mitra N, D'Andrea K, et al. Pathway-based analysis of GWAs data identifies association of sex determination genes with susceptibility to testicular germ cell tumors. Hum Mol Genet 2014;23:6061–8.

Kugler KG, Mueller LA, Graber A. MADAM: an open source meta-analysis toolbox for R and Bioconductor. Source Code Biol Med 2010;5:3.

Lang F, Stournaras C, Alesutan I. Regulation of transport across cell membranes by the serum- and glucocorticoid-inducible kinase SGK1. Mol Membr Biol 2014;31:29–36.

Lavin MF, Birrell G, Chen P, et al. ATM signaling and genomic stability in response to DNA damage. Mutat Res 2005;569:123–32.

Lizzul PF, Aphale A, Malaviya R, et al. Differential expression of phosphorylated NF-kappaB/RelA in normal and psoriatic epidermis and downregulation of NF-kappaB in response to treatment with etanercept. J Invest Dermatol 2005;124:1275–83.

Manolio TA. Genomewide association studies and assessment of the risk of disease. N Engl J Med 2010;363:166–76.

Martens JH, Stunnenberg HG. BLUEPRINT: mapping human blood cell epigenomes. Haematologica 2013;98:1487–9.

Mkhikian H, Grigorian A, Li CF, et al. Genetics and the environment converge to dysregulate N-glycosylation in multiple sclerosis. Nat Commun 2011;2:334.

Nair RP, Ruether A, Stuart PE, et al. Polymorphisms of the IL12B and IL23R genes are associated with psoriasis. J Invest Dermatol 2008;128:1653–61.

Nair RP, Stuart PE, Nistor I, et al. Sequence and haplotype analysis supports HLA-C as the psoriasis susceptibility 1 gene. Am J Hum Genet 2006;78:827–51.

Nestle FO, Kaplan DH, Barker J. Psoriasis. N Engl J Med 2009;361:496–509.

Park JH, Wacholder S, Gail MH, et al. Estimation of effect size distribution from genome-wide association studies and implications for future discoveries. Nat Genet 2010;42:570–5.

Purcell S, Neale B, Todd-Brown K, et al. PLINK: a tool set for whole-genome association and population-based linkage analyses. Am J Hum Genet 2007;81:559–75.

Ramanan VK, Shen L, Moore JH, et al. Pathway analysis of genomic data: concepts, methods, and prospects for future development. Trends Genet 2012;28:323–32.

Raney BJ, Cline MS, Rosenbloom KR, et al. ENCODE whole-genome data in the UCSC genome browser (2011 update). Nucleic Acids Res 2011;39:D871–5.

Roos WP, Kaina B. DNA damage-induced cell death by apoptosis. Trends Mol Med 2006;12:440–50.

Rudd PM, Elliott T, Cresswell P, et al. Glycosylation and the immune system. Science 2001;291:2370–6.

Ryan SO, Cobb BA. Roles for major histocompatibility complex glycosylation in immune function. Semin Immunopathol 2012;34:425–41.

Strange A, Capon F, Spencer CC, et al. A genome-wide association study identifies new psoriasis susceptibility loci and an interaction between HLA-C and ERAP1. Nat Genet 2010;42:985–90.

Takekoshi T, Wu X, Mitsui H, et al. CXCR4 negatively regulates keratinocyte proliferation in IL-23-mediated psoriasiform dermatitis. J Invest Dermatol 2013;133:2530–7.

Tsoi LC, Elder JT, Abecasis GR. Graphical algorithm for integration of genetic and biological data: proof of principle using psoriasis as a model. Bioinformatics 2015a;31:1243–9.

Tsoi LC, Spain SL, Ellinghaus E, et al. Enhanced meta-analysis and replication studies identify five new psoriasis susceptibility loci. Nat Commun 2015b;6:7001.

Tsoi LC, Spain SL, Knight J, et al. Identification of 15 new psoriasis susceptibility loci highlights the role of innate immunity. Nat Genet 2012;44:1341–8.

Tsuruta D. NF-kappaB links keratinocytes and lymphocytes in the pathogenesis of psoriasis. Recent Pat Inflamm Allergy Drug Discov 2009;3:40–8.

Vallabhajosyula RR, Chakravarti D, Lutfeali S, et al. Identifying hubs in protein interaction networks. PLoS One 2009;4:e5344.

Vig M, Kinet JP. Calcium signaling in immune cells. Nat Immunol 2009;10:21–7.

Vyse TJ, Todd JA. Genetic analysis of autoimmune disease. Cell 1996;85:311–8.

Wang K, Li M, Hakonarson H. Analysing biological pathways in genome-wide association studies. Nat Rev Genet 2010;11:843–54.

Weatherhead SC, Farr PM, Jamieson D, et al. Keratinocyte apoptosis in epidermal remodeling and clearance of psoriasis induced by UV radiation. J Invest Dermatol 2011;131:1916–26.

Wolf R, Orion E, Ruocco E, et al. Abnormal epidermal barrier in the pathogenesis of psoriasis. Clin Dermatol 2012;30:323–8.

Wu MC, Kraft P, Epstein MP, et al. Powerful SNP-set analysis for case-control genome-wide association studies. Am J Hum Genet 2010;86:929–42.

Yin X, Low HQ, Wang L, et al. Genome-wide meta-analysis identifies multiple novel associations and ethnic heterogeneity of psoriasis susceptibility. Nat Commun 2015;6:6916.

Yin X, Wineinger NE, Cheng H, et al. Common variants explain a large fraction of the variability in the liability to psoriasis in a Han Chinese population. BMC Genomics 2014;15:87.

Zgraggen S, Huggenberger R, Kerl K, et al. An important role of the SDF-1/CXCR4 axis in chronic skin inflammation. PLoS One 2014;9:e93665.

Zhang K, Cui S, Chang S, et al. i-GSEA4GWAS: a web server for identification of pathways/gene sets associated with traits by applying an improved gene set enrichment analysis to genome-wide association study. Nucleic Acids Res 2010;38:W90–5.

Zhong H, Yang X, Kaplan LM, et al. Integrating pathway analysis and genetics of gene expression for genome-wide association studies. Am J Hum Genet 2010;86:581–91.

Zhu KJ, Zhu CY, Shi G, et al. Meta-analysis of IL12B polymorphisms (rs3212227, rs6887695) with psoriasis and psoriatic arthritis. Rheumatol Int 2013;33:1785–90.

Zuo X, Sun L, Yin X, et al. Whole-exome SNP array identifies 15 new susceptibility loci for psoriasis. Nat Commun 2015;6:6793.

4) Julià, A., Blanco, F., Fernández-Gutierrez, B., González, A., Cañete, J. D., Maymó, J., … Codó, L., Gelpí, J. L. …, Marsal, S. (2016). Identification of IRX1 as a Risk Locus for Rheumatoid Factor Positivity in Rheumatoid Arthritis in a Genome-Wide Association Study. Arthritis & Rheumatology, 68(6), 1384–1391. https://doi.org/10.1002/art.39591

ARTHRITIS & RHEUMATOLOGY
Vol. 68, No. 6, June 2016, pp 1384–1391
DOI 10.1002/art.39591

Identification of *IRX1* as a Risk Locus for Rheumatoid Factor Positivity in Rheumatoid Arthritis in a Genome-Wide Association Study

Antonio Julià,[1] Francisco Blanco,[2] Benjamín Fernández-Gutierrez,[3] Antonio González,[4] Juan D. Cañete,[5] Joan Maymó,[6] Mercedes Alperi-López,[7] Alex Olivè,[8] Héctor Corominas,[9] Víctor Martínez-Taboada,[10] Isidoro González-Álvaro,[11] Antonio Fernandez-Nebro,[12] Alba Erra,[13] Simón Sánchez-Fernández,[14] Arnald Alonso,[1] María López-Lasanta,[1] Raül Tortosa,[1] Laia Codó,[15] Josep Lluis Gelpi,[15] Andrés C. García-Montero,[16] Jaume Bertranpetit,[17] Devin Absher,[18] Richard M. Myers,[18] Jesús Tornero,[19] and Sara Marsal[1]

Objective. **Rheumatoid factor (RF) is a well-established diagnostic and prognostic biomarker in rheumatoid arthritis (RA). However, ~20% of RA patients are negative for this anti-IgG antibody. To date, only variation at the HLA–DRB1 gene has been associated with the presence of RF. This study was undertaken to identify additional genetic variants associated with RF positivity.**

Methods. **A genome-wide association study (GWAS) for RF positivity was performed using an Illumina Quad610 genotyping platform. A total of 937 RF-positive and 323 RF-negative RA patients were genotyped for >550,000 single-nucleotide polymorphisms (SNPs). Association testing was performed using an allelic chi-square test implemented in Plink software. An independent cohort of 472 RF-positive and 190 RF-negative RA patients was used to validate the most significant findings.**

Results. **In the discovery stage, a SNP in the *IRX1* locus on chromosome 5p15.3 (SNP rs1502644) showed a genome-wide significant association with RF positivity ($P = 4.13 \times 10^{-8}$, odds ratio [OR] 0.37 [95% confidence interval (95% CI) 0.26–0.53]). In the validation stage, the association of *IRX1* with RF was replicated in an independent group of RA patients ($P = 0.034$, OR 0.58 [95% CI 0.35–0.97] and combined $P = 1.14 \times 10^{-8}$, OR 0.43 [95% CI 0.32–0.58]).**

Conclusion. **To our knowledge, this is the first GWAS of RF positivity in RA. Variation at the *IRX1* locus on chromosome 5p15.3 is associated with the presence of RF. Our findings indicate that *IRX1* and HLA–DRB1 are the strongest genetic factors for RF production in RA.**

Supported by the Spanish Ministry of Economy and Competitiveness (grants PSE-010000-2006-6 and IPT-010000-2010-36).

[1]Antonio Julià, PhD, Arnald Alonso, PhD, María López-Lasanta, MD, PhD, Raül Tortosa, PhD, Sara Marsal, MD, PhD: Vall d'Hebron Hospital Research Institute, Barcelona, Spain; [2]Francisco Blanco, MD, PhD: Instituto de Investigación Biomédica de A Coruña–Hospital Universitario A Coruña, A Coruña, Spain; [3]Benjamín Fernández-Gutierrez, MD, PhD: Hospital Clínico San Carlos, Madrid, Spain; [4]Antonio González, MD, PhD: Instituto de Investigación Sanitaria and Hospital Clinico Universitario de Santiago, Santiago de Compostela, Spain; [5]Juan D. Cañete, MD, PhD: Hospital Clínic de Barcelona, Barcelona, Spain; [6]Joan Maymó, MD, PhD: Hospital del Mar, Barcelona, Barcelona, Spain; [7]Mercedes Alperi-López, MD, PhD: Hospital Universitario Central de Asturias, Oviedo, Spain; [8]Alex Olivè, MD, PhD: Hospital Universitari Germans Trias i Pujol, Barcelona, Spain; [9]Héctor Corominas, MD, PhD: Hospital Moisès Broggi, Barcelona, Spain; [10]Víctor Martínez-Taboada, MD, PhD: Hospital Universitario Marqués de Valdecilla, Santander, Spain; [11]Isidoro González-Álvaro, MD, PhD: Hospital Universitario La Princesa, IIS Princesa, Madrid, Spain; [12]Antonio Fernandez-Nebro, MD, PhD: Instituto de Investigación Biomédica de Málaga, Hospital Regional Universitario de Málaga, and Universidad de Málaga, Málaga, Spain; [13]Alba Erra, MD, PhD: Hospital Sant Rafael, Barcelona, Spain; [14]Simón Sánchez-Fernández, MD: Hospital General La Mancha Centro, Ciudad Real, Spain; [15]Laia Codó, Josep Lluis Gelpi, PhD: Barcelona Supercomputing Center, Barcelona, Spain; [16]Andrés C. García-Montero, PhD: National DNA Bank Carlos III and University of Salamanca, Salamanca, Spain; [17]Jaume Bertranpetit, PhD: National Genotyping Center and Pompeu Fabra University, Barcelona, Spain; [18]Devin Absher, PhD, Richard M. Myers, PhD: HudsonAlpha Institute for Biotechnology, Huntsville, Alabama; [19]Jesús Tornero, MD, PhD: Hospital Universitario de Guadalajara, Guadalajara, Spain.

Address correspondence to Sara Marsal, MD, PhD, Rheumatology Research Group, Vall d'Hebron University Hospital, Passeig de la Vall d'Hebron, 119-129, 08035 Barcelona, Spain. E-mail: sara.marsal@vhir.org.

Submitted for publication June 26, 2015; accepted in revised form January 7, 2016.

Autoantibody generation by pathogenic B cells is one of the hallmark features of rheumatoid arthritis (RA) (1). Approximately 80% of RA patients are positive for rheumatoid factor (RF), and ~75% express anti–cyclic citrullinated protein antibodies (ACPAs). Although less specific than ACPAs, the presence of RF has been used to diagnose RA both in classic and in recently updated diagnostic criteria (2). Despite evidence of a genetic basis of this key trait (3), however, to date only variation at the HLA–DRB1 locus has been reproducibly associated with the presence of RF in RA (4).

RF is an antibody that targets the Fc portion of IgG. The ability to bind to other antibodies facilitated its identification in early studies investigating the high agglutination properties of serum from RA patients compared to that from controls (5). However, as was also noted very early on (6), not all RA patients are positive for this autoantibody. In addition, it was later shown that RF is not specific to RA; it is present in patients with other autoimmune and inflammatory diseases (7) and, although at a much lower frequency, in healthy individuals (8). Nonetheless, its association with key features of RA, such as disease severity (9–11) and treatment response (12,13), as well as the fact that its presence may precede the onset of the disease (14,15), have extended the relevance of this protein biomarker in the study and management of RA. Importantly, evidence of a direct pathogenic role of RF in RA etiology has recently emerged (16), renewing interest in the biologic mechanisms associated with this autoantibody.

In recent years, genome-wide association studies (GWAS) have drastically improved knowledge of the genetic basis of RA. To date, >100 genomic regions have been consistently associated with the risk of developing the disease (17). Of relevance, the loci showing stronger penetrance in RA risk are mainly associated with ACPA-positive RA (18,19). In ACPA-negative patients, a different set of risk loci is starting to be defined (20). Recently, GWAS directly comparing ACPA-positive patients to ACPA-negative patients have confirmed this genetic heterogeneity, suggesting its usefulness as a tool for classifying patients (21). To date, however, no GWAS has been performed to identify the genetic variation associated with the presence of RF in RA.

In the present study, we undertook for the first time a GWA analysis of RF positivity in RA. Using a discovery cohort of 1,260 patients and an additional validation cohort of 662 patients of Southern European ancestry, we sought to identify the genetic variation associated with the presence of this autoantibody in RA.

PATIENTS AND METHODS

A GWAS design was used to identify new genetic variants associated with RF positivity. In the discovery phase, a total of 1,260 patients with RA were recruited by the Immune-Mediated Inflammatory Disease Consortium (IMIDC) (22). The IMIDC is a biomedical research consortium that includes rheumatology departments from 15 Spanish university hospitals from different regions in Spain. All RA patients satisfied the American College of Rheumatology diagnostic criteria for RA (23) and were followed up for >2 years after diagnosis. Importantly, all patients had erosive disease, defined as ≥1 erosion in at least 2 joint groups in the hands and/or feet. All patients were Caucasian with all 4 grandparents born in Spain. The replication cohort was recruited following the same clinical and epidemiologic criteria as in the discovery phase. A total of 662 RA patients were recruited for the replication phase. The study was undertaken in compliance with the Declaration of Helsinki. Informed consent was obtained from all participants, and protocols were reviewed and approved by local institutional review boards.

Genome-wide genotyping of 937 RF-positive RA patients and 323 RF-negative RA patients was performed using Illumina Quad610 BeadChips. Quad610 arrays contain probes that genotype a total of 598,821 single-nucleotide polymorphisms (SNPs). GWA genotyping was performed at the National Genotyping Center. After excluding mitochondrial and X and Y chromosome SNPs, a total of 582,591 SNPs were genotyped for each patient in the discovery phase. SNP genotype calling was performed using Illumina GenomeStudio software version 2010.1. After excluding samples with <95% genotype completion rate (1.1%; n = 15), we selected SNPs with a >95% call rate (>99% of SNPs) and a minor allele frequency (MAF) of >0.05 (93.9% of SNPs). As an additional quality check, only those SNPs that showed Hardy-Weinberg equilibrium in samples of control individuals from the same ancestry and genotyped with the same platform (22) were included in the GWAS (99.5% of SNPs; $P > 0.0001$).

In order to exclude outlier individuals according to ancestry, we performed a principal components analysis (PCA) using EigenStrat software (24). In order to improve the estimation of the genetic variation of the Spanish population, we also included a previously described cohort of 1,493 healthy controls (22). Outlier patients were characterized as those individuals showing a clear deviation (i.e., >6 SD) in any of the 10 top principal components. A total of 35 outliers were identified and removed from the study (see Supplementary Figure 1, available on the *Arthritis & Rheumatology* web site at http://onlinelibrary.wiley.com/doi/10.1002/art.39591/abstract). The estimated genomic inflation was very close to 1 ($\lambda = 1.01$), indicating a very low probability of population stratification. Association analyses were nonetheless performed using logistic regression including the 10 first principal components, in order to exclude any potential confounding effect due to stratification. Association analyses were performed using Plink (version 1.07) (25). The statistical power of this study was estimated using Genetic Power Calculator software (26). Given the RF-positive and RF-negative sample sizes of the discovery stage, and assuming a MAF frequency of 0.2, the present study had 80% power to detect an effect size as low as an odds ratio (OR) of 1.7 at a significance level of $P = 5 \times 10^{-8}$ under a multiplicative model.

Table 1. RA susceptibility loci significantly associated with RF positivity*

Locus	SNP	Chr.	bp	Minor allele	Major allele	MAF in RF+ patients	MAF in RF− patients	Imputation quality score	*P*	OR (95% CI)
HLA–DRB1	rs9268839	6	32,428,772	G	A	0.61	0.52	1.00	0.00038	1.44 (1.19–1.75)
TEC	rs2664035	4	48,220,839	A	G	0.42	0.33	0.98	0.00088	1.43 (1.16–1.75)
ZNF438	rs793108	10	31,415,106	T	C	0.48	0.41	NA	0.0022	1.32 (1.09–1.6)
PLCL2	rs4452313	3	17,047,032	T	A	0.34	0.28	1.00	0.0047	1.33 (1.08–1.64)
CCR6	rs1571878	6	167,540,842	C	T	0.46	0.53	0.99	0.0079	0.75 (0.61–0.91)
ACOXL	rs6732565	2	111,607,832	G	A	0.37	0.31	1.00	0.013	1.31 (1.06–1.60)
IRF8	rs13330176	16	86,019,087	A	T	0.19	0.14	0.93	0.018	1.43 (1.05–1.94)
ARAP1	rs11605042	11	72,411,664	A	G	0.47	0.41	1.00	0.025	1.23 (1.02–1.49)
LOC100506023	rs2105325	1	173,349,725	A	C	0.21	0.26	1.00	0.027	0.76 (0.61–0.95)
CXCR5	rs10790268	11	118,729,391	A	G	0.22	0.28	0.93	0.034	0.73 (0.54–0.98)
RUNX1-LOC100506403	rs8133843	21	36,738,242	G	A	0.39	0.34	0.97	0.037	1.27 (1.02–1.58)

* Association results are listed for the single-nucleotide polymorphisms (SNPs) from the 10 established rheumatoid arthritis (RA) risk loci that show a significant ($P < 0.05$) association with rheumatoid factor (RF) positivity. SNPs are listed in order of the significance of the association. In all loci except *CCR6* and *ACOXL*, the risk allele is associated with the presence of RF. Chr. = chromosome; MAF = minor allele frequency; OR = odds ratio; 95% CI = 95% confidence interval; NA = not applicable (directly genotyped).

The association between established risk loci for RA and RF status was determined (17). A total of 100 SNPs, representing the strongest reported signal from each of the known RA risk loci, were tested for association with RF positivity. In those cases where the SNP was not genotyped in the Quad610 Illumina array, the marker genotype was imputed using SHAPEIT version 2 phasing software (27) and IMPUTE version 2 (28) imputation software. Data from the European cohort from the 1,000 Genomes Project was used as the reference panel for imputation (29).

Replication genotyping of 472 RF-positive patients and 190 RF-negative patients was performed at the HudsonAlpha Institute for Biotechnology (Huntsville, AL) using an Illumina GoldenGate assay. Five percent of the samples were genotyped in duplicate, giving an ~1% genotyping error rate.

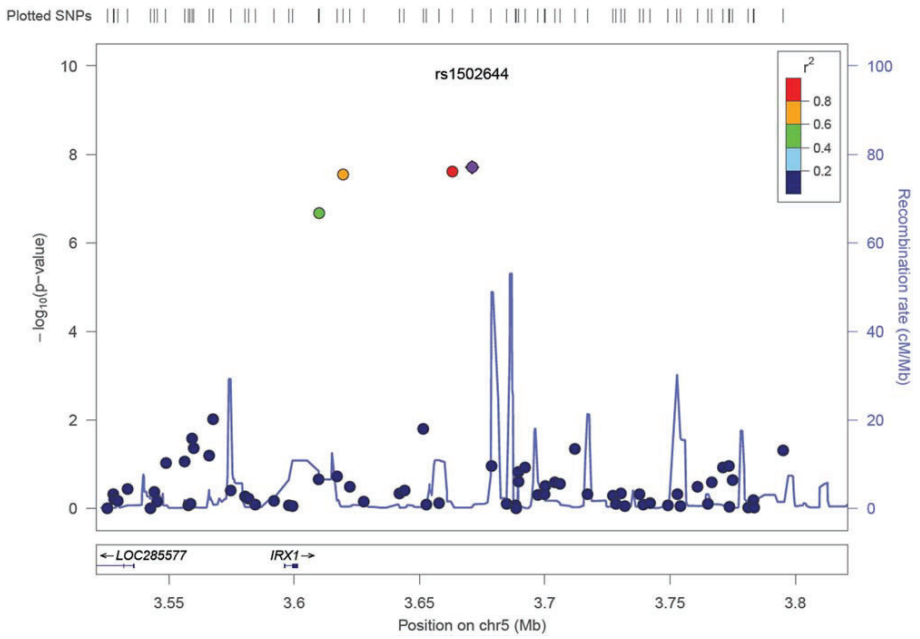


Figure 1. Association results for the *IRX1* locus with rheumatoid factor positivity in rheumatoid arthritis. A regional plot with the significance of the single-nucleotide polymorphisms (SNPs) in the *IRX1* locus, i.e., $-\log_{10}(P)$, as a function of location in chromosome 5p15.3 (basepair) is shown. The genome-wide significant SNP (rs1502644) is shown as a purple diamond, and the remaining SNPs are shown as circles, with color indicating the level of linkage disequilibrium (r^2) with rs1502644. The estimated recombination rates in this genomic region are plotted as a continuous background line (cM/Mb).

RESULTS

After quality-control analysis and PCA identification of genetic background outliers, 896 RF-positive RA patients and 282 RF-negative RA patients were included in the discovery stage. A total of 509,374 SNPs passed all quality and frequency filters and were used for GWA analysis.

In the discovery stage, the previously described significant association between the HLA–DRB1 locus and RF status was replicated ($P = 0.00038$; OR 1.44 [95% confidence interval (95% CI) 1.19–1.75]) (Table 1). Ten additional RA risk loci were found to be nominally associated with RF positivity ($P < 0.05$) (Table 1).

In the discovery stage, a SNP in chromosome 5p15.3 (rs1502644), a genomic region not previously associated with RA risk, showed a genome-wide significant association with RF positivity (MAF 0.051; $P = 4.13 \times 10^{-8}$) (OR 0.37 [95% CI 0.26–0.53]) (Figure 1). This genetic marker lies 69.5 kb downstream of the 3′ end of *IRX1*. No other genomic region, including the HLA region on chromosome 6, showed a highly significant association (defined as $P < 1 \times 10^{-6}$) with the presence of the autoantibody (Figure 2). Supplementary Table 1 (available on the *Arthritis & Rheumatology* web site at http://onlinelibrary.wiley.com/doi/10.1002/art.39591/abstract) lists the genomic regions showing moderate and high evidence of association with RF positivity ($P < 5 \times 10^{-5}$).

In the independent cohort of RA patients, we validated the association between the *IRX1* locus SNP rs1502644 and RF status, with the same direction of effect as in the discovery stage (replication $P = 0.034$, OR 0.58 [95% CI 0.35–0.97] and combined $P = 1.14 \times 10^{-8}$, OR 0.43 [95% CI 0.32–0.58]).

In order to functionally characterize *IRX1*, and in particular its association with B cell activity, we used a GenomicScape high-throughput data analyzer (30). GenomicScape is an online tool that allows rapid and easy characterization of the functionality of genes based on their expression in multiple human tissues available from large public repositories of gene-expression microarray studies. The GS-DT-1 microarray data set, which includes whole-genome expression profiling of human B cells during lymphopoiesis, was selected to analyze *IRX1* expression. This data set includes transcriptional data from 8 different differentiation states in B lymphocytes, from naive B cells to mature bone marrow plasma cells (31). Comparing the gene expression in different stages of lymphopoiesis, we found that *IRX1* is significantly overexpressed in bone marrow plasma cells compared to previous developmental stages ($P = 0.01$ by analysis of variance) (Figure 3).

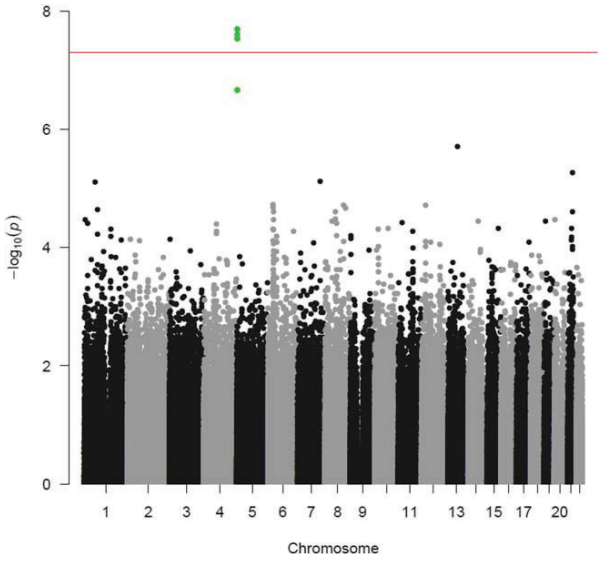


Figure 2. Genome-wide association results for rheumatoid factor (RF) positivity in rheumatoid arthritis. A Manhattan plot of the significance ($-\log_{10}[P]$) of the association of the 509,374 tested single-nucleotide polymorphisms (SNPs) with RF status is shown. Each dot corresponds to the significance of association of a SNP, and its position on the x-axis shows its position in the chromosome. Black and gray indicate SNPs from different chromosomes. Green indicates SNPs in the *IRX1* locus in chromosome 5p15.3, which show high levels of statistical significance. The horizontal line shows the threshold for significance at the genome-wide level (i.e., $P = 5 \times 10^{-8}$). Color figure can be viewed in the online issue, which is available at http://onlinelibrary.wiley.com/journal/doi/10.1002/art.39591/abstract.

DISCUSSION

RF is a key diagnostic and prognostic marker for RA. Recent studies have confirmed the initial hypothesis that RF participates directly in RA pathophysiology. To date, however, only variation at the HLA–DRB1 locus has been significantly associated with the presence of this autoantibody. In order to identify additional genetic risk factors, we performed the first GWAS of RF positivity. Using a large cohort of RA patients, we identified an association between the *IRX1* locus SNP rs1502644 at the genome-wide level of significance ($P < 5 \times 10^{-8}$). We replicated the association of *IRX1* with RF in RA in an independent cohort of patients. Additionally, 10 established RA risk loci showed a nominal level of association with RF positivity.

IRX1 encodes for a member of the Iroquois protein family. This group of genes has been associated with many developmental processes in vertebrates (32). Examination of the gene expression profile of *IRX1* in the different stages of B cell differentiation (31) showed that there is a clear increase in the expression of this

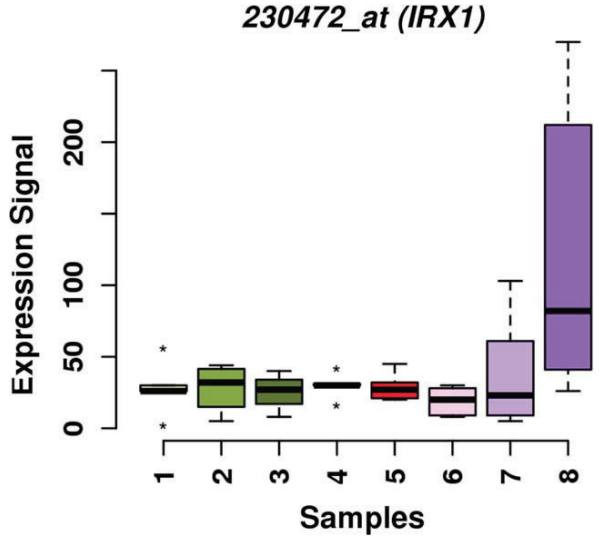


Figure 3. Gene expression levels of *IRX1* at different stages of B cell differentiation. Normalized gene expression levels of *IRX1* measured in human naive B cells (sample 1), centroblasts (sample 2), centrocytes (sample 3), memory B cells (sample 4), preplasmablasts (sample 5), plasmablasts (sample 6), early plasma cells (sample 7), and bone marrow plasma cells (sample 8) are shown. *IRX1* gene expression level is significantly increased in the latest stage of B cell differentiation ($P = 0.01$ versus all other cell types, by analysis of variance) when B cells are responsible for most long-term antibody production. Data are shown as box plots. Each box represents the upper and lower interquartile range (IQR). Lines inside the boxes represent the median. Whiskers represent the highest and lowest values. Asterisks indicate outliers (>1.5 times the upper and lower IQRs). Color figure can be viewed in the online issue, which is available at http://onlinelibrary.wiley.com/journal/doi/10.1002/art.39591/abstract.

gene in plasma cells compared to less-differentiated B cell subtypes. Bone marrow plasma cells are the end-stage differentiation of B lymphocytes and are responsible for the production of circulating antibodies. It is therefore possible that variants affecting the activity of plasma cells can also have an impact on the production of antibodies, including autoantibodies like RF. Of relevance, a recent GWAS of antibody levels generated by smallpox vaccination (33) identified a highly suggestive association signal at the *IRX1* locus ($P = 8.8 \times 10^{-7}$). Although the SNP identified in this study is likely to represent an independent association signal (linkage disequilibrium $r^2 = 0.07$ between the variants associated in the 2 studies), the presence of genetic associations for similar phenotypes supports the implication of this genomic region in the molecular mechanisms associated with antibody production.

Several RA risk loci, mainly in the HLA–DRB1 locus, have previously been shown to be associated with different clinical phenotypes in RA. Therefore, RA susceptibility loci also have a higher probability of being associated with RF positivity. In the present GWAS of RF, we validated the previously established association between RF positivity and HLA–DRB1. The estimated effect size of this main RA risk locus in RF positivity is, however, clearly lower than the effect size for *IRX1* (OR 1.44 for HLA–DRB1 versus OR 2.70 for *IRX1*), indicating that variation at the newly discovered locus could be more influential for the presence of this trait in RA. Additionally, 10 RA risk loci were also found to be associated with RF positivity at the nominal level ($P < 0.05$). In 8 of these loci—*TEC*, *ZNF438*, *PLCL2*, *LOC100506023, IRF8, RUNX1-LOC100506403, ARAP1,* and *CXCR5*—the allele associated with RA risk was also associated with RF positivity. In *CCR6* and *ACOXL*, however, the risk allele for RA was associated with RF-negative RA. To our knowledge, this is the first time that these RA risk loci have also been shown to be associated with the presence of RF. Of interest, several of these genes, such as *IRF8* (34), *TEC* (35), *CXCR5* (36), *CCR6* (37), and *PLCL2* (38), have been found to be directly associated with different key aspects of B cell functionality. These results further strengthen the hypothesis that genetic variation at B cell pathways influences the risk of expressing RF in RA.

In RA, there is a significant correlation between ACPA positivity and RF positivity. Despite this, there is a large fraction of patients positive for only 1 autoantibody (39,40), suggesting the presence of independent genetic factors. In the discovery stage of this study, we controlled for ACPA positivity and found that the association of *IRX1* with RF was still highly significant ($P = 0.0006$) (data not shown). In contrast, when testing for an association of *IRX1* SNP rs1502644 with ACPA positivity, controlling for RF, we found no evidence of association ($P > 0.1$) (data not shown). Our results therefore support the notion that variation at the *IRX1* locus is associated specifically with the presence of RF in RA. Additionally, we evaluated the association of loci recently found to be associated with ACPA positivity (41) in our discovery data set. Of all loci showing the most significant association with ACPA in the previous study (41) (after directly comparing autoantibody-negative positive patients to autoantibody-negative patients, as in the present study), only the HLA–DRB1 locus showed a significant association with RF positivity in the present study (data not shown). This result confirms the relevance of HLA region variation to autoantibody positivity, and suggests that non-HLA genetic variation associated with

ACPA positivity is independent of the variation associated with RF positivity.

Identification of the genomic variation that influences the expression of the autoantibodies RF and ACPA has been a challenge for studies of RA genetics. Before GWA techniques were available, family-based linkage studies attempted to identify the genomic regions associated with the presence of RF. In particular, the North American RA Consortium collection of 491 multiplex RA families was analyzed by multiple independent studies to try to identify genomic regions associated with RF expression. Similar to other complex traits, however, the linkage signals obtained from different studies were low and had little overlap, with the exception of the short arm of chromosome 6, which harbors the HLA region (42–44). Interestingly, in the original study where this multicase RA family cohort was recruited (42), one of the strongest linkage signals for RF titer was found on chromosome 5p15.2. *IRX1* is located in chromosome 5p15.33, and despite being more than 7.9 Mb away, it is possible that this original linkage peak could have captured part of the genetic association with RF production (45). However, the relatively low statistical evidence (logarithm of odds score 1.21) and the lack of replication in other linkage analyses might have precluded an in-depth analysis of the association of this chromosomal region with RF levels.

More recently, a GWAS of a large cohort of healthy individuals from a Japanese population evaluated the GWA of ACPA and RF positivity (46). In an analysis of >3,000 individuals, only ~6% of healthy individuals were found to be positive for RF, thereby greatly reducing the power to identify genetic variants associated with this trait. Consequently, no genetic variant surpassed the genome-wide level of significance in that study.

Identifying genetic variation associated with a specific trait in a heterogeneous disease like RA can be challenging. In order to increase the statistical power of such studies, reducing patient heterogeneity is considered a useful approach (47). In the present study, all recruited RA patients had erosive disease and were therefore homogeneous for this key pathologic feature. There is evidence that RF-negative patients have a lower likelihood of developing joint erosions (48). Thus, the inclusion of patients with nonerosive disease would have led to a higher proportion of such patients in the RF-negative group. Additional studies are needed to show how the inclusion of patients with no evidence of joint damage influences the association of the *IRX1* locus with RF positivity.

Recently, evidence has been found to support the notion of a direct role of RF in RA pathophysiology (16). Rather than being a mere surrogate of the chronic inflammation in RA, RF could directly participate in the amplification of the inflammation in the synovial membrane. In this disease model, immune complexes accumulating in the synovial joint and formed mainly by ACPAs would be recognized by macrophages via RF, leading to an increase in the production of proinflammatory cytokines, including tumor necrosis factor. Consequently, factors influencing the expression of RF could affect both the risk of developing RA and the risk of developing more severe forms of the disease. In the present study, *IRX1* SNP rs1502644 allele frequency was found to be significantly different from healthy controls in both the RF-positive and the RF-negative RA samples (MAF 0.10 for RF-negative RA patients, MAF 0.040 for RF-positive RA patients, and MAF 0.053 for controls; $P < 0.05$) (data not shown), suggesting the possibility of a pleiotropic effect of this locus in RA (17,49). Additional studies evaluating the influence of the *IRX1* locus with clinically relevant phenotypes, such as disease severity or response to anti-TNF therapy in RA, are therefore warranted.

In this study, we performed the first GWAS of RF positivity in RA and identified *IRX1* as a risk locus for this trait. We also showed that several established RA risk loci are associated with RF status. These findings are an important step in the characterization of the genetic basis of RF positivity in RA.

AUTHOR CONTRIBUTIONS

All authors were involved in drafting the article or revising it critically for important intellectual content, and all authors approved the final version to be published. Dr. Marsal had full access to all of the data in the study and takes responsibility for the integrity of the data and the accuracy of the data analysis.

Study conception and design. Julià, Blanco, Fernández-Gutierrez, Tornero, Marsal.

Acquisition of data. Blanco, Fernández-Gutierrez, González, Cañete, Maymó, Alperi-López, Olivè, Corominas, Martínez-Taboada, González-Álvaro, Fernandez-Nebro, Erra, Sánchez-Fernández, López-Lasanta, Tortosa, García-Montero, Bertranpetit, Absher, Myers, Tornero, Marsal.

Analysis and interpretation of data. Julià, González, Cañete, Alonso, López-Lasanta, Codó, Gelpi, Absher, Myers, Marsal.

REFERENCES

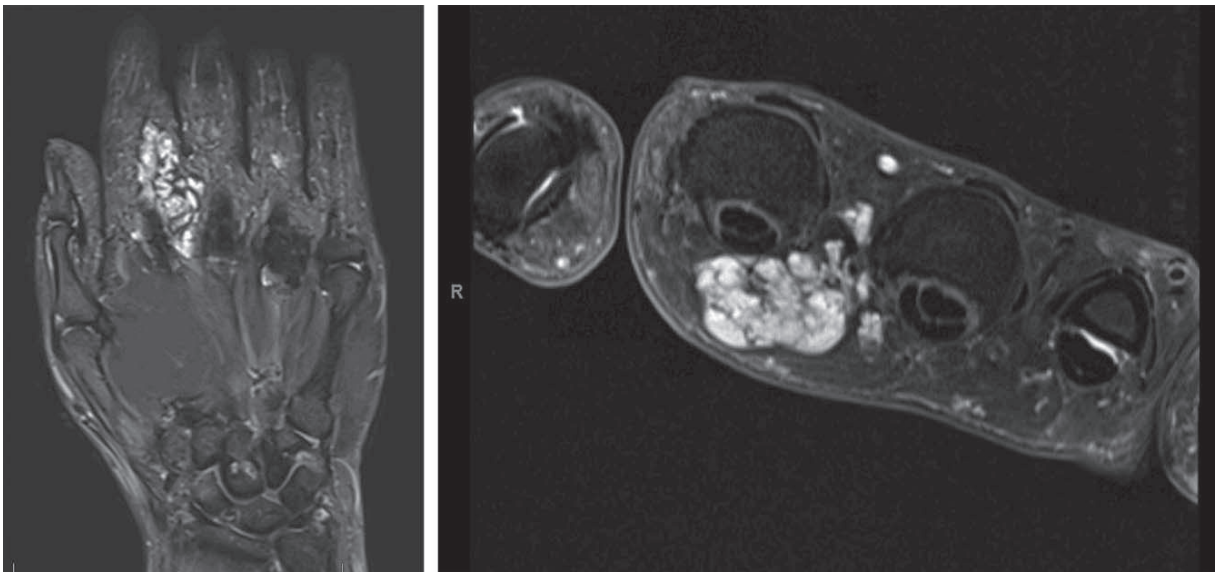
1. Firestein GS. Evolving concepts of rheumatoid arthritis. Nature 2003;423:356–61.
2. Arnett FC, Edworthy SM, Bloch DA, McShane DJ, Fries JF, Cooper NS, et al. The American Rheumatism Association 1987 revised criteria for the classification of rheumatoid arthritis. Arthritis Rheum 1988;31:315–24.
3. MacGregor AJ, Ollier WE, Venkovsky J, Mageed RA, Carthy D, Silman AJ. Rheumatoid factor isotypes in monozygotic and dizygotic twins discordant for rheumatoid arthritis. J Rheumatol 1995;22:2203–7.
4. Olsen NJ, Callahan LF, Brooks RH, Nance EP, Kaye JJ, Stastny P, et al. Associations of HLA-DR4 with rheumatoid factor and

radiographic severity in rheumatoid arthritis. Am J Med 1988; 84:257–64.
5. Rose HM, Ragan C, Pearce E, Lipman MO. Differential agglutination of normal and sensitized sheep erythrocytes by sera of patients with rheumatoid arthritis. Proc Soc Exp Biol Med 1948; 68:1–6.
6. Waaler E. On the occurrence of a factor in human serum activating the specific agglutination of sheep blood corpuscles. Acta Pathol Microbiol Scand 1940;17:172–88.
7. Dorner T, Egerer K, Feist E, Burmester GR. Rheumatoid factor revisited. Curr Opin Rheumatol 2004;16:246–53.
8. Newkirk MM. Rheumatoid factors: host resistance or autoimmunity? Clin Immunol 2002;104:1–13.
9. Masi AT, Maldonado-Cocco JA, Kaplan SB, Feigenbaum SL, Chandler RW. Prospective study of the early course of rheumatoid arthritis in young adults: comparison of patients with and without rheumatoid factor positivity at entry and identification of variables correlating with outcome. Semin Arthritis Rheum 1976;4:299–326.
10. Goronzy JJ, Matteson EL, Fulbright JW, Warrington KJ, Chang-Miller A, Hunder GG, et al. Prognostic markers of radiographic progression in early rheumatoid arthritis. Arthritis Rheum 2004;50:43–54.
11. Bukhari M, Lunt M, Harrison BJ, Scott DG, Symmons DP, Silman AJ. Rheumatoid factor is the major predictor of increasing severity of radiographic erosions in rheumatoid arthritis: results from the Norfolk Arthritis Register Study, a large inception cohort. Arthritis Rheum 2002;46:906–12.
12. Quartuccio L, Fabris M, Salvin S, Atzeni F, Saracco M, Benucci M, et al. Rheumatoid factor positivity rather than anti-CCP positivity, a lower disability and a lower number of anti-TNF agents failed are associated with response to rituximab in rheumatoid arthritis. Rheumatology (Oxford) 2009;48:1557–9.
13. Klaasen R, Cantaert T, Wijbrandts CA, Teitsma C, Gerlag DM, Out TA, et al. The value of rheumatoid factor and anti-citrullinated protein antibodies as predictors of response to infliximab in rheumatoid arthritis: an exploratory study. Rheumatology (Oxford) 2011;50:1487–93.
14. Nielen MM, van Schaardenburg D, Reesink HW, van de Stadt RJ, van der Horst-Bruinsma IE, de Koning MH, et al. Specific autoantibodies precede the symptoms of rheumatoid arthritis: a study of serial measurements in blood donors. Arthritis Rheum 2004;50:380–6.
15. Rantapaa-Dahlqvist S, de Jong BA, Berglin E, Hallmans G, Wadell G, Stenlund H, et al. Antibodies against cyclic citrullinated peptide and IgA rheumatoid factor predict the development of rheumatoid arthritis. Arthritis Rheum 2003;48:2741–9.
16. Laurent L, Anquetil F, Clavel C, Ndongo-Thiam N, Offer G, Miossec P, et al. IgM rheumatoid factor amplifies the inflammatory response of macrophages induced by the rheumatoid arthritis-specific immune complexes containing anticitrullinated protein antibodies. Ann Rheum Dis 2015;74:1425–31.
17. Okada Y, Wu D, Trynka G, Raj T, Terao C, Ikari K, et al. Genetics of rheumatoid arthritis contributes to biology and drug discovery. Nature 2014;506:376–81.
18. Lundstrom E, Kallberg H, Alfredsson L, Klareskog L, Padyukov L. Gene–environment interaction between the DRB1 shared epitope and smoking in the risk of anti–citrullinated protein antibody–positive rheumatoid arthritis: all alleles are important. Arthritis Rheum 2009;60:1597–603.
19. Padyukov L, Seielstad M, Ong RT, Ding B, Ronnelid J, Seddighzadeh M, et al. A genome-wide association study suggests contrasting associations in ACPA-positive versus ACPA-negative rheumatoid arthritis. Ann Rheum Dis 2011;70:259–65.
20. Klareskog L, Catrina AI, Paget S. Rheumatoid arthritis. Lancet 2009;373:659–72.
21. Bossini-Castillo L, de Kovel C, Kallberg H, van 't Slot R, Italiaander A, Coenen M, et al. A genome-wide association study of rheumatoid arthritis without antibodies against citrullinated peptides. Ann Rheum Dis 2015;74:e15.
22. Julia A, Domenech E, Chaparro M, Garcia-Sanchez V, Gomollon F, Panes J, et al. A genome-wide association study identifies a novel locus at 6q22.1 associated with ulcerative colitis. Hum Mol Genet 2014;23:6927–34.
23. Arnett FC. Revised criteria for the classification of rheumatoid arthritis. Bull Rheum Dis 1989;38:1–6.
24. Price AL, Patterson NJ, Plenge RM, Weinblatt ME, Shadick NA, Reich D. Principal components analysis corrects for stratification in genome-wide association studies. Nat Genet 2006;38: 904–9.
25. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira MA, Bender D, et al. PLINK: a toolset for whole-genome association and population-based linkage analyses. Am J Hum Genet 2007; 81:559–75.
26. Purcell S, Cherny SS, Sham PC. Genetic Power Calculator: design of linkage and association genetic mapping studies of complex traits. Bioinformatics 2003;19:149–50.
27. Delaneau O, Zagury JF, Marchini J. Improved whole-chromosome phasing for disease and population genetic studies. Nat Methods 2013;10:5–6.
28. Howie B, Marchini J, Stephens M. Genotype imputation with thousands of genomes. G3 (Bethesda) 2011;1:457–70.
29. Abecasis GR, Auton A, Brooks LD, DePristo MA, Durbin RM, Handsaker RE, et al. An integrated map of genetic variation from 1,092 human genomes. Nature 2012;491:56–65.
30. Kassambara A, Reme T, Jourdan M, Fest T, Hose D, Tarte K, et al. GenomicScape: an easy-to-use web tool for gene expression data analysis: application to investigate the molecular events in the differentiation of B cells into plasma cells. PLoS Comput Biol 2015;11:e1004077.
31. Jourdan M, Cren M, Robert N, Bollore K, Fest T, Duperray C, et al. IL-6 supports the generation of human long-lived plasma cells in combination with either APRIL or stromal cell-soluble factors. Leukemia 2014;28:1647–56.
32. Gomez-Skarmeta JL, Modolell J. Iroquois genes: genomic organization and function in vertebrate neural development. Curr Opin Genet Dev 2002;12:403–8.
33. Ovsyannikova IG, Kennedy RB, O'Byrne M, Jacobson RM, Pankratz VS, Poland GA. Genome-wide association study of antibody response to smallpox vaccine. Vaccine 2012;30: 4182–9.
34. Carotta S, Willis SN, Hasbold J, Inouye M, Pang SH, Emslie D, et al. The transcription factors IRF8 and PU.1 negatively regulate plasma cell differentiation. J Exp Med 2014;211: 2169–81.
35. Kitanaka A, Mano H, Conley ME, Campana D. Expression and activation of the nonreceptor tyrosine kinase Tec in human B cells. Blood 1998;91:940–8.
36. Legler DF, Loetscher M, Roos RS, Clark-Lewis I, Baggiolini M, Moser B. B cell-attracting chemokine 1, a human CXC chemokine expressed in lymphoid tissues, selectively attracts B lymphocytes via BLR1/CXCR5. J Exp Med 1998;187:655–60.
37. Elgueta R, Marks E, Nowak E, Menezes S, Benson M, Raman VS, et al. CCR6-dependent positioning of memory B cells is essential for their ability to mount a recall response to antigen. J Immunol 2015;194:505–13.
38. Takenaka K, Fukami K, Otsuki M, Nakamura Y, Kataoka Y, Wada M, et al. Role of phospholipase C-L2, a novel phospholipase C-like protein that lacks lipase activity, in B-cell receptor signaling. Mol Cell Biol 2003;23:7329–38.
39. Hecht C, Englbrecht M, Rech J, Schmidt S, Araujo E, Engelke K, et al. Additive effect of anti-citrullinated protein antibodies and rheumatoid factor on bone erosions in patients with RA. Ann Rheum Dis 2015:74:2151–6.
40. Van Steenbergen HW, Ajeganova S, Forslind K, Svensson B, van der Helm-van Mil AH. The effects of rheumatoid factor and

anticitrullinated peptide antibodies on bone erosions in rheumatoid arthritis. Ann Rheum Dis 2015;74:e3.
41. Padyukov L, Seielstad M, Ong RT, Ding B, Ronnelid J, Seddighzadeh M, et al. A genome-wide association study suggests contrasting associations in ACPA-positive versus ACPA-negative rheumatoid arthritis. Ann Rheum Dis 2011;70:259–65.
42. Criswell LA, Chen WV, Jawaheer D, Lum RF, Wener MH, Gu X, et al. Dissecting the heterogeneity of rheumatoid arthritis through linkage analysis of quantitative traits. Arthritis Rheum 2007;56:58–68.
43. Mukhopadhyay N, Halder I, Bhattacharjee S, Weeks DE. Two-dimensional linkage analyses of rheumatoid arthritis. BMC Proc 2007;1 Suppl 1:S68.
44. Oh C. A Bayesian genome-wide linkage analysis of quantitative traits for rheumatoid arthritis via perfect sampling. BMC Proc 2007;1 Suppl 1:S110.
45. Hirschhorn JN, Daly MJ. Genome-wide association studies for common diseases and complex traits. Nat Rev Genet 2005;6:95–108.
46. Terao C, Ohmura K, Ikari K, Kawaguchi T, Takahashi M, Setoh K, et al, on behalf of the Nagahama Study Group. Effects of smoking and shared epitope on the production of anti–citrullinated peptide antibody in a Japanese adult population. Arthritis Care Res (Hoboken) 2014;66:1818–27.
47. McCarthy MI, Abecasis GR, Cardon LR, Goldstein DB, Little J, Ioannidis JP, et al. Genome-wide association studies for complex traits: consensus, uncertainty and challenges. Nat Rev Genet 2008;9:356–69.
48. Bukhari M, Lunt M, Harrison BJ, Scott DG, Symmons DP, Silman AJ. Rheumatoid factor is the major predictor of increasing severity of radiographic erosions in rheumatoid arthritis: results from the Norfolk Arthritis Register Study, a large inception cohort. Arthritis Rheum 2002;46:906–12.
49. Solovieff N, Cotsapas C, Lee PH, Purcell SM, Smoller JW. Pleiotropy in complex traits: challenges and strategies. Nat Rev Genet 2013;14:483–95.

DOI: 10.1002/art.39610

Clinical Images: A digital vascular malformation masquerading as synovitis



The patient, a 49-year-old woman with a history of seronegative rheumatoid arthritis (RA), presented with swelling of her right index finger, which was most marked around the metacarpophalangeal joint and proximal phalanx. The swelling was warm and boggy. After previous unsuccessful treatment with hydroxychloroquine, the patient had been taking oral methotrexate 15 mg/week for the past 2 years. The remainder of her joints were not swollen or tender, and blood tests revealed normal levels of inflammation markers. There was no history of trauma, and the lesion had developed progressively over the past 9 months. An edema-sensitive coronal magnetic resonance imaging sequence (fat-suppressed, proton density) showed a multilobulated, high-signal, infiltrative lesion within the palmar subcutaneous tissues extending toward the second web space (left). An axial fat-suppressed (T1) postcontrast image again demonstrated an infiltrative, heterogeneously enhancing lesion abutting the second flexor tendon sheath, consistent with a low-flow venous malformation (right). No evidence of synovitis was noted, confirming the clinical impression of RA in remission. This case serves as a reminder to consider in a timely manner other pathologies in patients with RA who present with isolated RA-like symptoms.

Douglas H. N. White, FRACP
Waikato DHB
and University of Auckland
Glenn Coltman, MBChB
Waikato Hospital
Hamilton, New Zealand

5) Julià, A., González, I., Fernández-Nebro, A., Blanco, F., Rodriguez, L., González, A., … Codó, L., Gelpí, J. L. …, Marsal, S. (2016). A genome-wide association study identifies SLC8A3 as a susceptibility locus for ACPA-positive rheumatoid arthritis. Rheumatology, 55(6), 1106–1111. https://doi.org/10.1093/rheumatology/kew035

RHEUMATOLOGY

Rheumatology 2016;55:1106–1111
doi:10.1093/rheumatology/kew035
Advance Access publication 15 March 2016

Concise report

A genome-wide association study identifies *SLC8A3* as a susceptibility locus for ACPA-positive rheumatoid arthritis

Antonio Julià[1], Isidoro González[2], Antonio Fernández-Nebro[3], Francisco Blanco[4], Luis Rodriguez[5], Antonio González[6], Juan D. Cañete[7], Joan Maymó[8], Mercedes Alperi-López[9], Alejandro Olivé[10], Héctor Corominas[11], Víctor Martínez-Taboada[12], Alba Erra[13], Simón Sánchez-Fernández[14], , Arnald Alonso[1], Maria Lopez-Lasanta[1], Raül Tortosa[1], Laia Codó[15,16], Josep Lluis Gelpi[15,16], Andres C. García-Montero[17], Jaume Bertranpetit[18], Devin Absher[19], S. Louis Bridges Jr[20], Richard M. Myers[21], Jesus Tornero[22] and Sara Marsal[1]

Abstract

Objective. RA patients with serum ACPA have a strong and specific genetic background. The objective of the study was to identify new susceptibility genes for ACPA-positive RA using a genome-wide association approach.

Methods. A total of 924 ACPA-positive RA patients with joint damage in hands and/or feet, and 1524 healthy controls were genotyped in 582 591 single-nucleotide polymorphisms (SNPs) in the discovery phase. In the validation phase, the most significant SNPs in the genome-wide association study representing new candidate loci for RA were tested in an independent cohort of 863 ACPA-positive patients with joint damage and 1152 healthy controls. All individuals from the discovery and validation cohorts were Caucasian and of Southern European ancestry.

Results. In the discovery phase, 60 loci not previously associated with RA risk showed evidence for association at $P < 5 \times 10^{-4}$ and were tested for replication in the validation cohort. A total of 12 loci were replicated at the nominal level ($P < 0.05$, same direction of effect as in the discovery phase). When combining the discovery and validation cohorts, an intronic SNP in the Solute Carrier family 8 gene (*SLC8A3*) was found to be associated with ACPA-positive RA at a genome-wide level of significance RA [odds ratio (95% CI): 1.42 (1.25, 1.6), $P_{combined} = 3.19 \times 10^{-8}$].

Conclusions. *SLC8A3* was identified as a new risk locus for ACPA-positive RA. This study demonstrates the advantage of analysing relevant subsets of RA patients to identify new genetic risk variants.

[1]Vall d'Hebron Hospital Research Institute, Rheumatology Research Group, Barcelona, [2]Rheumatology Department, Hospital Universitario La Princesa. IIS La Princesa, Madrid, [3]UGC Reumatología, Instituto de Investigación Biomédica de Málaga (IBIMA), Hospital Regional Universitario de Málaga, Universidad de Málaga, Málaga, [4]Rheumatology Department, INIBIC-Hospital Universitario A Coruña, A Coruña, [5]Rheumatology Department, Hospital Clínico San Carlos, Madrid, Madrid, [6]Instituto de Investigación Sanitaria-Hospital Clínico Universitario de Santiago, Rheumatology Unit, Santiago de Compostela, [7]Rheumatology Department, Hospital Clínic de Barcelona, Barcelona, [8]Rheumatology Department, Hospital del Mar, Barcelona, Barcelona, [9]Rheumatology Department, Hospital Universitario Central de Asturias, Oviedo, [10]Rheumatology Department, Hospital Universitari Germans Trias i Pujol, [11]Rheumatology Department, Hospital Moisès Broggi, Barcelona, [12]Rheumatology Department, Hospital Universitario Marqués de Valdecilla, Cantabria, [13]Rheumatology Department, Hospital Sant Rafael, Barcelona, [14]Rheumatology Department, Hospital General La Mancha Centro, Ciudad Real, [15]Life Sciences, Barcelona Supercomputing Centre, [16]Department of Biochemistry and Molecular Biology, University of Barcelona, Barcelona, [17]Banco Nacional de ADN Carlos III, University of Salamanca, Salamanca, [18]Nacional Genotyping Centre (CeGen), Universitat Pompeu Fabra, Barcelona, Spain, [19]Hudson Alpha Institute for Biotechnology, Abshers lab, Huntsville, [20]Division of Clinical Immunology and Rheumatology, University of Alabama at Birmingham, Birmingham, [21]Hudson Alpha Institute for Biotechnology, Myers lab, Huntsville, AL, USA and [22]Rheumatology Department, Hospital Universitario De Guadalajara, Guadalajara, Spain

Submitted 19 June 2015; revised version accepted 4 February 2016

Correspondence to: Sara Marsal, Grup de Recerca de Reumatologia, Vall d' Hebron Hospital Research Institute, Pg Vall Hebron 119-129, 08035, Barcelona, Spain. E-mail: sara.marsal@vhir.org

CLINICAL SCIENCE

Key words: rheumatoid arthritis, anti-citrullinated protein antibodies, joint erosions, genetic risk, genome-wide association study

Rheumatology key messages

- ACPA-positive RA has a specific genetic risk background.
- Reducing patient heterogeneity is a powerful approach to identify new risk loci for RA .
- *SLC8A3* is a new risk locus for ACPA-positive and erosive RA.

Introduction

RA is the most common inflammatory arthritis in the western world, and it develops on the background of a complex genetic susceptibility. Genome wide association studies (GWAS) have radically improved our knowledge of the genetic variability associated with the risk of developing RA. To date more than 100 different loci have been associated with RA at the genome-wide level of statistical significance [1]. However, these new risk loci collectively explain $<$10% of the heritability of RA. Therefore, the likelihood that additional, undiscovered loci contribute to RA risk is very high.

One major advance in the understanding of the genetic basis of RA has been the identification of a differential genetic background between ACPA-positive patients and ACPA-negative patients [2]. In ACPA-positive RA, a larger number of genes influence the risk of developing the disease and, also, they show a stronger penetrance compared with ACPA-negative patients [3]. The most compelling example of this differential genetic component is that the two loci most strongly associated with RA, *HLA-DRB1* and *PTPN22* genes, are essentially not associated with ACPA-negative RA [4]. Integrating this acquired knowledge into the study of RA genetic aetiology is proving to be a powerful strategy to identify additional risk factors [5].

Although joint destruction is the hallmark of RA and is the focus of many actual therapeutic interventions, it is not present in all patients [6]. This variability in the clinical presentation of the disease might reflect the presence of underlying genetic heterogeneity. Therefore, analysing only RA patients with radiographic joint damage represents a useful strategy to reduce heterogeneity and so increase the power to identify new genetic factors associated with the disease. In the present study we performed a case-control GWAS using ACPA-positive RA patients with radiographic joint damage in hands and/or feet. The most significant results from the GWAS that were suggestive of new risk loci for RA were subsequently corroborated using an independent validation cohort of ACPA-positive RA patients, also with radiographic joint damage. The results of this study show that analysing relevant groups of RA patients can help to identify additional genetic variation associated with disease risk.

Methods

Study subjects

In the discovery phase, a total 924 RA patients were recruited by the Immune-Mediated Inflammatory Disease Consortium [7]. All RA patients satisfied the ACR diagnostic criteria for RA, were ACPA-positive and had $>$2 years of follow-up since diagnosis. Importantly, all patients had erosive disease defined as $\geqslant$1 erosions in, at least, two joint groups in hands and/or feet. All patients were Caucasian European and with all four grandparents born in Spain. Supplementary Table S1, available at *Rheumatology* Online, describes the main characteristics of the RA patient cohorts in this study.

The control cohort was also collected by the Immune-Mediated Inflammatory Disease Consortium, in collaboration with the Spanish National DNA biobank [7]. All healthy control individuals were $>$18 years old and without an autoimmune disease. In order to increase the power of the study, control individuals with a first or second degree relative affected with an autoimmune disease were excluded from the study. A total of 1524 healthy control individuals were recruited for analysis in the discovery phase. All controls were also Caucasian and with all four grandparents born in Spain.

The replication cohort was recruited following the same criteria as in the discovery phase. Anti-CCP-positive patients and healthy controls were all Caucasian and with all grandparents born in Spain. A total of 863 ACPA-positive patients with erosive disease and 1152 controls were recruited for the replication phase.

This study was undertaken in compliance with the Declaration of Helsinki. Informed consent was obtained from all participants, and both the protocols and study were reviewed and approved by the Vall d'Hebron University Hospital review board.

GWAS

Genome-wide genotyping was performed using Illumina Quad610 Beadchips (Illumina, San Diego, CA, USA) on 924 ACPA-positive RA patients and 1524 healthy controls. The Quad610 arrays genotype more than 550 000 single nucleotide polymorphisms (SNPs). GWAS genotyping was performed at the Centro Nacional de Genotipado (CeGen, Spain). Details on the quality control procedure used in this stage are described in 'GWAS and Replication Quality Control Procedures' in supplementary Fig. S1, available at *Rheumatology* Online.

Table 1 Association results for the new loci for ACPA-positive RA identified in our study

				GWAS		Validation		Combined	
Chr	SNP	Gene	MAF	P	OR (95%CI)	P	OR (95% CI)	P	OR (95% CI)
2	rs6435818	*SPAG16*	0.15	0.00044	1.36 (1.15, 1.62)	0.024	1.21 (1.01, 1.45)	6.76×10^{-5}	1.29 (1.14, 1.46)
3	rs2664122	*SRGAP3*	0.4	7.82×10^{-5}	1.28 (1.13, 1.44)	0.017	1.16 (1.01, 1.32)	1.91×10^{-5}	1.21 (1.11, 1.33)
3	rs807193	*CACNA1D*	0.22	1.85×10^{-5}	1.38 (1.19, 1.59)	0.043	1.15 (0.98, 1.34)	1.11×10^{-5}	1.27 (1.14, 1.41)
4	rs10517086	*LOC645481*	0.41	3.67×10^{-5}	1.29 (1.14, 1.46)	0.0044	1.19 (1.05, 1.36)	1.44×10^{-6}	1.24 (1.14, 1.36)
5	rs1991493	*EBF1*	0.21	0.00018	0.77 (0.67, 0.88)	0.013	0.84 (0.73, 0.98)	1.67×10^{-5}	0.8 (0.72, 0.89)
7	rs11496005	*AGR3*	0.35	1.43×10^{-5}	1.32 (1.17, 1.5)	0.027	1.15 (1, 1.31)	6.07×10^{-6}	1.24 (1.13, 1.36)
8	rs870615	*SGCZ*	0.28	7.41×10^{-5}	1.31 (1.15, 1.5)	0.045	1.13 (0.98, 1.31)	3.79×10^{-5}	1.23 (1.11, 1.35)
9	rs11788776	*BNC2*	0.5	7.65×10^{-5}	1.27 (1.13, 1.43)	0.048	1.11 (0.98, 1.26)	4.30×10^{-5}	1.2 (1.1, 1.3)
12	rs789331	*C12orf28*	0.38	3.41×10^{-5}	0.78 (0.69, 0.87)	0.044	0.89 (0.79, 1.01)	4.41×10^{-5}	0.83 (0.76, 0.91)
13	rs927788	*CLDN10*	0.42	6.95×10^{-5}	1.28 (1.13, 1.44)	0.039	1.12 (0.99, 1.28)	3.52×10^{-5}	1.2 (1.1, 1.31)
14	rs17175346	*SLC8A3*	0.16	8.53×10^{-5}	1.4 (1.18, 1.66)	5.97×10^{-5}	1.44 (1.2, 1.73)	3.19×10^{-8}	1.42 (1.25, 1.6)
14	rs7146876	*SERPINA13*	0.23	0.00042	1.29 (1.12, 1.49)	0.025	1.17 (1, 1.37)	8.16×10^{-5}	1.24 (1.11, 1.37)

Association results of the 12 new candidate loci for ACPA-positive RA identified in the GWAS stage and nominally replicated ($P < 0.05$) in the validation stage. GWAS: association statistics for the discovery cohort; Validation: association statistics for the validation cohort; Combined: association statistics for GWAS and validation cohorts combined; Chr: chromosome; MAF: minor allele frequency; OR (95%CI): odds ratio of SNP and 95% confidence interval; P: P-values for association; SNP: single-nucleotide polymorphism.

Replication study

Replication genotyping was performed at the HudsonAlpha Institute for Biotechnology (Huntsville, AL, USA) using the Illumina GoldenGate assay (Illumina, San Diego, CA, USA) on 863 ACPA-positive patients with joint damage and 1152 controls. Details on the quality control measures are included in 'GWAS and Replication Quality Control Procedures' in the supplementary data, available at *Rheumatology* Online.

Results

After quality-control analysis, a final number of 890 ACPA-positive RA patients and 1493 controls were used in the discovery stage. A total of 506 950 SNPs passed all quality and frequency filters and were used for association analysis.

In the discovery cohort, 25 of the established risk loci for RA were found to be significantly associated ($P < 0.05$) with ACPA-positive RA [1] (supplementary Table S2, available at *Rheumatology* Online). From these, 10 loci had not been previously associated to this specific group of RA patients. To our knowledge, *MTF1-INPP5B*, *PLCL2*, *ATG5*, *ZNF348*, *WDFY4*, *PLD4-AHNAK2* and *MED1* loci have not been previously associated with ACPA-positive RA. *RCAN1* had been analysed in a previous GWAS in ACPA-positive patients from Asian ancestry [5] but was not significantly associated.

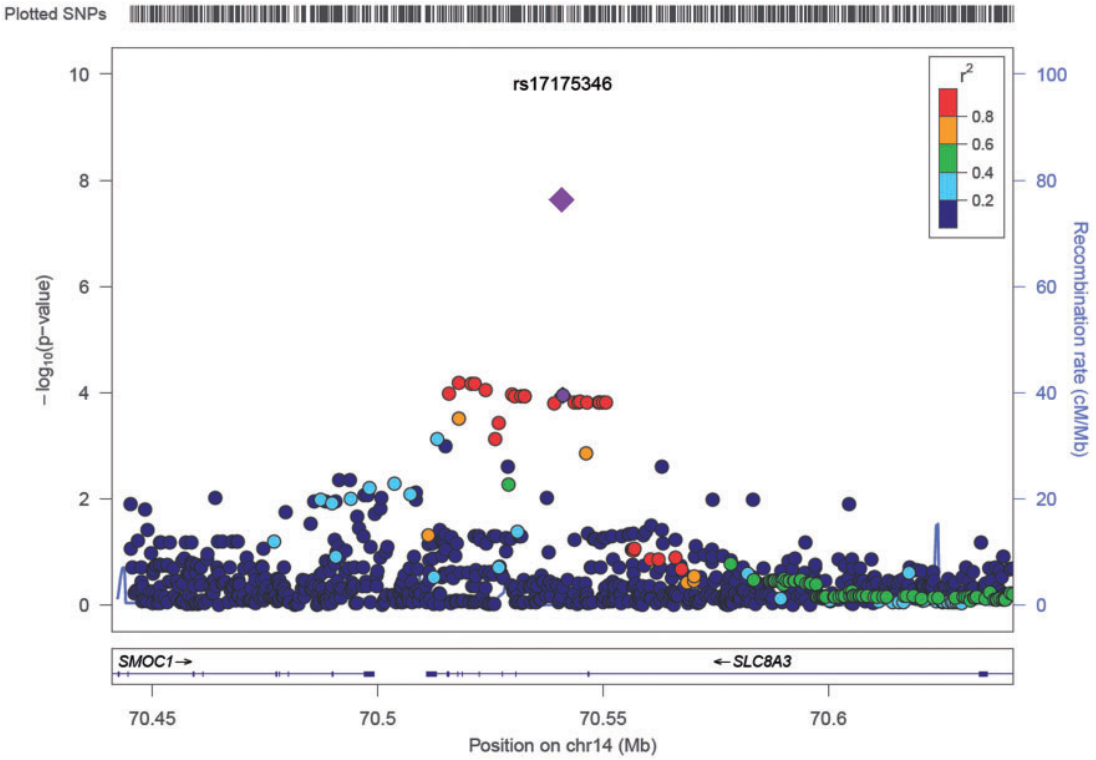
In the discovery phase, 60 genomic regions not previously associated to RA or ACPA-positive RA showed high statistical evidence of association ($P < 5 \times 10^{-4}$, supplementary Table S3, available at *Rheumatology* Online). This group of candidate risk loci was selected for validation in the independent case-control cohort. Using the validation cohort, a total of 12 loci were replicated at the nominal level loci ($P < 0.05$, same direction of effect as in the GWAS, Table 1).

When combining the data from the discovery and validation phases, SNP rs17175346, located in an intron of the solute carrier family 8 member 3 gene (*SLC8A3*) in chromosome 14q24.1, reached a genome-wide level of significance [$P = 3.19 \times 10^{-8}$, odds ratio, OR (95% CI): 1.44 (1.2, 1.73)] (Fig. 1).

In order to gain insights to the possible regulatory potential of the *SLC8A3* SNP rs17175346 associated with ACPA-positive RA we screened public functional annotation databases [8]. We found that the chromosome 14q24.1 region where rs17175346 lies has strong regulatory evidence (supplementary Fig. S2, available at *Rheumatology* Online). For example, DNAseI screening studies performed by the ENCODE project have shown that this genomic region is hypersensitive to cleavage in 112 out of 125 different human cell types, strongly supporting its role as an active genetic regulatory site. Similar evidence obtained using other genomic regulation characterization approaches like the Roadmap Epigenomics Consortium (http://www.roadmapepigenomics.org/, supplementary Fig. S3, available at *Rheumatology* Online) also support the existence of an important regulatory element in the chromosome region that harbours the SNP associated to ACPA-positive RA.

Given that the biological role of the *SLC8A3* gene is still poorly understood, we used the GeneNetwork genomic database (www.genenetwork.nl) to predict its function. This functional analysis tool uses the co-expression patterns in more than 80 000 genome-wide expression analyses in mouse and human to predict significant biological functions of genes. Using this method we found that, from all tested biological annotations, the Gene Ontology database biological processes regulation of ion transmembrane transport

Fig. 1 Association results for *SLC8A3* locus with ACPA-positive RA



Regional plot with the significance [i.e. −log10 (P-values), *y*-axis] of the SNPs in *SLC8A3* gene region in the discovery phase as a function of basepair location in chromosome 14q24.1 (*x*-axis). The validated SNP (rs17175346) is shown as a purple diamond with significance value from the combined (GWAS and validation) cohort association analysis. The remaining SNPs are shown as circles with colour coding indicating the level of LD (i.e. r^2, legend) with respect to rs17175346. The estimated recombination rates (centimorgans/megabase, right y-axis) are plotted as a continuous background line. LD: linkage disequilibrium; SNP: single-nucleotide polymorphism.

(GO:0034765) and ossification (GO:0001503) showed the most significant associations for *SLC8A3* function ($P = 1.26 \times 10^{-10}$ and $P = 1.85 \times 10^{-9}$, respectively; supplementary Table S4, available at *Rheumatology* Online).

Discussion

Using a GWAS approach, we have identified *SLC8A3* as a new risk locus for ACPA-positive RA. Analysing 890 ACPA-positive patients with joint damage and 1493 healthy controls we have identified several candidate risk loci. Using an independent cohort of 863 ACPA-positive patients with joint damage and 1152 healthy controls, we replicated the association of 12 of these new candidate risk loci for RA at the nominal level ($P < 0.05$, same direction as in GWAS). When combining the data from the discovery and validation phases, we have found a genome-wide significant association for rs17175346 (combined $P = 3.19 \times 10^{-8}$), an intronic SNP from *SLC8A3* gene located on chromosome *14q24.1*.

SLC8A3, also known as *NCX3*, encodes a highly conserved protein that mediates sodium and calcium ion exchange across the cell membrane [9]. To date, little is known about the biological processes and cell types that depend on *SLC8A3*. Recent evidence, however, indicates that it is a gene that is constitutively expressed in monocytes/macrophages [10]. Importantly, *SLC8A3* activation in cultured macrophages has been associated to an increase of TNF cytokine production [10]. TNF secretion by macrophages is clearly one of the main pathophysiological mechanisms associated with RA aetiology [11]. Therefore, genetic variants influencing the regulation of TNF secretion in this key cell type in RA could increase the risk of the disease.

In silico prediction of *SLC8A3* biological activity also suggests an association of this Na^+-dependent Ca^{2+} transporter with bone metabolism. Also, the associated SNP rs17175346 lies in a CTCF binding site, a regulatory variant that insulates from enhancer and silencer signals, and it has been characterized in bone forming cells (osteoblasts) (supplementary Table S5, available at *Rheumatology* Online). In RA, the disequilibrium between enhanced osteoclast differentiation and the inhibition of osteoblast-mediated bone repair contributes to bone

erosion, which is the hallmark of the disease. *SLC8A3* has been shown to be expressed in osteoblasts during their differentiation and following bone formation [12]. Furthermore, there is increasing evidence that SLC8A3 is the main cellular translocator of Ca^{2+} from osteoblasts into the bone extracellular matrix [13]. Our results therefore suggest that genetic variation in the biological pathways affecting the target tissue in RA can also increase the risk of developing the disease.

Joint destruction is the most important severity feature of RA. In the present study we recruited ACPA-positive patients with radiographic joint damage, thus increasing the homogeneity of the patient cohort. To our knowledge, this is the first GWAS for RA where all patients both in the discovery and in the replication cohort are positive for ACPA and joint destruction in hands and/or feet. Using this approach we have increased the homogeneity of the patient cohort and we have therefore significantly increased the power to identify new genetic variants relevant for this predominant group of patients. A recent meta-analysis with Caucasian European and Asian RA cohorts increased to 101 the number of genetic variants associated to RA [1]. Despite the large sample size of this study, *SLC8A3* SNP rs17175346 did not show evidence of statistical association ($P > 0.05$, supplementary Table S6, available at *Rheumatology* Online). In this meta-analysis, however, patients were selected neither for positivity to ACPA nor for the presence of erosions. ACPA-negative patients and patients without erosions can represent up to 30% of individuals diagnosed with RA [6]. As suggested previously, including different patient subsets in the genetic analysis can clearly undermine the statistical power to identify new risk variants in RA [14, 15]. Of relevance, one of the nominally replicated genes in this study, *SPAG16*, has been recently found to be associated with the radiological progression rate in RA at the genome-wide level of significance [16]. Despite being a GWAS for a RA phenotype (and therefore a case-only study), it shares several features with the present GWAS for disease risk. Like in our study, in this recent GWAS only ACPA-positive RA patients were analysed. Also, similar to our study, the associated *SPAG16* variant does not show a significant association in the global RA meta-analysis study ($P > 0.05$, data not shown). Together, these results highlight the importance of patient selection criteria in the identification of additional relevant genetic variants in RA.

In the validation phase we replicated the association of 11 additional risk loci with ACPA-positive RA at the nominal level ($P < 0.05$). Although none of these additional loci reached a genome-wide level of statistical significance after combining both cohorts (i.e. $P < 5 \times 10^{-8}$), there is a clear enrichment of nominally significant genes ($P = 0.00017$, binomial test). This result clearly supports that within this group of replicated genes there are additional true risk factors for ACPA-positive RA. Apart from *SPAG16*, another highly suggestive candidate for ACPA-positive RA risk based on its biological function is early B cell factor (*EBF1*) gene. *EBF1* activity has shown to be crucial for B cell lineage commitment to mature antibody-secreting cells [17]. Variation at this gene has been recently associated with the risk of SS [18]. If validated in an independent dataset, this gene would add to the group of B cell pathway genes that have been previously associated with RA susceptibility [1].

In this study, we performed a GWAS in ACPA-positive RA with joint damage. We have identified *SLC8A3* as a new risk locus for ACPA-positive RA and we have also identified several additional loci with suggestive evidence of association with this prevalent disease group. These findings underline the importance of patient selection to characterize the missing heritability of RA.

Acknowledgements

We thank the patients and clinical specialists collaborating in the Immune-Mediated Inflammatory Disease Consortium for participation.

Funding: This study was supported by the Spanish Ministry of Economy and Competitiveness [grant numbers PSE-010000-2006-6, IPT-010000-2010-36]. The study sponsor had no role in the writing, study design, collection, analysis or interpretation of the data.

Disclosure statement: The authors have declared no conflicts of interest.

Supplementary data

Supplementary data are available at *Rheumatology* Online.

References

1 Okada Y, Wu D, Trynka G *et al*. Genetics of rheumatoid arthritis contributes to biology and drug discovery. Nature 2014;506:376–81.

2 Kurreeman F, Liao K, Chibnik L *et al*. Genetic basis of autoantibody positive and negative rheumatoid arthritis risk in a multi-ethnic cohort derived from electronic health records. Am J Hum Genet 2011;88:57–69.

3 Padyukov L, Seielstad M, Ong RT *et al*. A genome-wide association study suggests contrasting associations in ACPA-positive versus ACPA-negative rheumatoid arthritis. Ann Rheum Dis 2011;70:259–65.

4 Raychaudhuri S, Sandor C, Stahl EA *et al*. Five amino acids in three HLA proteins explain most of the association between MHC and seropositive rheumatoid arthritis. Nat Genet 2012;44:291–6.

5 Kim K, Bang SY, Lee HS *et al*. High-density genotyping of immune loci in Koreans and Europeans identifies eight new rheumatoid arthritis risk loci. Ann Rheum Dis 2015;74:e13.

6 van der Heijde DM. Joint erosions and patients with early rheumatoid arthritis. Br J Rheumatol 1995;34 (Suppl 2): 74–8.

7 Julia A, Domenech E, Chaparro M *et al*. A genome-wide association study identifies a novel locus at 6q22.1

associated with ulcerative colitis. Hum Mol Genet 2014;23:6927–34.

8 Rosenbloom KR, Armstrong J, Barber GP *et al*. The UCSC Genome Browser database: 2015 update. Nucleic Acids Res 2015;43(Database issue):D670–81.

9 Lytton J. Na^+/Ca^{2+} exchangers: three mammalian gene families control Ca^{2+} transport. Biochem J 2007;406:365–82.

10 Staiano RI, Granata F, Secondo A *et al*. Expression and function of Na^+/Ca^{2+} exchangers 1 and 3 in human macrophages and monocytes. Eur J Immunol 2009;39:1405–18.

11 Firestein GS. Evolving concepts of rheumatoid arthritis. Nature 2003;423:356–61.

12 Stains JP, Weber JA, Gay CV. Expression of Na^+/Ca^{2+} exchanger isoforms (NCX1 and NCX3) and plasma membrane Ca^{2+} ATPase during osteoblast differentiation. J Cell Biochem 2002;84:625–35.

13 Sosnoski DM, Gay CV. NCX3 is a major functional isoform of the sodium-calcium exchanger in osteoblasts. J Cell Biochem 2008;103:1101–10.

14 Viatte S, Plant D, Bowes J *et al*. Genetic markers of rheumatoid arthritis susceptibility in anti-citrullinated peptide antibody negative patients. Ann Rheum Dis 2011;71:1984–90.

15 Bossini-Castillo L, de Kovel C, Kallberg H *et al*. A genome-wide association study of rheumatoid arthritis without antibodies against citrullinated peptides. Ann Rheum Dis 2015;74:e15.

16 Knevel R, Klein K, Somers K *et al*. Identification of a genetic variant for joint damage progression in autoantibody-positive rheumatoid arthritis. Ann Rheum Dis 2014;73:2038–46.

17 Thal MA, Carvalho TL, He T *et al*. Ebf1-mediated down-regulation of Id2 and Id3 is essential for specification of the B cell lineage. Proc Natl Acad Sci U S A 2009;106:552–7.

18 Nordmark G, Kristjansdottir G, Theander E *et al*. Association of EBF1, FAM167A(C8orf13)-BLK and TNFSF4 gene variants with primary Sjogren's syndrome. Genes Immun 2011;12:100–9.

6) Julià, A., Pinto, J. A., Gratacós, J., Queiró, R., Ferrándiz, C., Fonseca, E., … Codó, L., Gelpí, J. L. …, Marsal, S. (2015a). A deletion at ADAMTS9-MAGI1 locus is associated with psoriatic arthritis risk. Annals of the Rheumatic Diseases, 74(10), 1875–1881. https://doi.org/10.1136/annrheumdis-2014-207190

EXTENDED REPORT

A deletion at *ADAMTS9-MAGI1* locus is associated with psoriatic arthritis risk

Antonio Julià,[1] José Antonio Pinto,[2] Jordi Gratacós,[3] Rubén Queiró,[4] Carlos Ferrándiz,[5] Eduardo Fonseca,[6] Carlos Montilla,[7] Juan Carlos Torre-Alonso,[8] Lluís Puig,[9] José Javier Pérez Venegas,[10] Antonio Fernández Nebro,[11] Emilia Fernández,[12] Santiago Muñoz-Fernández,[13] Esteban Daudén,[14] Carlos González,[15] Daniel Roig,[16] José Luís Sánchez Carazo,[17] Pedro Zarco,[18] Alba Erra,[19] José Luís López Estebaranz,[20] Jesús Rodríguez,[21] David Moreno Ramírez,[22] Pablo de la Cueva,[23] Francisco Vanaclocha,[24] Enrique Herrera,[25] Santos Castañeda,[26] Esteban Rubio,[27] Georgina Salvador,[28] César Díaz-Torné,[29] Ricardo Blanco,[30] Alfredo Willisch Domínguez,[31] José Antonio Mosquera,[32] Paloma Vela,[33] Jesús Tornero,[34] Simón Sánchez-Fernández,[35] Héctor Corominas,[16] Julio Ramírez,[36] María López-Lasanta,[1] Raül Tortosa,[1] Nuria Palau,[1] Arnald Alonso,[1] Andrés C García-Montero,[37] Josep Lluís Gelpí,[38] Laia Codó,[39] Kenneth Day,[39] Devin Absher,[39] Richard M Myers,[39] Juan D Cañete,[36] Sara Marsal[1]

Handling editor Tore K Kvien

▶ Additional material is published online only. To view please visit the journal online (http://dx.doi.org/10.1136/annrheumdis-2014-207190).

For numbered affiliations see end of article.

Correspondence to
Professor Sara Marsal, Rheumatology Research Group, Vall d'Hebron University Hospital, Pg Vall d'Hebron, 119-129, Barcelona 08035, Spain; sara.marsal@vhir.org and Dr Juan D Cañete, Rheumatology Department, Hospital Clínic i Provincial and IDIBAPS, c/Villarroel 170, 08036, Barcelona, Spain; jcanete@clinic.ub.es

SM and JDC are co-corresponding authors.

Received 19 December 2014
Revised 23 April 2015
Accepted 23 April 2015
Published Online First 19 May 2015

▶ http://dx.doi.org/10.1136/annrheumdis-2014-207187

To cite: Julià A, Pinto JA, Gratacós J, *et al*. *Ann Rheum Dis* 2015;**74**:1875–1881.

ABSTRACT

Objective Copy number variants (CNVs) have been associated with the risk to develop multiple autoimmune diseases. Our objective was to identify CNVs associated with the risk to develop psoriatic arthritis (PsA) using a genome-wide analysis approach.

Methods A total of 835 patients with PsA and 1498 healthy controls were genotyped for CNVs using the Illumina HumanHap610 BeadChip genotyping platform. Genomic CNVs were characterised using CNstream analysis software and analysed for association using the χ^2 test. The most significant genomic CNV associations with PsA risk were independently tested in a validation sample of 1133 patients with PsA and 1831 healthy controls. In order to test for the specificity of the variants with PsA aetiology, we also analysed the association to a cohort of 822 patients with purely cutaneous psoriasis (PsC).

Results A total of 165 common CNVs were identified in the genome-wide analysis. We found a highly significant association of an intergenic deletion between *ADAMTS9* and *MAGI1* genes on chromosome *3p14.1* (p=0.00014). Using the independent patient and control cohort, we validated the association between *ADAMTS9-MAGI1* deletion and PsA risk (p=0.032). Using next-generation sequencing, we characterised the 26 kb associated deletion. Finally, analysing the PsC cohort we found a lower frequency of the deletion compared with the PsA cohort (p=0.0088) and a similar frequency to that of healthy controls (p>0.3).

Conclusions The present genome-wide scan for CNVs associated with PsA risk has identified a new deletion associated with disease risk and which is also differential from PsC risk.

INTRODUCTION

Psoriatic arthritis (PsA) is a chronic inflammatory arthritis that affects 10–30% of patients with psoriasis.[1 2] To date, genome-wide association studies (GWAS) as well as candidate gene studies have shown that both diseases share a substantial genetic component. However, sibling recurrence rates (λ_s) are much higher for PsA than psoriasis (PsA λ_s~37 vs psoriasis λ_s~7),[3–5] indicating that additional, perhaps disease-specific, risk factors need to be identified.

GWAS based on single-nucleotide polymorphisms (SNPs) have been highly successful in identifying >30 loci associated with psoriasis and PsA susceptibility.[6–8] The cumulative risk exerted by these loci, however, is <50%,[9] and additional genetic factors still need to be identified in order to explain the missing heritability. Strategies to complete the characterisation of the genetic architecture of psoriasis and PsA include the use of large sample sizes or the combination of different studies through metaanalysis,[10] the deep sequence characterisation to identify rare variants with large effect sizes[11] and, also, the analysis of other types of genetic variation that cannot be completely captured by SNP-based genotyping platforms such as copy number variants (CNVs).

CNVs are fragments of DNA with sizes that range from hundreds of bases to several megabases, and that can either be absent (ie, deletions), repeated a certain number of times (ie, amplifications) or even rearranged.[12] Psoriasis was one of the first chronic inflammatory diseases where CNVs were found to be associated with the disease risk. The amplifications of the β-defensin genes on 8p23.1 region[13] and the deletion affecting *LCE3B* and *LCE3C* genes[14] have been clearly associated with psoriasis aetiology. The association of these CNVs with PsA aetiology, however, is still not clear,[2 15] suggesting that they could participate in the chronic inflammatory processes in the skin rather than in the pathological process occurring in the joint.

In the present study, we have performed the first genome-wide analysis of CNVs in PsA. We have first analysed a discovery panel of 835 patients with PsA and 1498 healthy controls from the Spanish population using a microarray platform. The CNVs showing a more significant association to PsA risk were subsequently selected and validated in an independent cohort of 1133 patients with PsA and 1831 healthy controls. In order to test for the specificity of the CNV association with PsA aetiology, we have also analysed a set of 822 psoriasis patients without arthritis. Using this approach, we have identified a new deletion associated with PsA risk that is not associated with purely cutaneous psoriasis (PsC).

PATIENTS AND METHODS

Study subjects

To identify new loci associated with psoriasis risk using the GWAS approach, we recruited 835 patients with PsA and 1498 healthy controls from the Spanish population. Patient and control individuals were obtained by the Immune-Mediated Inflammatory Disease Consortium (IMIDC).[16] The IMIDC is a Spanish biomedical research collaboration project that includes biomedical and clinical researchers on rheumatology, dermatology and gastroenterology, and that is devoted to the study of prevalent autoimmune diseases. In the present study, a total of 26 rheumatology departments—15 in the GWAS stage and 11 additional in the replication stage—and 11 dermatology departments from different university hospitals in Spain participated in the patient recruitment and clinical data collection. All patients with PsA included in this study had a clinical diagnosis made by a consultant rheumatologist. All patients with PsA were diagnosed according to the Classification Criteria for Psoriatic Arthritis criteria,[17] were >18 years old—although the disease could have started earlier in life—and had at least 1 year of evolution of the disease. Exclusion criteria for the present study were (i) presence of any other inflammatory joint disease, (ii) presence of any inflammatory bowel disease and (iii) positivity of rheumatoid factor.

Control individuals for the GWAS stage were recruited from blood bank donors attending at 13 hospitals from different regions in Spain in collaboration with the Spanish National DNA Bank (http://www.bancoadn.org). Eligible individuals were screened for the presence of PsA or any other autoimmune disorder, as well as for history of autoimmune disorders in first-degree relatives, and positive individuals were discarded from this study. Additionally, in order to increase the 'hypernormality' of the control cohort,[18] only individuals who were >30 years old were included. In total, 1498 controls, 40% of whom were women, were analysed in the GWAS. Of note, >96% of the control individuals were >40 years old at the time of recruitment.

All patients and controls in the GWAS and replication cohorts were Caucasian European. In those cases where any of the four grandparents was not born in Spain, the individual was discarded from the study. The DNA samples from patients and controls in both stages of the study were obtained from whole blood samples.

A total of 1131 patients with PsA and 1831 controls were used to validate the most significant loci identified in the GWAS phase. Both cohorts were collected using the same clinical and epidemiological selection criteria as for the GWAS. Additionally, a sample of 822 patients diagnosed with psoriasis and without PsA (ie, PsC) was also analysed in the validation phase. All patients with PsC were diagnosed and recruited by a consultant dermatologist participating in the IMID Consortium.[2] Psoriasis patients with plaque psoriasis affecting torso and/or extremities and with at least one year of duration were included. Patients with a single clinical localisation of plaque psoriasis (ie, scalp, face, palmoplantar), with exclusively inverse plaque psoriasis or with an inflammatory bowel disease, were excluded from the study. Finally, psoriasis patients diagnosed with PsA by a rheumatologist were excluded from this group.

Genome-wide CNV analysis

We performed a CNV genome-wide scan by using Illumina 610Quad Beadchips (Illumina, San Diego, California, USA), which contains a total of 620 901 probes. Sample genotyping was performed at the HudsonAlpha Institute for Biotechnology (Alabama, USA). After excluding mitochondrial as well as X and Y chromosome SNPs, a total of 600 470 probes were considered for GWAS CNV analysis.

Before proceeding to perform PsA risk analysis, we performed several quality control analysis steps. First, only those samples that had a >95% genotype completion rate were considered for analysis (99% of samples). Second, we used the SNPs genotype information to estimate the main axes of variation using the principal component analysis implemented in the Eigenstrat software.[19] With this approach, individuals showing a high deviation in any of the 10 top principal axes of variation were considered outliers and were consequently removed (>6 SDs from the centre of each component, n=42 outliers). Online supplementary figure S1 shows the patient with PsA and control distributions according to the first and second principal components after excluding the outliers.

Since CNV genotyping is subject to more technical biases than SNP genotyping, several additional quality control filters must be applied to the GWAS data. Following previous GWAS CNV studies,[20] samples with a substantial deviation from the mean log Ratio ($|\mu$ log R ratio$|$ > 0.1) or with an excessive variability (σ log R ratio > 0.2), were excluded (n=556 individuals, 23.8%). After applying all the quality control filters, a total of 658 patients with PsA and 1063 controls were finally available for the CNV GWAS. Online supplementary figure S2 shows a schematic representation of the global CNV analysis workflow.

CNV identification and genotyping was performed using CNstream software (figure 1A).[21] CNstream first applies a normalisation procedure to control for the presence of potential intensity biases from samples processed at different time points (ie, batch normalisation). It then applies a second normalisation step to minimise the difference of sensitivity between the two colour channels used to analyse each probe (ie, intensity normalisation).[22] Once the data are normalised, CNstream jointly analyses sets of consecutive probes (n=5 in this study) to identify the presence of a CNV in a particular region of the genome and generate a genotype call for each individual.

After genome-wide CNV identification and calling, the association with disease risk was tested using the genotypic χ^2 test. In the case of low-frequency CNVs (minor allele frequency <5%), the genotype counts for CNV homozygous (0N) as well as individuals with 1 deletion (1N) were merged in a single CNV-positive group and compared with non-CNV carriers (2N). Statistical association analyses were performed using R software V.3.0.1.[23]

Targeted sequencing of *ADAMTS9-MAGI1* locus

In order to characterise the chromosome 3q14.2 sequence harbouring the deletion associated with PsA risk in the CNV GWAS, we performed a targeted resequencing analysis in

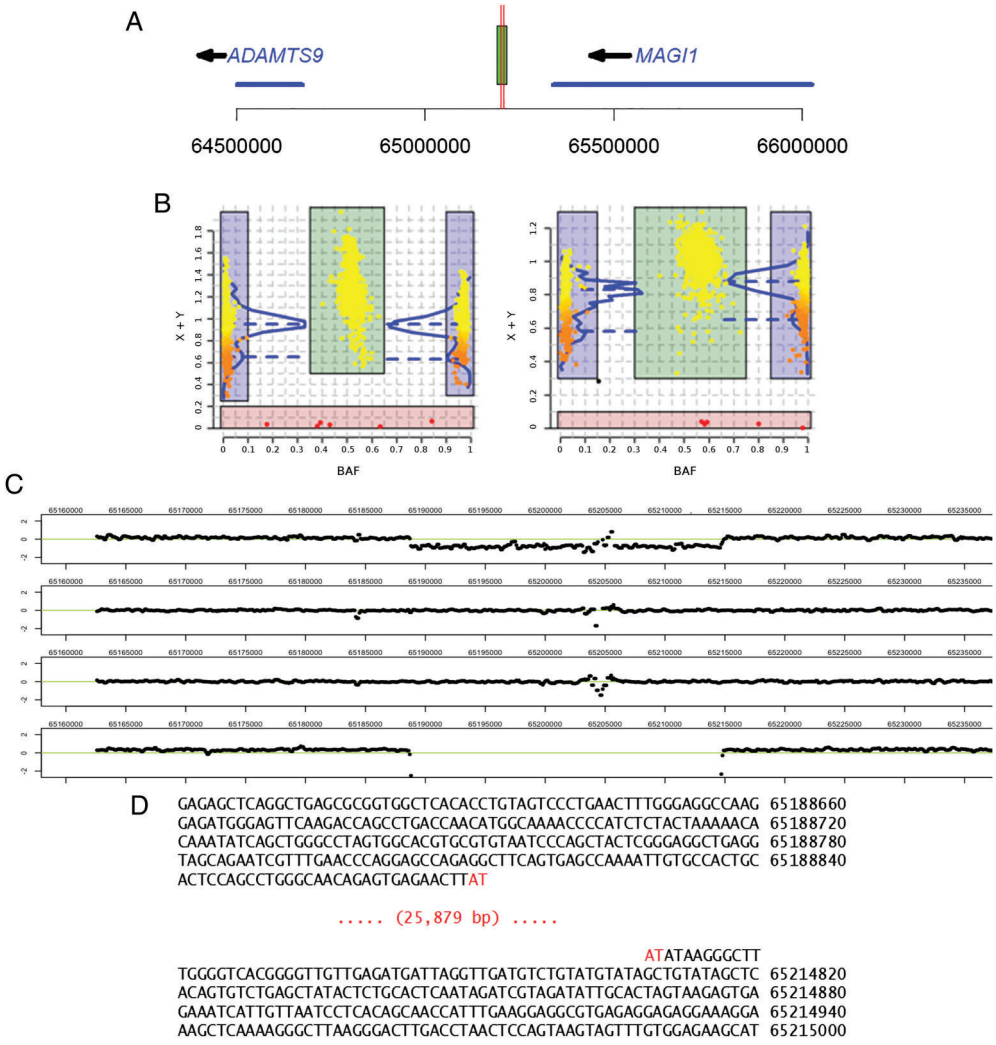


Figure 1 *ADAMTS9-MAGI1* deletion characterisation. (A) Map of chromosome *3p14.1* region showing *ADAMTS9* and *MAGI1* genes (blue segments), and the intergenic deletion (green rectangle) associated with psoriatic arthritis (PsA) risk. The horizontal red lines indicate the location of the two Taqman assays (Hs03225015_cn and Hs03225295_cn, red vertical lines) used to genotype the deletion. (B) CNStream deletion calling using multiple probes within the deletion sequence. This software method combines the copy number estimation of consecutive probes (left: probe at chr3:65 191 847 pb; right: probe at chr3: 65 196 123 pb). (C) Log2 intensity reads of six individuals for the chromosome 3 region harbouring the deletion. Starting from the top, the first two individuals show the characteristic drop in intensity corresponding to hemizygous (ie, 1N) individuals. The third, fourth and sixth individuals show the expected log2 reads of a 2N individual, with most of the intensities centred around 0. Finally, the fifth individual clearly shows the presence of an individual homozygous for the deletion, with no sequence reads mapping to this region of the chromosome. (D) Sequence of the reference and deletion alleles. Physical coordinates are on the reference human genome (build 37).

selected samples from the discovery phase. Next-generation sequencing was performed at the HudsonAlpha Institute for Biotechnology (Alabama, USA). A total of 100 patients with PsA and 100 control individuals were selected for sequencing of *ADAMTS9-MAGI1* locus. The individuals were selected so that deletion carriers and non-carriers—as determined by GWAS genotyping—were equally present in both groups. Consequently, 100 sequenced individuals carried one or two deletions (ie, 1N or 0N) and 100 individuals had no deletion (ie, 2N individuals).

The Illumina sequencing platform (Illumina, San Diego, USA) was used to characterise the deletion sequence. In order to identify yet undiscovered variants in the two flanking genes that could be responsible for the observed association with PsA risk, we also sequenced *ADAMTS9* and *MAGI1* genes and their 5′ and 3′ flanking sequences. First, the DNA quality of these samples was assessed by running 1–3 µL on a 1% agarose gel that contained 1× Sybr Green I dye (Life Technologies, USA). Next we followed the CATCH-Seq procedure we reported recently.[24] In brief, we purified CTD-2216H2, CTD-2255G2, CTD-2517E23, RP11-841H13, RP11-1080G20, RP11-411F5 and RP11-257J13 BAC DNAs that are commercially available (Life Technologies). BAC DNAs were pooled by percentage of the total target size (1.1 Mb) according to a 4 µg total input mass, and the pool was sheared by E220 Covaris sonication. Linkers containing T7 promoter sequences were ligated to sheared BAC fragments, and biotinylated RNA probes were synthesised by in vitro transcription using a MEGAscript kit (Ambion) and biotin-11-UTP (Life Technologies) with T7-BAC fragments as template. Illumina libraries were prepared according to standard protocol using 24 inline barcoded adapters.

For capture of library within the *ADAMTS9/MAGI1* chromosome 3 region, hybridisation reactions were assembled with 4 barcoded libraries (125 ng of each), 20 µg of Cot-1 DNA (Life Technologies), 236 ng probe, 20 U SUPERase-In (Life Technologies) and 2× hybridisation buffer in a final volume of

26 µL incubated at 65°C for 70 h. Hybridisation reactions were incubated with 25 µL MyOne Streptavidin C1Dynabeads (Life Technologies) for 30 min with frequent pulse vortexing. Bead captures were washed twice for 15 min each at room temperature in 0.5 mL wash buffer 1 (1×SSC, 0.1% SDS), followed by four stringency wash steps at 65°C in 0.5 mL preheated wash buffer 2 (0.1×SSC, 0.1% SDS) for 10 min each. Captured libraries were eluted in 50 µL 0.1 M NaOH and neutralised in 70 µL 1 M Tris pH 7.5. Final libraries were cleaned with 1.8× solid-phase reversible immobilisation beads and eluted in 33 µL water for assembly of library amplification PCR containing 1 µL Platinum Taq (Life Technologies) and 5 µL 5 M betaine (Sigma) in 50 µL reactions (98°C 1 min, 95°C 30 s and 62°C 3.5 min for 20 cycles). Final 4-plex library concentrations were determined by KAPA QPCR (KAPA Biosystems) and adjusted to 15 nM each. Stock 4-plex libraries were pooled appropriately for final 24-plex libraries each for a single lane on HiSeq2000 sequencer (Illumina) using 50 bp paired end conditions.

Sequencing reads from individual samples were demuxed based on inline barcode sequences and aligned to the human reference genome (hg19) with BWA.[25] Relative read depth was calculated as the number of bases mapped to 100 bp windows per total bases mapped for a given sample. Then a log2 ratio between each sample at each 100 bp window to the mean of all samples at that window was used to plot a normalised read depth, representing the read depth relative to a theoretical diploid reference. The plots of these normalised read depths across the locus were used to confirm the presence of the deletion (figure 1B).

CNV replication analysis

Replication genotyping was performed using the TaqMan Genotyping System (Applied Biosystems, Foster City, California, USA). Two pre-designed Taqman CNV assays Hs03225015_cn and Hs03225295_cn were found to be located within the estimated deletion boundaries. In order to validate the two assays, we genotyped the group of 200 individuals that were previously used to sequence the deletion. The correspondence between the calls of the two CNVs between the Taqman and sequencing analysis was 100%. Consequently, we used the two Taqman assays to genotype the *ADAMTS9-MAGI1* deletion in an independent group of 1133 patients with PsA, 1831 healthy controls and 822 patients with PsC. Quality control measures similar to the GWAS were applied, including genotyping call rate >95%, sample completion rate >90% and Hardy–Weinberg disequilibrium p value of control group p>0.001. The CNV genotype concordance between the two Taqman assays was >99%. Meta-analysis of the GWAS and replication association statistics was performed using METAL software.[26]

RESULTS

CNV identification and genotyping

Table 1 summarises the main features of the GWAS and replication PsA patient cohorts.

Using a total of 658 patients with PsA and 1063 healthy controls, we identified a total of 2674 CNV segments. After merging segments belonging to the same genomic region (distance <10 kb and/or r^2>0.9), we performed the genotype calling in a total of 1953 different CNV regions. Among them, 165 CNVs appeared in >5% of the samples and were subsequently used to test for association with PsA risk. Online supplementary table S1 describes the characteristics of these CNVs.

Table 1 Phenotypic summary of GWAS and replication patient cohorts

Characteristic	GWAS cohort	Replication cohort
Gender (% female)	45.3%	48.6%
Age (mean±SD)	52.9 ±12.8	54.4±13.2
Chronic plaque psoriasis (%)	92.0%	90.9%
Age at onset psoriasis (mean±SD)	33.9 ±15.4	35.7±15.5
Age at onset PsA (mean±SD)	42.5 ±13.5	42.6±15.1
RF positive (%)	0%	0%
Fulfilling Classification Criteria for Psoriatic Arthritis criteria	100%	100%

GWAS, genome-wide association studies; PsA, psoriatic arthritis; RF, rheumatoid factor.

CNV GWAS for PsA risk

Using the group of common CNVs, we performed a GWAS for PsA risk. We found a very strong association signal at an intergenic deletion located between the *HLA-C* and *HLA-B* genes (p=5.37e-11, OR (95% CI) 2.08 (1.47 to 2.95), MAF=0.18, figure 2). Since *HLA-C* locus is an established risk locus for PsA we sought to estimate the association of the deletion after correcting for the HLA haplotypes associated with psoriasis risk as recently described.[27] We found that, after correcting for *HLA-C*0602* and *HLA-B*3801* alleles, the deletion was no longer associated with PsA risk (p=0.81, see online supplementary table S2). Consequently, this CNV was considered a proxy for the HLA allele association and was not included for replication.

We also found another highly significant association for a deletion located in the chromosome 3q14.2 intergenic region, between *ADAMTS9* and *MAGI1* genes (p=0.00014, OR (95% CI) 1.94 (1.37 to 2.75), MAF=0.04, figure 2). This genomic region had not been previously associated to any SNP-based

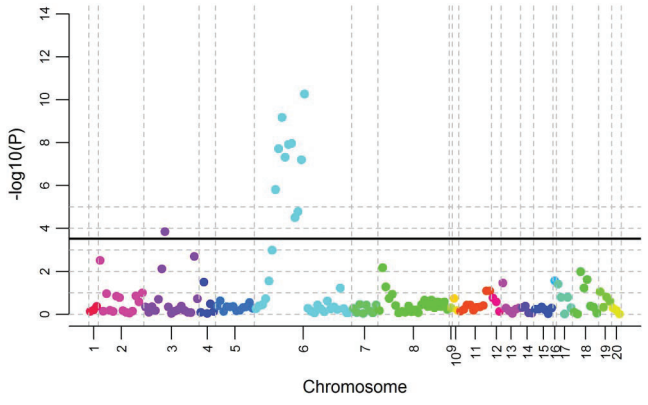


Figure 2 Plot of the copy number variant (CNV) genome-wide association studies results. The −log10 p values (y-axis) are plotted for each of the CNVs identified by CNSstream. Each chromosome is coded in a different colour. The probes mapping the intergenic deletion in *HLA-C/-B* locus in chromosome 6 (light blue dots) were found to have a high significance; however, after correcting for *HLA-C* and *HLA-B* alleles this association disappeared. In chromosome 3, an intergenic deletion between genes *ADAMTS9* and *MAGI1* (purple dot) shows a significant association that withstands multiple test correction.

GWAS in PsA or any other related disease. After correcting for the number of CNVs analysed, the deletion association was still found to be significantly associated with PsA risk (p=0.023, Bonferroni multiple test correction). Consequently, we selected this region for replication in the independent data set of patients and controls. In the remaining group of CNVs, we found 12 additional CNV regions nominally associated with PsA risk (p<0.05, see online supplementary table S3). However, after multiple test correction none of these variants was statistically significant and therefore they were not selected for replication. Additionally, evaluating the concordance between these CNVs and neighbouring SNPs, we did not find a strong linkage disequilibrium (LD) (r^2>0.8) with markers previously associated with PsA, Ps or other autoimmune diseases.

CNV replication in the validation cohorts

The deletion genotypes determined using the quantitative RT-PCR assays showed a 100% concordance with the number of copies (0, 1 or 2) estimated using sequencing. We subsequently used these two RT-PCR assays to genotype an independent cohort of 1133 patients with PsA and 1831 controls. We replicated the association of the *ADAMTS9-MAGI1* intergenic deletion with PsA risk (p=0.032, OR (95% CI) 1.3 (1.0 to 1.7), meta-analysis p=5.97e-5, OR (95% CI) 1.48 (1.21 to 1.82)).

Finally, comparing the frequencies of *ADAMTS9-MAGI1* deletion of patients with PsA with patients with PsC, we also found a statistically significant increase of the deletion in the group of patients with PsA similar to that observed when comparing to healthy controls (freq PsA=11.0%, freq PsC=7.7%, freq controls=8.8%, p=0.0088). Accordingly, when comparing patients with PsC to healthy controls, we did not find a statistically significant difference between the deletion frequencies of groups (p=0.33).

ADAMTS9-MAGI1 deletion sequence characterisation

Using a next-generation sequencing approach, we characterised the deletion region in chromosome 3q14.2 associated with PsA risk. Using a sample of 100 patients with PsA and 100 controls selected to have a higher frequency of the deletion (in total, 100 deletion carriers vs 100 2N individuals), we determined the physical extent and molecular nature of *ADAMTS9-MAGI1* deletion polymorphism. PCR capture and sequencing of the deletion breakpoints revealed that the deletion removes 25 879 nucleotides (figure 1D).

In order to explore the LD pattern associated with the deletion and the relation with the two flanking genes, we sequenced *ADAMTS9* and *MAGI1* genes and their proximal 5′ and 3′ untranslated regions. LD analysis showed clearly that only very few close polymorphisms have moderate to high correlation with the deletion (r^2>0.7, n=2′ SNPs, figure 3). From the >18 000 DNA variants identified after sequencing both *ADAMTS9* and *MAGI1* genes, none showed a significant LD with the deletion (figure 3).

DISCUSSION

GWAS based on SNPs have been highly successful in identifying genetic variants associated with the risk to develop psoriasis and PsA. However, there is still a large fraction of the genetic basis of PsA that has not been identified. In the present study, we have performed a GWAS of CNVs to identify new genomic loci associated with PsA risk. Using a discovery cohort of Spanish patients with PsA and controls, we have found a significant association of a deletion located between *ADAMTS9* and *MAGI1* genes with PsA risk. We have subsequently validated this association in an independent cohort of patients and controls. Furthermore, using a cohort of patients with psoriasis without concomitant arthritis we provide evidence that the deletion is specific for PsA.

ADAM metallopeptidase with thrombospondin type 1 motif 9 (*ADAMTS9*) gene belongs to a family of enzymes specialised in the degradation of the extracellular matrix. In particular, ADAMTS9 belongs to the family of the aggrecanases (which also includes ADAMTS-1, ADAMTS-4, ADAMTS-6, ADAMTS-8 and ADAMTS-15) that are specialised in the degradation of aggrecan, one of the main proteoglycan constituents of the cartilage extracellular matrix.[28] Aggrecanase activity has been found to be associated with cartilage degradation in inflammatory joint diseases including PsA.[29 30] Importantly, in vitro studies with chondrocytes have found that after stimulation with tumour necrosis factor (TNF)-α and interleukin (IL)-1, two of the most abundant cytokines in PsA synovium,[31]

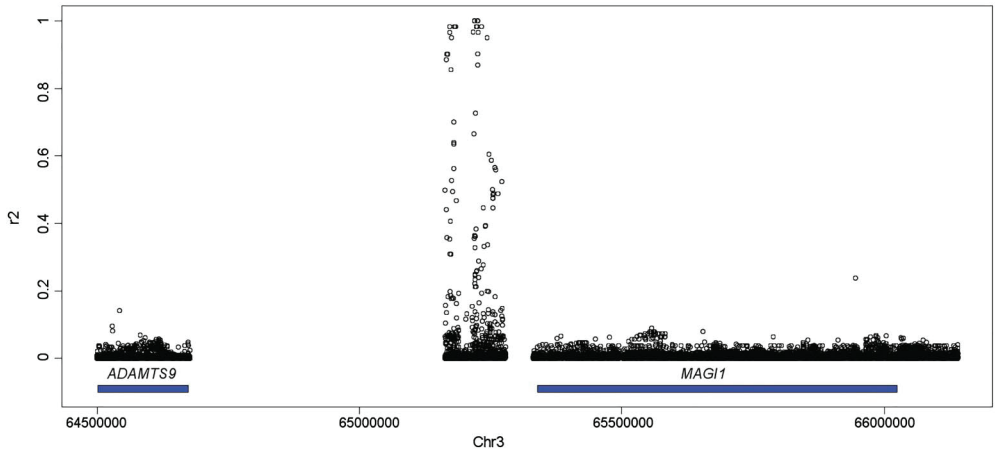


Figure 3 Pairwise linkage disequilibrium (LD) between the deletion and flanking sequence variants. The LD (r^2) between the deletion and the >18 000 variants identified after sequencing the regions flanking the copy number variants as well as *ADAMTS9* and *MAGI1* loci. From these results, it is clear that only a small proportion of close variants show a high correlation with the deletion (ie, r^2>0.9, centre region of the plot) and that there is no variant within the transcribed or the flanking regions of *ADAMTS9* (left region) or *MAGI1* (right region) that could explain the observed association. The *3p14.1* chromosome regions between these three loci and that were not sequenced are left blank.

ADAMTS9 was clearly the most highly induced among all the different aggrecanases.[32 33] Consequently, genetic variants that influence the rates of matrix turnover in the cartilage and bone of the inflamed synovial joint could be crucial in determining the level of tissue degradation in PsA.

Membrane-associated guanylate kinase, WW and PDZ domain containing 1 (*MAG1*) is a member of the membrane-associated guanylate kinase family of genes. MAGI1 is known to be expressed in cell-to-cell contacts, acting as a scaffold protein to stabilise cadherin-mediated adhesions and has been found to be expressed in epithelial and endothelial tight junctions.[34] MAGI1 activity has been related to several pathological junction-associated processes, including oncogenic[35] as well as virus-associated invasiveness.[36] To date, there is no evidence of a direct implication of MAGI1 in PsA pathology or, even, in autoimmune diseases. There is evidence, however, that MAGI1 interacts with phosphatase and tensin homologue protein, a signalling protein that participates in the negative regulation of regulatory T cells (Tregs),[37] which are master inhibitors of autoimmunity. While the implication of Treg has been clearly defined in rheumatoid arthritis or psoriasis aetiology,[38] there are yet no studies directly analysing the implication of this key immune regulator in PsA,[39] although there is recent evidence of their activity in the disease pathology.[40] Clearly, future studies aimed at characterising the implication of *MAGI1* activity in autoimmunity are necessary.

The deep sequence analysis found that there is very little correlation between the ~26 kb intergenic deletion associated with PsA risk and the polymorphisms located in the transcribed sequences and proximal regions of *ADAMTS9* and *MAGI1* genes. Also, our sequencing analysis clearly shows that very few neighbouring markers are in moderate or high LD with this CNV, suggesting that the deletion itself is the genetic variant implicated in the aetiology of PsA. Exploration of the chromosome 3q14.2 deleted region in multiple biomedical databases including ENCODE[41] did not show regulatory evidence associated with this variation. Also analysing available cis and trans-eQTL regulatory data sets[42] we did not find an association between this deletion and the expression of other genes. However, it is increasingly becoming evident that a large fraction of regulatory variants in the genome will be only detected under the specific target tissue where they are expressed and, perhaps, only under the adequate stimulation.[43] For example, *ADAMTS9* expression in chondrocytes was found to be expressed only after stimulation by proinflammatory cytokines TNF and IL-1.[32] Additional studies aimed at characterising the functional implications of this deletion and their role in PsA aetiology are therefore warranted.

To date, there is evidence that the frequency and penetrance of multiple risk loci in PsA and psoriasis risk is different populations with different ancestries.[44] It will be therefore necessary to evaluate the frequency and effect size of this deletion at *3p14.2* in other non-Caucasian populations. Also, the association of the *ADAMTS9-MAGI1* CNV with different PsA phenotypes could be of high relevance. In our discovery cohort, we analysed the association of the deletion with axial disease, arthritis mutilans, gender, age of start of the disease and PsA familial aggregation, but we did not find a significant association (data not shown). These results support that the deletion at chromosome *3q14.2* is specifically associated to the risk to develop PsA. It is possible, however, that once the specific biological mechanisms influenced by this genetic variation are identified, more targeted analysis will reveal association to other PsA phenotypes.

In the present study, we have performed the first CNV GWAS in PsA. We have identified a new deletion in *ADAMTS9-MAGI1* locus associated with PsA risk and we have validated this association in an independent patient and control cohort. Additionally, using a cohort of patients with PsC we have demonstrated that the variation is specifically associated with the development of PsA. The present study represents an important step in the characterisation of the common genetic variation associated with PsA.

Author affiliations
[1]Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, Spain
[2]Rheumatology Department, Complejo Hospitalario Juan Canalejo, A Coruña, Spain
[3]Rheumatology Department, Hospital Parc Taulí, Sabadell, Barcelona, Spain
[4]Rheumatology Department, Hospital Universitario Central de Asturias, Oviedo, Spain
[5]Dermatology Department, Hospital Universitari Germans Trias i Pujol, Badalona, Barcelona, Spain
[6]Dermatology Department, Complejo Hospitalario Universitario de A Coruña, A Coruña, Spain
[7]Rheumatology Department, Hospital Virgen de la Vega, Salamanca, Spain
[8]Rheumatology Department, Hospital Monte Naranco, Oviedo, Spain
[9]Dermatology Department, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain
[10]Rheumatology Department, Hospital de Jerez de la Frontera, Cádiz, Spain
[11]UGC Reumatología, Instituto de Investigación Biomédica de Málaga (IBIMA), Hospital Regional Universitario de Málaga, Málaga, Spain
[12]Department of Dermatology, Hospital Universitario de Salamanca, Salamanca, Spain
[13]Rheumatology Department, Hospital Universitario Infanta Sofía, Madrid, Spain
[14]Dermatology Department, Hospital Universitario La Princesa, Madrid, Spain
[15]Rheumatology Department, Hospital Universitario Gregorio Marañón, Madrid, Spain
[16]Rheumatology Service, Hospital Moisès Broggi, Barcelona, Spain
[17]Dermatology Department, Hospital General Universitario de Valencia, Valencia, Spain
[18]Rheumatology Department, Hospital Universitario Fundación Alcorcón, Madrid, Spain
[19]Rheumatology Department, Hospital Sant Rafael, Barcelona, Spain
[20]Dermatology Department, Hospital Universitario Fundación Alcorcón, Madrid, Spain
[21]Rheumatology Department, Hospital Universitari de Bellvitge, Barcelona, Spain
[22]Dermatology Department, Hospital Universitario Virgen Macarena, Sevilla, Spain
[23]Department of Dermatology, Hospital Universitario Infanta Leonor, Madrid, Spain
[24]Dermatology Department, Hospital Universitario 12 de Octubre, Madrid, Spain
[25]Dermatology Department, Hospital Universitario Virgen de la Victoria, Málaga, Spain
[26]Rheumatology Department, Hospital Universitario de La Princesa, IIS-Princesa, Madrid, Spain
[27]Rheumatology Department, Centro de Salud Virgen de los Reyes, Sevilla, Spain
[28]Rheumatology Department, Hospital Mútua de Terrassa, Terrassa, Spain
[29]Rheumatology Unit, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain
[30]Rheumatology Department, Hospital Universitario Marqués de Valdecilla, Santander, Spain
[31]Rheumatology Department, Complexo Hospitalario de Ourense, Ourense, Spain
[32]Rheumatology Department, Complejo Hospitalario Hospital Provincial de Pontevedra, Pontevedra, Spain
[33]Rheumatology Department, Hospital General Universitario de Alicante, Alicante, Spain
[34]Rheumatology Department, Hospital Universitario Guadalajara, Guadalajara, Spain
[35]Rheumatology Department, Hospital La Mancha Centro, Alcázar de San Juan, Spain
[36]Rheumatology Department, Hospital Clínic de Barcelona and IDIBAPS, Barcelona, Spain
[37]Banco Nacional de ADN Carlos III, University of Salamanca, Salamanca, Spain
[38]Life Sciences, Barcelona Supercomputing Centre, National Institute of Bioinformatics, Barcelona, Spain
[39]HudsonAlpha Institute for Biotechnology, Alabama, USA

Correction notice This article has been corrected since it was published Online First. Dr Juan D Cañete has been included as a co-corresponding author.

Acknowledgements We thank the patients and clinical specialists collaborating in the IMID Consortium for participation.

Contributors All authors were involved in the design, analysis and interpretation of data. All authors revised the manuscript and gave final approval for its submission. SM and JDC shared senior authorship.

Funding This study was funded by the Spanish Ministry of Economy and Competitiveness, grant numbers: PSE-010000-2006-6 and IPT-010000-2010-36.

Competing interests None declared.

Patient consent Obtained.

Ethics approval Local Institutional Review boards from all participating centres. This study was conducted in accordance with the Declaration of Helsinki principles

Provenance and peer review Not commissioned; externally peer reviewed.

REFERENCES

1 Gladman DD, Antoni C, Mease P, *et al*. Psoriatic arthritis: epidemiology, clinical features, course, and outcome. *Ann Rheum Dis* 2005;64(Suppl 2):ii14–17.
2 Julia A, Tortosa R, Hernanz JM, *et al*. Risk variants for psoriasis vulgaris in a large case-control collection and association with clinical subphenotypes. *Hum Mol Genet* 2012;21:4549–57.
3 Gladman DD, Farewell VT, Pellett F, *et al*. HLA is a candidate region for psoriatic arthritis. Evidence for excessive HLA sharing in sibling pairs. *Hum Immunol* 2003;64:887–9.
4 Myers A, Kay LJ, Lynch SA, *et al*. Recurrence risk for psoriasis and psoriatic arthritis within sibships. *Rheumatology (Oxford)* 2005;44:773–6.
5 Bhalerao J, Bowcock AM. The genetics of psoriasis: a complex disorder of the skin and immune system. *Hum Mol Genet* 1998;7:1537–45.
6 Liu Y, Helms C, Liao W, *et al*. A genome-wide association study of psoriasis and psoriatic arthritis identifies new disease Loci. *PLoS Genet* 2008;4:e1000041.
7 Ellinghaus E, Ellinghaus D, Stuart PE, *et al*. Genome-wide association study identifies a psoriasis susceptibility locus at TRAF3IP2. *Nat Genet* 2010;42:991–5.
8 Strange A, Capon F, Spencer CC, *et al*. A genome-wide association study identifies new psoriasis susceptibility loci and an interaction between HLA-C and ERAP1. *Nat Genet* 2010;42:985–90.
9 Chen H, Poon A, Yeung C, *et al*. A genetic risk score combining ten psoriasis risk loci improves disease prediction. *PLoS One* 2011;6:e19454.
10 Ellinghaus E, Stuart PE, Ellinghaus D, *et al*. Genome-wide meta-analysis of psoriatic arthritis identifies susceptibility locus at REL. *J Invest Dermatol* 2012;132:1133–40.
11 Sheng Y, Jin X, Xu J, *et al*. Sequencing-based approach identified three new susceptibility loci for psoriasis. *Nat Commun* 2014;5:4331.
12 Cook EH Jr, Scherer SW. Copy-number variations associated with neuropsychiatric conditions. *Nature* 2008;455:919–23.
13 Hollox EJ, Huffmeier U, Zeeuwen PL, *et al*. Psoriasis is associated with increased beta-defensin genomic copy number. *Nat Genet* 2008;40:23–5.
14 de Cid R, Riveira-Munoz E, Zeeuwen PL, *et al*. Deletion of the late cornified envelope LCE3B and LCE3C genes as a susceptibility factor for psoriasis. *Nat Genet* 2009;41:211–15.
15 Huffmeier U, Estivill X, Riveira-Munoz E, *et al*. Deletion of LCE3C and LCE3B genes at PSORS4 does not contribute to susceptibility to psoriatic arthritis in German patients. *Ann Rheum Dis* 2010;69:876–8.
16 Julia A, Domenech E, Ricart E, *et al*. A genome-wide association study on a southern European population identifies a new Crohn's disease susceptibility locus at RBX1-EP300. *Gut* 2012;62:1440–5.
17 Taylor W, Gladman D, Helliwell P, *et al*. Classification criteria for psoriatic arthritis: development of new criteria from a large international study. *Arthritis Rheum* 2006;54:2665–73.
18 Morton NE, Collins A. Tests and estimates of allelic association in complex inheritance. *Proc Natl Acad Sci U S A* 1998;95:11389–93.
19 Price AL, Patterson NJ, Plenge RM, *et al*. Principal components analysis corrects for stratification in genome-wide association studies. *Nat Genet* 2006;38:904–9.
20 Peiffer DA, Le JM, Steemers FJ, *et al*. High-resolution genomic profiling of chromosomal aberrations using Infinium whole-genome genotyping. *Genome Res* 2006;16:1136–48.
21 Alonso A, Julia A, Tortosa R, *et al*. CNstream: a method for the identification and genotyping of copy number polymorphisms using Illumina microarrays. *BMC Bioinformatics* 2010;11:264.
22 Steemers FJ, Chang W, Lee G, *et al*. Whole-genome genotyping with the single-base extension assay. *Nat Methods* 2006;3:31–3.
23 R Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing: Vienna, 2013.
24 Day K, Song J, Absher D. Targeted Sequencing of Large Genomic Regions with CATCH-Seq. *PLoS One* 2014;9:e111756.
25 Li H, Durbin R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* 2009;25:1754–60.
26 Willer CJ, Li Y, Abecasis GR. METAL: fast and efficient meta-analysis of genomewide association scans. *Bioinformatics* 2010;26:2190–1.
27 Okada Y, Han B, Tsoi LC, *et al*. Fine mapping major histocompatibility complex associations in psoriasis and its clinical subtypes. *Am J Hum Genet* 2014;95:162–72.
28 Kevorkian L, Young DA, Darrah C, *et al*. Expression profiling of metalloproteinases and their inhibitors in cartilage. *Arthritis Rheum* 2004;50:131–41.
29 Lohmander LS, Neame PJ, Sandy JD. The structure of aggrecan fragments in human synovial fluid. Evidence that aggrecanase mediates cartilage degradation in inflammatory joint disease, joint injury, and osteoarthritis. *Arthritis Rheum* 1993;36:1214–22.
30 Arner EC, Pratta MA, Decicco CP, *et al*. Aggrecanase. A target for the design of inhibitors of cartilage degradation. *Ann N Y Acad Sci* 1999;878:92–107.
31 Partsch G, Steiner G, Leeb BF, *et al*. Highly increased levels of tumor necrosis factor-alpha and other proinflammatory cytokines in psoriatic arthritis synovial fluid. *J Rheumatol* 1997;24:518–23.
32 Demircan K, Hirohata S, Nishida K, *et al*. ADAMTS-9 is synergistically induced by interleukin-1beta and tumor necrosis factor alpha in OUMS-27 chondrosarcoma cells and in human chondrocytes. *Arthritis Rheum* 2005;52:1451–60.
33 Uysal S, Unal ZN, Erdogan S, *et al*. Augmentation of ADAMTS9 gene expression by IL-1beta is reversed by NFkappaB and MAPK inhibitors, but not PI3 kinase inhibitors. *Cell Biochem Funct* 2013;31:539–44.
34 Laura RP, Ross S, Koeppen H, *et al*. MAGI-1: a widely expressed, alternatively spliced tight junction protein. *Exp Cell Res* 2002;275:155–70.
35 Zaric J, Joseph JM, Tercier S, *et al*. Identification of MAGI1 as a tumor-suppressor protein induced by cyclooxygenase-2 inhibitors in colorectal cancer cells. *Oncogene* 2012;31:48–59.
36 Kolawole AO, Sharma P, Yan R, *et al*. The PDZ1 and PDZ3 domains of MAGI-1 regulate the eight-exon isoform of the coxsackievirus and adenovirus receptor. *J Virol* 2012;86:9244–54.
37 Walsh PT, Buckler JL, Zhang J, *et al*. PTEN inhibits IL-2 receptor-mediated expansion of CD4+ CD25+ Tregs. *J Clin Invest* 2006;116:2521–31.
38 Buckner JH. Mechanisms of impaired regulation by CD4(+)CD25(+)FOXP3(+) regulatory T cells in human autoimmune diseases. *Nat Rev Immunol* 2010;10:849–59.
39 Nograles KE, Brasington RD, Bowcock AM. New insights into the pathogenesis and genetics of psoriatic arthritis. *Nat Clin Pract Rheumatol* 2009;5:83–91.
40 Ryder LR, Bartels EM, Woetmann A, *et al*. FoxP3 mRNA splice forms in synovial CD4+ T cells in rheumatoid arthritis and psoriatic arthritis. *APMIS* 2012;120:387–96.
41 ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature* 2012;489:57–74.
42 Lappalainen T, Sammeth M, Friedlander MR, *et al*. Transcriptome and genome sequencing uncovers functional variation in humans. *Nature* 2013;501:506–11.
43 Fairfax BP, Humburg P, Makino S, *et al*. Innate immune activity conditions the effect of regulatory variants upon monocyte gene expression. *Science* 2014;343:1246949.
44 Roberson ED, Bowcock AM. Psoriasis genetics: breaking the barrier. *Trends Genet* 2010;26:415–23.

7) Julià, A., Domènech, E., Ricart, E., Tortosa, R., García-Sánchez, V., Gisbert, J. P., … Codó, L., Gelpí, J. L. …, Marsal, S. (2013). A genome-wide association study on a southern European population identifies a new Crohn's disease susceptibility locus at RBX1-EP300. Gut, 62(10), 1440–1445. https://doi.org/10.1136/gutjnl-2012-302865

ORIGINAL ARTICLE

A genome-wide association study on a southern European population identifies a new Crohn's disease susceptibility locus at *RBX1-EP300*

Antonio Julià,[1] Eugeni Domènech,[2,3] Elena Ricart,[3,4] Raül Tortosa,[1] Valle García-Sánchez,[5] Javier P Gisbert,[3,6] Pilar Nos Mateu,[3,7] Ana Gutiérrez,[3,8] Fernando Gomollón,[3,9] Juan Luís Mendoza,[10] Esther Garcia-Planella,[3,11] Manuel Barreiro-de Acosta,[12] Fernando Muñoz,[13] Maribel Vera,[14] Cristina Saro,[15] Maria Esteve,[3,16] Montserrat Andreu,[17] Arnald Alonso,[1] María López-Lasanta,[1] Laia Codó,[18] Josep Lluís Gelpí,[18,19] Andres C García-Montero,[20] Jaume Bertranpetit,[21] Devin Absher,[22] Julián Panés,[3,4] Sara Marsal[1]

▶ Additional supplementary data are published online only. To view these files please visit the journal online(http://dx.doi.org/10.1136/gutjnl-2012-302865).

For numbered affiliations see end of article.

Correspondence to
Dr Eugeni Domènech, Digestive System Service, Hospital Universitari Germans Trias i Pujol, Badalona, 08196, Spain; edomenech.germanstrias@gencat.cat

Revised 27 July 2012
Accepted 27 July 2012
Published Online First 30 August 2012

ABSTRACT

Objective Genome-wide association studies (GWAS) have identified multiple risk loci for Crohn's disease (CD). However, the cumulative risk exerted by these loci is low, and the likelihood that additional, as-yet undiscovered loci contribute to the risk of CD is very high. We performed a GWAS on a southern European population to identify new CD risk loci.

Design We genotyped 620 901 genome markers on 1341 CD patients and 1518 controls from Spain. The top association signals representing new candidate risk loci were subsequently analysed in an independent replication cohort of 1365 CD patients and 1396 controls.

Results We identified a genome-wide significant association on chromosome 22q13.2 in the intergenic region between the *RBX1* and *EP300* genes (single nucleotide polymorphism rs4820425, OR 1.27, 95% CI 1.17 to 1.38, p=3.42E-8). We also found suggestive evidence for the association of the *IFNGR2* (21q22.11), *FOXP2* (7q31), *MACROD2* (20p12.1) and *AIF1* (6p21.3) loci with CD risk.

Conclusions In this GWAS performed on a southern European cohort, we have identified a new risk locus for CD between *RBX1* and *EP300*. This study demonstrates that using populations of different ancestry is a useful strategy to identify new risk loci for CD.

Significance of this study

What is already known on this subject?

- CD is a common inflammatory bowel disease with high heritability.
- Over the past few years, GWAS in populations of northern European ancestry have been successful in identifying more than 70 new loci convincingly associated with CD risk.
- The cumulative risk exerted by the CD risk loci identified so far is small and the likelihood that additional, undiscovered loci contribute to CD risk is very high.

What are the new findings?

- Using a GWAS on a large southern European cohort we have found a new risk locus for CD on chromosome 22q13.2 in the intergenic region between the *RBX1* and *EP300* genes.
- We have found suggestive evidence of an association with four additional loci with CD risk. One of these loci, *FOXP2*, is significantly associated with UC in a large GWAS meta-analysis of populations of northern European ancestry.
- Analysing large case–control cohorts of previously unscreened populations is a powerful strategy to uncover the genetic architecture of CD.

How might it impact on clinical practice in the foreseeable future?

- GWAS have been fundamental in highlighting the role of specific biological pathways in CD aetiology, some of which had not previously been associated with this inflammatory disease.
- Efforts are currently under way to translate this knowledge into new therapeutic approaches.
- The results from this GWAS increase our knowledge of the genetic basis of CD and therefore could contribute to the development of new and improved treatments.

INTRODUCTION

Crohn's disease (CD, OMIM (266600)) is one of the most common chronic inflammatory disorders of the gastrointestinal tract, with an average prevalence of approximately 70 cases per 100 000 in European populations.[1] Compared to other chronic inflammatory disorders, the contribution of genetic variation to CD susceptibility is high (sibling relative risk, λ_s=17–35).[2] Until very recently, however, only *NOD2* variants on chromosome 16q21 and the *IBD5* locus on chromosome 5q31 had been convincingly associated with CD susceptibility.[3] Over the past 5 years, genome-wide association studies (GWAS) have identified more than 70 new loci convincingly associated with CD.[4] Many of

To cite: Julià A, Domènech E, Ricart E, *et al*. *Gut* 2013;**62**:1440–1445.

these newly discovered loci involve key pathological features of CD, such as the epithelial barrier defence, immune cell recruitment or T-cell regulation.[3 5 6] Importantly, other associated loci involve specific biological pathways, such as autophagy, which had not previously been associated with CD pathology.[7] Together, these results confirm the success of the GWAS approach for the discovery of the genetic basis of CD.

The cumulative risk exerted by the CD risk loci identified so far is less than 30%.[4] Therefore, the likelihood that additional, undiscovered loci contribute to CD risk is very high. GWAS on large cohorts of previously unscreened populations should therefore be a powerful means to identify these new risk variants. In the present work, we have performed a GWAS on a large case–control cohort of southern European ancestry to identify new CD genetic risk factors. We have subsequently performed a replication study on an independent case–control cohort on those loci showing the strongest significance signals. Using a combined cohort of 5620 individuals, we have found a new risk locus for CD on chromosome 22q13.2 in the intergenic region between the *RBX1* and *EP300* genes as well as suggestive evidence of an association with four additional loci.

MATERIALS AND METHODS

Study subjects

To identify new loci associated with CD risk using the GWAS approach, we recruited 1341 CD patients and 1518 controls from the Spanish population. Both cohorts were collected as part of the Immune-Mediated Inflammatory Disease Consortium (IMIDC) between June 2007 and December 2010. The IMIDC is a network of Spanish researchers working on the genomic basis of immune-mediated inflammatory diseases. For this study, a total of 15 gastroenterology departments from different hospitals participated. All selected CD patients fulfilled the standard Lennard–Jones diagnostic criteria for CD.[8] To increase the homogeneity of the case cohort, only patients with at least 5 years of follow-up since diagnosis were included.

Control individuals were recruited from blood bank donors attending at 13 hospitals from different regions in Spain in collaboration with the Spanish National DNA Bank. All the controls included in this study were screened for the presence of CD or any autoimmune disorder, as well as for a family history of autoimmune disorders in first-degree relatives. Individuals with an autoimmune disease or a family history of autoimmune disease were excluded. To increase the hypernormality in this cohort, only control individuals who were more than 30 years old and who fulfilled the previous criteria were included in the study. In total, 1518 controls, 40% of whom were women, were genotyped. Of note, more than 95% of the control individuals were over 40 years old at the time of recruitment (mean age ±SD 49.7±8 years).

All case and control individuals in the GWAS and replication cohorts were Caucasian European and born in Spain. In the Spanish population, the impact of genetic variation due to ancestry is relatively small and comparable to other central Europe populations.[9 10] However, to reduce the potential impact of recent demographic events that could introduce stratification, the grandparents of both patients and controls also had to be born in Spain.

A total of 1365 CD patients and 1396 controls was used to validate the most significant loci identified in the GWAS phase. CD patients were selected using the same clinical and epidemiological selection criteria as for the GWAS. Of the replication cohort patients, 60% were collected from the IMIDC, whereas the remaining 40% were obtained from the ENEIDA project collection.[11] Control individuals were selected from the Spanish DNA bank repository using the same epidemiological criteria as in the discovery phase.

Informed consent was obtained from all participants, and protocols were reviewed and approved by local institutional review boards. This study was conducted in accordance with the Declaration of Helsinki principles.

Genotyping and quality control

The genome-wide scan was performed using Illumina Quad610 Beadchips (Illumina, San Diego, California, USA) on 1341 CD patients and 1518 controls. The Quad610 arrays scan over 550 000 single nucleotide polymorphisms (SNP) and have approximately 60 000 probes specific for copy number variant (CNV) detection. GWAS genotyping was performed at the Centro Nacional de Genotipado (CeGen, Spain). After excluding mitochondrial, X and Y chromosome SNP, a total of 600 470 markers was considered for GWAS analysis, of which 17 879 were exclusively CNV probes. The SNP genotype calling was performed using Illumina GenomeStudio software v2010.1 (Illumina, San Diego, California, US), and CNV calls were performed using CNstream software.[12] Only samples that had a greater than 95% genotype completion rate were considered for analysis (99.6% of samples), and only SNP that had a greater than 95% call rate (98.3% of SNP) and a minor allele frequency (MAF) greater than 0.05 (94.9% of SNP) and that showed Hardy–Weinberg equilibrium (99.6% of SNP, $p>0.0001$ in controls) were considered for association analysis. Also, SNP showing a differential rate of missingness between cases and controls (0.7% SNP significantly different at $p<1e\text{-}7$) were excluded from any further analysis. From this set of QC-filtered SNP (n=508 934), we selected those markers showing low pairwise linkage disequilibrium (LD; $r^2<0.2$, n=118 294 SNP) to infer the main axis of variation using the principal component approach implemented in EIGENSTRAT software.[13] Using the 10 first principal components, n=82 outlying samples were excluded. Supplementary figure S1A (available online only) shows the case and control distribution according to the first and second principal components after excluding the outliers. Supplementary figure S1B (available online only) also shows the projection of our GWAS cohort onto the eigenvectors calculated from the Hapmap reference samples. The genomic inflation factor was close to 1 ($\lambda=1.052$, see supplementary figure S2, available online only), and consequently, no adjustment was performed over the GWAS association statistics.

Replication genotyping was performed at the HudsonAlpha Institute for Biotechnology (Huntsville, Alabama, USA) using the TaqMan OpenArray Genotyping System (Applied Biosystems, Foster City, California, USA) on 1365 CD patients and 1396 controls. Similar quality control measures were applied, including genotyping call rate over 95%, sample completion rate over 90%, Hardy–Weinberg disequilibrium p value of control group $p>0.001$ and MAF greater than 5%. In the control group, three markers were excluded for low genotyping call rate, and one was excluded for significant Hardy–Weinberg disequilibrium . For the remaining SNP (n=28), the average call rate was 99%.

Statistical analyses

Using the set of QC-filtered GWAS data, we performed a case–control association analysis using an allelic χ^2 test of association using PLINK software V.1.07.[14] A total of 32 SNP was selected for replication. Of these, 31 SNP were the most significant association signals in the GWAS study ($p<6E\text{-}5$) representing new

candidate loci for CD. From the group of markers showing a moderate level of association (p>6E-5 to p<5E-4), we also selected SNP rs1869839 from the *FOXP2* locus. Recently, the forkhead box (FOX) transcription factors *FOXP1* and *FOXP3* were found to participate in key immune regulatory processes, and therefore we considered *FOXP2* to be a plausible candidate gene for this autoimmune disease.[15 16] The one-tailed p value of replicating markers was combined with the discovery p value using Fisher's combined probability method to produce the combined p value from both study cohorts.

We evaluated 71 loci associated with CD risk in previous GWAS on populations of northern European ancestry.[4] For each locus, we estimated the statistical power for replication of our GWAS cohort using genetic power calculator software.[17] For each of the known CD risk loci, we selected the marker showing the highest evidence of an association, as described in the recent large-scale GWAS meta-analysis performed by Franke *et al*.[4] Power estimation was performed using the reported allele frequencies and risk estimates and considering a significant level of replication at α=0.05. Next, for each of the 71 CD risk loci, we performed an allelic test of association to determine their association in our GWAS cohort. Of the 71 SNP, 39 were not directly genotyped by the Illumina Quad610 microarray and had to be imputed. For genotype imputation, we used MACH imputation software on Caucasian European Hapmap phased data (ie, CEU) as described previously.[18 19]

RESULTS

Table 1 summarises the main epidemiological variables of both the GWAS and the replication cohorts. After QC filtering, a total of 508 934 SNP was tested for their association with CD risk using 1277 patients and 1488 controls. Thirty-one out of the 71 previously known CD risk loci were replicated at p<0.05 (see supplementary table S1, available online only). After discarding previously associated loci, we found a new strong association signal on chromosome 22q13.2 (SNP rs4820425, OR 1.36, CI 1.21 to 1.54, p<5E-7) in the intergenic region between the genes for the RING-box protein 1 (*RBX1*) and the E1A binding protein p300 (*EP300*) (figure 1). The second most significant new association signal in the CD GWAS was found in a SNP in the third intron of the gene for the interferon gamma receptor 2 (*IFNGR2*) (rs2834215, OR 0.88, CI 0.79 to 0.98, p<1.3 E-6). In addition, 29 other loci were identified at significance levels of p>1.3 E-6 to p<6E-5.

Table 1 GWAS and replication cohort epidemiological and clinical statistics

	GWAS CD	GWAS controls	Replication CD	Replication controls
% Female	50%	40%	49%	48%
Age (median±SD)	42.6 (12.8)	49.7 (8)	42.9 (14.6)	44.4 (14.5)
% Familial autoimmunity	32.7%	0%	32.9%	0%
% Familial IBD	17.8%	0%	17.5%	0%
% Familial CD	11.7%	0%	12%	0%
% ≥5 years of disease duration	100%	NA	93%	NA
Age at diagnosis (IQR)	22–37	NA	22–41	NA

Familial autoimmunity: presence of one or more first or second order relatives with an autoimmune disease (including IBD).
CD, Crohn's disease; GWAS, genome-wide association study; IBD, inflammatory bowel disease; NA, not applicable.

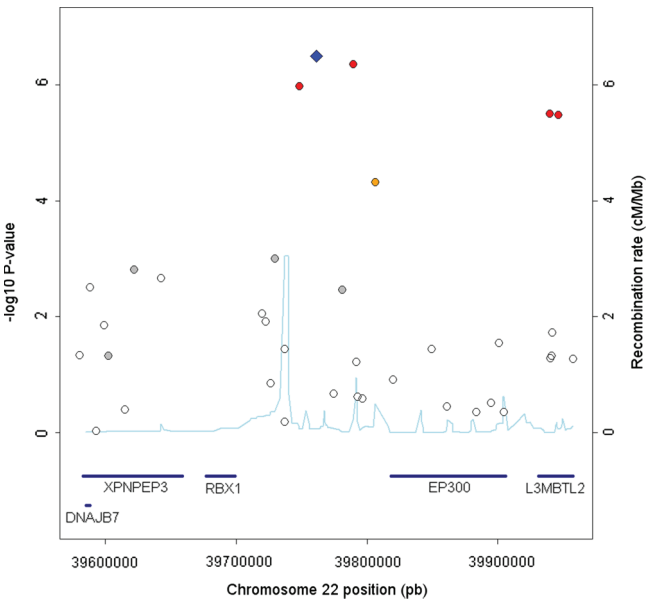


Figure 1 Recombination plot showing the association between Crohn's disease risk and the *RBX1-EP300* locus. Plot showing the pattern of associations at chromosome 22q13.2 in the genome-wide association studies stage, the recombination rate (build 36 HapMap Caucasian European) and the genes located in the region. The single nucleotide polymorphism (SNP) showing the main signal is labelled as a blue diamond (rs4820425); other SNP are colour-coded according to their linkage disequilibrium with rs4820425 (red r^2>0.8, orange 0.5–0.8, grey 0.2–0.5, white <0.2).

From the group of SNP showing a moderate level of association (p>6E-5 to p<5E-4), we also selected SNP rs1869839 from the *FOXP2* locus (p<2.2 E-4) for replication.

Using an independent cohort of 1365 CD patients and 1396 controls, the SNP from *RBX1-EP300*, *IFNGR2*, *AIF1*, *MACROD2* and *FOXP2* were nominally replicated (p<0.05, table 2) (see supplementary figure S3, available online only). Applying a standard genome-wide significance threshold of 5E-8 for the combined GWAS and replication signals, we found that SNP rs4820425 from *RBX1-EP300* surpassed this threshold. The second most associated SNP in the GWAS analysis, *IFNGR2*, was also significant in the combined sample, although it did not reach a genome-wide level of statistical association ($p_{combined}$=4.43E-07, OR 0.82, CI 0.76 to 0.89).

A total of 2674 CNV was identified in our GWAS cohort using CNstream software. For this group of CNV, we found no evidence of an association with CD risk at p<1E-4 (see supplementary figure S4, available online only).

With the sample size of our GWAS case–control cohort and according to previously reported effect sizes, we estimated a greater than 80% power of replication in 17 out of 71 known CD risk loci (see supplementary table S1, available online only). Testing all 71 SNP in our GWAS cohort, we replicated the association of 31 of these markers with CD risk (p<0.05, see supplementary table S1, available online only).

The associated *RBX1-EP300* locus SNP, rs4820425, is 62.3 kb from the *RBX1* 3′ end and 57.3 kb from the *EP300* 5′ end. The LD patterns in this region both in our GWAS data and in the Hapmap CEU data indicate that the haplotype block that includes rs4820425 is close to the *EP300* promoter region (see supplementary figure S5, available online only). Examining publicly available expression quantitative trait locus (eQTL) databases, we found that the strongest eQTL for *EP300* gene

Table 2 GWAS and replication association statistics

SNP	Chr	Coordinate	Gene(s)	Minor allele	MAF cases	MAF controls	OR GWAS (95% CI)	p GWAS	OR replication (95% CI)	p Replication	OR combined (95% CI)	p Combined
rs9348876	6	31683255	*AIF1*	T	0.09	0.06	1.63 (1.32 to 2)	5.30E-06	1.22 (1 to 1.5)	0.029	1.41 (1.22 to 1.63)	2.60E-06
rs1869839	7	114144779	*FOXP2*	G	0.32	0.36	0.81 (0.72 to 0.9)	0.00021	0.85 (0.76 to 0.95)	0.0022	0.83 (0.77 to 0.9)	7.30E-06
rs6105269	20	14348165	*MACROD2*	A	0.39	0.34	1.25 (1.12 to 1.39)	7.92E-05	1.13 (1.01 to 1.26)	0.017	1.19 (1.1 to 1.28)	1.99E-05
rs2834215	21	33718756	*IFNGR2*	A	0.36	0.42	0.76 (0.68 to 0.85)	1.29E-06	0.88 (0.79 to 0.98)	0.012	0.82 (0.76 to 0.89)	2.89E-07
rs4820425	22	39761288	*RBX1-EP300*	A	0.3	0.24	1.36 (1.21 to 1.54)	3.76E-07	1.18 (1.04 to 1.33)	0.0043	1.27 (1.17 to 1.38)	3.42E-08

GWAS, genome-wide association study; MAF, minor allele frequency in the GWAS cohort.

expression is cis SNP rs4821990 ($p<1.6E-7$), which is only 7.4 kb from rs4820425. This gene expression correlation was identified on a recent eQTL GWAS performed on transcriptomes from circulating monocytes of 1490 individuals genotyped with Affymetrix Genome-Wide Human SNP Array 6.0.[20] Although this genotyping platform does not include the SNP associated in our study, both SNP are in perfect LD ($r^2=1$, CEU Hapmap data), suggesting that they represent a common eQTL association with *EP300* expression.

Evaluating the significance of the loci associated in our study with the association data available at the International IBD Genetics Consortium (IIBDGC, http://www.ibdgenetics.org/), we were able to confirm that *RBX1-EP300* is also associated with CD risk in this large meta-analysis ($p=0.0035$, see supplementary data, available online only). As in our GWAS study, SNP rs4820425 is the strongest association signal within the *RBX1-EP300* locus.

The other four loci nominally replicated in our study do not attain genome-wide significance, and therefore, additional evidence is necessary to confirm their association with CD risk. In the IIBDGC data, SNP rs2834215 (*IFNGR2*), rs1869839 (*FOXP2*) rs6105269 (*MACROD2*) are not significantly associated ($p>0.05$). Importantly, however, other markers in the same LD block show nominally significant association with CD ($p<0.001$, see supplementary figure S5, available online only). In particular, SNP rs2284553 from the first intron of *IFNGR2* shows a remarkable nominal association signal with CD risk in northern European ancestry cohorts ($p<1.7E-5$). The fact that the LD blocks from Hapmap CEU data encompass our association signal as well as the significant signals from the IIBDGC data increases the likelihood that these are risk loci for CD. This result also raises the possibility that as yet unidentified genetic variants within these loci are associated with CD. Therefore, additional studies using more dense sequence information in these regions will be needed to confirm definitively their implication with disease risk.

In a recent meta-analysis of three GWAS in CD[6] a genome-wide association was identified for the first time within the *HLA* region (SNP rs3763313, $p<5e-8$). In our GWAS, however, the association for this SNP was considerably lower ($p=0.033$). Instead, our strongest signal within the *HLA* region was 800 kb upstream, close to the *AIF1* gene (SNP rs9348876, $p=5.3e-6$). Importantly, when the previous CD meta-analysis was expanded to include three additional GWAS[4] the strongest signal within the *HLA* region was notably closer to our signal (SNP rs1799964, at only 32 kb from SNP rs9348876). SNP rs1799964 was not available in our genotyping platform; however, after imputing its genotype we found a lower significance compared to our most associated SNP ($p=0.0007$). Furthermore, conditioning the association of *AIF1* SNP rs9348876 to rs1799964 genotype, we found that the association with CD risk was still strong ($p=0.0001$). Therefore, while both our study and the meta-analysis of Franke *et al*[4] identify the strongest signal close to the *AIF1* gene, our study further suggests that the risk conferred by this region does not depend on a single variation but, instead, it may be influenced by multiple independent genetic variants in this region.

Many CD risk genes are also associated with ulcerative colitis (UC) susceptibility.[21] On evaluating the association signals in the IIBDGC meta-analysis for UC, we found that *EP300*, *IFNGR2* and *MACROD2* do not seem to be common risk factors for UC. The *FOXP2* SNP, rs1869839, however, shows significant evidence of replication with UC risk ($p=0.00051$). The same allele associated with CD risk in our study is associated with UC in that meta-analysis, suggesting a common biological pathway for both diseases. To our knowledge, this is the first study to suggest an association of the *FOXP2* locus with an autoimmune disease.

DISCUSSION

Using a GWAS on a large case–control cohort of southern European ancestry, we have identified a new genome-wide significant risk locus for CD in the intergenic region between the *RBX1* and *EP300* genes on chromosome 22q13.2. We have also found suggestive evidence of an association with four other loci not previously associated with CD risk. As expected, a significant number of loci previously associated with CD in GWAS in populations of northern European ancestry were also associated. As more large-scale GWAS are performed in populations of different ancestry, we can expect the number of newly identified CD risk loci to increase.

The *EP300* gene encodes an acetyltransferase that interacts with several transcription factors.[22 23] Mutations in this gene have been associated with colorectal cancer (OMIM #114500).[24] The EP300 protein has been shown to interact with STAT3, a risk locus for CD,[25] and therefore, it could contribute to CD risk through the previously associated IL-23 signalling pathway.[26] There is functional evidence, however, that EP300 is a critical factor in CD4 T-cell differentiation, stabilising Th-inducing POK, a key transcription factor in the CD4 lineage.[27] More important, there is recent evidence that EP300 is crucial for the FOXP3-dependent differentiation of CD4 regulatory T cells.[28] Therefore, genetic variations influencing *EP300* regulation could affect self-tolerance and consequently increase the risk of an autoimmune process. Very recently, genome-wide eQTL analyses on immune cell transcriptomes have shown that lysozyme gene (ie, *LYZ*) expression in monocytes is tighly regulated by a promoter SNP that is likely to harbour an EP300 protein binding site.[29] Gene expression correlation between *LYZ* and *EP300* under the eQTL ancestral allele support this finding. Of relevance, we found suggestive

evidence that *RBX1-EP300* locus SNP, rs4820425, is a strong cis-eQTL for *EP300*, and this evidence was also found specifically in monocytes. Therefore, these results also suggest that *EP300* could contribute to CD risk through the regulation of this antimicrobial peptide in monocytes/macrophages.[30]

The analysis of populations with different genetic backgrounds can increase our ability to identify the genetic risk components for CD. To date, however, most large-scale studies of CD have focused on populations of northern European origin.[3 4 6 31 32] More recently, a GWAS performed on a population of Jewish origin has identified new risk loci for CD.[33] Our study is the first GWAS performed on a large case–control cohort of southern European ancestry. The observation of genetic heterogeneity with CD risk factors within European populations is not new and has previously been demonstrated for *NOD2*.[34] In particular, it has been shown that the *NOD2* Leu1007fsinC frameshift variation is substantially lower in northern populations such as Ireland (4%) or Sweden (2%) but is frequent in southern countries such as Spain (14%) or France (12%). A more extreme case for population-specific variation in CD genetic aetiology is present in the Asian population, in which there has been scarce replication of risk loci identified in European populations.[35] For *IRGM1* locus, this lack of association even extends to major functional differences between both populations, suggesting the interplay of complex genetic mechanisms in the risk of developing CD.[36] Therefore, as yet unidentified genetic and environmental factors could influence our ability to identify new risk loci for CD.

In a highly heterogeneous disease such as CD, other factors besides the population origin of samples can influence the power to detect genetic associations.[37] The existence of several subphenotypes strongly associated with one or another risk variant could impair or maximise our ability to associate them with CD risk.[38 39] The study, for example, of early-onset patients has provided additional power to identify new genetic variants associated with CD risk that could not be detected by previous genome-wide scans.[40] Therefore, additional studies evaluating the association of genetic factors with specific CD clinical subphenotypes are needed. In our study, both the GWAS and the replication cohorts had very similar demographic and clinical parameters, which should have reduced the level of phenotype heterogeneity and consequently increased our power to identify and replicate new genetic variants. Finally, we used hypernormal controls (ie, individuals with a low liability to develop an autoimmune disease) that, while a more costly recruitment strategy, can lead to a substantial gain in statistical power.[31]

In summary, this CD GWAS on a southern European population has led to the identification of a novel risk locus for this disease in the region between *RBX1* and *EP300*. Our study has identified suggestive evidence for four additional risk loci for CD, one of which, *FOXP2*, could also be a risk factor for UC. This study increases the knowledge of inflammatory bowel disease risk factors and demonstrates the power of using populations of different origin.

Author affiliations
[1]Rheumatology Research Group, Vall d'Hebron Research Institute, Barcelona, Spain
[2]Digestive System Service, Hospital Universitari Germans Trias i Pujol, Badalona, Spain
[3]Centro de Investigación Biomédica en Red de Enfermedades Hepáticas y Digestivas (CIBEREHD), Spain
[4]Gastroenterology Department, Hospital Clínic de Barcelona, IDIBAPS, Barcelona, Spain
[5]Digestive System Service, Hospital Universitario Reina Sofía, Córdoba, Spain
[6]Gastroenterology Service, Hospital Universitario de la Princesa and IP, Madrid, Spain
[7]Digestive Medicine Service, Hospital la Fe, Valencia, Spain
[8]Gastroenterology Service, Hospital General de Alicante, Alicante, Spain
[9]Digestive System Service, Hospital Clínico Universitario, Zaragoza, Spain
[10]Gastroenterology Service, Hospital Clínico San Carlos, Madrid, Spain
[11]Gastroenterology Department, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain
[12]Gastroenterology Service, Hospital Clínico Universitario, Santiago de Compostela, Spain
[13]Gastroenterology Service, Complejo Hospitalario de León, León, Spain
[14]Gastroenterology Service, Hospital Universitario Puerta de Hierro, Madrid, Spain
[15]Internal Medicine Service, Hospital de Cabueñes, Gijón, Spain
[16]Gastroenterology Service, Hospital Universitari Mutua de Terrassa, Barcelona, Spain
[17]Department of Gastroenterology, IMIM, Institute of Research Hospital del Mar, Parc de Salut Mar. Pompeu Fabra University, Barcelona, Spain
[18]Barcelona Supercomputing Centre, Life Sciences, National Institute of Bioinformatics, Barcelona, Spain
[19]Department of Biochemistry and Molecular Biology, University of Barcelona, Barcelona, Spain
[20]Banco Nacional de ADN Carlos III, University of Salamanca, Salamanca, Spain
[21]Nacional Genotyping Centre (CeGen), Universitat Pompeu Fabra, Barcelona, Spain
[22]HudsonAlpha Institute for Biotechnology, Huntsville, Alabama, USA

Correction notice This article has been corrected since it was published Online First. The author surname Bertanpetit has been amended to Bertranpetit.

Acknowledgements The authors would like to acknowledge the use of the published results from the International Inflammatory Bowel Disease Genetics Consortium (IIBDGC) meta-analysis (results available at: http://www.ibdgenetics.org/). They wish to thank Genoma España Foundation, the Barcelona Supercomputing Centre (BSC), Instituto Nacional de Bioinformática (INB), the Centro Nacional de Genotipado (CeGEN), the Banco Nacional de ADN (USAL) and the Hudson Alpha Institute for Biotechnology for their support. They are also grateful to Roche Pharma, Pfizer and Merck Sharp and Dome for the support of the IMID Consortium, as well as Abbott, Bristol Myers Squibb and UCB.

Contributors AJ, ED, JPG, JP and SM conceived and designed the study; ED, ER, VGS, JPG, PNM, AG, FG, JLM, EGP, MBdA, FM, MV, CS, ME, MA, JP and MLL acquired the data; AJ and SM analysed the data; AJ, ED, JPG, JP and SM interpreted the data; AJ, ED, JPG, JP and SM drafted the manuscript; RT, AA, LC, JLG, ACGM, JB and DA provided technical support.

Funding This study was funded by of the Spanish Science and Innovation Ministry, grant no. PSE-010000-2006-6. The study sponsor had no role in the collection, analysis or interpretation of the data.

Competing interests None.

Ethics approval Protocols were reviewed and approved by local institutional review boards. This study was conducted in accordance with the principles of the Declaration of Helsinki.

Patient consent Obtained.

Provenance and peer review Not commissioned; externally peer reviewed.

REFERENCES

1 Loftus EV Jr. Clinical epidemiology of inflammatory bowel disease: incidence, prevalence, and environmental influences. *Gastroenterology* 2004;126:1504–17.
2 Khor B, Gardet A, Xavier RJ. Genetics and pathogenesis of inflammatory bowel disease. *Nature* 2011;474:307–17.
3 Duerr RH, Taylor KD, Brant SR, *et al*. A genome-wide association study identifies IL23R as an inflammatory bowel disease gene. *Science* 2006;314:1461–3.
4 Franke A, McGovern DP, Barrett JC, *et al*. Genome-wide meta-analysis increases to 71 the number of confirmed Crohn's disease susceptibility loci. *Nat Genet* 2011;42:1118–25.
5 Lees CW, Barrett JC, Parkes M, *et al*. New IBD genetics: common pathways with other diseases. *Gut* 2011;60:1739–53.
6 Barrett JC, Hansoul S, Nicolae DL, *et al*. Genome-wide association defines more than 30 distinct susceptibility loci for Crohn's disease. *Nat Genet* 2008;40:955–62.
7 Hirschhorn JN. Genomewide association studies – illuminating biologic pathways. *N Engl J Med* 2009;360:1699–701.
8 Lennard-Jones JE. Classification of inflammatory bowel disease. *Scand J Gastroenterol Suppl* 1989;170:2–6; discussion 16–19.
9 Julià A, Ballina J, Cañete J, *et al*. Genome-wide association study of rheumatoid arthritis in the Spanish population: KLF12 as a risk locus for rheumatoid arthritis susceptibility. *Arthritis Rheum* 2008;58:2275–86.
10 Nelis M, Esko T, Magi R, *et al*. Genetic structure of Europeans: a view from the north-east. *PLoS One* 2009;4:e5472.

11 Chaparro M, Panes J, Garcia V, *et al*. Long-term durability of infliximab treatment in Crohn's disease and efficacy of dose "escalation" in patients losing response. *J Clin Gastroenterol* 2011;45:113–18.
12 Alonso A, Julia A, Tortosa R, *et al*. CNstream: a method for the identification and genotyping of copy number polymorphisms using Illumina microarrays. *BMC Bioinformatics* 2010;11:264.
13 Price AL, Patterson NJ, Plenge RM, *et al*. Principal components analysis corrects for stratification in genome-wide association studies. *Nat Genet* 2006;38:904–9.
14 Purcell S, Neale B, Todd-Brown K, *et al*. PLINK: a tool set for whole-genome association and population-based linkage analyses. *Am J Hum Genet* 2007;81:559–75.
15 Feng X, Wang H, Takata H, *et al*. Transcription factor Foxp1 exerts essential cell-intrinsic regulation of the quiescence of naive T cells. *Nat Immunol* 2011;12:544–50.
16 Sakaguchi S, Yamaguchi T, Nomura T, *et al*. Regulatory T cells and immune tolerance. *Cell* 2008;133:775–87.
17 Purcell S, Cherny SS, Sham PC. Genetic power calculator: design of linkage and association genetic mapping studies of complex traits. *Bioinformatics* 2003;19:149–50.
18 Hung RJ, McKay JD, Gaborieau V, *et al*. A susceptibility locus for lung cancer maps to nicotinic acetylcholine receptor subunit genes on 15q25. *Nature* 2008;452:633–7.
19 Li Y, Willer CJ, Ding J, *et al*. MaCH: using sequence and genotype data to estimate haplotypes and unobserved genotypes. *Genet Epidemiol* 2010;34:816–34.
20 Zeller T, Wild P, Szymczak S, *et al*. Genetics and beyond–the transcriptome of human monocytes and disease susceptibility. *PLoS One* 2010;5:e10693.
21 Anderson CA, Boucher G, Lees CW, *et al*. Meta-analysis identifies 29 additional ulcerative colitis risk loci, increasing the number of confirmed associations to 47. *Nat Genet* 2011;43:246–52.
22 Liu L, Scolnick DM, Trievel RC, *et al*. p53 sites acetylated in vitro by PCAF and p300 are acetylated in vivo in response to DNA damage. *Mol Cell Biol* 1999;19:1202–9.
23 Perkins ND, Felzien LK, Betts JC, *et al*. Regulation of NF-kappaB by cyclin-dependent kinases associated with the p300 coactivator. *Science* 1997;275:523–7.
24 Gayther SA, Batley SJ, Linger L, *et al*. Mutations truncating the EP300 acetylase in human cancers. *Nat Genet* 2000;24:300–3.
25 Chen LF, Greene WC. Shaping the nuclear action of NF-kappaB. *Nat Rev Mol Cell Biol* 2004;5:392–401.
26 Wang K, Zhang H, Kugathasan S, *et al*. Diverse genome-wide association studies associate the IL12/IL23 pathway with Crohn disease. *Am J Hum Genet* 2009;84:399–405.
27 Zhang M, Zhang J, Rui J, *et al*. P300-mediated acetylation stabilizes the Th-inducing POK factor. *J Immunol* 2010;185:3960–9.
28 van Loosdregt J, Vercoulen Y, Guichelaar T, *et al*. Regulation of Treg functionality by acetylation-mediated Foxp3 protein stabilization. *Blood* 2010;115:965–74.
29 Fairfax BP, Makino S, Radhakrishnan J, *et al*. Genetics of gene expression in primary immune cells identifies cell type-specific master regulators and roles of HLA alleles. *Nat Genet* 2012;44:502–10.
30 Davis KM, Nakamura S, Weiser JN. Nod2 sensing of lysozyme-digested peptidoglycan promotes macrophage recruitment and clearance of *S. pneumoniae* colonization in mice. *J Clin Invest* 2011;121:3666–76.
31 WTCC. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature* 2007;447:661–78.
32 Libioulle C, Louis E, Hansoul S, *et al*. Novel Crohn disease locus identified by genome-wide association maps to a gene desert on 5p13.1 and modulates expression of PTGER4. *PLoS Genet* 2007;3:e58.
33 Kenny EE, Pe'er I, Karban A, *et al*. A genome-wide scan of Ashkenazi Jewish Crohn's disease suggests novel susceptibility loci. *PLoS Genet* 2012;8:e1002559.
34 Gaya DR, Russell RK, Nimmo ER, *et al*. New genes in inflammatory bowel disease: lessons for complex diseases? *Lancet* 2006;367:1271–84.
35 Nakagome S, Takeyama Y, Mano S, *et al*. Population-specific susceptibility to Crohn's disease and ulcerative colitis; dominant and recessive relative risks in the Japanese population. *Ann Hum Genet* 2010;74:126–36.
36 Prescott NJ, Dominy KM, Kubo M, *et al*. Independent and population-specific association of risk variants at the IRGM locus with Crohn's disease. *Hum Mol Genet* 2010;19:1828–39.
37 Gasche C, Grundtner P. Genotypes and phenotypes in Crohn's disease: do they help in clinical management? *Gut* 2005;54:162–7.
38 Elding H, Lau W, Swallow DM, *et al*. Dissecting the genetics of complex inheritance: linkage disequilibrium mapping provides insight into Crohn disease. *Am J Hum Genet* 2011;89:798–805.
39 Parkes M, Barrett JC, Prescott NJ, *et al*. Sequence variants in the autophagy gene IRGM and multiple other replicating loci contribute to Crohn's disease susceptibility. *Nat Genet* 2007;39:830–2.
40 Imielinski M, Baldassano RN, Griffiths A, *et al*. Common variants at five new loci associated with early-onset inflammatory bowel disease. *Nat Genet* 2009;41:1335–40.

8) Buitrago, D†, . Codó, L.†, Illa, R., de Jorge, P., Battistini, F., Flores, O., … Orozco, M. (2019). Nucleosome Dynamics: a new tool for the dynamic analysis of nucleosome positioning. Nucleic Acids Research. https://doi.org/10.1093/nar/gkz759

† *The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors*

Nucleic Acids Research, 2019 **1**
doi: 10.1093/nar/gkz759

Nucleosome Dynamics: a new tool for the dynamic analysis of nucleosome positioning

Diana Buitrago[1,†], Laia Codó[2,†], Ricard Illa[1], Pau de Jorge[1], Federica Battistini[1], Oscar Flores[1], Genis Bayarri[1], Romina Royo[2], Marc Del Pino[2], Simon Heath[3], Adam Hospital[1], Josep Lluís Gelpí[2,4], Isabelle Brun Heath[1] and Modesto Orozco[1,4,*]

[1]Institute for Research in Biomedicine (IRB Barcelona), The Barcelona Institute of Science and Technology, Baldiri Reixac 10, Barcelona 08028, Spain, [2]Barcelona Supercomputing Center (BSC), Jordi Girona 31, Barcelona 08028. Spain, [3]Centro Nacional de Análisis Genómico (CNAG-CRG), Centre de Regulacio Genómico (CRG), The Barcelona Institute of Science and Technology, Barcelona, Spain and [4]Departament de Bioquímica i Biomedicina. Facultat de Biologia, Universitat de Barcelona, Avgda Diagonal 647, Barcelona 08028, Spain

Received June 03, 2019; Revised July 23, 2019; Editorial Decision August 06, 2019; Accepted August 22, 2019

ABSTRACT

We present Nucleosome Dynamics, a suite of programs integrated into a virtual research environment and created to define nucleosome architecture and dynamics from noisy experimental data. The package allows both the definition of nucleosome architectures and the detection of changes in nucleosomal organization due to changes in cellular conditions. Results are displayed in the context of genomic information thanks to different visualizers and browsers, allowing the user a holistic, multidimensional view of the genome/transcriptome. The package shows good performance for both locating equilibrium nucleosome architecture and nucleosome dynamics and provides abundant useful information in several test cases, where experimental data on nucleosome position (and for some cases expression level) have been collected for cells under different external conditions (cell cycle phase, yeast metabolic cycle progression, changes in nutrients or difference in MNase digestion level). Nucleosome Dynamics is a free software and is provided under several distribution models.

INTRODUCTION

Eukaryotic chromatin is organized in a hierarchical manner, where the basic structural units are repetitive elements named nucleosomes. Each of them is defined by around 147 base pairs of DNA wrapped around a protein octamer, the histones. The position of the nucleosomes in the cell is not random and recurrent patterns have been detected in cell populations (1–3), indicating a maintenance of the nucleosome architecture which seems to be crucial for a correct regulation of genome activity (4). The protein octamer serves as an anchoring point for proteins recognizing histone epigenetic signals, while unwrapped DNA is targeted by transcription factors and enhancers (5,6). Thus, nucleosomes shifting due to alterations in the sequence (7), DNA methylation (8) or the action of chromatin remodelers (5,9–13) can result in dramatic changes in gene expression. Characterizing such changes is crucial for the understanding of the connection between chromatin structure and genome functionality (6).

Experimental determination of nucleosome positioning is typically performed by treating a group of cells (in the range 10^6-10^9) with enzymes acting on nucleosome-free DNA. ATAC-seq (14) uses a hyperactive transposase for tagging nucleosome-free DNA segments for sequencing (the linkers). MNase-seq, the most widely used technique for nucleosome localization, uses Micrococcal nuclease to degrade linker DNA preserving the DNA segments wrapped in the nucleosomes, which are then sequenced. Both MNase-seq and ATAC-seq, after filtering nucleosomal reads by size (14), provide at the end the same type of information: DNA reads that need to be grouped into individual nucleosomes using a variety of computational approaches (15–18), which in all cases suffer from the intrinsic dispersion in read coverage. The resulting nucleosome maps show well defined depleted regions (the nucleosome free regions, NFR), well-positioned (W) nucleosomes, and a large number of 'fuzzy' (F) nucleosomes giving partial protection signals longer than 147 bps (19). Fuzzy positioning signals are the result of nucleosomes not being in exactly the same

*To whom correspondence should be addressed. Tel: +34 93 40 37156; Fax: +34 93 40 37157; Email: modesto.orozco@irbbarcelona.org
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

genomic position in the cell population, and are intrinsically difficult to annotate by any nucleosome calling algorithms (15,17). While it is known (20,21) that this technique can be affected by MNase concentration and sequence-preference biases that affect the detection of the so called 'fragile' nucleosomes, it is still the most widely used to detect nucleosome positioning for its versatility and accuracy. In 2012, Brogaard *et al.* developed a chemical cleavage method that provides a very accurate positioning of nucleosomes (22). However, this technique requires to do genetic engineering replacing the endogenous histone H4 (or H3 (23)) by a mutated version, therefore restricting its use (24–27). Moreover, it has been shown that the MNase sequence bias can be corrected using digested naked DNA as baseline (20,21), obtaining more pronounced nucleosome coverage peaks.

The noisy nature of experimental data such as MNase-seq, makes very difficult to compare nucleosome architecture in two samples, as the signal is masked by the intrinsic fuzziness of the maps. Methods available such as DANPOS and Dimnp (15,28) can detect only a limited number of changes affecting large percentage of the cells, as they work at the level of the fragment coverage, missing the opportunity to work with the raw data: the fragments themselves.

We present here Nucleosome Dynamics, a complete virtual framework to characterize the structure and dynamics of nucleosome architectures. The package consists of two main blocks: an improved version of our nucleR algorithm for nucleosome location (17), and NucDyn, an algorithm specifically created to detect changes (shifts, evictions and insertions) in nucleosome architectures based on the direct processing of raw data (the sequencing reads) obtained from pairs of experiments. The Nucleosome Dynamics package (available under the Apache 2.0 License) can be installed from the source code, obtained from BioConductor (29), or run as a web tool hosted by the MuGVRE workspace (30) as well as in a Galaxy server (31), where additional analysis algorithms, browsers and visualization tools are included.

MATERIALS AND METHODS

Package overview

The input data for Nucleosome Dynamics is one or several files containing sequence reads aligned to the reference genome and stored in BAM format. The user can select (see Figure 1) between: (i) processing a single file to define the consensus nucleosome architecture using an extended version of nucleR (17) or (ii) detecting changes in nucleosome distribution between two experiments, by comparing pairs of mapped sequence files using the newly developed NucDyn module. For a complete description of the parameters of the available functionalities, see Supplementary Table S1. In the MuGVRE implementation (see Table 1), the user has access to a wide range of analysis and visualization tools to characterize the nucleosome patterns, their changes across different conditions, and to put all the data in the context of other information mapped to the genome (genome structure, expression, epigenetic signals, etc). We have evaluated the performance of nucleR and NucDyn generating synthetic nucleosome maps and have tested their descriptive power using publicly available nucleosome positioning experimental data.

Single experiment analysis

Nucleosome positioning and coverage. BAM files from a single MNase-seq experiment are processed to define nucleosome coverage, which can be directly visualized using a genome browser (Figure 2A) or processed to obtain nucleosome positions. Accordingly, following signal theory the read coverage is described as a combination of periodic waves, which are then subjected to Fast Fourier Transformation (FFT) to remove the high frequency components responsible for the noise (see Supplementary Figure S1). The parameters for FFT filtering can be adjusted taking into account the nucleosome repeat length and noise level of each organism and cell type (see Supplementary Table S1). Clean profiles are processed to annotate the nucleosome dyads (located at the local maxima of the distributions). Putative nucleosomes are then scored based on the shape of the associated peaks (see Supplementary Figure S1). Those leading to sharp signals are labelled as well-positioned (W) nucleosomes (high localization score), while flat peaks are labelled as 'fuzzy' (F) nucleosomes (low localization score). Once all nucleosomes are localized, the software analyses the nucleosome architecture (see Supplementary Figure S2A) around the transcription start sites (TSS) and classifies the nucleosome architecture for each gene based on (19): the extension of the nucleosome free region (NFR) around the core promoter (open (o), closed (c) or missing −1 or +1 nucleosome) and the degree of localization of the +1 (downstream the TSS) and −1 (upstream the TSS) nucleosomes (see Supplementary Figure S2A). Data are presented at the individual gene level as well as summarized at global level (Figure 2). Nucleosome Dynamics performs a global detection of all NFRs, as these regions usually are the main recognition sites for effector proteins, and well-defined and extended NFRs typically signal active regions in the genome.

Periodicity at coding regions. The software evaluates the periodicity in the nucleosome pattern inside the genes, following signal propagation theory from two 'emitting sites' located at well-positioned nucleosomes. The first signal comes from the 5′ end of the gene (the +1 nucleosome located just downstream the TSS) and the second from the 3′ end of the gene (the –last nucleosome located just upstream the transcription termination site; TTS). We assume that both signals proceed in opposite directions (from +1 to –last nucleosome) following an exponential decay periodic function (32). We found out that when the +1 and –last originated waves are in phase the signals sum up and nucleosomes are well located inside the gene body, while when they are in antiphase the signals cancel out and the gene typically shows fuzzy nucleosomes. The periodicity (T) of the signal is obtained by maximizing the autocorrelation function (Equation 1: see an example in Supplementary Figure S2):

$$R(T) = \int_{X_2}^{X_1} I(x) * I(x - T)dx \quad (1)$$

where X_1 and X_2 are the intervals of the window, $I(x)$ stands for the coverage. This value will be dependent on the nucleosome repeat length of each species and cell type (see Supplementary Table S2 for suggested T in different cell types).

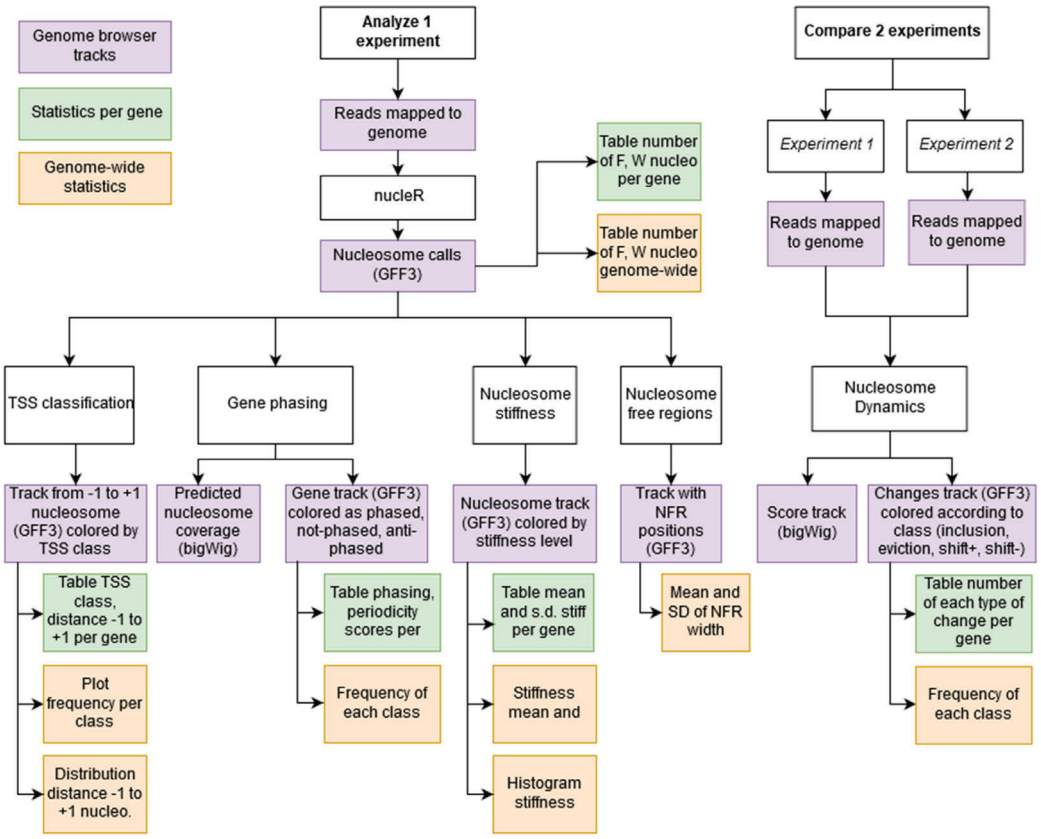


Figure 1. Analysis pipeline for Nucleosome Dynamics. A single MNase-seq experiment can be analysed, obtaining: nucleosome calls with nucleR, their fuzzy/well-positioned classification and stiffness estimation, Nucleosome Free Regions location, classification of TSS according to −1 and +1 nucleosomes, and nucleosome phasing along the gene body. Comparing two MNase-seq experiments, NucDyn identifies hotspots of changes (SHIFT +, SHIFT -, INCLUSION and EVICTION), and reports a significance score of the difference in the coverage profiles at base-pair level. Summary statistics per gene as well as genome-wide are also reported for each calculation.

Table 1. Implementation models for Nucleosome Dynamics

Code distribution		
Standalone installation	Nucleosome Dynamics CLI	https://github.com/nucleosome-dynamics/nucleosome_dynamics
	nucleR R package	https://github.com/nucleosome-dynamics/nucleR Bioconductor: http://bioconductor.org/packages/nucleR/
	NucDyn R package	https://github.com/nucleosome-dynamics/NucDyn Bioconductor: (*in review*)
Containerized installation	Docker	https://github.com/nucleosome-dynamics/docker Docker-hub: mmbirb/nucleosome-dynamics
	Singularity	https://github.com/nucleosome-dynamics/nucleosome_dynamics_singularity Singularity-hub: https://singularity-hub.org/collections/2579
Platforms in use		
MuG Virtual Research Environment		https://vre.multiscalegenomics.eu/workspace/?from=nucldynwf
Galaxy Platform		https://dev.usegalaxy.es (in development) Galaxy ToolShed: https://toolshed.g2.bx.psu.edu/repository?repository_id=822e9c879cf92fd0

A 'phased' gene is defined when the distance between +1 and –last nucleosome is close to a multiple of T (Supplementary Figure S2B). An 'antiphased' gene is defined when the modulus of the ratio between the distance of the +1 and –last nucleosomes and T is close to T/2 (Supplementary Figure S2C). The package provides a theoretical nucleosome map based on signal propagation theory with +1 and –last nucleosomes as emitting sites. Comparison of the predicted and the real nucleosome maps helps to detect anomalies in the gene nucleosome distribution emerging from interacting proteins or from the effect of the remodelling machinery.

Nucleosome stiffness. Nucleosome Dynamics also analyses the sliding propensities of nucleosomes by computing its apparent resistance to be displaced along the sequence. For this purpose, we map the original reads around located nucleosomes and estimate the normalized Gaussian that better fits the distribution of reads (see Supplementary Figure

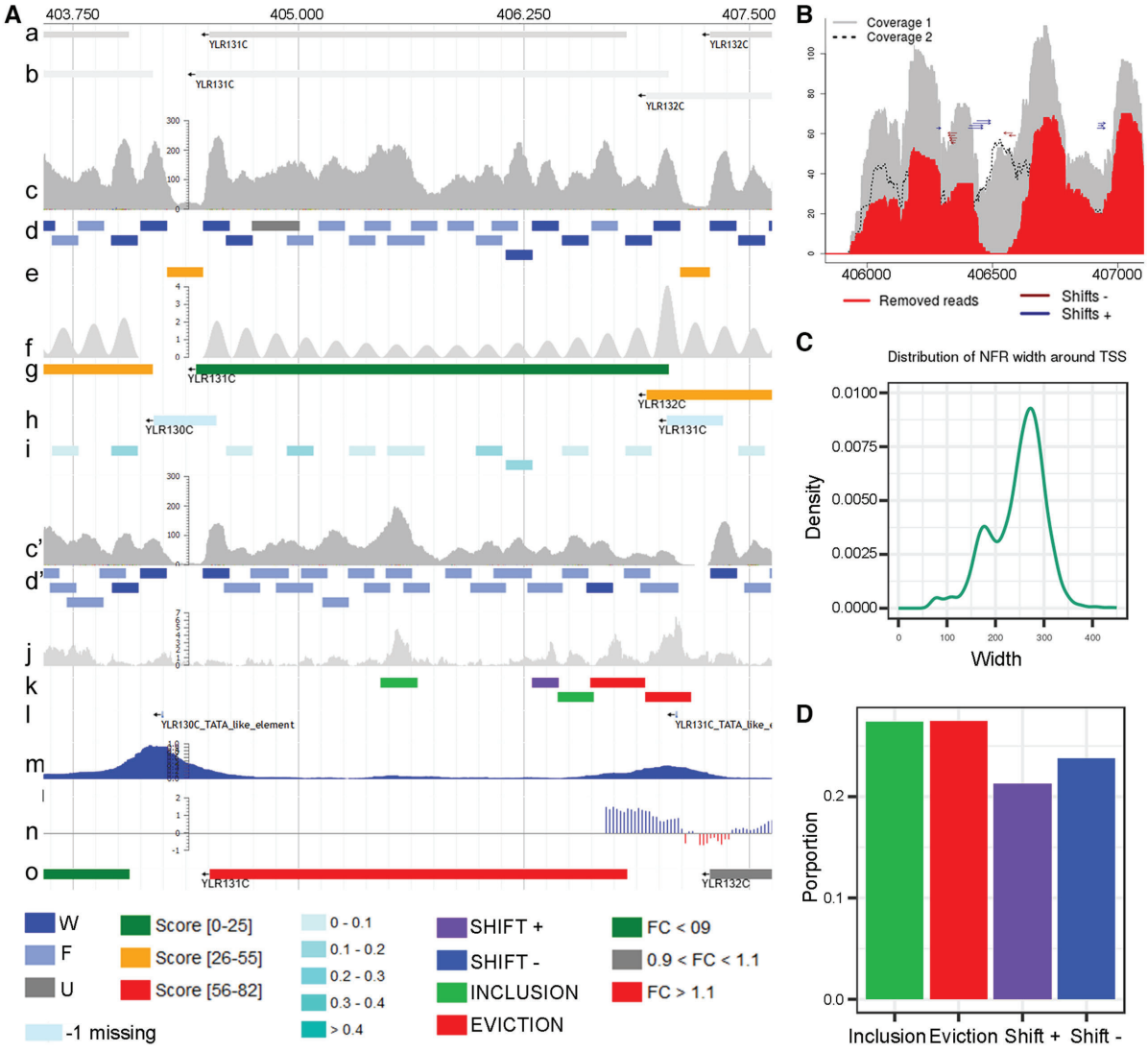


Figure 2. Visualization of Nucleosome Dynamics results in MuGVRE. (**A**) Nucleosome positioning along ACE2 (YLR131C) gene from *S. cerevisiae* between G2 and M cell cycle phases (a) YLR131C open reading frame; (b) ACE2 full length transcripts; (c), (c') coverage of MNase-seq reads aligned to reference genome (in G2 and M phase, respectively); (d), (d') nucleosome calls obtained with nucleR (G2 and M phase, respectively); (e) NFR coordinates in G2 phase; (f) prediction of the nucleosome coverage along each gene, using signals from +1 and -last nucleosomes; (g) genes shown as coloured boxes according to the phasing between the +1 and -last nucleosomes; (h) TSS classification based on nucleosomes −1 and +1 (*W*, *F*, *missing*) and the distance between them (*open* or *close*) represented as coloured boxes, with an arrow indicating the direction of the gene; (i) nucleosomes are coloured by their apparent stiffness value: darker blue nucleosomes are more stiff and lighter are less stiff; (j) significance of the differences in nucleosome coverage between G2 and M phases ($-\log_{10}$ of the p-value) (k) movement hotspots represented as colour coded boxes: purple for *shift* +, blue for *shit* -, green for *inclusion* and red for *eviction*; (l,m, n, o) Tracks from publicly available data representing (l) TATA elements (Rhee *et al.*, 2012), (m) TFIIB binding sites (Mayer *et al.*, 2010), (n) H3K4me3 histone mark enrichment (Liu *et al.*, 2005) and (o) gene expression changes during cell cycle (Deniz *et al.*, 2016). (**B**) Detailed view around a hotspot identified by NucDyn in chrXII. (**C**) Genome wide statistics of NFR width around TSS, in G2 phase. (**D**) Genome wide frequency of changes detected by NucDyn between G2 and M.

S2) from which stiffness is derived by the elastic approximation as shown in Equation (2):

$$\theta = 2\frac{k_B T}{sd^2} \tag{2}$$

where $k_B T$ is the thermal energy at room temperature and sd is the standard deviation of the Gaussian fitted to reads associated to the nucleosome.

Defining changes in nucleosome distribution

Pairs of BAM files are processed to determine changes in the nucleosome architecture between two experiments. For this purpose (see Supplementary Figure S3) the program pairs the reads obtained from one experiment to the other to discard those that are unchanged. It also removes reads that share the same starting or ending point or those that can be fitted in longer read in the paired experiment, as they are likely to be generated by spurious differences in

nuclease degradation activity. The remaining reads are then paired between the two experiments using a dynamic programming algorithm designed to maximize: (i) the number of matches, (ii) the proximity in the middle points of the paired reads, (iii) the assignment of the paired reads to the same nucleosome. To achieve these objectives the dynamic programming highly penalizes gaps and scores read pairs inversely proportional to their distance, with a –infinite score when the distance between the middle point of the reads is longer than half the length of the nucleosome. The final output of the procedure is a set of read pairs shifted in one experiment with respect to the other. These shifts are accumulated to define hotspots that are further analysed to determine their statistical significance as markers of shifts in the nucleosome architecture.

A second type of changes detected by the program is related to differences in occupancy (insertions and evictions) between the two experiments, that are determined directly from the coverage. To reduce the impact of experimental noise we analyse the coverage data by computing a Z-score for every position x across the genome, normalizing it in 10 000 bp windows, which allows us to find locally normalized differences in coverage (Equation 3).

$$Z = \frac{m - E(m)}{(V(m))^{\frac{1}{2}}} \quad (3)$$

where m is the number of reads covering position x in experiment 1, $E(m) = nf$ (with f being the fraction of total reads in the window (N) that corresponds to experiment 1 (M) and n is the number of reads covering position x in both experiments) and $V(m)$ is the expected variance of a hypergeometric distribution, i.e. $V(m) = nf(1-f)\frac{N-n}{n-1}$. Positive Z-score peaks mean that at that point the read coverage found at experiment 1 is higher than the coverage at experiment 2 and an eviction hotspot is annotated. Similarly, negative Z- peaks signal inclusions.

The statistical significance of the detected hotspots (shifts, inclusions and evictions) is scored using the P-values derived from Fisher's test from a contingency table between the reads in each experiment (columns) and the reads at a given position compared to reads within the window (rows):

	Exp 1.	Exp. 2	Total
Covering x	$M-m$	$N-M-n+m$	$N-n$
Not covering x	m	$n-m$	n
Total	M	$N-M$	N

Software availability and implementation

The Nucleosome Dynamics package is available in different deployment models to fulfil the needs of different users. Moreover, it is also offered as a service in two different research platforms. All available distributions are explained at Nucleosome Dynamics landing page: http://mmb.irbbarcelona.org/NucleosomeDynamics/, and summarized in Table 1.

Code distributions.

- Nucleosome Dynamics is written in R and composed of two packages (nucleR and NucDyn), and a series of R wrappers providing a unified interface to such core functionalities and other additional analyses (TSS classification, NFR, Phasing, Stiffness, etc., see above). Source code and documentation are available for standalone installation (see Table 1). Both nucleR and NucDyn packages are also distributed via BioConductor. Although the native R interface is recommended for experienced R users, other deployments built on top of the R software are also provided for further accessibility and portability.
- Nucleosome Dynamics package depends on a series of other R packages and helper applications. To minimize the possibilities of collision with existing installations, and to avoid installation issues to the non-experts, the packages are offered as software containers in both, the well-known Docker implementation and the Singularity format, the latter intended for multi-user systems where running Docker containers natively is not trivial – i.e. HPC systems. A single container allows the user to obtain all functions of the package directly from the command-line, and additionally, the launcher is able to accept a list of Nucleosome Dynamics analysis commands in bash to orchestrate a custom workflow. Furthermore, the use of the containers allows seamless software update. The images are registered at the corresponding hubs (see Table 1).

Use in research platforms.

- MuG Virtual Research Environment (MuGVRE) is an integrated workspace designed to put together a series of applications related to the study of 3D/4D genomics (30). The MuGVRE workspace allows to combine data, either uploaded to the workspace or obtained from public repositories such as ArrayExpress (33). MuGVRE includes applications covering a wide range of levels in the study of chromatin, from atomistic simulation or protein-nucleic acids docking to coarse-grained simulation of large nucleic acids molecules or chromatin fibers, as well as the analysis of Hi-C data. All those applications share a common data space where interoperability is assured through a common data model, and a specific protocol to incorporate new tools. MuGVRE is a cloud-based application that simplifies the deployment and provides user access to visualization tools, additional data in external repositories, and to a variety of other programs for the analysis of chromatin at different levels of resolution.
- The server provides a graphical interface based on an embedded sequence browser, Jbrowse (34), that allows visualization of nucleosome architectures in the context of other omics data (see Figure 2A). For this purpose, the outputs of all calculations are generated in GFF3 or bigWig format. Nucleosomes are represented as boxes coloured with different tones of blue according to their positioning score, with regions where nucleosome are too fuzzy displayed in a lighter colour. Similarly, nucleosome-

free regions are highlighted by yellow boxes in a different row (see Figure 2A). In both cases, numerical information (scores, characteristics of the nucleosome or NFR) can be obtained by clicking on the corresponding boxes. The nucleosome architecture around TSS is classified based on the length of the NFR and the location score of the +1 and −1 nucleosome (see above). The results are shown as boxes between the −1 and +1 nucleosome dyads, color-coded by the corresponding architecture class. The analysis of nucleosome phasing generates a bigWig file with the theoretical prediction of the nucleosome positions inside the gene body, based on periodicity considerations and the +1 and –last nucleosomes (see above), and a GFF3 file which is displayed as a coloured box indicating whether the gene shows 'phased', 'antiphased' or intermediate nucleosome phasing (see Figure 2A). Similarly, the stiffness associated to the nucleosomes is represented (through a GFF3 file) as a box mapping to the nucleosome, coloured according to the estimated stiffness (see Figure 2A). Nucleosome Dynamics data can be put into genomic context by a series of additional tracks (Supplementary Table S3) providing gene annotations and relevant literature data. Further analyses can be obtained from the web server, such as detailed plots of nucleosome coverage and changes in nucleosome distribution between the two experiments (Figure 2B), genome-wide statistics of nucleosome architecture around TSS (Figure 2C), and overall frequency of inclusions, evictions and shifts between the two experiments (Figure 2D). Finally, Nucleosome Dynamics also generates a table listing the number of nucleosomes, their status (fuzzy/well-positioned), the identified nucleosome changes (inclusions, evictions, shifts), the classification of the promoter and the width of the NFR at the TSS for every single gene in the genome (Supplementary Table S4). These analyses are useful to test the effect of a treatment/growth conditions on nucleosome positioning both globally, or at gene level.

- *Running Nucleosome Dynamics on the galaxy platform:* Galaxy is a web-based scientific analysis platform widely used by scientists to analyse biomedical datasets such as genomics, proteomics, metabolomics or imaging (31). Nucleosome Dynamics docker has been wrapped in a series of Galaxy tools, one for each analysis. Users can launch them individually, or as part of a Galaxy workflow, building a custom pipeline that may integrate other Galaxy applications. The tools are published in the Galaxy ToolShed (see Table 1) and adopted by the ELIXIR_ES Galaxy server (currently in development phase), together with a complete ready-to-use Nucleosome Dynamics workflow. The output calculations, mainly GFF3 and bigWig files, are treated in the platform as any other sequence annotation file. Plain files like GFFs are locally displayed using a column-based visualization, while the genomic-context analysis is based on the UCSC genome browser (35). Galaxy transparently loads the data to the central UCSC browser service, and there, the sequences are loaded as custom tracks and visualized together with the other UCSC annotations.

Benchmarking data sets

The ability of the package to determine the location and nature of nucleosomes and nucleosome architectures was evaluated using synthetic maps from single cell nucleosome architectures that are combined to create *in silico* MNase digestion maps approaching closely to those found in real yeast MNase-seq experiments. For this purpose, we created multiple single cell nucleosome architectures by first placing NFRs at specific positions separated by ∼2000 bp (the typical range of NFR-NFR distances in yeast) in a 10 kb DNA fragment. As the NFR are highly conserved, their positions are located with small noise in the different cells. Once the NFRs for a single cell have been placed, we defined windows for nucleosome positioning using the known average nucleosome periodicity (165 bp for yeast in the experiments simulated here). Each window was associated to either a W or F nucleosome following probability functions reproducing their expected populations at different distances from NFR. Windows that (in a given cell) appeared to be associated to W nucleosomes have a high probability to be occupied by a nucleosome, which is placed within a narrow range from the centre of the window (see Supplementary Figure S4). On the contrary, windows associated to F nucleosomes have a higher probability to be empty in a given cell, and once the nucleosome is assigned, its position is variable within the window. Once obtained, the population of *in silico* cells was processed by *in silico* MNase digestion repeating this process many times, introducing 'digestion noise' to reproduce the distributions of read lengths observed in typical yeast experiments. The integration of the reads for the entire population provides an *in silico* MNase-seq map where we know exactly the real population and fuzziness of all nucleosomes at all positions in the pool of cells. This constitutes an unambiguous benchmark to validate the performance of nucleosome annotation software. The probability functions used to generate the different cell nucleosome architectures were adjusted to qualitatively reproduce reads obtained in real MNase-seq experiments (Supplementary Figure S4). Synthetic data simulating ATAC-seq experiments can be derived in a very similar manner.

The synthetic data obtained as explained above were the starting point to generate pairs of *in silico* experiments simulating changes in nucleosome architectures. To this end, a percentage of the reads was either shifted or removed for a given nucleosome. Shifts from 1 to 5 turns of DNA (1 DNA turn = 10 bp) were introduced generating 100 replicates in each case to evaluate the sensitivity of the method to detect shifts of different lengths and affecting different percentage of the total population.

RESULTS

Performance using *in silico* datasets

NucleR. We explored the ability of the software to position nucleosomes using as reference our highly controlled synthetic data (see Methods). As a control, we repeated the exercise using another widely used program for nucle-

osome annotation, DANPOS (15). Both packages show a good ability to represent the nucleosome architecture using MNase-seq data. In terms of occupancy DANPOS performs slightly better than the nucleR module implemented in our Nucleosome Dynamics package ($R^2 = 0.97$ for DANPOS versus $R^2 = 0.93$ for nucleR), while nucleR can detect better the nucleosome fuzziness ($R^2 = 0.94$ for nucleR versus $R^2 = 0.87$ for DANPOS; see Supplementary Methods. for description of the metrics). The location of W nucleosomes is nearly identical in both methods, but for F nucleosomes the results are quite different, as DANPOS annotates a wide region of sequence reads as a single nucleosome positioned with a large uncertainty, while nucleR can assign several nucleosomes to the wide signal, even when in some cases the two nucleosomes can partially overlap (see Figure 3A). As a result, DANPOS provides, probably, the best 'average' distribution of nucleosomes, but nucleR provides a more realistic picture of the cellular variability, capturing the presence of alternative nucleosome architectures in the cellular population. As it can be seen in Figure 3A, where selected examples of DANPOS and nucleR nucleosome distributions are compared with the real nucleosome architecture existing in our synthetic data; in Figure 3B, where we compare the ability of DANPOS and nucleR to detect the presence of a percentage of cells showing a different nucleosome architecture and in Figure 3C, where we report the average distance between the real position of the dyads of the synthetic nucleosomes and those located by DANPOS or nucleR.

NucDyn. We tested the ability of our method to detect rearrangements in the nucleosome architecture using again our controlled *in-silico* MNase-seq data, simulating displacement (shift), insertion or eviction of one nucleosome, occurring in a different percentage of the cells. Sizeable changes such as nucleosome insertion or eviction are detected with good sensitivity by our method, even when they affect a relatively small percentage of cells (Figure 3D), while DANPOS or Dimnp only detect such changes when affecting a very large proportion of cells. Small nucleosome shifts (implying less than one turn of DNA) are not detectable by our algorithm unless they occur in a large percentage of the cells; while shifts implying a displacement of at least two turns of DNA (20 base pairs) are detectable with good sensitivity, even when affecting less than half of the cellular population (see Figure 3E). In this case, the comparison with other programs is difficult, as only DANPOS (15) allows an indirect way to detect nucleosome shifts by looking at distances between nucleosome peaks in both experiments. Unfortunately, with our synthetic data, DANPOS achieved poor sensitivity (less than 0.20 for 5 turns of DNA shift in 70% of cells and <0.1 for 3 turns shifts affecting also 70% of cells; see Supplementary Figure S5).

In summary, analysis of well-controlled *in silico* data shows that the Nucleosome Dynamics package (including NucDyn and nucleR) is not only a very powerful tool to define nucleosome families from MNase-seq experiments performed with a population of cells, but also a powerful approach to detect subtle changes in nucleosome architecture affecting a percentage of the cells in the studied sample.

Test cases

In order to illustrate the information derived from Nucleosome Dynamics, we applied our method in different real cases where experimental MNase-seq data were available. It is important to mention that the biological relevance of this type of comparison depends on the quality of the data and especially on the similar level of MNase digestion of the samples being compared. Indeed, several groups, including ours, have demonstrated the impact of the level of MNase digestion on the final nucleosome maps in several organisms, essentially at the level of the so-called 'fragile' nucleosomes (19,36–38). To illustrate this observation, we took advantage of the extensive study made by (36) and used Nucleosome Dynamics to compare nucleosome positioning in the input of two H2B and two H4 MNase-ChIP-seq samples, one under-digested and one over-digested (50U and 400 U of MNase respectively for H2B; 25 U and 300 U MNase respectively for H4). First, we focused on the H2B-input samples and confirmed that the number of nucleosome detected by nucleR decreases as the amount of MNase increases (from 80 160 down to 72 775, Supplementary Table S5) which is corroborated by the detection by NucDyn of 3559 evictions genome wide (Supplementary Table S6). At the promoter level, the proportion of W-open-W TSS increases from 123 to 2026 while the W-close-W TSS decreases from 2656 down to 346 (Supplementary Figure S6A). Regarding the phasing analysis, the percentage of phased genes does not change significantly due to the level of MNase digestion (Supplementary Figure S6B). Similar numbers were obtained for the H4-input samples. Hence, it is important to control MNase digestion level before using Nucleosome Dynamics package. Another technique that is not influenced by the level of MNase digestion is chemical cleavage mapping. NucleR can be applied to map nucleosome positions using the coverage obtained from these experiments (Supplementary Figure S7),

Cell cycle. The first example comes from the analysis of the changes in nucleosome organization occurring along the cell cycle in yeast, using our own previously published data (39). As shown in Figure 4A, the nucleR module of the Nucleosome Dynamics package suggests that nucleosomes tend to be fuzzier (F) in S and M phases compared to G1 and G2 phases. The increase in fuzziness in S and M phases impacts on the promoter classification as the number of W-open-W and W-closed-W promoters decreases compared to G1 and G2 (Figure 4B), but overall the ratio of closed/open NFRs (nucleosome free regions) is not dramatically altered along cell cycle (Supplementary Figure S8). Very interestingly, the changes in nucleosome architecture detected by NucDyn are not randomly distributed along the genome, but appear to be localized in specific families of genes, which are related to cell cycle progression, as shown by Gene Ontology (GO) Enrichment Analysis (40) in Figure 4C. Examples of the detailed information provided by Nucleosome Dynamics for some specific genes are shown in Figure 4D, where we report nucleosome maps of *PRY2* (a gene related to lipid transport), whose expression peaks in G1 phase, *YHP1* (involved in negative regulation of transcription of certain cell cycle genes), and *GIC1* (a GTPase-interaction

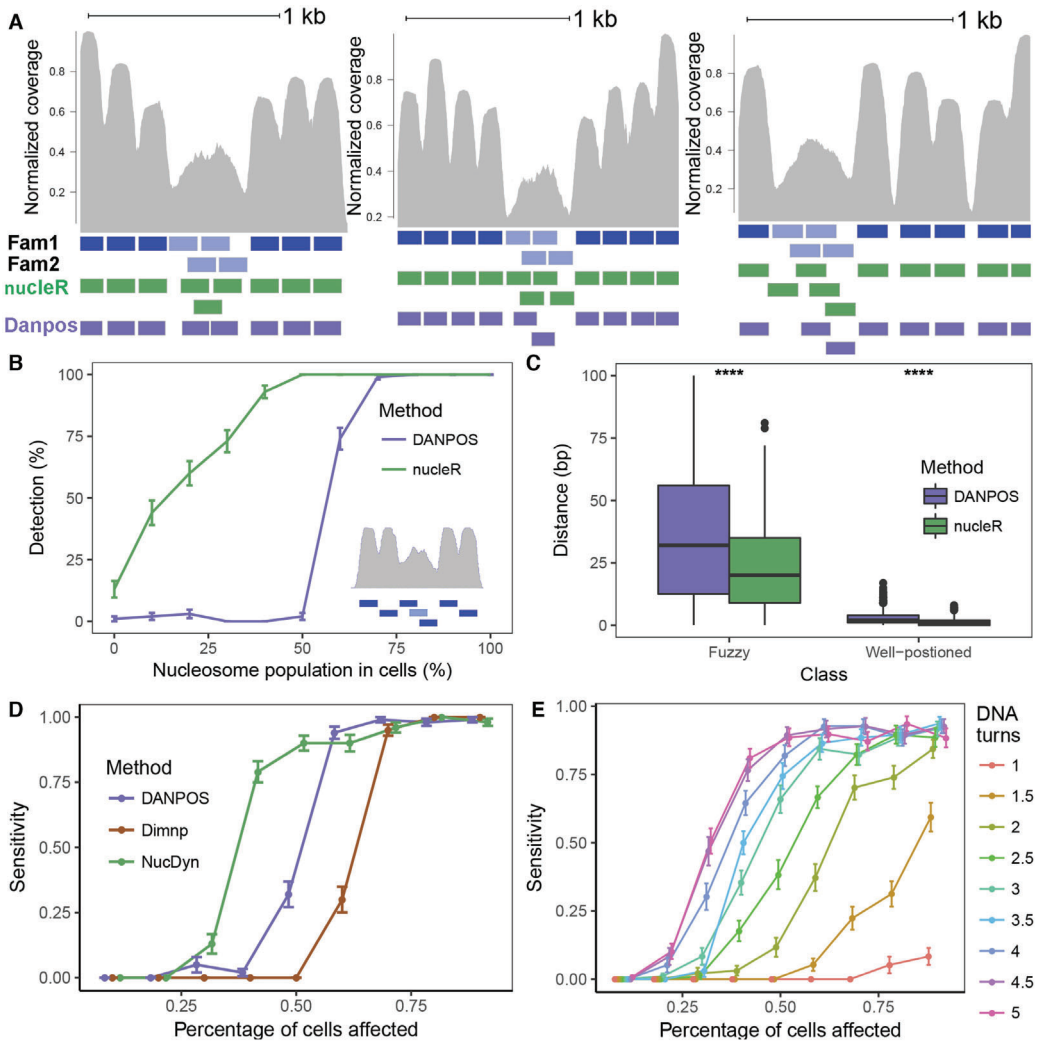


Figure 3. Performance of nucleR and NucDyn. (**A**) Coverage of three synthetic nucleosome maps (shown in grey), containing well-positioned (dark blue) and fuzzy nucleosomes (light blue). Two possible nucleosome families generate different nucleosome positioning in fuzzy regions (Fam1 and Fam2). Predicted nucleosome positions using nucleR and Danpos are shown in green and purple, respectively. (**B**) Comparison of nucleR and Danpos for detection of a second family of nucleosomes (light blue nucleosome in the bottom-right panel). Y-axis shows the number of cells required in the second family in order to be detected by the algorithm. (**C**) Distance between the dyads identified by nucleR (green) and DANPOS (purple) to the dyad position in the true synthetic nucleosome map for fuzzy and well positioned nucleosomes. (**D**) Sensitivity of the EVICTION prediction for NucDyn, DANPOS and Dimnp. Evictions were simulated removing reads from a given percentage of families (10%, 20%, …, 90%) and were identified from DANPOS output as a nucleosome with point_$\log_2$FC < -1 and point_diff_FDR < 0.01 (point with highest difference in the two samples, as reported by the software), and with default parameters for Dimnp. (*E*) Sensitivity of the SHIFT prediction computed on synthetic nucleosome maps. Shifts were introduced displacing reads from 1 to 5 DNA turns and modifying different percentages of the families (10%, 20%, …, 90%).

component involved in mitosis regulation) expressed in S phase (39). In the three cases, expression changes correlate with significant variation in nucleosome architecture at the promoter region between two stages of cell cycle. Typically, eviction or shifts reducing the presence of nucleosome in the core promoter region are related to active states of the genes (4,19,28,41,42).

Yeast metabolic cycle. A second example of use of our tool was the comparison of nucleosome architecture amongst cells at different stages of the yeast metabolic cycle (YMC). We took advantage here of high resolution experiments (43) in which the authors analysed simultaneously gene expression and MNase-seq maps at regularly spaced periods of time after adding fresh culture media. At two of these time points (T9 and T12 in Nocetti and Whitehouse 2016) dramatic changes of expression in genes related to reductive charging (poorly transcribed at T9 and highly transcribed at T12) and oxidative phase (highly transcribed at T9 and poorly transcribed at T12) have been detected. Analysis of global nucleosome architecture shows moderate changes between T9 and T12 (Figure 5A), but differences are more noticeable when the analysis is focused on Ox-genes (involved in amino acid synthesis, sulphur metabolism, ribosome and RNA metabolism (44), which are expressed in T9 and repressed in T12) and R/C genes (involved in non-

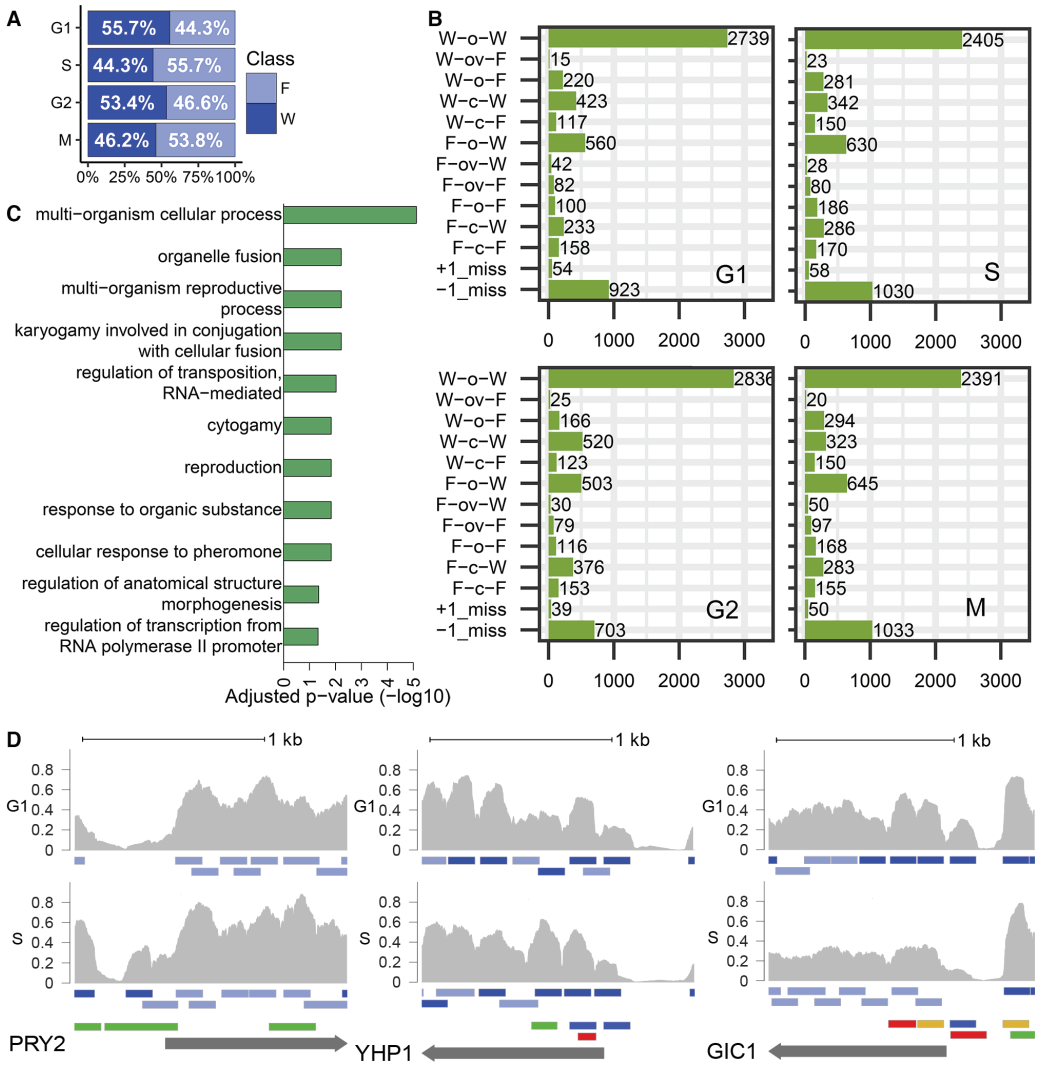


Figure 4. Nucleosome Dynamics along the cell cycle. (**A**) Percentage of fuzzy and well-positioned nucleosomes and (**B**) promoter classification (number of genes in each class) for every cell cycle stage. (**C**) GO terms enriched in genes with nucleosome changes between G1 and S detected by NucDyn. (**D**) Example of three cell-cycle dependent genes that present differential nucleosome architectures between G1 and S. In gray, the normalized coverage from the BAM files of the two cell cycle stages, 500 bp upstream and 1000 bp downstream the TSS. Below each BAM file, the nucleosome calls obtained with nucleR are represented (dark blue for well-positioned nucleosomes, light blue for fuzzy nucleosomes). The fifth track contains shifts (yellow for positive, blue for negative), inclusions (green) and evictions (red) identified by NucDyn.

respiratory metabolism, protein degradation, autophagy and vacuole (44), which are expressed in T12 and repressed in T9). Nucleosome Dynamics allows the detection and quantification of the alterations in nucleosome architecture coupled to such changes in expression. Thus, for Ox-genes (Figure 5B), the fuzziness at the −1 nucleosome decreases when moving from T9 to T12, in agreement with the general rule that reduced NFR upstream the well positioned +1 nucleosome correlates with inactive genes. On the contrary, for R/C genes (Figure 5C) the NFR upstream well positioned +1 nucleosome enlarges, since the −1 nucleosomes become fuzzier, again in perfect agreement with the changes of expression. To discard any biases resulting from the MNase digestion conditions, we confirmed that the length of the sequenced fragments was comparable in both samples (Supplementary Figure S9). Examples of the detailed information provided by Nucleosome Dynamics for three Ox-genes (*NSR1*, *RRP4* and *SNU13,* all of them related to ribosomal biogenesis and RNA metabolism) are shown in Figure 5D, where upon T9→T12 transition, shifts and even insertions are shown leading to a reduction in the width of the NFR upstream the TSS: a fingerprint of gene deactivation. Similarly, Figure 5E provides the same type of information for three R/C genes (*CTA1*, *SUE1* and *SAF1*), which are associated respectively with peroxisome, cytochrome C degradation and proteasome (see above). In the three cases, T9→T12 transition is coupled with massive nucleosome eviction upstream the TSS, leading to open configurations of NFR, typical of highly expressed genes.

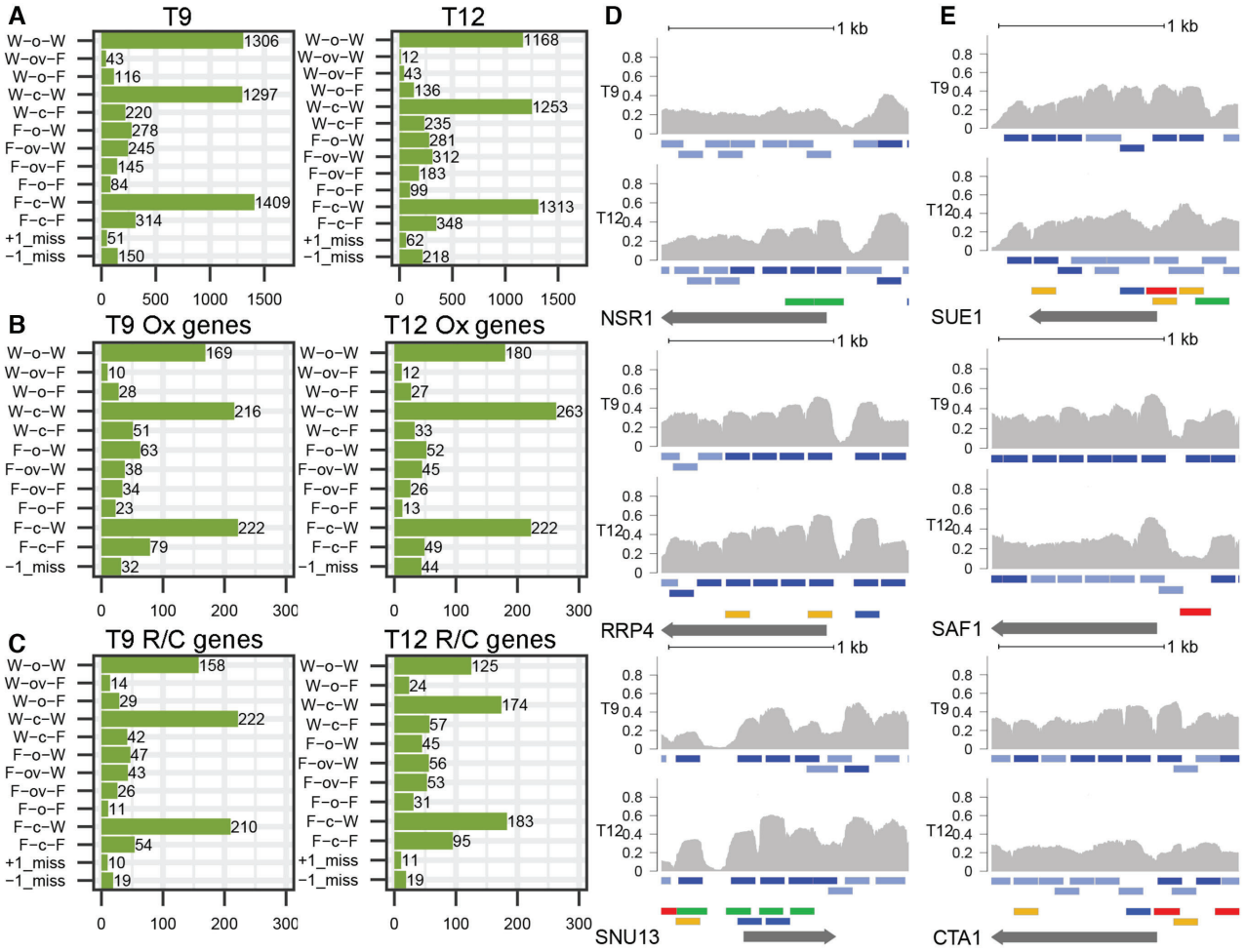


Figure 5. Nucleosome Dynamics in two points of the YMC. Promoter classification in the two time points (T9 and T12) for (**A**) all genes, (**B**) genes from the Ox cluster and (**C**) genes from the R/C cluster. (**D** and **E**) Example of three genes from Ox and 3 from R/C clusters that present differential nucleosome architectures between T9 and T12. In grey, the normalized coverage from the BAM files of the two time points, 500 bp upstream and 1000 bp downstream the TSS. Below each BAM file, the nucleosome calls obtained with nucleR are represented (dark blue for well-positioned nucleosomes, light blue for fuzzy nucleosomes). The fifth track contains shifts (yellow for positive, blue for negative), inclusions (green) and evictions (red) identified by NucDyn.

Changes in nutrients. As a last test, we applied Nucleosome Dynamics to explore the modifications in nucleosome architecture in yeast, linked to the change in the media from glucose-rich to either galactose-rich or ethanol-rich (45). Changes in the TSS nucleosome architecture classification occur among the three conditions (Figure 6A). There are not complete expression data in Kaplan *et al.* 2009, but we expected that replacement of glucose by galactose in the media would imply changes in expression in genes related to carbohydrate metabolism and transport which encouragingly, are those where sizeable changes in nucleosome architecture are detected by Nucleosome Dynamics (Figure 6B). Similarly, replacement of glucose by ethanol was expected to have an impact on the cell through: (i) expression of stress response genes, (ii) changing completely hexose metabolism in the absence of hexoses and (iii) eliminating ethanol through oxidation generating changes in the redox state of the cell which need to be corrected (46). Very encouragingly again, genes involved in stress response, hexose metabolism and redox activities are those for which the largest changes in nucleosome architecture have been detected (Figure 6C).

We analysed in detail some genes which are expected to change dramatically their expression upon glucose→galactose substitution, as they are crucial to integrate galactose in normal hexose metabolism: *GAL1*, coding for a Galactokinase, *GAL10*, coding for the UDP-glucose-4-epimerase, and *GAL7*, coding for the galactose-1-phosphate uridyl transferase (Figure 6D). In the three cases evictions and shifts (in some cases noticeably) generate wider NFRs upstream the gene, changes that in some cases extend to the coding regions and that signal a pronounced increase in expression of these galactose-related genes.

A similar detailed analysis was made for three genes which are expected to be overexpressed when ethanol substitutes glucose as energy source: two stress response genes *HSP26* and *HSP12*, and *HXK1*, a hexokinase activated when cells are shifted to a non-fermentable carbon source

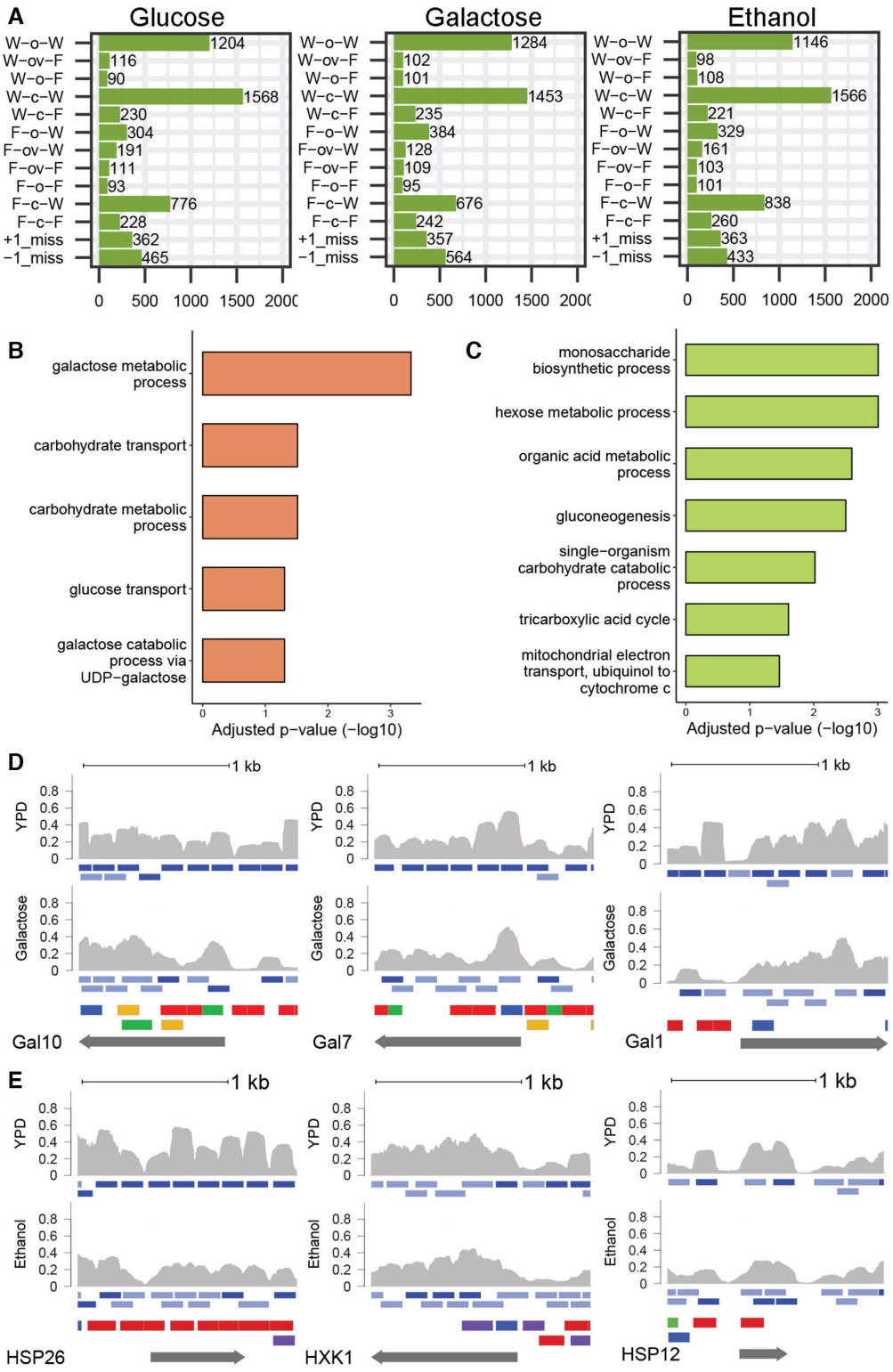


Figure 6. Nucleosome Dynamics under different nutrient conditions. (**A**) Promoter classification in glucose, galactose and ethanol rich media. (**B** and **C**) GO terms enriched in genes with nucleosome changes detected by NucDyn, changing the medium from glucose to galactose or ethanol, respectively. (**D** and **E**) Example of three genes involved in galactose and ethanol metabolism, respectively, that present differential nucleosome architectures depending on the carbon source. In gray, the normalized coverage from the BAM files of the two cell cycle stages, 500 bp upstream and 1000 bp downstream the TSS. Below each BAM file, the nucleosome calls obtained with nucleR are represented (dark blue for well-positioned nucleosomes, light blue for fuzzy nucleosomes). The fifth track contains shifts (yellow for positive, blue for negative), inclusions (green) and evictions (red) identified by NucDyn.

such as ethanol (47). Results in Figure 6E illustrate the magnitude of the changes (mainly evictions) detected by Nucleosome Dynamics, which affect the NFR, and even in some cases the coding regions.

DISCUSSION

Different studies demonstrated that nucleosome architecture is coupled to gene function (4,43,48) and that transcriptional activity and nucleosome architecture are tightly coupled. Unfortunately, detecting changes in nucleosome architecture is complex as nucleosomes are dynamic and even a population of 'identical' well synchronized and grown under identical conditions cells might have nucleosomes placed at different positions. This, combined with the intrinsic uncertainties of MNase- or ATAC-seq experiments, generate noisy data which are difficult to process for precisely locating nucleosomes and even more difficult to detect significant changes in nucleosome arrangements due to internal or external stimuli. The suite of programs incorporated in Nucleosome Dynamics allows not only a robust location of nucleosomes, even in cases of heterogeneous pools of cells, but also the detection of changes in nucleosome arrangements, even those affecting a moderate population of cells. To increase its utility, Nucleosome Dynamics is integrated into a powerful virtual research environment, where it is combined with different tools for analysis of data and visualization in the context of genomic metadata, which help the user not only to analyse nucleosome architecture and dynamics, but also to put them in the context of known genomic information (Figure 2).

We validated the methodology using synthetic data that mimic typical MNase-seq maps, in which the positions of the nucleosome in the different cells are unambiguously known. The two main modules of Nucleosome Dynamics (nucleR and NucDyn) perform very well capturing cellular diversity and detecting shifts, evictions and inclusions that affect a moderate percentage of the cellular population. Furthermore, we tested the power of the methodology by exploring nucleosome rearrangements occurring along cell cycle, yeast metabolic cycle, and those linked to the change in the source of energy from glucose to galactose or ethanol. In the tested cases, Nucleosome Dynamics provides accurate global and local descriptions of nucleosome structure and dynamics and deciphers the nature of the connection between nucleosome organization and gene expression.

DATA AVAILABILITY

Raw MNase-seq datasets reported as test cases in this study were obtained from the *ENA-SRA* website (http://www.ebi.ac.uk/ena) and the GEO repository under accession numbers: PRJEB6970 for the cell cycle data, GSE77631 for the yeast metabolic cycle, GSE13622 for the nucleosome maps from yeast cultivated in glucose, galactose and ethanol media, and GSE83123 for the different levels of MNase digestion. Processed chemical cleavage data was obtained from GSE97290.

The processed test data and benchmarking synthetic data supporting the conclusions of this article are available in Zenodo repository (10.5281/zenodo.2632999), and can be incorporated to both MuGVRE and Galaxy Nucleosome Dynamics installations for additional testing.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We are indebted all members of the MuG consortia for help and for acting as a beta-tester of the software.

FUNDING

M.O. is an ICREA (Institució Catalana de Recerca i Estudis Avancats) academia researcher; Spanish Ministry of Science [RTI2018-096704-B-100]; Catalan Government [2017-SGR-134]; Instituto de Salud Carlos III–Instituto Nacional de Bioinformática, the European Union's Horizon 2020 research and innovation program, and the Biomolecular and Bioinformatics Resources Platform [ISCIII PT 17/0009/0007 co-funded by the Fondo Europeo de Desarrollo Regional FEDER; Grants Elixir-Excelerate: 676559 and BioExcel2: 823830; ERC:812850; MuG-676566]; MINECO Severo Ochoa Award of Excellence from the Government of Spain (awarded to IRB Barcelona). Funding for open access charge: Spanish Ministry of Science [RTI2018-096704-B-100].
Conflict of interest statement. None declared.

REFERENCES

1. Barski,A., Cuddapah,S., Cui,K., Roh,T.-Y., Schones,D.E., Wang,Z., Wei,G., Chepelev,I. and Zhao,K. (2007) High-Resolution profiling of histone methylations in the human genome. *Cell*, **129**, 823–837.
2. Mavrich,T.N., Jiang,C., Ioshikhes,I.P., Li,X., Venters,B.J., Zanton,S.J., Tomsho,L.P., Qi,J., Glaser,R.L., Schuster,S.C. *et al.* (2008) Nucleosome organization in the Drosophila genome. *Nature*, **453**, 358–362.
3. Yuan,G.-C. (2005) Genome-Scale Identification of Nucleosome Positions in S. cerevisiae. *Science*, **309**, 626–630.
4. Lai,W.K.M. and Pugh,B.F. (2017) Understanding nucleosome dynamics and their links to gene expression and DNA replication. *Nat. Rev. Mol. Cell Biol.*, **18**, 548–562.
5. Rando,O.J. and Ahmad,K. (2007) Rules and regulation in the primary structure of chromatin. *Curr. Opin. Cell Biol.*, **19**, 250–256.
6. Jiang,C. and Pugh,B.F. (2009) Nucleosome positioning and gene regulation: advances through genomics. *Nat. Rev. Genet.*, **10**, 161–172.
7. Raveh-Sadka,T., Levo,M., Shabi,U., Shany,B., Keren,L., Lotan-Pompan,M., Zeevi,D., Sharon,E., Weinberger,A. and Segal,E. (2012) Manipulating nucleosome disfavoring sequences allows fine-tune regulation of gene expression in yeast. *Nat. Genet.*, **44**, 743–750.
8. Collings,C.K. and Anderson,J.N. (2017) Links between DNA methylation and nucleosome occupancy in the human genome. *Epigenet. Chromatin*, **10**, 18.
9. Kubik,S., O'Duibhir,E., de Jonge,W.J., Mattarocci,S., Albert,B., Falcone,J.-L., Bruzzone,M.J., Holstege,F.C.P. and Shore,D. (2018) Sequence-directed action of RSC remodeler and general regulatory factors modulates +1 nucleosome position to facilitate transcription. *Mol. Cell*, **71**, 89–102.
10. Mellor,J. and Morillon,A. (2004) ISWI complexes in Saccharomyces cerevisiae. *Biochim. Biophys. Acta (BBA) - Gene Struct. Expression*, **1677**, 100–112.
11. Knight,B., Kubik,S., Ghosh,B., Bruzzone,M.J., Geertz,M., Martin,V., Dénervaud,N., Jacquet,P., Ozkan,B., Rougemont,J. *et al.*

(2014) Two distinct promoter architectures centered on dynamic nucleosomes control ribosomal protein gene transcription. *Genes Dev.*, **28**, 1695–1709.
12. Whitehouse,I., Flaus,A., Cairns,B.R., White,M.F., Workman,J.L. and Owen-Hughes,T. (1999) Nucleosome mobilization catalysed by the yeast SWI/SNF complex. *Nature*, **400**, 784–787.
13. Whitehouse,I., Stockdale,C., Flaus,A., Szczelkun,M.D. and Owen-Hughes,T. (2003) Evidence for DNA translocation by the ISWI chromatin-remodeling enzyme. *Mol. Cell Biol.*, **23**, 1935–1945.
14. Buenrostro,J.D., Giresi,P.G., Zaba,L.C., Chang,H.Y. and Greenleaf,W.J. (2013) Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nat. Methods*, **10**, 1213–1218.
15. Chen,K., Xi,Y., Pan,X., Li,Z., Kaestner,K., Tyler,J., Dent,S., He,X. and Li,W. (2013) DANPOS: Dynamic analysis of nucleosome position and occupancy by sequencing. *Genome Res.*, **23**, 341–351.
16. Chen,W., Liu,Y., Zhu,S., Green,C.D., Wei,G. and Han,J.-D.J. (2014) Improved nucleosome-positioning algorithm iNPS for accurate nucleosome positioning from sequencing data. *Nat. Commun.*, **5**, 4909.
17. Flores,O. and Orozco,M. (2011) nucleR: a package for non-parametric nucleosome positioning. *Bioinformatics*, **27**, 2149–2150.
18. Teif,V.B. (2016) Nucleosome positioning: resources and tools online. *Brief. Bioinform.*, **17**, 745–757.
19. Flores,O., Deniz,O., Soler-Lopez,M. and Orozco,M. (2014) Fuzziness and noise in nucleosomal architecture. *Nucleic Acids Res.*, **42**, 4934–4946.
20. Deniz,Ö., Flores,O., Battistini,F., Pérez,A., Soler-López,M. and Orozco,M. (2011) Physical properties of naked DNA influence nucleosome positioning and correlate with transcription start and termination sites in yeast. *BMC Genomics*, **12**, 489.
21. Gutiérrez,G., Millán-Zambrano,G., Medina,D.A., Jordán-Pla,A., Pérez-Ortín,J.E., Peñate,X. and Chávez,S. (2017) Subtracting the sequence bias from partially digested MNase-seq data reveals a general contribution of TFIIS to nucleosome positioning. *Epigenet. Chromatin*, **10**, 58.
22. Brogaard,K., Xi,L., Wang,J.-P. and Widom,J. (2012) A map of nucleosome positions in yeast at base-pair resolution. *Nature*, **486**, 496–501.
23. Chereji,R.V., Ramachandran,S., Bryson,T.D. and Henikoff,S. (2018) Precise genome-wide mapping of single nucleosomes and linkers in vivo. *Genome Biol.*, **19**, 19.
24. Voong,L.N., Xi,L., Sebeson,A.C., Xiong,B., Wang,J.-P. and Wang,X. (2016) Insights into nucleosome organization in mouse embryonic stem cells through chemical mapping. *Cell*, **167**, 1555–1570.
25. Thakur,J., Talbert,P.B. and Henikoff,S. (2015) Inner kinetochore protein interactions with regional centromeres of fission yeast. *Genetics*, **201**, 543–561.
26. Moyle-Heyrman,G., Zaichuk,T., Xi,L., Zhang,Q., Uhlenbeck,O.C., Holmgren,R., Widom,J. and Wang,J.-P. (2013) Chemical map of Schizosaccharomyces pombe reveals species-specific features in nucleosome positioning. *Proc. Natl. Acad. Sci. U.S.A.*, **110**, 20158–20163.
27. Henikoff,S., Ramachandran,S., Krassovsky,K., Bryson,T.D., Codomo,C.A., Brogaard,K., Widom,J., Wang,J.-P. and Henikoff,J.G. (2014) The budding yeast Centromere DNA Element II wraps a stable Cse4 hemisome in either orientation in vivo. *eLife*, **3**, e01861.
28. Liu,L., Xie,J., Sun,X., Luo,K., Qin,Z.S. and Liu,H. (2017) An approach of identifying differential nucleosome regions in multiple samples. *BMC Genomics*, **18**, 135.
29. R Core Team. (2016) *R-A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna.
30. Codó,L., Bayarri,G., Cid-Fuentes,J.A., Conejero,J., Hospital,Adam, Royo,R., Repchevsky,D., Pasi,M., Meletiou,A., McDowall,M.D. *et al.* (2019) MuGVRE. A virtual research environment for 3D/4D genomics Bioinformatics.
31. Afgan,E., Baker,D., Batut,B., van den Beek,M., Bouvier,D., Čech,M., Chilton,J., Clements,D., Coraor,N., Grüning,B.A. *et al.* (2018) The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic Acids Res.*, **46**, W537–W544.
32. Giancoli,D.C. (2000) *Physics for Scientists & Engineers with Modern Physics*. 3rd edn. Prentice Hall, Upper Saddle River.
33. Kolesnikov,N., Hastings,E., Keays,M., Melnichuk,O., Tang,Y.A., Williams,E., Dylag,M., Kurbatova,N., Brandizi,M., Burdett,T. *et al.* (2015) ArrayExpress update—simplifying data submissions. *Nucleic Acids Res.*, **43**, D1113–D1116.
34. Buels,R., Yao,E., Diesh,C.M., Hayes,R.D., Munoz-Torres,M., Helt,G., Goodstein,D.M., Elsik,C.G., Lewis,S.E., Stein,L. *et al.* (2016) JBrowse: a dynamic web platform for genome visualization and analysis. *Genome Biol.*, **17**, 66.
35. Kent,W.J., Sugnet,C.W., Furey,T.S., Roskin,K.M., Pringle,T.H., Zahler,A.M. and Haussler,A. D. (2002) The human genome browser at UCSC. *Genome Res.*, **12**, 996–1006.
36. Chereji,R.V., Ocampo,J. and Clark,D.J. (2017) MNase-sensitive complexes in yeast: nucleosomes and non-histone barriers. *Mol. Cell*, **65**, 565–577.
37. Chereji,R.V., Kan,T.-W., Grudniewska,M.K., Romashchenko,A.V., Berezikov,E., Zhimulev,I.F., Guryev,V., Morozov,A.V. and Moshkin,Y.M. (2016) Genome-wide profiling of nucleosome sensitivity and chromatin accessibility in *Drosophila melanogaster*. *Nucleic Acids Res.*, **44**, 1036–1051.
38. Jeffers,T.E. and Lieb,J.D. (2017) Nucleosome fragility is associated with future transcriptional response to developmental cues and stress in *C.elegans*. *Genome Res.*, **27**, 75–86.
39. Deniz,Ö., Flores,O., Aldea,M., Soler-López,M. and Orozco,M. (2016) Nucleosome architecture throughout the cell cycle. *Sci. Rep.*, **6**, 19729.
40. Falcon,S. and Gentleman,R. (2007) Using GOstats to test gene lists for GO term association. *Bioinformatics*, **23**, 257–258.
41. Teif,V.B., Vainshtein,Y., Caudron-Herger,M., Mallm,J.-P., Marth,C., Höfer,T. and Rippe,K. (2012) Genome-wide nucleosome positioning during embryonic stem cell development. *Nat. Struct. Mol. Biol.*, **19**, 1185–1192.
42. Chen,J., Li,E., Zhang,X., Dong,X., Lei,L., Song,W., Zhao,H. and Lai,J. (2017) Genome-wide nucleosome occupancy and organization modulates the plasticity of gene transcriptional status in maize. *Mol. Plant*, **10**, 962–974.
43. Nocetti,N. and Whitehouse,I. (2016) Nucleosome repositioning underlies dynamic gene expression. *Genes Dev.*, **30**, 660–672.
44. Tu,B.P., Kudlicki,A., Rowicka,M. and McKnight,S.L. (2005) Logic of the yeast metabolic Cycle: Temporal compartmentalization of cellular processes. *Science*, **310**, 1152–1158.
45. Kaplan,N., Moore,I.K., Fondufe-Mittendorf,Y., Gossett,A.J., Tillo,D., Field,Y., LeProust,E.M., Hughes,T.R., Lieb,J.D., Widom,J. *et al.* (2009) The DNA-encoded nucleosome organization of a eukaryotic genome. *Nature*, **458**, 362–366.
46. Stanley,D., Bandara,A., Fraser,S., Chambers,P.J. and Stanley,G.A. (2010) The ethanol stress response and ethanol tolerance of *Saccharomycescerevisiae*. *J. Appl. Microbiol.*, **109**, 13–24.
47. Rodríguez,A., Cera,T. de la, Herrero,P. and Moreno,F. (2001) The hexokinase 2 protein regulates the expression of the GLK1, HXK1 and HXK2 genes of Saccharomyces cerevisiae. *Biochem. J.*, **355**, 625–631.
48. Bai,L. and Morozov,A.V. (2010) Gene regulation by nucleosome positioning. *Trends Genet.*, **26**, 476–483.

9) Codó, L., Bayarri, G., Cid-Fuentes, J. A., Conejero, J., Hospital, A., Royo, R., … Gelpí, J. L. (2019). MuGVRE. A virtual research environment for 3D/4D genomics. BioRxiv, 602474. https://doi.org/10.1101/602474

MuGVRE. A virtual research environment for 3D/4D genomics

Laia Codó[1], Genís Bayarri[2], Javier Alvarez Cid-Fuentes[1], Javier Conejero[1], Adam Hospital[2], Romina Royo[1], Dmitry Repchevsky[1], Marco Pasi[3], Athina Meletiou[3], Mark D. McDowall[4], Fatima Reham[4], José A. Alcantara[2], Brian Jimenez-Garcia[1], Jürgen Walther[2], Ricard Illa[2], François Serra[5], Michael Goodstadt[5], David Castillo[5], Satish Sati[6], Diana Buitrago[2], Isabelle Brun-Heath[2], Juan Fernandez-Recio[1,7], Giacomo Cavalli[6], Marc Marti-Renom[5,8], Andrew Yates[4], Charles A. Laughton[3], Rosa M. Badia[1], Modesto Orozco[2,8], Josep Ll. Gelpí[1,8]*

1. Barcelona Supercomputing Center, Barcelona, Spain,
2. Institute for Research in Biomedicine, the Barcelona Institute of Science and Technology. Barcelona, Spain,
3. Sch. of Pharmacy and Centre for Biomolecular Sciences, Nottingham, UK,
4. European Molecular Biological Laboratory, European Bioinformatics Institute, Wellcome Genome Campus, Hinxton, UK
5. CNAG-CRG, Centre for Genomic Regulation (CRG), Barcelona Institute of Science and Technology (BIST), Baldiri i Reixac 4, 08028 Barcelona, Spain. Gene Regulation, Stem Cells and Cancer Program, Centre for Genomic Regulation (CRG), Dr. Aiguader 88, 08003 Barcelona, Spain. Universitat Pompeu Fabra (UPF), Barcelona, Spain.
6. Institute of Human Genetics, CNRS, Univ. Montpellier, Montpellier, France.
7. ICREA, Pg. Lluís Companys 23, 08010 Barcelona, Spain,
8. Dept. of Biochemistry and Molecular Biomedicine, University of Barcelona, Barcelona, Spain.

* To whom correspondence should be addressed. Tel: 34934034009; Fax: 34934021559; Email: gelpi@ub.edu.

Marco Pasi, present address: Ecole Normale Superieure de Cachan, Laboratory of Biology and Applied Pharmacology, Paris, Île-de-France, FR

ABSTRACT

Multiscale Genomics (MuG) Virtual Research Environment (MuGVRE) is a cloud-based computational infrastructure created to support the deployment of software tools addressing the various levels of analysis in 3D/4D genomics. Integrated tools tackle needs ranging from high computationally demanding applications (e.g. molecular dynamics simulations) to high-throughput data analysis applications (like the processing of next generation sequencing). The MuG Infrastructure is based on openNebula cloud systems implemented at the Institute for

research in Biomedicine, and the Barcelona Supercomputing Center, and has specific interfaces for users and developers. Interoperability of the tools included in MuGVRE is maintained through a rich set of metadata allowing the system to associate tools and data in a transparent manner. Execution scheduling is based in a traditional queueing system to handle demand peaks in applications of fixed needs, and an elastic and multi-scale programming model (pyCOMPSs, controlled by the PMES scheduler), for complex workflows requiring distributed or multi-scale executions schemes. MuGVRE is available at https://vre.multiscalegenomics.eu and documentation and general information at https://www.multiscalegenomics.eu. The infrastructure is open and freely accessible.

INTRODUCTION

While major milestones have been achieved in determining the sequence of DNA, understanding its 3D folding, the connection between chromatin structure and genome functionality and the links between changes in chromatin structure and pathology are still major challenges that are attracting a large research effort and have created a new area of knowledge: 3D/4D genomics. Opposite to traditional sequencing projects, 3D/4D genomics faces a major problem related to the diversity of data types and formats generated, the variety of analysis methods, and the multi-resolution nature of the navigation in a multi-scale data space. Data used in the field range from simple sequence (1D genomics), typically enriched by functional and structural annotations (2D Genomics) to contact maps, single or multiple structures (at a very wide range of resolutions) and images.

ENA (1), EGA (2) and Ensembl (3) are the most common sources of 1D genomic data in Europe, whose contents are visualized thanks to tools such as “Genome browsers” (4-8) that provide an integrated view of both sequence and annotations (2D genomics). Many research infrastructures deal with this level of data, Galaxy (9), being by far the most popular.

A second level of data includes nucleic acid and protein structures determined at the atomistic level by X-Ray crystallography or NMR spectroscopy and deposited in the Protein Data Bank (10). Analysis and visualization of data at this level have, in practice, little overlap with the 2D genomics level. This represents a major caveat for people interested in specific DNA-protein complexes. An additional type of data includes simulation trajectories, i.e. a set of structures defining the conformational ensemble of DNA (and associated proteins), which is typically obtained through atomistic or coarse-grained molecular dynamics (MD) simulations (11-17). Specific databases on DNA simulation data (18) or protein-DNA complexes (19, 20), and tools to perform flexibility analysis of nucleic acids like NAFlex (21) or 3DNA (22) are available. Few tools allow mapping 3D structures onto the genomic sequences (23-25).

A third level of data is represented by nucleosome and protein positioning studies (26-30). Raw data is obtained from sequencing analysis (MNAse-seq, Chip-seq) and is usually limited to the annotation of specific binding sites. Data at this level is available from public repositories like ArrayExpress (31), where it is presented in a 1D manner without any connection to the 3D structure and flexibility of the chromatin fiber. At the upper end of the scale the main experimental strategies include FISH (32), and chromosome conformation capture (3C-like techniques (33)), which provide chimeric sequences representing interactions within the genome, from which structural insight can be derived after a complex manipulation of the data. Data visualization at this level is complex (34); several tools are already available (35-38), but they ignore any atomistic detail, or even the general description of the nucleosome string.

In summary, 3D (structure) and 4D (dynamics) genomics cover a large variety of data types and experimental and theoretical strategies. Tools and data repositories exist; but they are not integrated. As a result, research at the different levels is performed in isolation, and researchers almost ignore alternative views coming from different levels. The Multiscale Complex Genomics project (MuG, http://www.multiscalegenomics.eu/MuG) is one of the initiatives aiming to bridge the gap among the different levels of chromatin study and provide such an integrated view. Here, we present the MuG Virtual Research Environment (MuGVRE) infrastructure, designed to provide researchers with a single access point where data and tools covering the full spectrum, from sequence and atomistic data to chromosome capture results, can be used and combined to obtain a holistic picture of chromatin. MuGVRE is cloud based, allowing for an easy deployment and extension at the technical level. It is a base infrastructure where additional tools can be plugged-in to extend the functionality. The initial offer of tools already hosted, includes sequence, structure and Hi-C data analysis and visualizers; tools to handle nucleosome positioning, and tools for performing and analyzing simulation data from atomistic to coarse-grained levels. All the tools are accessible through an intuitive personal workbench, where most technical decisions are taken automatically by the system, allowing the user to concentrate on the scientific aspects of the analysis. We believe that MugVRE is the missing element to integrate the different views of physiological DNA.

INFRASTRUCTURE DESIGN AND COMPONENTS

The MuGVRE infrastructure has been designed to fulfil the following principles:

Technical:

1. Flexible structure, able to adapt to the specific needs of the analysis tools, both in terms of software requirements, and computational resources.

2. Software scheduler(s), able to manage analysis workflows, and computational resources in a transparent and adaptable manner. This should constitute an elastic infrastructure with automatic adaptation to user loads.
3. Multi-scale execution. Analysis workflows should be executed either at the cluster level, in HPC environments, or distributed infrastructures like EGI (https://www.egi.eu), and eventually in the forthcoming European Open Science Cloud (EOSC) ecosystem.

Usage:

4. Web-based access centered on the user workspace and complemented by full programmatic access using well-established interfaces.
5. Data should be kept private, through the appropriate Authentication and Authorization Infrastructure applied to all data transactions.

Supplementary Figure S1 shows a general schema of the computational infrastructure underlying MuGVRE.

MuGVRE Main components

Cloud deployments: MuGVRE infrastructure has been designed as a fully virtualized environment. At its present state, MuGVRE has been deployed in at the Starlife cloud infrastructure (https://www.bsc.es/marenostrum/star-life), at the Barcelona Supercomputing Center, using OpenNebula (https://opennebula.org/) and the KVM hypervisor (https://www.linux-kvm.org).

Process management: MuGVRE uses two complementary layouts for process management: i) Sun Grid Engine (SGE, https://sourceforge.net/projects/gridscheduler/), in combination with OneFlow (https://docs.opennebula.org/5.4/advanced_components/application_flow_and_auto-scaling/appflow_use_cli.html), a component of the OpenNebula framework that allows managing Multi-VM applications and auto-scaling. SGE is used to manage applications where no complex workflows are necessary, requiring only to deploy additional workers on peaks of demand, ii) the COMPS Superscalar (COMPSs) programming model (and its python binding pyCOMPSs (39)), managed by the Programming Model Enactment Service (PMES) (40), which interacts with cloud infrastructures through Open Cloud Computing Interface (OCCI, http://occi-wg.org/) servers. PMES/pyCOMPSs are used to control complex workflows and distributed execution.

Database manager: Operational data and metadata regarding installed tools, public repositories, and user's files are maintained in a MongoDB database (https://www.mongodb.com). The MongoDB server also contains reference data as a copy of

Protein Data Bank (10), Uniprot (41), and BiGNASim (18) database. User's data is stored in a standard filesystem in its original format.

Authentication and authorization system: Data privacy is maintained using the authentication and authorization server Keycloak (http://www.keycloak.org/) to handle all internal communications and user access. Keycloak implements OpenID Connect which allows for Web access on the code authorization flow of OAuth2, and a token-based authentication for the REST services. For registered users, authentication schemes based on username/password, but also third-party identity providers (Google, LinkedIn, Elixir) are accepted.

See Supplementary Material Section 1 for additional information about MuGVRE software components.

USER WORKSPACE

User access

MuGVRE can be accessed without authentication. Users are granted a private workspace to hold data and analysis results. Data is maintained for one week after the last access and can be recovered during this period through a unique URL address. Users desiring a longer interaction with MuGVRE are advised to register to get a permanent workspace. In such cases the user space is maintained up to two months after the last access.

Personal workspace

The MuGVRE personal workspace is the central environment for user activity. It is based on a filesystem-based layout (see Figure 2), where both uploaded data and analysis results are available. Uploaded data should be annotated to specify data types and formats. This allows the MuGVRE workspace to offer an adapted toolkit for each file, including only compatible tools and visualizers (see Suppl. Material Section 2 for a description of the procedure, and Suppl.Table S3 for a description of accepted data types and formats). The user workspace has been laid out to provide an intuitive look-and-feel. The workspace itself is structured in projects, to keep data logically organized. Within each project the input data is distributed in folders: *Uploads* (uploaded data), and *Repository* (data obtained from public repositories). The remaining folders correspond to analysis operations (a new folder is generated for any new process started in the VRE). File lists can be filtered by any of the fields (name, format, data type, or project). Additionally, a tool-based filter is available to select only valid input files for the given tool.

Three interactive toolkits containing the following options are available:

- File toolkit: Download, rename, move, compress, delete files and folders, edit their metadata, rerun jobs.

- Visualization toolkit: Available visualizers for the specific file/s selection (based on their data type and format).
- Tools toolkit: Available tools for the specific file/s selection (based on their data type and format).

Tools based access

In addition to the data centric approach, more experienced user may prefer to access directly to the analysis operation. To this end, user may select the desired operation from an ontology, covering all the operations provided by the installed tools. The user should then select the desired tool and fill in the required parameters. At this point, a selection of input files filtered out from the user workspace are suggested according to the metadata accompanying both, user's data and tool input restrictions.

Applications and developer access

MuGVRE is designed as an infrastructure open to any application designed for the analysis of 3D/4D genomics data. Tools installed in MuGVRE should accept free, unrestricted usage. Guidelines are available for developers (see Suppl. Material, Section 2). Developers wishing to include their applications are granted access to a specific workspace to manage tool definitions and execution details, and to edit the tool's help pages, and perform execution tests. Also developers get access to usage statistics, execution logs, and job associated files and metadata. The current offer of tools is detailed in Suppl. Table S4. Several data visualizers are available (Suppl, Table S5). Figure 2 shows a representative set of the types of data than can be visualized in the infrastructure. JBrowse (5) is used for sequence data (see Figure 2A). In the figure, data from a Nucleosome Dynamics analysis of MNase-seq data is depicted (blue blocks for nucleosome calls, and coverage plot). The NGL visualizer (42) is used for 3D atomistic (see Figure 2B for a transcription factor-DNA complex) or coarse-grained structure data (output from Chromatin Dynamics, Figure 2D, depicting the structure of a fragment of chromatin fiber containing nucleosomes at positions shown in Fig. 2A). Finally, the TadKit visualizer (http://3DGenomes.org/tadkit, Figure 2C) allows the combination of 1D to 3D information taken from chromosome capture data.

USAGE

Users can populate the MuGVRE workspace in several ways:

- *Direct upload into the workspace*, using the HTTPS protocol
- *Create files using an embedded text editor*. This is intended for data or metadata of reduced size.
- *Upload from an External URL*: MuGVRE can access external sites to download. This is the recommended procedure to obtain data from public repositories, or import bulky

data, as the upload process becomes a batch job. HTTP and FTP protocols are accepted, also when user credentials are required.

- *From repository:* Data imported from the list of public repositories made browsable at the infrastructure (currently ArrayExpress (31) and BigNASim(18)).

- From sample data: selected input and output examples for the available tools are provided as help to start using the interface.

Files can be selected anywhere in the workspace, and added to the execution list, where a specific tool should be then selected. Alternatively, the user can select first an analysis tool from a list of available operations. Either selection mode opens a configuration screen where the user can assign data files (Suppl. Figure S7) to the appropriate input parameters, define additional settings, and launch the tool. Executions are performed in the background and do not require the user to keep the interactive session open. Results of the analysis are added to the workspace under a separate folder that contains the output files generated by the tool, log files, and a customized results page (Suppl. Figure S10). For a complete usage example, see Supplementary Material Section 3, where we show a series of screenshots of a session centered on the analysis of MNase-seq data from yeast Chr II on phases M and G2 of the cell cycle (data taken from (43)), analyzed with the Nucleosome Dynamics tool.

DISCUSSION

3D/4D Genomics is an emerging field originated from the unplanned aggregation of different disciplines which have developed their tools, and associated data types and formats, independently. This diversity is a major obstacle towards the generation of a complete picture of chromatin structure and dynamics. MuGVRE has been designed as an integration space. It follows the traditional concept of the personal workbench, already used in general genomic workbenches like Galaxy (9) or GenePattern (44), or spaces designed for simulation data analysis like NAFlex (21). In this kind of environment, data and tools are available and the user has the freedom to design his/her own analysis pipelines. MuGVRE has an initial offer of tools and visualizers covering all levels of resolution, from atomistic simulation to chromatin fiber simulation, or Hi-C data analysis. Tools and visualizers are offered in a single space where, for example, chromosome conformation capture or nucleosome positioning data can be visualized along with sequence annotations and the structures and binding modes of the transcription factors affecting the same DNA region. A strong commitment of MuGVRE design is to free the user from understanding the technical side of the infrastructure. With this aim, not only the computational layout is hidden, but also most technical decisions are taken automatically by the system. For example, we have designed a comprehensive ontology of data types and formats that are checked internally to configure the options offered to the user. The user can just select a tool, and the workspace selects those files that match its input requirements. Output from the

analyses are reusable following the same philosophy. As a result, the user can easily configure a pipeline taking only scientific decisions and not bothering about technicalities.

MuGVRE has been designed as a large and sustainable infrastructure, which relies now on the computational capabilities of the Barcelona Supercomputing Center (http://www.bsc.es), but technical decisions have been taken to assure the compatibility with other infrastructures like Elixir computational platforms (http://elixir-europe.org), EGI (https://www.egi.eu), or EUDAT (https://eudat.eu). The choice of a fully flexible cloud system, controlled with a multiscale software scheduler, and linked to HPC facilities, assures the usage of the optimal computational environment for each specific analysis.

MuGVRE is presented as an open platform with the aim of growing in functionality, since new tools can be easily incorporated by external developers. Hosting at MuGVRE can be an option for developers alternative to build a dedicated web site to run their tools.

MuGVRE was presented to the multiscale genomics community in November 2017, and has performed already over 6,500 analysis runs.

MuGVRE is a unique tool that aims to help researchers in the 3D/4D genomics field to gain an integrated view of discipline, sharing data among the diverse analysis levels and providing a complete and integrated view on DNA. We hope that MuGVRE will foster the development, deployment and use of new strategies for the analysis of the chromatin structure that were not envisioned simply because data was kept in separate silos.

AVAILABILITY

MuGVRE: https://vre.multiscalegenomics.eu

General information and documentation: https://www.multiscalegenomics.eu

ACKNOWLEDGEMENT

We are indebted to the entire MuG consortium and to the β-testers of the application for suggestions and comments.

FUNDING

This work has been supported by the Spanish MINECO [grants BIO2015-64802-R; BFU2015-61670-EXP, TIN2015-65316-P, TEC2015-67774-C2-2-R, BFU2013-47736-P and BFU2017-85926-P], the Catalan Government [grants 2014-SGR-134, 2014-SGR-1051]; the Instituto de Salud Carlos III-Instituto Nacional de Bioinformática [INB; grants PT13/0001/0019 and PT13/0001/0028]; France: The Fondation pour la Recherche Médicale, [grant

DEI20151234396], Laboratory of Excellence EpiGenMed; European Union, H2020 programme [grants Elixir-Excelerate: 676559; BioExcel: 674728 and MuG: 676566]. ERC Council [grants 291433, 609989]; IRB, CRG, and BSC are recipients of a Severo Ochoa Award of Excellence from MINECO (Government of Spain).

Funding for open access charge: European Union and Spanish Ministry of Science.

REFERENCES

1. Karsch-Mizrachi, I., Takagi, T., Cochrane, G. and Collaboration, I.N.S.D. (2018) The international nucleotide sequence database collaboration. Nucleic Acids Res, 46, D48-D51.
2. Lappalainen, I., Almeida-King, J., Kumanduri, V., Senf, A., Spalding, J.D., Ur-Rehman, S., Saunders, G., Kandasamy, J., Caccamo, M., Leinonen, R. et al. (2015) The European Genome-phenome Archive of human data consented for biomedical research. Nat Genet, 47, 692-695.
3. Zerbino, D.R., Achuthan, P., Akanni, W., Amode, M.R., Barrel,l D., Bhai, J., Billis K, Cummins C, Gall A, Girón CG, et al.(2018) Ensembl 2018. Nucleic Acids Res, 46, D308-D314.
4. Barrios, D. and Prieto, C. (2017) D3GB: An Interactive Genome Browser for R, Python, and WordPress. J Comput Biol, 24, 447-449.
5. Buels, R., Yao, E., Diesh, C.M., Hayes, R.D., Munoz-Torres, M., Helt, G., Goodstein, D.M., Elsik, C.G., Lewis, S.E., Stein, L. et al. (2016) JBrowse: a dynamic web platform for genome visualization and analysis. Genome Biol, 17, 66.
6. Medina, I., Salavert, F., Sanchez, R., de Maria, A., Alonso, R., Escobar, P., Bleda, M. and Dopazo, J. (2013) Genome Maps, a new generation genome browser. Nucleic Acids Res, 41, W41-46.
7. Robinson, J.T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E.S., Getz, G. and Mesirov, J.P. (2011) Integrative genomics viewer. Nat Biotechnol, 29, 24-26.
8. Casper, J., Zweig, A.S., Villarreal, C., Tyner, C., Speir, M.L., Rosenbloom, K.R., et al. (2018) The UCSC Genome Browser database: 2018 update. Nucleic Acids Res. 46, D762-D769
9. Afgan, E., Baker, D., van den Beek, M., Blankenberg, D., Bouvier, D., Čech, M., Chilton, J., Clements, D., Coraor, N., Eberhard, C. et al. (2016) The Galaxy platform for accessible, reproducible and cFollaborative biomedical analyses: 2016 update. Nucleic Acids Res, 44, W3-W10.
10. Berman, H., Henrick, K., Nakamura, H. and Markley, J.L. (2007) The worldwide Protein Data Bank (wwPDB): ensuring a single, uniform archive of PDB data. Nucleic Acids Research, 35, D301-D303.

11. Dans, P.D., Zeida, A., Machado, M.R., and Pantano, S. (2010) A Coarse Grained Model for Atomic-Detailed DNA Simulations with Explicit Electrostatics. J Chem Theory Comput. 6, 1711-1725.
12. Dans, P.D., Perez, A., Faustino, I., Lavery, R. and Orozco, M. (2012) Exploring polymorphisms in B-DNA helical conformations. Nucleic acids res. 40, 10668-10678.
13. Dans, P.D., Walther, J., Gómez, H., Orozco, M. (2016) Multiscale simulation of DNA. Curr Opin Struct Biol. 37, 29-45.
14. Pasi, M., Maddocks, J.H., Beveridge, D., Bishop, T.C., Case, D.A., Cheatham, T., III, Dans, P.D., Jayaram, B., Lankas, F., Laughton, C. et al. (2014) mu ABC: a systematic microsecond molecular dynamics study of tetranucleotide sequence effects in B-DNA. Nucleic Acids Research, 42, 12272-12283.
15. Beveridge, D.L., Barreiro, G., Byun, K.S., Case, D.A., Cheatham, T.E., Dixit, S.B., et al. (2004) Molecular dynamics simulations of the 136 unique tetranucleotide sequences of DNA oligonucleotides. I. Research design and results on d(C(p)G) steps. Biophys. J. 87, 3799-7813.
16. Lavery, R., Zakrzewska, K., Beveridge, D., Bishop, T.C., Case, D.A., Cheatham, T. III, et al. (2010) A systematic molecular dynamics study of nearest-neighbor effects on base pair and base pair step conformations and fluctuations in B-DNA. Nucleic Acids Res. 38, 299-313.
17. Zakrzewska, K., Lavery, R. (2012) Towards a molecular view of transcriptional control. Curr Opin Struct Biol. 22, 160-167.
18. Hospital, A., Andrio, P., Cugnasco, C., Codo, L., Becerra, Y., Dans, P.D., Battistini, F., Torres, J., Goñi, R., Orozco, M. et al. (2016) BIGNASim: a NoSQL database structure and analysis portal for nucleic acids simulation data. Nucleic Acids Res, 44, D272-278.
19. Park, B., Kim, H. and Han, K. (2014) DBBP: database of binding pairs in protein-nucleic acid interactions. BMC Bioinformatics, 15 Suppl 15, S5.
20. Kirsanov, D.D., Zanegina, O.N., Aksianov, E.A., Spirin, S.A., Karyagina, A.S. and Alexeevski, A.V. (2013) NPIDB: Nucleic acid-Protein Interaction DataBase. Nucleic Acids Res, 41, D517-523.
21. Hospital, A., Faustino, I., Collepardo-Guevara, R., Gonzalez, C., Gelpi, J.L., Orozco, M. (2013) NAFlex: a web server for the study of nucleic acid flexibility. Nucleic Acids Research. 41, W47-W55.
22. Lu, X.J., Olson, W.K. (2008) 3DNA: a versatile, integrated software system for the analysis, rebuilding and visualization of three-dimensional nucleic-acid structures. Nature Protocols. 3, 1213-1227.
23. Duran, E., Djebali, S., Gonzalez, S., Flores, O., Maria Mercader, J., Guigo, R., Torrents, D., Soler-Lopez, M. and Orozco, M. (2013) Unravelling the hidden DNA

structural/physical code provides novel insights on promoter location. Nucleic Acids Research, 41, 7220-7230.

24. Goni, J.R., Perez, A., Torrents, D. and Orozco, M. (2007) Determining promoter location based on DNA structure first-principles calculations. Genome Biol, 8, R263.
25. Goni, J.R., de la Cruz, X. and Orozco, M. (2004) Triplex-forming oligonucleotide target sequences in the human genome. Nucleic Acids Res, 32, 354-360.
26. Chereji, R.V., Ramachandran, S., Bryson, T.D. and Henikoff, S. (2018) Precise genome-wide mapping of single nucleosomes and linkers in vivo. Genome Biol, 19, 19.
27. Flores, O., Deniz, O., Soler-Lopez, M. and Orozco, M. (2014) Fuzziness and noise in nucleosomal architecture. Nucleic Acids Research, 42, 4934-4946.
28. Sharma, S., Ding, F., Dokholyan, N.V. (2007) Multiscale modeling of nucleosome dynamics. Biophys J. 92, 1457-1470.
29. Zuiddam, M., Everaers, R., Schiessel, H. (2017) Physics behind the mechanical nucleosome positioning code. Phys Rev E. 96, 052412.
30. Johnson, D.S., Mortazavi, A., Myers, R.M. and Wold, B. (2007) Genome-wide mapping of in vivo protein-DNA interactions. Science, 316, 1497-1502.
31. Kolesnikov, N., Hastings, E., Keays, M., Melnichuk, O., Tang, Y.A., Williams, E., Dylag, M., Kurbatova, N., Brandizi, M., Burdett, T. et al. (2015) ArrayExpress update--simplifying data submissions. Nucleic Acids Res, 43, D1113-1116.
32. Cremer, M., Grasser, F., Lanctôt, C., Müller, S., Neusser, M., Zinner, R., Solovei, I. and Cremer, T. (2008) Multicolor 3D fluorescence in situ hybridization for imaging interphase chromosomes. Methods Mol Biol, 463, 205-239.
33. Dekker, J., Rippe, K., Dekker, M. and Kleckner, N. (2002) Capturing chromosome conformation. Science, 295, 1306-1311.
34. Goodstadt, M. and Marti-Renom, M.A. (2017) Challenges for visualizing three-dimensional data in genomic browsers. FEBS Lett, 591, 2505-2519.
35. Zhou, X., Lowdon, R.F., Li, D., Lawson, H.A., Madden, P.A., Costello, J.F. and Wang, T. (2013) Exploring long-range genome interactions using the WashU Epigenome Browser. Nat Methods, 10, 375-376.
36. Durand, N.C., Robinson, J.T., Shamim, M.S., Machol, I., Mesirov, J.P., Lander, E.S. and Aiden, E.L. (2016) Juicebox Provides a Visualization System for Hi-C Contact Maps with Unlimited Zoom. Cell Syst, 3, 99-101.
37. Yang, D., Jang, I., Choi, J., Kim, M.S., Lee, A.J., Kim, H., Eom, J., Kim, D., Jung, I. and Lee, B. (2018) 3DIV: A 3D-genome Interaction Viewer and database. Nucleic Acids Res, 46, D52-D57.
38. Tang, B., Li, F., Li, J., Zhao, W. and Zhang, Z. (2017) Delta: a new Web-based 3D genome visualization and analysis platform. Bioinformatics.

39. Tejedor, E., Becerra, Y., Alomar, G., Queralt, A., Badia, R.M., Torres, J., Cortes, T. and Labarta, J. (2017) PyCOMPSs: Parallel computational workflows in python. Intl. J. High Perf. Comput. Appl., 31, 66-82.

40. Lordan, F., Tejedor, E., Ejarque, J., Rafanell, R., Alvarez, J., Marozzo, F., Lezzi, D., Sirvent, R., Talia, D. and Badia, R.M. (2013) ServiceSs: An Interoperable Programming Framework for the Cloud. J. Grid. Comput., 12, 67-91.

41. (2010) The Universal Protein Resource (UniProt) in 2010. Nucleic Acids Res, 38, D142-148.

42. Rose, A.S. and Hildebrand, P.W. (2015) NGL Viewer: a web application for molecular visualization. Nucleic Acids Res, 43, W576-579.

43. Deniz, Ö., Flores, O., Aldea, M., Soler-López, M. and Orozco, M. (2016) Nucleosome architecture throughout the cell cycle. Sci Rep, 6, 19729.

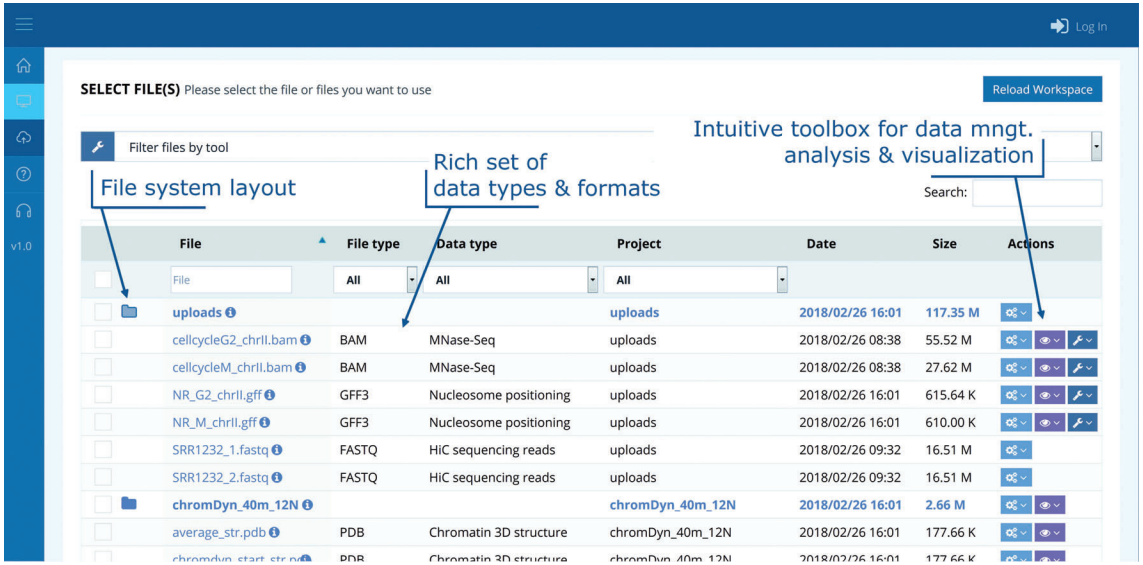
44. Reich, M., Tabor, T., Liefeld, T., Thorvaldsdóttir, H., Hill, B., Tamayo, P. and Mesirov, J.P. (2017) The GenePattern Notebook Environment. Cell Syst, 5, 149-151.e141.

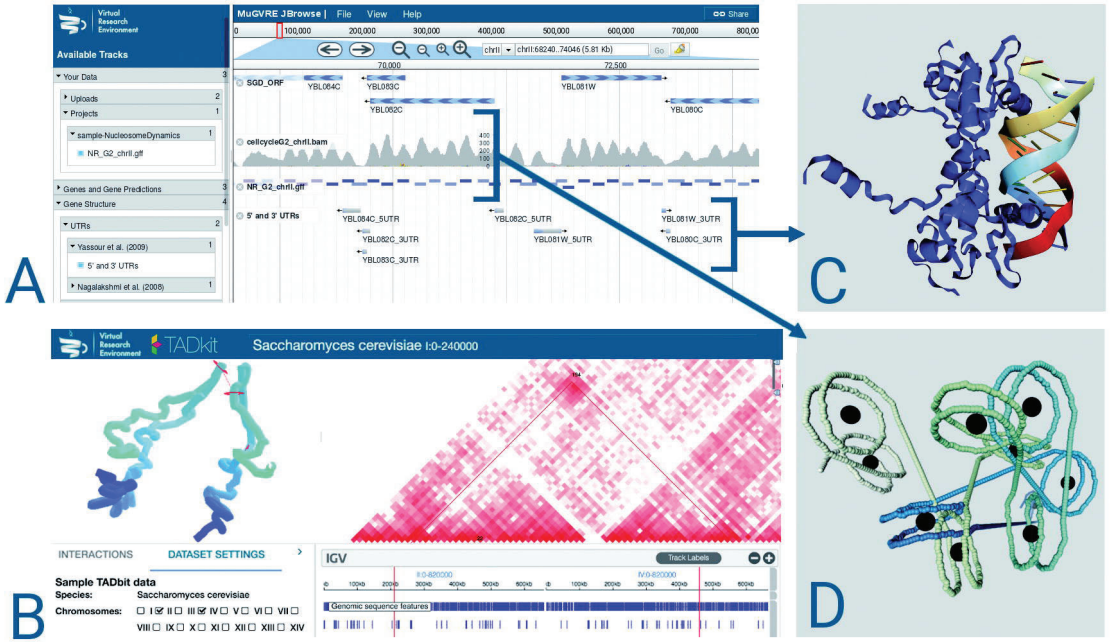
LEGENDS TO FIGURES

Figure 1. Screenshot of MuGVRE personal workspace

Figure 2. Sample visualizations at MuGVRE.

A**:** Genome browser showing reference sequences, annotations, including a nucleosome positioning track (blue marks) (Jbrowse). B: Transcription Factor structure bound to cognate DNA fragment (NGL). C. HiC adjacent matrix, combined with genomics annotations, and a 3D model of chromatin structure (Tadkit), and D: Coarse-grained model of DNA fiber with several attached nucleosomes, from ChromatinDynamics (NGL).


Log In
SELECT FILE(S) Please select the file or files you want to use
Reload Workspace
Filter files by tool
Intuitive toolbox for data mngt.
analysis & visualization
Rich set of
data types & formats
File system layout
Search:
File
File type
Data type
Project
Date
Size
Actions
File
All
All
All
uploads
uploads
2018/02/26 16:01
117.35 M
cellcycleG2_chrII.bam
BAM
MNase-Seq
uploads
2018/02/26 08:38
55.52 M
cellcycleM_chrII.bam
BAM
MNase-Seq
uploads
2018/02/26 08:38
27.62 M
NR_G2_chrII.gff
GFF3
Nucleosome positioning
uploads
2018/02/26 16:01
615.64 K
NR_M_chrII.gff
GFF3
Nucleosome positioning
uploads
2018/02/26 16:01
610.00 M
SRR1232_1.fastq
FASTQ
HiC sequencing reads
uploads
2018/02/26 09:32
16.51 M
SRR1232_2.fastq
FASTQ
HiC sequencing reads
uploads
2018/02/26 09:32
16.51 M
chromDyn_40m_12N
chromDyn_40m_12N
2018/02/26 16:01
2.66 M
average_str.pdb
PDB
Chromatin 3D structure
chromDyn_40m_12N
2018/02/26 16:01
177.66 K
v1.0


A
MuGVRE JBrowse | File View Help
Share
Available Tracks
Your Data
Uploads
Projects
sample-NucleosomeDynamics
NR_G2_chrII.gff
Genes and Gene Predictions
Gene Structure
UTRs
Yassour et al. (2009)
5' and 3' UTRs
Nagalakshmi et al. (2008)
SGD_ORF
YBL064C
YBL063C
YBL062C
YBL061W
YBL060C
cellcycleG2_chrII.bam
NR_G2_chrII.gff
5' and 3' UTRs
YBL064C_5UTR
YBL062C_5UTR
YBL061W_3UTR
YBL062C_3UTR
YBL061W_5UTR
YBL060C_3UTR
YBL063C_3UTR
C
B
TADkit
Saccharomyces cerevisiae I:0-240000
INTERACTIONS
DATASET SETTINGS
Sample TADbit data
Species:
Saccharomyces cerevisiae
Chromosomes:
IGV
Track Labels
Genomic sequence features
D

10) Hospital, A., Andrio, P., Cugnasco, C., Codó, L., Becerra, Y., Dans, P. D., … Gelpí, J. L. (2016). BIGNASim: a NoSQL database structure and analysis portal for nucleic acids simulation data. Nucleic Acids Research, 44(D1), D272–D278. https://doi.org/10.1093/nar/gkv1301

D272–D278 *Nucleic Acids Research, 2016, Vol. 44, Database issue* *Published online 26 November 2015*
doi: 10.1093/nar/gkv1301

BIGNASim: a NoSQL database structure and analysis portal for nucleic acids simulation data

Adam Hospital[1,2], Pau Andrio[2,3], Cesare Cugnasco[3,4], Laia Codo[2,3], Yolanda Becerra[3,4], Pablo D. Dans[1,2], Federica Battistini[1,2], Jordi Torres[3,4], Ramón Goñi[2,3], Modesto Orozco[1,2,3,5,*] and Josep Ll. Gelpí[2,3,5,*]

[1]Institute for Research in Biomedicine (IRB Barcelona), The Barcelona Institute of Science and Technology, Baldiri Reixac 10-12, 08028 Barcelona, Spain, [2]Joint BSC-IRB Research Program in Computational Biology, Baldiri Reixac 10-12, 08028 Barcelona, Spain, [3]Barcelona Supercomputing Center, Jordi Girona 29, 08034 Barcelona, Spain, [4]Dept. Computer Architecture, Technical University of Catalonia (UPC-BarcelonaTech), 08034 Barcelona, Spain and [5]Department of Biochemistry and Molecular Biology, University of Barcelona, 08028 Barcelona, Spain

Received August 27, 2015; Revised October 30, 2015; Accepted November 02, 2015

ABSTRACT

Molecular dynamics simulation (MD) is, just behind genomics, the bioinformatics tool that generates the largest amounts of data, and that is using the largest amount of CPU time in supercomputing centres. MD trajectories are obtained after months of calculations, analysed *in situ,* and in practice forgotten. Several projects to generate stable trajectory databases have been developed for proteins, but no equivalence exists in the nucleic acids world. We present here a novel database system to store MD trajectories and analyses of nucleic acids. The initial data set available consists mainly of the benchmark of the new molecular dynamics force-field, parmBSC1. It contains 156 simulations, with over 120 μs of total simulation time. A deposition protocol is available to accept the submission of new trajectory data. The database is based on the combination of two NoSQL engines, Cassandra for storing trajectories and MongoDB to store analysis results and simulation metadata. The analyses available include backbone geometries, helical analysis, NMR observables and a variety of mechanical analyses. Individual trajectories and combined meta-trajectories can be downloaded from the portal. The system is accessible through http://mmb.irbbarcelona.org/BIGNASim/. Supplementary Material is also available *on-line* at http://mmb.irbbarcelona.org/BIGNASim/SuppMaterial/.

INTRODUCTION

After almost 40 years since the first biomolecular simulation, molecular dynamics (MD) has become a mature technique to assess the dynamic properties of macromolecules. Modern MD simulations are reaching, in a routine manner, the multi-nanosecond and even the microsecond scale, approaching then the biologically relevant time scale. These huge trajectory files need to be processed, and ideally stored for further analysis. However, in practice most of these trajectories are lost after a typically rather superficial analysis. This leads to duplication of efforts, lack of reference data sets for benchmarking and the impossibility to perform genome-scale analysis involving hundreds or thousands of trajectories.

Strategies in building simulation databases

Three large simulation databases have been reported: Dynameomics (1), oriented to the study of protein folding and stability, reporting over 7000 simulations in the nanosecond range (although only 100 were distributed); MoDEL (2), aiming to cover a representative subset of the protein space, with over 1800 10-ns simulations, distributed in a compressed format; and Dynasome (3), reporting a comprehensive collection of protein dynamics properties obtained from over 110 0.1-μs simulations, also representative of the protein space. They had a significant coverage of the proteome, and involved the generation of Terabytes of trajectory data. Two of them reported details about the strategies used to handle data. Dynameomics chose a particular database engine (MOLAP) (4), with the capability of being assessable using complex points of view, like time slice and specific molecular fragments. Although MOLAP was flexible on the criteria to retrieve a trajectory, its use required specific software for the analysis. On the other hand,

*To whom correspondenceshould be addressed. Tel: +34 934034009; Fax: +34 934021559; Email: gelpi@ub.edu
Correspondence may also beaddressed to Modesto Orozco. Tel: +34 934037155; Fax: +34 034037175; Email: modesto.orozco@irbbarcelona.org

MoDEL (2) used a more conservative approach where trajectory data were kept in their original format. This allowed to use existing analysis software. Although the access to data was less flexible, MoDEL relied on a special file system layout to speed up data retrieval. Also, a complete SQL-based metadata storage allowed to define specific time-slices or molecular fragments, making possible to pre-calculate, and store, relevant analysis data. Not being a simulation database *per se*, more recently, the iBIOMES project (5,6) reported an infrastructure to manage and share distributed simulation data, based in the iRODS framework (https://irods.org/). In the nucleic acids world, the ABC consortia recently reported microsecond simulations of all unique DNA tetramers (7), generating near 10 TB of data. The project did not report the implementation of any formal database structure, and data are stored in their original flat files in a series of computers from the European and American participating groups.

Analysis portals for molecular dynamics simulations

Trajectory analysis is usually done using software provided together with simulation codes, which is typically refined to analyse protein dynamics. For nucleic acids, specific software, independent from simulation engines, has been developed and used as *de facto* standard (8–11). Particularly, Curves or 3DNA are widely used to obtain helical parameters, the basis of nucleic acids conformational analysis. Using the experience gained in MoDEL (2) and MDWeb (12) projects, our group recently developed a new portal, NAFlex (13), which allows a non-experienced user to setup simulations starting from either DNA sequences, or 3D structures, and providing a wide repertoire of post-trajectory analysis both general to macromolecules and specific of nucleic acids.

We present here BIGNASim, a comprehensive platform including a database system and analysis portal, aimed to be a general database for handling nucleic acids simulations. At its initial stage, the database has been populated with the trajectories prepared during the development and validation of the parmBSC1 force-field (14) The database allows direct access to trajectory data, and contains a complete set of pre-computed analyses. Additionally, the database is provided with a flexible NAFlex-based engine allowing users to perform their own analysis pipelines. BIGNASim accepts the submission of new trajectory data. A simpler version of the database managing software and analysis package is also available for download.

DATABASE DESIGN AND IMPLEMENTATION

BIGNASim (Figure 1) is based on the combination of two NoSQL database engines, Cassandra (http://cassandra.apache.org/) and MongoDB (https://www.mongodb.org), and an adapted version of the analysis section of our Nucleic Acids MD portal NAFlex (13). For trajectory data manipulation, the platform uses MDPlus, an in-house python library that integrates MDAnalysis tools (15) with a developed Cassandra interface.

The design of the BIGNASim platform has followed a similar dual approach as done in the case of the MoDEL project. The main specifications of the design were:

1. Storage, using a consistent structure, of trajectory data, simulation metadata and analysis results.
2. Retrieval of the trajectory data on the three coordinates: simulation, time-slice and molecular fragments. Trajectory data should be retrieved in a format that is compatible with the existing analysis software.
3. Storage of analysis and simulation metadata. Database structure should allow storing any kind of analysis result and being flexible enough to incorporate new analysis data without the need of reconfiguration.

The database structure has been divided in two subsystems: (i) the trajectory subsystem, based on Cassandra, and (ii) the analysis and metadata subsystem, based on MongoDB. These two types of databases show characteristics that are appropriate for the BIGNASim DB purposes. On one side, Cassandra is a column-oriented database, especially efficient when data can be represented in key-value pairs. The simplicity of trajectory data structure, a uniform series of Cartesian coordinates that should be retrieved in well-known groups of data, makes it ideal to be handled by the Cassandra engine. On the other hand, MongoDB is a document oriented database, where data should not follow a rigid schema. MongoDB may store from single values, to 2D or 3D data, or even full length trajectory videos within a single document. Its flexibility, especially with respect to the structure of the stored documents, allows the use of a common data structure both in the database and in the analysis software. The capacity of both systems scales horizontally and they can coexist in the same computer equipment. Finally, simulation metadata has been placed in the MongoDB database allowing an easier interaction with analysis data. Indexing coordinates used in both subsystems of the BIGNASim database are fully consistent, in the way that analysis and trajectory data match naturally. A detailed description of the database structure and capabilities can be found in Supplementary Material.

DATA PORTAL

We have developed a data portal to offer several ways of accessing the data:

1. Browse and search on the trajectory data set following a rich set of options, including available metadata, sequence, or molecular fragments.
2. Access to simulation details, and quality control of the trajectories.
3. Access to both standard MD analysis results, and also nucleic acids specific ones.
4. Access to global meta-simulation analysis results.
5. Possibility of downloading trajectories or meta-trajectories for further in-house analysis.
6. Possibility of re-analysing trajectory fragments (either by time slice, or molecular fragment) within the portal.

Searching and browsing the database

Simulations can be located by: (i) sequence, (ii) sequence fragments and (iii) simulation metadata (Figure 2A). In the case of sequence search, regular expressions are accepted,

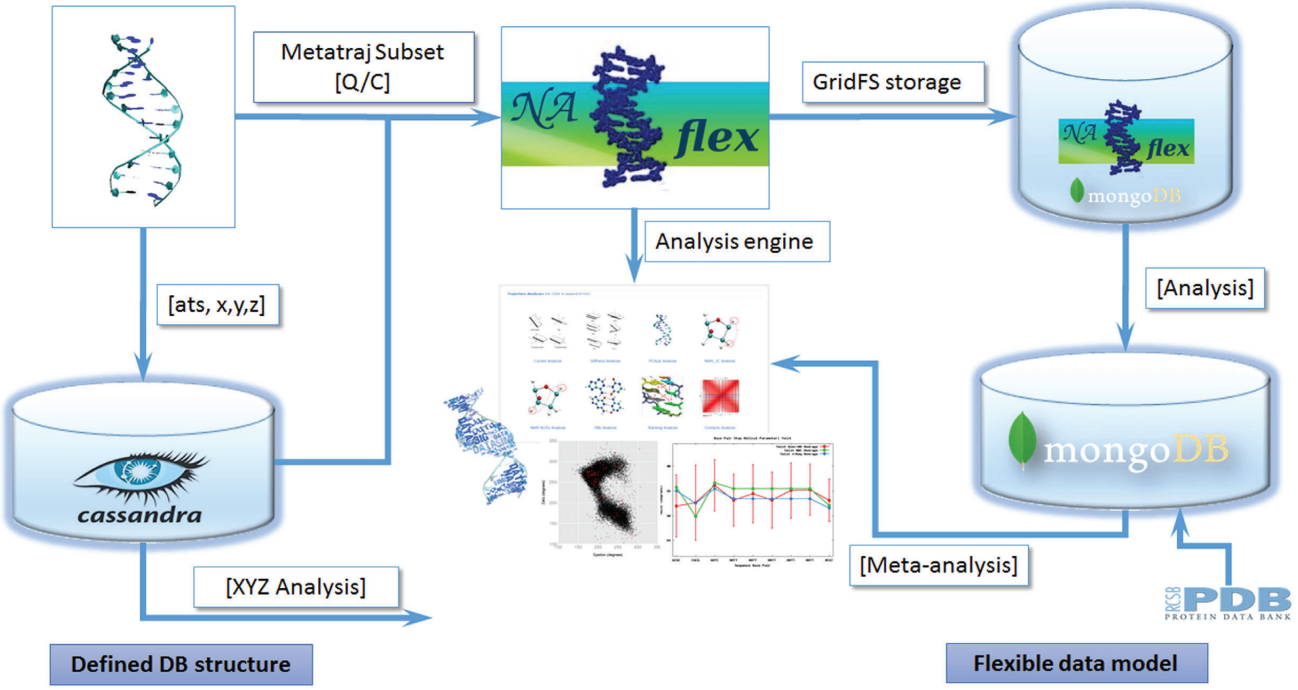


Figure 1. Global outline of the database platform and data flow.

allowing searching for degenerated sequence strings. Additionally, sequences corresponding to structures in the PDB structures can be retrieved and inserted automatically. Simulations containing defined sequence fragments (i.e. bases, base-pairs or base-pair steps) can be specifically located (see examples in Supplementary Material). After selection, simulations are shown in the browser screen (Figure 2B). Database browser includes an advanced filtering engine to make the navigation easier. From this screen, individual or combined analyses, and also meta-trajectories combining the selected simulations and fragments, can be obtained. Once a simulation is selected, its description screen contains four sections (see Supplementary Figure S1): (i) **Nucleic acid data.** Information about sequence, molecular details, and links to PDB, and Nucleic Acids Database (NDB) (16), when applicable (Supplementary Figure S1A); (ii) **MD simulation.** Information about simulation, trajectory (video and interactive JSmol, http://wiki.jmol.org/index.php/JSmol) (Supplementary Figure S1B); (iii) **Trajectory analyses.** Access to the available pre-computed analyses (Supplementary Figure S1C) and (iv) **Trajectory selection.** Possibility to extract a particular slice and/or atom selection of the trajectory (Supplementary Figure S1D). Figure 2C shows a representative screen to access to analysis data for one, several selected trajectories, or global analysis including all database data (each bullet links to the analyses of the indicated molecular fragment). As an example, Figure 3 shows details of the four consecutive steps required to obtain the twist helical parameter analysis of a CG bp-step. A complete help with tutorials can be found at the BIGNASim Web site, and in the Examples of Use (Section 5 in Supplementary Material).

On-demand trajectories and analysis

The BIGNASim portal allows the user to download dry/imaged trajectories from the available simulations for further analysis. Additionally, the Cassandra's infrastructure offers the possibility of generating new trajectories choosing either time-slices, or molecular fragments (see Supplementary Figure S1D). Those trajectories can be downloaded for 'in house' analysis, but also can be sent to the NAFlex engine (see Figure 1) accessing to specific nucleic acids oriented analysis. Additionally, meta-trajectories containing data for the same molecular fragment in different simulations can be constructed and analysed in a similar way (see Example of use in Section 5.3 of Supplementary Material). This flexibility opens a nearly infinite number of possibilities to post-analyse the stored simulation data.

Personal workspace. BIGNASim provides a personal workspace to allow users to manage simulation data. Default, anonymous, users are provided with a temporary workspace where they can store data downloaded from the database. The temporary workspace holds data retrieved in a single session, however, using a specific URL provided in the download operation, it is accessible for a defined period of time. Alternatively, users may register on the system and get a filesystem-like permanent workspace. In this case, users will find all downloaded data in a single place. The same structure can be used to upload data to be submitted to the database.

Available analysis

BIGNASim includes a variety of analysis, specific suited to nucleic acids (see Supplementary Table S1 for a complete

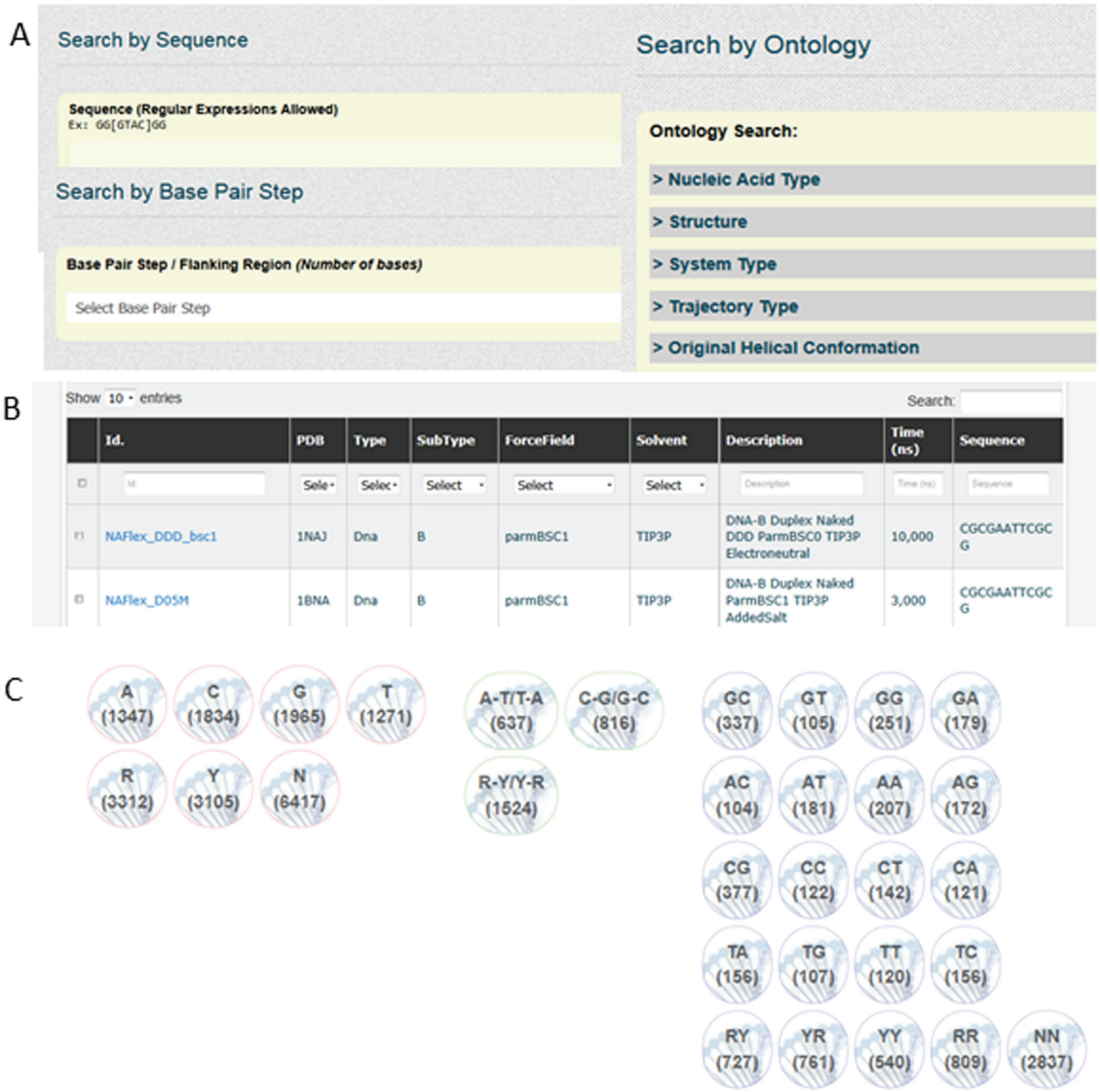


Figure 2. Details of screenshots of the BIGNASim portal. (**A**) Details of the three search options. (**B**) Browser table. Column selectors and top search box allow filtering contents. (**C**) Portal to available analyses, after trajectory selection. Also available for global database analyses. Each bullet leads to analyses of the indicated molecular fragment. Number in parentheses indicates the available data items on each option. Full screenshots are available in the Supplementary Material examples

list). Those include standard Cartesian and helical analyses (9,12,17), principal component (PCA) (18,19) and helical stiffness analysis (20,21). The server offers also the possibility to determine stacking and hydrogen bonding interaction energies along the trajectory, as well as NMR observables. In the case of protein-nucleic acid complexes, analysis are performed on the nucleic acid component, and the protein component is sent to our flexibility analysis portal, FlexServ (http://mmb.irbbarcelona.org/FlexServ). New analysis protocols, including specific methods for RNA or protein-nucleic acid complexes, are expected to be added to the platform in a near future.

DATA DESCRIPTION AND STATISTICS

BIGNASim has been designed to become a long term archival platform for Nucleic Acids simulations, and its content is expected to be in constant growth, incorporating validated simulations from other groups. Both scientific and technical quality of the stored data will be assured through a series of requirements (see Section 3 in Supplementary Ma-

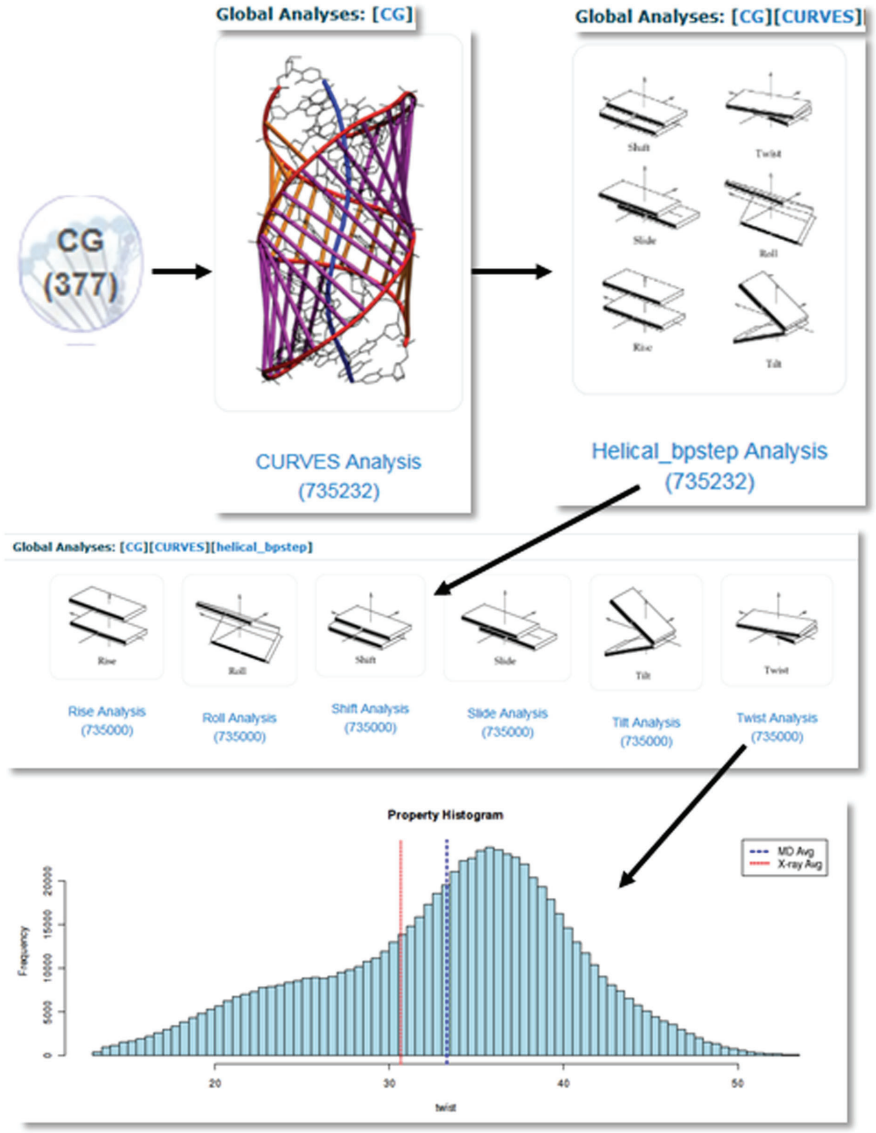


Figure 3. Screenshots of the BIGNASim portal. Example of navigation in the analysis structure for obtaining the twist parameter of CG bp-steps. (i) Selection of series of analysis based on curves. (ii) Selection of helical parameters. (iii) Selection of the twist parameter calculated for CG steps on all individual frames. Numbers in parentheses indicate the amount of available data items on each option. Raw histogram data are available for downloading. Full screenshots are available in the Supplementary Material examples.

terial for the procedure and instructions to submit new trajectories to the database). Simulations on BIGNASim are grouped internally in logical data sets that can be eventually made public or kept on-hold depending on project's requirements. Its initial public data set corresponds mainly to the benchmarking of the parmBSC1 force-field (14). Table 1 shows the present global statistics of BIGNASim contents. Detailed statistics are kept updated at the BIGNASim site. To avoid bandwidth problems the data directly available from the web portal consist of 5000 frames of dry, imaged trajectories. Downloadable trajectories are fully consistent with the pre-calculated analyses available at the web site. Full trajectories, and direct access to the database are available on request.

DISCUSSION

Data management is a major concern in modern bioinformatics. Most of the large scale bioinformatics data projects, usually in the genomics or biomedical field, invest significant efforts in data organization and provide specialized structures to this aim (the Data Control Centres). However, little effort has been made on finding similar solutions in the biosimulation world, where also large volume of data is generated. This leads to the loss of precious information and to the continuous recalculation of trajectories that have been already obtained many times before in different laboratories around the world.

The major issues of making such database open to the community, in the same way as the Protein Data Bank (22)

Table 1. Global statistics of BIGNASim

Type of simulation	Number of simulations	Cumulated simulation time
Total	156	120 μs
DNA simulations	136	99 μs
RNA simulations	14	15.6 μs
Prot-DNA complexes	6	5.5 μs
Type of analysed group	**Number of groups**	**Number of stored data items**
Total	12 449	18 092 839
Nucleotides (A, C, T, G)	6 516	9 643 652
Base Pairs (AT/TA, GC/CG)	3 043	4 155 377
Base Pair Steps (XpY)	2 890	4 293 810

Up to date statistics are available at BIGNASim portal.

is used to deposit experimental structures, are the limitation of the database platforms used (mainly SQL based systems), the lack of standards to describe the data and, the lack of tools to analyse trajectories at a high-throughput regime. Several initiatives in this direction exist though. In 2013 the European Scalalife project published a white-paper on Standards for Data Handling, also available at BIGNASim web site. Later, Cheatham's group presented a different but compatible ontology for simulation data (23). The latter ontology has a better coverage on the simulation process concepts while the Scalalife document was centred in data description. In this work, a variant of the Scalalife ontology has been used, and completed, for the generation of simulation metadata (see Ontology, Section 2 in Supplementary Material).

Our aim here is to build a generally usable simulation database that will be specially suited for storage and analysis of nucleic acids trajectories.

The first decision made in our design was on the nature of the database platform. It became clear from previous experiences that traditional SQL based systems were too limited for two main reasons: the inability to grow indefinitely, and the need of a rigid schema. Modern NoSQL systems have solved these two issues: they can scale horizontally and do not require the previous setting of a data schema.

As noted above, we have chosen Cassandra to store trajectories and MongoDB for analysis and metadata. Both systems have specific advantages. Cassandra is very efficient in data retrieval, especially for simple data structures. Data are stored as raw atom-per-time coordinates, so no specific format is required. For coordinates I/O, we rely on MDPlus, an extended version of MDAnalysis (15), which is compatible with most used trajectory formats. Therefore our system is able to select the output data format on-the-fly, and hence to interact with most analysis software. The Cassandra subsystem allows recovering any set of molecular fragments and time-slices and even generating meta-trajectories joining together data from several trajectories sharing a common molecular fragment. This is a particularly interesting feature in the simulation of nucleic acids, as it is common to analyse a single sequence fragment in different environments (7,24). On the analysis side, MongoDB was selected due to its flexibility in data representation. Analysis data can be referred to single snapshots, or averaged over trajectories, or meta-trajectories; they can correspond to several types of molecular fragments (single nucleotides, base pairs, base-pair steps); and can lead to any data type, from single values to 3D grids. MongoDB offers a very flexible data layout in a way that any data objects could be easily mapped. Additionally, its powerful indexing engine allows searches at any level (see examples of use in Supplementary Material). MongoDB's GridFS is used as a file-system substitute to communicate the Cassandra with the MongoDB subsystems and to support user workspace. This provides an increased performance over traditional file systems. BIGNASim is sharing the server with a complete replica of the Protein Data Bank, which allows enriching analyses with experimental data in a transparent manner (see Use Case 4 in Supplementary Material).

Last, but not least, we cannot ignore that the MD analysis world is in continuous evolution. Our database structure and analysis portal have been designed to allow the easy incorporation of new analysis types without the need of reconfiguration, which guarantees the long-term suitability of our project.

CONCLUSION

We have presented here a complete platform to hold and analyse nucleic acids simulation data. It is based on two NoSQL database engines, Cassandra to hold trajectory data and MongoDB for analyses and metadata. At its initial release, the database included the complete data set used for the validation of the new parmBSC1 force-field (more than 120 μs of cumulated trajectory data), but its structure is open to grow to integrate new simulations and analysis strategies. The system is not limited in size, as the database engines used scale horizontally, or complexity, as MongoDB allows for a fully flexible data schema. Trajectory data can be translated to the desired data format on-the-fly, using the MDPlus package. Most common analyses (helical parameters, NMR observables, stiffness, hydrogen bonding and stacking energies and geometries) are precalculated for the trajectories available, to speed up their retrieval, but any of those analyses can be also done interactively using our NAFlex interface, directly connected to the platform. Additionally, whole trajectories, fragments or meta-trajectories can be analysed or downloaded for further in-house processing. To our knowledge this is the most ambitious database initiative in the world of nucleic acids simulations, and we expect that it will set the basis for a

more general strategy in developing distributed simulation databases.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We are indebted to the members of the ABC consortium for many discussions, and the parmBSC1 contributors for the BIGNASim initial data set. We thank Dmitry Repchevsky for useful discussions about BIGNASim ontology.

FUNDING

Spanish Ministry of Science [BIO2012-32868, SEV-2011-00067, TIN2012-34557]; Catalan Government [2014-SGR-134, 2014-SGR-1051]; Institut Català de Recerca I Estudis Avançats, ICREA Academia [to M.O.], Instituto de Salud Carlos III-Instituto Nacional de Bioinformática [PT13/0001/0019, PT13/0001/0028]; European Research Council [ERC_SimDNA]; European Union, H2020 programme [Elixir-Excellerate: 676559; BioExcel: 674728, MuG: 676566]; PEDECIBA and SNI (ANII, Uruguay) [to P.D.D.]. Funding for open access charge: European Union [MuG: 676566].
Conflict of interest statement. None declared.

REFERENCES

1. van der Kamp,M.W., Schaeffer,R.D., Jonsson,A.L., Scouras,A.D., Simms,A.M., Toofanny,R.D., Benson,N.C., Anderson,P.C., Merkley,E.D., Rysavy,S. *et al.* (2010) Dynameomics: a comprehensive database of protein dynamics. *Structure*, **18**, 423–435.
2. Meyer,T., D'Abramo,M., Hospital,A., Rueda,M., Ferrer-Costa,C., Perez,A., Carrillo,O., Camps,J., Fenollosa,C., Repchevsky,D. *et al.* (2010) MoDEL (Molecular Dynamics Extended Library): a database of atomistic molecular dynamics trajectories. *Structure*, **18**, 1399–1409.
3. Hensen,U., Meyer,T., Haas,J., Rex,R., Vriend,G. and Grubmüller,H. (2012) Exploring protein dynamics space: the dynasome as the missing link between protein structure and function. *PLoS One*, **7**, e33931.
4. Kehl,C., Simms,A.M., Toofanny,R.D. and Daggett,V. (2008) Dynameomics: a multi-dimensional analysis-optimized database for dynamic protein data. *Protein Eng. Des. Sel.*, **21**, 379–386.
5. Thibault,J.C., Facelli,J.C. and Cheatham,T.E. III (2013) iBIOMES: managing and sharing biomolecular simulation data in a distributed environment. *J. Chem. Inf. Model.*, **53**, 726–736.
6. Thibault,J.C., Cheatham,T.E. III and Facelli,J.C. (2014) iBIOMES Lite: summarizing biomolecular simulation data in limited settings. *J. Chem. Inf. Model.*, **54**, 1810–1819.
7. Pasi,M., Maddocks,J.H., Beveridge,D., Bishop,T.C., Case,D.A., Cheatham,T. III, Dans,P.D., Jayaram,B., Lankas,F., Laughton,C. *et al.* (2014) mu ABC: a systematic microsecond molecular dynamics study of tetranucleotide sequence effects in B-DNA. *Nucleic Acids Res.*, **42**, 12272–12283.
8. Blanchet,C., Pasi,M., Zakrzewska,K. and Lavery,R. (2011) CURVES+ web server for analyzing and visualizing the helical, backbone and groove parameters of nucleic acid structures. *Nucleic Acids Res.*, **39**, W68–W73.
9. Lavery,R., Moakher,M., Maddocks,J.H., Petkeviciute,D. and Zakrzewska,K. (2009) Conformational analysis of nucleic acids revisited: Curves. *Nucleic Acids Res.*, **37**, 5917–5929.
10. Kumar,R. and Grubmuller,H. (2015) do_x3dna: a tool to analyze structural fluctuations of dsDNA or dsRNA from molecular dynamics simulations. *Bioinformatics*, **31**, 2583–2585.
11. Lu,X.J. and Olson,W.K. (2008) 3DNA: a versatile, integrated software system for the analysis, rebuilding and visualization of three-dimensional nucleic-acid structures. *Nat. Protoc.*, **3**, 1213–1227.
12. Hospital,A., Andrio,P., Fenollosa,C., Cicin-Sain,D., Orozco,M. and Gelpi,J.L. (2012) MDWeb and MDMoby: an integrated web-based platform for molecular dynamics simulations. *Bioinformatics*, **28**, 1278–1279.
13. Hospital,A., Faustino,I., Collepardo-Guevara,R., Gonzalez,C., Gelpi,J.L. and Orozco,M. (2013) NAFlex: a web server for the study of nucleic acid flexibility. *Nucleic Acids Res.*, **41**, W47–W55.
14. Ivani,I., Dans,P.D., Noy,A. and Pérez,A. (2015) Parmbsc1: a refined force field for DNA simulations. *Nature Methods*, doi:10.1038/nmeth.3658.
15. Michaud-Agrawal,N., Denning,E.J., Woolf,T.B. and Beckstein,O. (2011) Software News and Updates MDAnalysis: a toolkit for the analysis of molecular dynamics simulations. *J. Comput. Chem.*, **32**, 2319–2327.
16. Coimbatore Narayanan,B., Westbrook,J., Ghosh,S., Petrov,A.I., Sweeney,B., Zirbel,C.L., Leontis,N.B. and Berman,H.M. (2014) The Nucleic Acid Database: new features and capabilities. *Nucleic Acids Res.*, **42**, D114–D122.
17. Blanchet,C., Pasi,M., Zakrzewska,K. and Lavery,R. (2011) CURVES plus web server for analyzing and visualizing the helical, backbone and groove parameters of nucleic acid structures. *Nucleic Acids Res.*, **39**, W68–W73.
18. Amadei,A., Linssen,A.B.M. and Berendsen,H.J.C. (1993) Essential dynamics of proteins. *Proteins-Struct. Funct. Genet.*, **17**, 412–425.
19. Noy,A., Meyer,T., Rueda,M., Ferrer,C., Valencia,A., Perez,A., de la Cruz,X., Lopez-Bes,J.M., Pouplana,R., Fernandez-Recio,J. *et al.* (2006) Data mining of molecular dynamics trajectories of nucleic acids. *J. Biomol. Struct. Dyn.*, **23**, 447–455.
20. Lankas,F., Sponer,J., Hobza,P. and Langowski,J. (2000) Sequence-dependent elastic properties of DNA. *J. Mol. Biol.*, **299**, 695–709.
21. Olson,W.K., Gorin,A.A., Lu,X.J., Hock,L.M. and Zhurkin,V.B. (1998) DNA sequence-dependent deformability deduced from protein-DNA crystal complexes. *Proc. Natl. Acad. Sci. U.S.A.*, **95**, 11163–11168.
22. Berman,H.M., Battistuz,T., Bhat,T.N., Bluhm,W.F., Bourne,P.E., Burkhardt,K., Iype,L., Jain,S., Fagan,P., Marvin,J. *et al.* (2002) The Protein Data Bank. *Acta Crystallogr. D Biol. Crystallogr.*, **58**, 899–907.
23. Thibault,J.C., Roe,D.R., Facelli,J.C. and Cheatham,T.E. III (2014) Data model, dictionaries, and desiderata for biomolecular simulation data indexing and sharing. *J. Cheminformatics*, **6**, 4.
24. Beveridge,D.L., Barreiro,G., Byun,K.S., Case,D.A., Cheatham,T.E., Dixit,S.B., Giudice,E., Lankas,F., Lavery,R., Maddocks,J.H. *et al.* (2004) Molecular dynamics simulations of the 136 unique tetranucleotide sequences of DNA oligonucleotides. I. Research design and results on d(C(p)G) steps. *Biophys. J.*, **87**, 3799–3813.

11) Andrio, P., Hospital, A., Conejero, J., Jordà, L., del Pino, M., Codó, L., … Gelpí, J. L. (2019). BioBB: BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows. Nature Scientific Data, 6(1), 169. https://doi.org/10.1038/s41597-019-0177-4

SCIENTIFIC DATA

OPEN

ARTICLE

BioExcel Building Blocks, a software library for interoperable biomolecular simulation workflows

Received: 25 March 2019
Accepted: 16 August 2019
Published online: 10 September 2019

Pau Andrio[1], Adam Hospital[2], Javier Conejero[1], Luis Jordá[1], Marc Del Pino[1], Laia Codo[1], Stian Soiland-Reyes[3], Carole Goble[3], Daniele Lezzi[1], Rosa M. Badia[1], Modesto Orozco[2,4] & Josep Ll. Gelpi[1,4]

In the recent years, the improvement of software and hardware performance has made biomolecular simulations a mature tool for the study of biological processes. Simulation length and the size and complexity of the analyzed systems make simulations both complementary and compatible with other bioinformatics disciplines. However, the characteristics of the software packages used for simulation have prevented the adoption of the technologies accepted in other bioinformatics fields like automated deployment systems, workflow orchestration, or the use of software containers. We present here a comprehensive exercise to bring biomolecular simulations to the "bioinformatics way of working". The exercise has led to the development of the BioExcel Building Blocks (BioBB) library. BioBB's are built as Python wrappers to provide an interoperable architecture. BioBB's have been integrated in a chain of usual software management tools to generate data ontologies, documentation, installation packages, software containers and ways of integration with workflow managers, that make them usable in most computational environments.

Introduction

Biomolecular simulations have attained in the last years a level of maturity that allows to use them as "computational microscopes" to gain insight in biological processes. Atomistic simulations extend now to the μs range, approaching the time range of biological processes[1,2]. Coarse-grained simulations can go even further, in the length of simulations, and the size of the systems that can be analysed[3–6]. The traditional scope of simulations has overpassed the single protein or small nucleic acid systems to deal with relevant multiprotein and protein-nucleic acid complexes, nucleosomes, long segments of RNA, sections of chromatin or even full chromosomes[5]. This scenario envisions now a clear bridge between biomolecular simulations and genomics. Multiscale approaches can now bring together, for instance, Chip-seq data with simulation of protein-DNA complexes, or HiC or oligo-paint FISH experiments with large scale simulations of chromatin fibers[5]. However, the type of tools, and the way they are used differ between genomics and biomolecular simulations. Simulations have been traditionally based on a reduced number of well optimized codes run in HPC systems, where they indeed occupy a large amount of resources (over 60 M CPU-hours of BSC's MareNostrum supercomputer were dedicated to biomolecular simulations in 2018). On the other hand, traditional bioinformatics uses many competing tools usually orchestrated in complex workflows. Considering data, genomics mobilizes indeed the major amount of it, however, the storage of a typical μs-range trajectory on a mid-sized system requires already some hundreds of GB like a human whole genome obtained by Next-Generation Sequencing (NGS).

Workflow orchestration is a well-accepted concept in bioinformatics. No single, universal, solution exists, and the number of available frameworks to build and run workflows is large (https://github.com/common-workflow-language/common-workflow-language/wiki/Existing-Workflow-systems). Initiatives in the past like myGrid[7] and BioMoby[8], or more recent initiatives like CWL[9], or WDL (https://software.broadinstitute.org/wdl/), have attempted to define an interoperable ecosystem to run bioinformatics tools, web-services and the

[1]Barcelona Supercomputing Center (BSC), Jordi Girona 29, 08034, Barcelona, Spain. [2]Institute for Research in Biomedicine (IRB Barcelona), The Barcelona Institute of Science and Technology (BIST), Baldiri Reixac 10, Barcelona, 08028, Spain. [3]School of Computer Science, The University of Manchester, Manchester, United Kingdom. [4]Department Biochemistry and Molecular Biomedicine, University of Barcelona, Barcelona, Spain. Correspondence and requests for materials should be addressed to J.L.G. (email: gelpi@ub.edu)

workflows made out of them. Managers like Taverna[10], Rabix[11], Cromwell (https://cromwell.readthedocs.io/en/stable/), KNIME[12] or repositories like myExperiment[13] allow to execute or store workflow definitions for further re-usage. In this context, the ELIXIR (http://elixir-europe.org) organization is working to put in place recommendations to organize such ecosystem. At the level of registration, bio.tools[14] and Fairsharing[15] provide repositories for tools and standards. Specification languages like openAPI (https://www.openapis.org/), and CWL[9] are being recommended to document APIs and workflows, respectively. In terms of workflow execution, Galaxy[16] appears as the most popular framework, although other managers are also commonly used (e.g. Nextflow[17], PyCOMPSs[18], Snakemake)[19]. To formalize this scenario, the FAIR principles[20], initially presented to improve the quality of scientific data, are now being extended to research software. The key requirements for that (registries, standards, software managers and open repositories) are already available. Several organizations including the Software Sustainability Institute (https://www.software.ac.uk/), Research Software Engineers' associations, or ELIXIR itself are participating actively in the discussion.

Bioinformatics initiatives have little application to the simulation world. Simulations themselves are run in HPC systems in highly optimized environments. Most of the work, like setting up the simulation, a key step to assure the quality of the results[21–24], and the analysis of trajectories, is done almost manually. Modelers use *in-house* scripts, typically based on the software included in the simulation packages. In this situation, researchers usually limit themselves to a single package for all steps: setup, simulation, and analysis. Therefore, since the possibility of complementing software functionalities across packages is limited, developers should provide complete sets, re-implementing what other packages provide already. Additionally, since data formats are also diverse, data conversion modules proliferate, what in turn raises the question of which combinations of tools (although theoretically compatible) would give correct scientific answers.

Efforts to automate simulation setup and analysis do exist. Several graphical interfaces have been designed to ease the interaction with specific simulation packages[25–30]. These tools are especially useful for non-experts as they simplify the learning process. However, these utilities are still linked to specific simulation packages. One of the attempts, by our group, was MDWeb[31]. This was the first approach to offer a unified workbench allowing to setup a protein system for atomistic molecular dynamics simulation, able to work for GROMACS[32], NAMD[33], and Amber[34], three of the most popular simulation packages. Remarkably, MDWeb is powered internally by a series of web services built within the BioMoby framework and uses a common ontology of data types for the three simulation packages (http://mmb.irbbarcelona.org/MDWeb2/help.php?id=ontology). In this sense, this attempt, still in use with over 3,000 registered users, was rather unique. MDWeb was extended to the nucleic acids world with a nucleic-acids specific analysis portal, NAFlex[35]. At the large-scale end, systems have been designed to manage large scale simulation projects. Copernicus[36] combines peer-to-peer communication strategies with a simulation specific workflow management system, able to control large simulation sets in a distributed computational network. The iBIOMES project[37,38] reported an infrastructure to manage and share distributed simulation data, based in the iRODS framework (https://irods.org/). iBioMES has been used recently to manage nucleosome simulation data[39], in a clear example for the growing overlap between simulation and genomics. Some simulation databases have also been built. Dynameomics[40], centered in analysing protein folding and stability, MoDEL[22] offering an initial attempt of covering a significant sample of known protein structures, and BigNASim[23], specialized in Nucleic Acids. Remarkably, MoDEL and BigNASim provided ontologies for representing simulation data (https://mmb.irbbarcelona.org/BIGNASim/help.php?id=onto).

Even though a large set of tools are normally combined, the concept of workflow, as understood in general bioinformatics, is of limited usage. As said, most systems are setup and analyzed using *in-house* scripts. Recently, the BioExcel Center of Excellence (http://bioexcel.eu) has taken the objective of pushing the concept and usage of workflows into the biomolecular research field. In this work, we present a comprehensive exercise joining ELIXIR's recommendations and services, FAIR principles, and biomolecular simulations. We have selected the automatic setup for molecular dynamics simulations of a protein system including sequence variants, as case for demonstration. The aim of the exercise is to assess the feasibility of working according the FAIR principles and ELIXIR's recommendations in a field that is considered out of the scope of common bioinformatics. We will present a fully interoperable software library (the BioExcel Building Blocks, BioBBs) based mainly on (but not limited to) GROMACS[32] software components. For the deployment of BioBBs, we have leveraged existing platforms and services commonly used in bioinformatics, like BioConda[41], BioContainers[42] or Galaxy[16]. Workflows built using components of such library have been executed in several complementary computational environments, including personal desktops, virtualized systems, public e-infrastructures, and HPC systems. Besides, the components are documented using CWL and openAPI, what opens the possibility of run them in CWL complaint workflow managers.

Results and Discussion

Moving toward FAIR principles. FAIR principles[20] were defined with the aim of improving the quality of bioinformatics data repositories. Main principles include (1) *Findability:* Data should be findable, i.e. identified by permanent identifiers and included in searchable registries; (2) *Accessibility*: Data should be stored in permanent repositories and accessible in a machine readable form, (3) *Interoperability*: Data should use well-documented formats and standards to allow to interoperate with complementary datasets; and (4) *Reusability*: Documentation about the conditions and limitations of data reusability should be provided. Adherence to these principles has become part of the best-practices in bioinformatics data management and begins to be generally understood and accepted by the research community. They cannot be applied blindly to research software, but the general guidelines can be adapted.

Findability. A primary requirement for findability in the case of software is the availability of a software registry. Traditional software repositories like GitHub (https://github.com), are suitable for such usage although they are not usually seen as data resources, and the amount of available scientific metadata is limited. To overcome this

limitation, registries with different degrees of acceptance exist (https://www.genscript.com/tools.html; https://omictools.com/; https://www.fda.gov/ScienceResearch/BioinformaticsTools/default.htm). ELIXIR has pushed its own tools registry (bio.tools)[14]. It includes a large set of metadata that allows to search for tools according to their scientific utility, and provides extended metadata regarding publications, documentation and support. It is linked to ELIXIR's software benchmarking platform, openEBench (https://openebench.bsc.es), which in turn provides data for technical and scientific quality assessment of bioinformatics applications. One of the most remarkable features of bio.tools is the use of an extended ontology (EDAM[43]) for annotation. EDAM annotations allow to classify tools according the type of data they consume or produce and provides a controlled vocabulary to define their precise functionality. This information has been used to derive tools' annotation for CWL, or Galaxy[44,45] automatically. Unfortunately, ontology terms for structural bioinformatics, in general, and biomolecular simulation specifically, were scarce in EDAM. The generation of ontologies on simulation have been attempted in the past[22,23], but such ontologies have been seldom used outside the projects that generated them. However, the interest for addressing simulation data management has increased recently[24].

The first step of this exercise was to essay the registration in bio.tools of tools required for setup and analysis of a protein simulation. From this assay, several missing data types, file formats, and functionalities were collected (see Supplementary Table S1). We have taken the experience in MDWeb[31], MoDEL[22], and BigNASim[23] ontologies to fill the gaps in EDAM. The additions included setup, simulation and analysis operations, specific data types like system topology, trajectories, or principal components, and file formats covering the most popular simulation codes (Supplementary Table S1). These new terms have been already included in EDAM v1.22 (https://raw.githubusercontent.com/edamontology/edamontology/master/EDAM_dev.owl) and will be available for tools annotation in short. More than thirty simulation related tools, besides the BioBB library components have been registered in bio.tools. To provide an additional means for findability, a BioSchemas – based specification (http://bioschemas.org/) has been included in the appropriate places of BioBB's documentation.

A software architecture for interoperability. The recipe for full tool interoperability is theoretically simple: the use of a common, universal, data model. Past attempts like myGrid[7] and BioMoby[8] put foundations to this concept, by building a community-based data ontology and suggesting tool developers to stick to it when generating new tools. However, this attempt was not successful. The community-based approach made difficult to keep control on the evolution of the ontology. Similarly, in Galaxy servers, for instance, system administrators may add *ad-hoc* types and formats, hence contributing to make the scenario even more complex. In summary, attempting to generate a common data model for bioinformatics remains as a hard issue. Fortunately, when we focus on specific fields (NGS, array analysis, etc.), the options are limited, and *de-facto* standards do exist (bam, vcf, gff file formats in NGS analysis, for instance). Similarly, in biomolecular simulation the limited number of software packages makes the scenario less complicated. In any case, however, tool interoperability is an issue; a large set of operations in bioinformatics are, in fact, format conversions, and there is no security that an input data file is compatible with a given tool, even though that the format is the correct one.

In this exercise, we have defined a specific software architecture to contribute to the interoperability (Fig. 1). We use simple wrappers, written in Python, to encapsulate software components. Wrappers are organized in layers. The inner layer corresponds to the original tool, unaltered. Command-line tools, web services, software containers, or even remote calls to HPC systems, can be included here. A second one, *the compatibility layer*, provides the module with a well-defined interface for input, output, configuration, and provenance. It performs internally the necessary format conversions at input and output and launches the tool. This interface can be fully documented and specified using accepted standards like openAPI or CWL and can remain stable even when the associated tool needs to be updated. These two-layer wrappers can be already integrated in scripts as Python modules or executed as standalone command-line tools. A third layer, *the adaptor*, may be required for the integration in execution engines or e-infrastructures. BioBBs adaptors for Galaxy, PyCOMPSs, and CWL compliant managers are provided. Such adaptors can be used as templates to extend the usability of the library to other environments.

This architecture, even though it does not provide a common data model, do provide a uniform and stable interface, with enough information to plug the components into interoperable workflows (see below). Besides, any updates in the inner software tool would require only to update the wrapper, maintaining compatibility with previous versions, workflows, and with the chosen deployment options. Table 1 shows the present list of BioBBs with indication of their functionalities and associated tools.

Providing accessibility and enabling (re)usability. In the case of tools, the *accessibility* requirement is even stricter than for data: software not only should be accessible, it needs to be installed and executed. Different execution scenarios should be considered in the case of biomolecular simulations. They include personal workstations, used mainly for setup and analysis, or HPC systems where simulations are usually obtained. To address this principle, BioBB's use several deployment possibilities. Figure 2 shows a global information flow, and Online-only Table 1 summarizes the URLs corresponding to the different BioBB deployment alternatives. The main software repository used is available on Github. Information embedded in the code allows to generate (1) documentation using the ReadTheDocs platform (https://readthedocs.org/), (2) a JSON schema for library specification using openAPI, and (3) a reference CWL specification. To ease the deployment in a complete set of environments we have put together several packaging systems and services (Fig. 2). From the code deposited in Github, BioBBs have been uploaded to the Python Packaging Index Pypi (https://pypi.org/). Also, BioConda[41] packages have been prepared. These will allow to handle software dependencies in a transparent way, including the installation of the embedded tools. Considering only these two options, the package would be already available for installation where command-line is the main execution procedure, like personal workstations, clusters, virtual machines, or HPC. Installation can be done both as system-wide Python packages or using Python virtual environments. This kind of installation is illustrated by the execution of the lysozyme test (see below) in a Jupyter Notebook (https://jupyter.org/). Following from BioConda packages, and due to its integration with the BioContainers project[42],

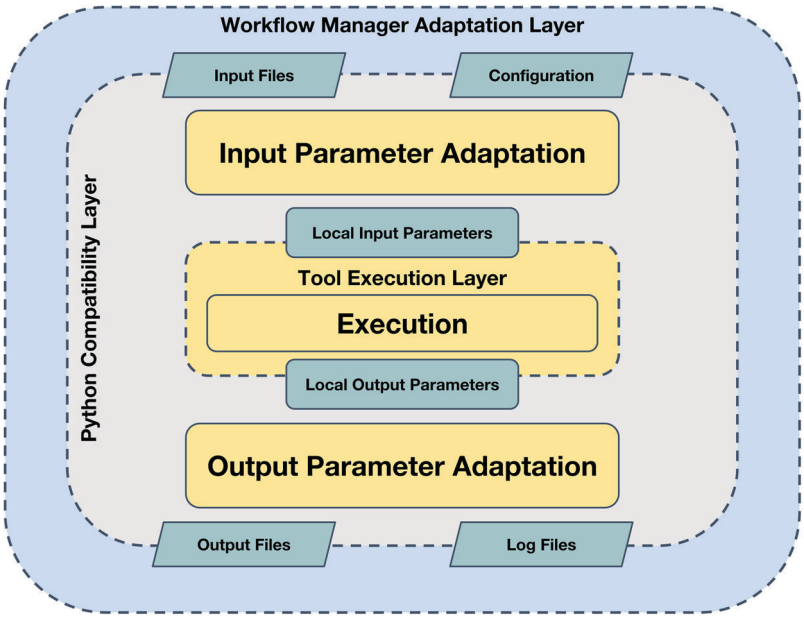


Fig. 1 BioExcel building block architecture. BioBB's structure split in three main layers: The inner layer corresponds to the original tool unaltered, the second one, the Python compatibility layer provides a standardized interface, the third one, the outer workflow manager adaptation layer translates the Python standard interface to each specific WF manager.

Docker containers are automatically generated and deposited in the quay.io repository. Offered Docker containers provide functionality for either individual packages to be integrated in more complex layouts, or complete workflows. Docker containers, in turn, are converted to Singularity containers that can be used in security demanding environments like HPC. Containers allow the non-expert user to deploy the software easily. For instance, Docker containers have been used to deploy BioBBs in a test Galaxy installation (http://dev.usegalaxy.es). BioBBs, encapsulated as Virtual Machines, are also available on BioExcel cloud portal (https://bioexcel.ebi.ac.uk), and EGI's appDB (https://appdb.egi.eu). Table 2 summarizes the recommended installation and execution options for the environments tested in the project.

BioBBs are fully open source, distributed under the Apache-2 license. Wrapped applications have their own licensing schemes, but for the library provided at present only open source software has been included.

Testing BioBBs in several environments. Setup for simulation for protein variants workflow. To test the feasibility of the software architecture, we have chosen a well-known procedure, the setup in standard conditions for molecular dynamics simulations of a protein system with sequence variants. We have used two biological systems: Lysozyme (PDB id 1AKI)[46], and Pyruvate kinase (PDB id 2VGB)[47]. Lysozyme is a small protein (129 res), which structure is available at a high resolution. The second system, Pyruvate kinase is a 200 kDa homo-tetramer, meaning a ~400,000 atom system after setup. Pyruvate kinase is a well-studied system with relevance in the understanding of allosteric regulation, but also of biomedical interest: more than 200 sequence variants related to pathogenic effects have been reported[48]. The test-cases consisted in a standard setup for NPT simulation with explicit solvent of several selected variants, followed with 5 ns long simulations, and a simple RMSd comparative analysis (see Method section). Supplementary Figs S1 and S2 show a schema of the simulation setup workflow as rendered by CWL viewer (https://view.commonwl.org) and Galaxy respectively. We have tested (1) the feasibility of running the workflow (including software installation, and workflow execution) in a variety of computational environments (Lysozyme test) and (2) its scalability on HPC systems (Pyruvate kinase test). Supplementary Table S3 shows a summary of the architectures and the executions performed. Execution times are shown just for illustration purposes and are totally dependent on the hardware used. Since most of the execution time corresponds to the simulation phases, no significant overhead in using the different execution approaches was detected. Parallelization has been carried out at different levels. PyCOMPSs has been used to deal with simulations of different protein variants, at a ratio of 1 variant per process. GROMACS parallelization schemes (OpenMP for intra-node parallelization and MPI when several nodes were involved) were used in the simulation phase. Linear scaling has been observed in all cases (note the similar wall-clock times between the two extreme executions made at BSC's MareNostrum, ranging from 2 variants, 384 cores, to 200 variants, 38,400 cores).

Conclusions

Biomolecular simulations are seldom considered as part of the field known as bioinformatics, even structural bioinformatics. Reasons for that come not only from the use of a different kind of tools and computational resources, but also from the traditional lack of applicability of simulation results to day-to-day biology. In the recent years, simulation has attained a significant level of maturity, and simulation results are now compatible with biologically

Block group	Block Id	Wrapped software	Functionality description
biobb_io	MmbPdb	API Call	Downloads a PDB file from the RCSB or MMB REST APIs
	MmbPdbVariants	API Call	Creates a text file containing a list of all the variants mapped to a RSCB PDB code from the corresponding UNIPROT entries.
	MmbPdbClusterZip	API Call	Creates a zip file containing all the PDB files in the given sequence similarity cluster percentage of the given RSCB PDB code
biobb_model	fix_side_chain	in house	Reconstructs the missing side chains and heavy atoms of the given PDB file
	mutate	in house	Creates a new PDB file performing the mutations given in a list of amino acid mutations to the input PDB file.
biobb_md	Pdb2gmx	gmx pdb2gmx	Creates a compressed (ZIP) Gromacs topology (TOP and ITP files) from a given PDB file.
	Editconf	gmx editconf	Creates a Gromacs structure file (GRO) adding the information of the solvent box to the input structure file.
	Genion	gmx genion	Creates a new compressed Gromacs topology adding ions until reaching the desired concentration to the input compressed Gromacs topology.
	Genrestr	gmx genrestr	Creates a new Gromacs compressed topology applying the indicated force restrains to the given input compressed topology.
	Grompp	gmx grompp	Creates a Gromacs portable binary run input file (TPR) applying the desired properties from the input compressed Gromacs topology.
	Mdrun	gmx mdrun	Performs molecular dynamics simulations from an input Gromacs TPR file.
	Make_ndx	gmx make_ndx	Creates a Gromacs index file (NDX) from an input selection and an input Gromacs structure file.
	Solvate	gmx solvate	Creates a new compressed Gromacs topology file adding solvent molecules to a given input compressed Gromacs topology file.
	Ndx2resttop	in house	Creates a new Gromacs compressed topology applying the force restrains to the input groups in the input index file to the given input compressed topology.
biobb_analysis	cluster	gmx cluster	Creates cluster structures from a given input trajectory.
	rms	gmx rms	Performs an RMS analysis of the given input trajectory.
	cpptraj	cpptraj	Performs multiple analysis of a given trajectory.
biobb_common	—	—	BioBB Base structure & common elements
biobb_template	—	—	Generic template to build new blocks

Table 1. List of available BioExcel Building blocks. Blocks are grouped by the type of operation and external tool.

relevant systems and time scales. Biomolecular simulations are already tackling questions that can be relevant for genomics, or transcriptomics. However, the isolation of biomolecular simulations in the context of bioinformatics has prevented the adoption by this community of normal software trends in bioinformatics, like automatic software deployment or the use of workflow managers. We have presented here the exercise of treating biomolecular simulations as normal bioinformatics operations. To this end, we have decorated standard simulation operations with a series of concepts and procedures, like an initial adherence to FAIR principles, the usage and documentation of workflows and stable interfaces, and the availability of a variety of deployment options, that are becoming routine in bioinformatics. FAIR principles for software have not yet been defined in the way as they exist for data. The exercise has led to an approach to the selection of software features (registration, methods of installations and deployment, documentation, licenses) that can be considered as an initial approach to them. The main outcome of the exercise is a complete software library (the BioBBs) that can be installed, deployed, and used as traditional bioinformatics applications, but provides a set of operations related to biomolecular simulations. BioBBs have been incorporated to the bioinformatics ecosystem: (1) The necessary new terms have been added to EDAM ontology, and tools included in the bio.tools registry. Bio.tools would provide a permanent identifier for them and the required metadata to assure their findability; (2) Interoperability has been addressed by the design of BioBBs architecture, but also through the use of recommended standards for specification (OpenAPI, CWL); and (3) Accessibility and usability have been addressed by using the set of well-known utilities, like Pipy, BioConda, BioContainers, or Galaxy, allowing the deployment and test of the library in a variety of alternative environments, from personal workstations to HPC.

BioBBs align with a variety of software that focus in similar functionality, however it opens the integration of biomolecular simulation operations into a more general bioinformatics landscape using similar, and compatible, software management procedures.

Methods

Atomistic simulations. *Lysozyme test.* Simulation of two sequence variants (Val2Tyr, and Val2Ala) of chicken Lysozyme (PDB code 1AKI)[46] were prepared as follows. Protein structure was obtained from the Protein Data Bank[49]. Amino acid side chains were modified as appropriate using the biobb_model package. Hydrogen atoms were added to the structure using standard ionization at pH 7.0. Protein was placed in a Cubic box of explicit water solvent (SPC/E water model)[50] with the appropriate size to allow 1 nm from the outermost protein atom. Periodic Boundary Conditions were applied. Cl^- and Na^+ ions were added to reach an ion concentration of 0.05 M and neutralize the system. Simulations were run using GROMACS 2018, and the Amber99sb-ILDN forcefield[51]. Temperature was maintained at 300 K and pressure to 1 atm. Setup was completed by 5,000 steps of

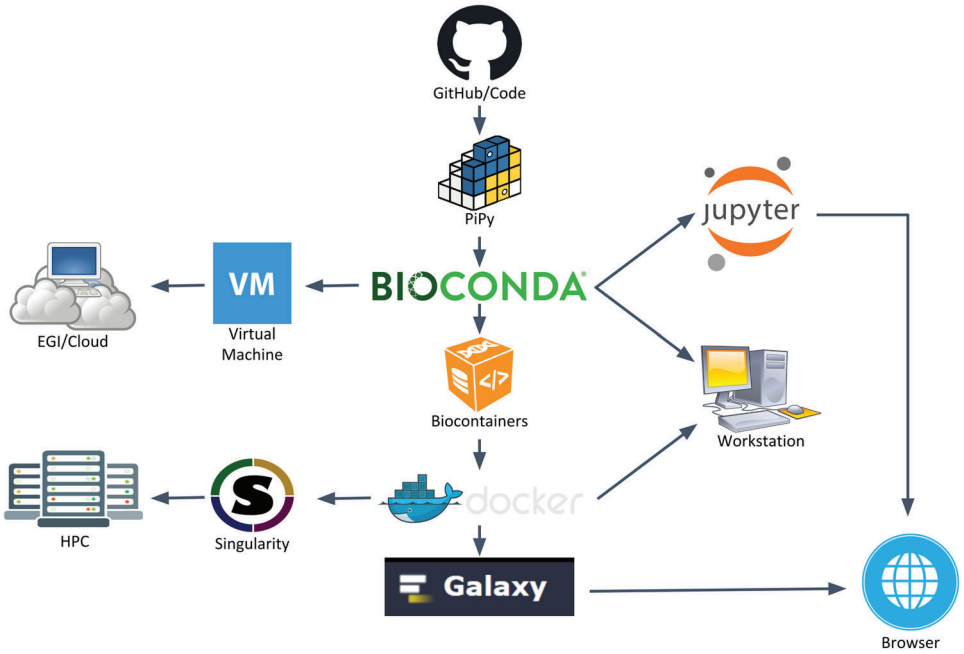


Fig. 2 Recommended distribution and deployment flow of the BioBBs. Distribution and packaging tools used to facilitate BioBB's installation and execution in a wide range of platforms: HPC, Cloud computing, user workstations and even browser interfaces.

	Installation alternatives				Workflow Execution alternatives			
Architecture	**PyPI**	**BioConda**	**VM**	**Container**	**Script**	**CWLtool**	**PyCOMPSs**	**Galaxy**
Workstation	A	R (T)	A	A	R (T)	A	A	
Cloud	A	A	R (T)	R	R (T)	A	A	
MareNostrum (HPC)	A	A (T)		R	A		R (T)	
Galaxy		A		R (T)				R (T)

Table 2. (A)vailable and (R)ecommended alternatives for Biobb installation and workflow execution. (T)est executions performed. Container generic denomination corresponds to Docker containers in the workstation, cloud and Galaxy cases and to Singularity containers in the MareNostrum HPC case.

steepest-descent energy minimization, followed by a 10 ps-long NVT equilibration, and a 10 ps-long NPT equilibration runs with a restriction of 1,000 kJ/mol.nm^2 put on heavy atoms. Production phase for the test consisted in 5 ns of unbiased NPT simulation at 2 fs time step. The LINCS algorithm[52] was used to keep covalent bonds at their equilibrium distances. Simulation setup and equilibration were done using components of the biobb_md package.

Pyruvate kinase test. 200 sequence variants for Human erythrocyte Pyruvate kinase (PDB code 2VGB)[47] were obtained from UniprotKB[53] (biobb_io package). Protein structure was obtained from the Protein Data Bank[49]. All non-protein components of the structure were removed, and protein variants were prepared by modification of the appropriate amino acid side chains using biobb_model package. Hydrogen atom were added considering standard ionization states at pH 7.0. Simulation was done in a truncated octahedron box placed at a distance of 1.5 nm from the outermost atom of the protein, using TIP3P water molecules[54], and using Periodic Boundary Conditions. Ions Cl^- and Na^+ were added to reach an ion concentration of 0.05 M and neutralize the system. The Particle mesh Ewald method[55] was used to calculate electrostatic and Van der Waals interactions, with 0.12 nm of FF grid spacing and a cut-off distance of 1 nm for both Coulomb and Lennard-Jones interactions. The LINCS algorithm[52] was used to keep covalent bonds at their equilibrium distances. Simulations were run using GROMACS 2018, and the Amber99sb-ILDN forcefield[51]. Temperature was maintained constant at 300 K (except in gradual heating), in two separate baths for the protein and non-protein groups, with the V-rescale thermostat[56] and a coupling constant of 0.1 ps. Pressure was isotropically maintained at 1 bar in NPT ensembles through Parrinello-Rahman coupling[57] with a constant of 1 ps, and applying a scaling of the center of mass of the reference coordinates with the scaling matrix. Given the size and complexity of the system, the Pyruvate kinase equilibration was performed with a more extended procedure: Setup was completed with two 5,000 steps energy minimizations, the first with a restrained potential of 500 kJ.mol^{-1}.nm^{-2} on all heavy atoms except those in the side chain of the mutated residue, and the second with all heavy atoms restrained. Systems were then equilibrated with the following steps: (1) 100 ps of gradual heating from 0 to 300 K with 1,000 kJ.mol^{-1}.nm^{-2} of restrained

potential in heavy atoms except for mutated side chains, (2) four 20 ps steps of equilibration with descending restrain force constants in the same atoms (from 1,000 to 300 kJ.mol^{-1}.nm^{-2}), (3) two 10 ps steps of NPT equilibration with restraints in all backbone atoms (200 and 100 kJ.mol^{-1}.nm^{-2} respectively) and (4) a 100 ps NPT equilibration without restraints. After equilibration, we ran 5 ns of unbiased NPT simulation. Simulation setup and equilibration were done using components of the biobb_md package.

Computational systems used. Systems used on Lysozyme test were: *Workstation:* ThinkStation E30 (LENOVO). Operating system: Linux Ubuntu 18.04. 8 CPU Intel(R) Xeon(R) CPU E31230 @ 3.20 GHz (1 socket, 4 cores/socket, 2 threads/core). 16 GB of RAM. *Virtual Machine:* 12 CPU QEMU Virtual CPU version 2.5+. 24 GB of RAM. *Galaxy:* 2 CPU QEMU Virtual CPU version 2.5+. Pyruvate kinase test was performed on BSC's MareNostrum supercomputer using from 2 to 800 nodes of 2x Intel Xeon Platinum 8160 24C at 2.1 GHz, 12 × 8 GB of RAM. Largest test used 4 nodes per simulation with a total of 38,400 cores.

Data Availability

The test data of each building block is available in the correspondent Github repository, see Online-only Table 1. The full data collection on the testing phase for BioBBs is available at ref.[58].

Code Availability

BioBB's source code is available at GitHub. URLs for the code and documentation repositories and the alternative installation and execution options are summarized in Online-only Table 1.

References

1. Hospital, A. & Gelpi, J. L. High-throughput molecular dynamics simulations: toward a dynamic view of macromolecular structure. *Wiley Interdisciplinary Reviews-Computational Molecular Science* **3**, 364–377 (2013).
2. Orozco, M. A theoretical view of protein dynamics. *Chem. Soc. Rev.* **43**, 5051–5066 (2014).
3. Ayton, G. S., Noid, W. G. & Voth, G. A. Multiscale modeling of biomolecular systems: in serial and in parallel. *Current Opinion in Structural Biology* **17**, 192–198 (2007).
4. Emperador, A., Carrillo, O., Rueda, M. & Orozco, M. Exploring the suitability of coarse-grained techniques for the representation of protein dynamics. *Biophysical Journal* **95**, 2127–2138 (2008).
5. Dans, P. D., Walther, J., Gómez, H. & Orozco, M. Multiscale simulation of DNA. *Curr Opin Struct Biol* **37**, 29–45 (2016).
6. Dans, P. D. *et al*. Modeling, Simulations, and Bioinformatics at the Service of RNA Structure. *Chem* **5**, 51–73 (2019).
7. Stevens, R. D., Robinson, A. J. & Goble, C. A. myGrid: personalised bioinformatics on the information grid. *Bioinformatics* **19**(Suppl 1), i302–4 (2003).
8. Wilkinson, M. D. *et al*. Interoperability with Moby 1.0–it's better than sharing your toothbrush! *Brief Bioinform* **9**, 220–231 (2008).
9. Amstutz, P. *et al*. Common Workflow Language, v1.0. *figshare*. https://doi.org/10.6084/M9.FIGSHARE.3115156.V2 (2016).
10. Wolstencroft, K. *et al*. The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud. *Nucleic acids research* **41**, W557–W561 (2013).
11. Kaushik, G. *et al*. RABIX: An Open-Source Workflow Executor Supporting Recomputability and Interoperability of Workflow Descriptions. *Pacific Symposium on Biocomputing* **22**, 154–165 (2016).
12. Beisken, S. *et al*. KNIME-CDK: Workflow-driven cheminformatics. *BMC bioinformatics* **14**, 257–257 (2013).
13. Goble, C. A. *et al*. myExperiment: a repository and social network for the sharing of bioinformatics workflows. *Nucleic acids research* **38**, W677–W682 (2010).
14. Ison, J. *et al*. Tools and data services registry: a community effort to document bioinformatics resources. *Nucleic acids research* **44**, D38–D47 (2016).
15. McQuilton, P. *et al*. BioSharing: curated and crowd-sourced metadata standards, databases and data policies in the life sciences. *Database: the journal of biological databases and curation* **2016**, baw075 (2016).
16. Afgan, E. *et al*. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic acids research* **46**, W537–W544 (2018).
17. Di Tommaso, P. *et al*. Nextflow enables reproducible computational workflows. *Nature Biotechnology* **35**, 316 (2017).
18. Tejedor, E. *et al*. PyCOMPSs: Parallel computational workflows in Python. *The International Journal of High Performance Computing Applications* **31**, 66–82 (2015).
19. Köster, J. & Rahmann, S. Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics* **28**, 2520–2522 (2012).
20. Wilkinson, M. D. *et al*. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data* **3**, 16008 (2016).
21. Rueda, M. *et al*. A consensus view of protein dynamics. *Proc Natl Acad Sci USA* **104**, 796–801 (2007).
22. Meyer, T. *et al*. MoDEL (Molecular Dynamics Extended Library): a database of atomistic molecular dynamics trajectories. *Structure* **18**, 1399–1409 (2010).
23. Hospital, A. *et al*. BIGNASim: a NoSQL database structure and analysis portal for nucleic acids simulation data. *Nucleic Acids Res* **44**, D272–278 (2016).
24. Elofsson, A. *et al*. Ten simple rules on how to create open access and reproducible molecular simulations of biological systems. *PLoS computational biology* **15**, e1006649–e1006649 (2019).
25. Kota, P. GUIMACS - a Java based front end for GROMACS. *In Silico Biol* **7**, 95–99 (2007).
26. Miller, B. T. *et al*. CHARMMing: a new, flexible web portal for CHARMM. *Journal of chemical information and modeling* **48**, 1920–1929 (2008).
27. Jo, S. *et al*. CHARMM-GUI 10 years for biomolecular modeling and simulation. *Journal of computational chemistry* **38**, 1114–1124 (2017).
28. Sellis, D., Vlachakis, D. & Vlassi, M. Gromita: a fully integrated graphical user interface to gromacs 4. *Bioinformatics and biology insights* **3**, 99–102 (2009).
29. Roopra, S., Knapp, B., Omasits, U. & Schreiner, W. jSimMacs for GROMACS: A Java Application for Advanced Molecular Dynamics Simulations with Remote Access Capability. *J. Chem. Inf. Model.* **49**, 2412–2417 (2009).
30. Ribeiro, J. V. *et al*. QwikMD - Integrative Molecular Dynamics Toolkit for Novices and Experts. *Scientific reports* **6**, 26536–26536 (2016).
31. Hospital, A. *et al*. MDWeb and MDMoby: an integrated web-based platform for molecular dynamics simulations. *Bioinformatics* **28**, 1278–1279 (2012).
32. Pronk, S. *et al*. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics (Oxford, England)* **29**, 845–854 (2013).
33. Nelson, M. T. *et al*. NAMD: a Parallel, Object-Oriented Molecular Dynamics Program. *The International Journal of Supercomputer Applications and High Performance Computing* **10**, 251–268 (1996).

34. Pearlman, D. A. *et al*. AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules. *Computer Physics Communications* **91**, 1–41 (1995).
35. Hospital, A. *et al*. NAFlex: a web server for the study of nucleic acid flexibility. *Nucleic Acids Res* **41**, W47–55 (2013).
36. Pronk, S. *et al*. Molecular Simulation Workflows as Parallel Algorithms: The Execution Engine of Copernicus, a Distributed High-Performance Computing Platform. *J. Chem. Theory Comput.* **11**, 2600–2608 (2015).
37. Thibault, J. C., Facelli, J. C. & Cheatham, T. E. 3rd. iBIOMES: managing and sharing biomolecular simulation data in a distributed environment. *J Chem Inf Model* **53**, 726–736 (2013).
38. Thibault, J. C., Cheatham, T. E. 3rd. & Facelli, J. C. iBIOMES Lite: summarizing biomolecular simulation data in limited settings. *J Chem Inf Model* **54**, 1810–1819 (2014).
39. Sun, R., Li, Z. & Bishop, T. C. TMB-iBIOMES: An iBIOMES-Lite Database of Nucleosome Trajectories and Meta-Analysis. Preprint at, https://doi.org/10.26434/chemrxiv.7793939.v1 (2019).
40. van der Kamp, M. W. *et al*. Dynameomics: A Comprehensive Database of Protein Dynamics. *Structure* **18**, 423–435 (2010).
41. Grüning, B. *et al*. Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nature Methods* **15**, 475–476 (2018).
42. da Veiga Leprevost, F. *et al*. BioContainers: an open-source and community-driven framework for software standardization. *Bioinformatics (Oxford, England)* **33**, 2580–2582 (2017).
43. Ison, J. *et al*. EDAM: an ontology of bioinformatics operations, types of data and identifiers, topics and formats. *Bioinformatics (Oxford, England)* **29**, 1325–1332 (2013).
44. Hillion, K.-H. *et al*. Using bio.tools to generate and annotate workbench tool descriptions. *F1000Research* **6**, ELIXIR-2074 (2017).
45. Doppelt-Azeroual, O. *et al*. ReGaTE: Registration of Galaxy Tools in Elixir. *GigaScience* **6**, 1–4 (2017).
46. Carter, D., He, J., Rubble, J. R. & Wright, B. The structure of the orthorhombic form of hen egg-white lysosome at 1.5 angstroms resolution. *Protein Data Bank*, *Rutgers University*, https://identifiers.org/pdb:1AKI (1997).
47. Valentini, G. *et al*. Human erythrocyte pyruvate kinase. *Protein Data Bank*, *Rutgers University*, https://identifiers.org/pdb:2VGB (2007).
48. Canu, G., De Bonis, M., Minucci, A. & Capoluongo, E. Red blood cell PK deficiency: An update of PK-LR gene mutation database. *Blood Cells, Molecules, and Diseases* **57**, 100–109 (2016).
49. wwPDB consortium. Protein Data Bank: the single global archive for 3D macromolecular structure data. *Nucleic Acids Research* **47**, D520–D528 (2018).
50. Berendsen, H. J. C., Grigera, J. R. & Straatsma, T. P. The missing term in effective pair potentials. *J. Phys. Chem.* **91**, 6269–6271 (1987).
51. Hornak, V. *et al*. Comparison of multiple Amber force fields and development of improved protein backbone parameters. *Proteins* **65**, 712–725 (2006).
52. Hess, B., Bekker, H., Berendsen, H. J. C. & Fraaije, J. G. E. M. LINCS: A linear constraint solver for molecular simulations. *Journal of Computational Chemistry* **18**, 1463–1472 (1997).
53. The UniProt Consortium. UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Research* **47**, D506–D515 (2018).
54. Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W. & Klein, M. L. Comparison of simple potential functions for simulating liquid water. *The Journal of Chemical Physics* **79**, 926–935 (1983).
55. Darden, T., York, D. & Pedersen, L. Particle mesh Ewald: An N·log(N) method for Ewald sums in large systems. *J. Chem. Phys.* **98**, 10089–10092 (1993).
56. Bussi, G., Donadio, D. & Parrinello, M. Canonical sampling through velocity rescaling. *J. Chem. Phys.* **126**, 014101 (2007).
57. Parrinello, M. & Rahman, A. Polymorphic transitions in single crystals: A new molecular dynamics method. *Journal of Applied Physics* **52**, 7182–7190 (1981).
58. Andrio, P. *et al*. Bioexcel building blocks test cases. *Zenodo*. https://doi.org/10.5281/zenodo.2581362 (2019).

Acknowledgements

We thank The Barcelona Supercomputing Center for providing all the HPC resources to launch the test executions presented in this manuscript. The project was supported by the following grants: BioExcel Center of Excellence (Horizon 2020 Framework Programme. Grants 675728, and 823830). Elixir-Excelerate (Horizon 2020 Framework Programme. Grant 676559). Spanish National Institute for Bioinformatics (Institute of Health Carlos III. Grants PT13/0001/0019, PT13/0001/0028, PT17/0009/0001). Spanish Government, Severo Ochoa Grant SEV2015-0493, TIN2015-65316-P, Generalitat de Catalunya. 2014-SGR-1051, 2017-SGR-1110.

Author Contributions

J.L.G. designed the strategy and wrote the manuscript with contributions of all authors. P.A. and A.H. were responsible of software development, J.C., D.L. and R.M.B. were responsible of PyCOMPSs adaptation and implementation, L.J. developed and validated HPC workflows, M.P. and L.C. developed and tested the Galaxy implementation. S.S.-R. helped to implement CWL and Jupyter Notebook. C.G., R.M.B., M.O. check the manuscript and provided useful additions.

Additional Information

Supplementary Information is available for this paper at https://doi.org/10.1038/s41597-019-0177-4.

Competing Interests: The authors declare no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

12) Georgieva, M. V, Yahya, G., Codó, L., Ortiz, R., Teixidó, L., Claros, J., … Aldea, M. (2015). Inntags: small self-structured epitopes for innocuous protein tagging. Nature Methods, 12(10), 955–958. https://doi.org/10.1038/nmeth.3556

Inntags: small self-structured epitopes for innocuous protein tagging

Maya V Georgieva[1,9], Galal Yahya[1,2,9], Laia Codó[3,4], Raúl Ortiz[1], Laura Teixidó[5], José Claros[6], Ricardo Jara[6], Mònica Jara[5], Antoni Iborra[5], Josep Lluís Gelpí[3,4,7], Carme Gallego[1,10], Modesto Orozco[4,7,8,10] & Martí Aldea[1,10]

Protein tagging is widely used in approaches ranging from affinity purification to fluorescence-based detection in live cells. However, an intrinsic limitation of tagging is that the native function of the protein may be compromised or even abolished by the presence of the tag. Here we describe and characterize a set of small, innocuous protein tags (inntags) that we anticipate will find application in a variety of biological techniques.

Proteins and peptides of various sizes and shapes have been used since the early 1980s[1] to tag proteins for many different purposes, ranging from affinity purification[2] to microscopic detection in whole organisms[3]. Whereas large, well-folded tags can serve to solubilize proteins, small, unfolded peptide tags have become invaluable tools for protein purification as well as protein-protein interaction studies. Yet both types of tagging strategies run the risk that the native function of the protein may be abolished or compromised through interactions with the tag. Large self-structured tags are more likely to cause steric interference than small tags[4]. However, as they are intrinsically disordered, short peptide tags can adopt a wide variety of conformations[5], which may facilitate interactions with the target protein and have unpredictable and adverse effects at different levels (**Fig. 1a**). First, spurious tag-driven interactions at intermediate steps of folding may prevent the target protein from reaching its final, mature conformation and favor its accumulation in large aggregates[6]. Second, both the misfolded and correctly folded fusion protein may be more susceptible to degradation. In this regard, one of the most commonly used small tags, HA, contains a cryptic caspase-cleavage site[7]. Finally, if unwanted interactions occur at essential target protein domains, the tag may modify the protein's structure[8], proper function[9], interactions[10] or normal localization in the cell[11]. Not surprisingly, a recent analysis of more than 400 proteins found that up to 20% showed different localization patterns in a comparison between fluorescence-tagged proteins and those detected by immunofluorescence methods[3].

Here we describe and characterize a set of small protein domains, which we term inntags, developed with the goal of minimizing the risk of functional and structural interference in protein tagging. To identify candidate inntags, we performed a search for protein domains that have a known three-dimensional (3D) structure and a set of specific properties (Online Methods and **Supplementary Table 1**). These structures were further scrutinized with the aid of additional criteria (Online Methods) to obtain 12 inntag candidates, which correspond to nine proteins from plants and insects, two fimbria-associated bacterial proteins and one viral envelope protein (**Supplementary Fig. 1a**).

Misfolded proteins commonly accumulate as aggregates that can be easily observed by microscope as varying types of inclusion bodies[12]. Although most inntag fusions attained expression levels similar to that of the GFP control construct, many of them displayed different degrees of aggregation (**Supplementary Fig. 1b–d**), and only inntags IT5, IT6 and IT10 produced a diffuse pattern in all cells. Thus, we selected these three inntags (**Supplementary Table 2**) to further test their suitability as protein tags.

Despite their small size, the structure of the IT5, IT6 and IT10 inntags is based on a packed β-sheet conformation that, in the case of the two smaller domains (IT6 and IT10), is probably stabilized by several disulfide bridges (**Fig. 1b**). Integrity and stability of GFP and human cyclin E were unaffected by these inntags, and their presence did not modify the intracellular mobility of GFP as deduced by fluorescence loss in photobleaching (FLIP) analysis, thus confirming the absence of stable associations of the selected inntags with cytoskeletal or other large cellular structures (**Supplementary Fig. 2**). Finally, as both fluorescence and protein levels were not affected by the presence of any of the inntags (**Supplementary Fig. 2**), we conclude that the three selected inntags do not grossly affect the folding and maturation rates of the fused GFP moiety.

We hypothesize that, given their low disorder propensity, inntags IT5, IT6 and IT10 should be less likely to produce unwanted effects than unstructured peptides such as HA, Flag and MYC, three commonly used protein tags that have much higher disorder propensity values (**Fig. 1c**). Intrinsic protein disorder has been associated with deleterious gene-dosage effects[13], and gene overexpression in budding yeast has proved to be an invaluable

[1]Molecular Biology Institute of Barcelona (IBMB), Consejo Superior de Investigaciones Científicas (CSIC), Barcelona, Spain. [2]Department of Microbiology and Immunology, School of Pharmacy, Zagazig University, Zagazig, Egypt. [3]Barcelona Supercomputing Center (BSC), Barcelona, Spain. [4]Joint BSC-CRG-IRB Programme in Computational Biology, Barcelona, Spain. [5]AbBcn Ltd, Bellaterra, Spain. [6]Immunostep Ltd, Salamanca, Spain. [7]Departament de Bioquimica, Facultat de Biologia, Universitat de Barcelona, Barcelona, Spain. [8]Institute for Research in Biomedicine (IRB Barcelona), Barcelona, Spain. [9]These authors contributed equally to this work. [10]These authors jointly directed this work. Correspondence should be addressed to C.G. (carme.gallego@ibmb.csic.es), M.O. (modesto.orozco@irbbarcelona.org) or M.A. (marti.aldea@ibmb.csic.es).

RECEIVED 14 NOVEMBER 2014; ACCEPTED 6 JULY 2015; PUBLISHED ONLINE 31 AUGUST 2015; DOI:10.1038/NMETH.3556

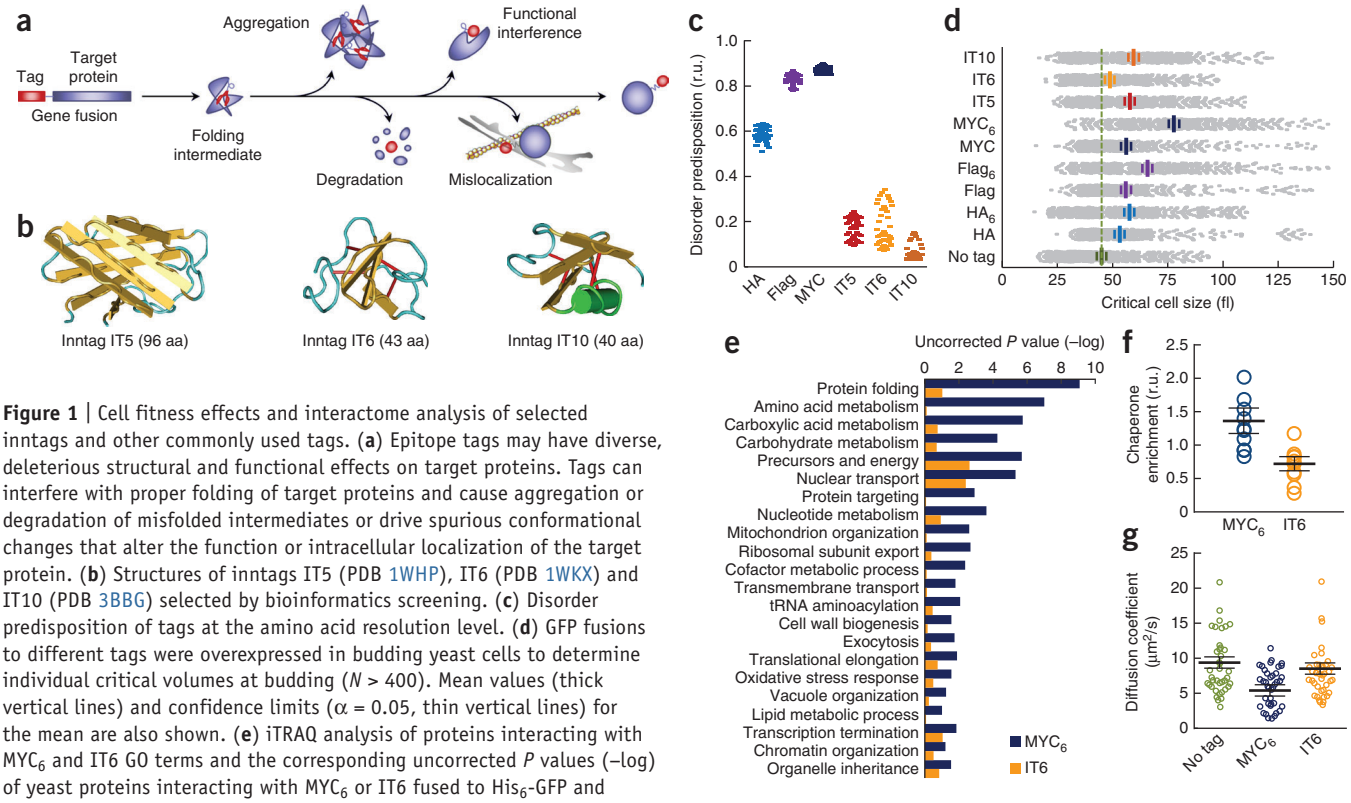


Figure 1 | Cell fitness effects and interactome analysis of selected inntags and other commonly used tags. (**a**) Epitope tags may have diverse, deleterious structural and functional effects on target proteins. Tags can interfere with proper folding of target proteins and cause aggregation or degradation of misfolded intermediates or drive spurious conformational changes that alter the function or intracellular localization of the target protein. (**b**) Structures of inntags IT5 (PDB 1WHP), IT6 (PDB 1WKX) and IT10 (PDB 3BBG) selected by bioinformatics screening. (**c**) Disorder predisposition of tags at the amino acid resolution level. (**d**) GFP fusions to different tags were overexpressed in budding yeast cells to determine individual critical volumes at budding ($N > 400$). Mean values (thick vertical lines) and confidence limits ($\alpha = 0.05$, thin vertical lines) for the mean are also shown. (**e**) iTRAQ analysis of proteins interacting with MYC_6 and IT6 GO terms and the corresponding uncorrected P values (–log) of yeast proteins interacting with MYC_6 or IT6 fused to His_6-GFP and identified by iTRAQ. (**f**) iTRAQ ratios obtained for MYC_6 and IT6 interactors belonging to the family of molecular chaperones (Hsp60, Sse1, Ssa1, Ssa2, Ssb2, Ssc1, Sti1, Aha1, Ydj1 and Sis1). Mean enrichment values (thick horizontal lines) relative to untagged GFP and confidence limits ($\alpha = 0.05$, thin horizontal lines) for the mean are also shown. (**g**) Yeast cells overexpressing GFP fused to MYC_6 and IT6 were analyzed by time-lapse confocal microscopy under FLIP conditions. Protein mobility mean values ($N = 40$ cells) and confidence limits for the mean ($\alpha = 0.05$) are shown. r.u., relative units. Comparable quantitative differences were obtained from a replicate experiment (**d**,**g**); statistical data were obtained from three independent pulldown experiments (**e**,**f**).

tool to establish functional relationships and pathways[14]. Thus, we analyzed the fitness effects of these tags when overexpressed as fusions with GFP in yeast cells. Because HA, Flag and MYC epitopes are routinely used as tandem repeats to improve the efficiency of immunodetection, we also analyzed hexameric versions of these tags, which have similar polypeptide lengths to inntags. Whereas monomeric versions of HA, Flag and MYC did not produce noticeable effects, the MYC_6 tag caused a twofold reduction in growth rate and a tenfold reduction in chronological lifespan (**Supplementary Fig. 3a**). Although the effects were not as pronounced, $Flag_6$, HA_6, IT5 and IT10 also decreased these two cell-fitness parameters, and only IT6 did not cause any effect when compared to untagged GFP in these assays.

Cell size is one of the main attributes that determine functional efficiency in multicellular organisms and fitness in unicellular organisms[15]. We found that the MYC_6 tag clearly dysregulated this fitness parameter, causing a nearly twofold increase in critical size when overexpressed as a GFP fusion (**Fig. 1d**). HA_6, $Flag_6$, IT5 and IT10 also caused an increase in the critical size to different extents, but the difference due to IT6 overexpression was not significant ($P < 0.05$). Although the monomeric HA, Flag and MYC tags produced only minor effects, these tags produced a clear increase in the intrinsic variability of critical size (**Supplementary Fig. 3b**). Notably, IT6 did not alter this physiological parameter.

To further characterize the differential deleterious effects of tags on cell fitness we chose MYC_6 and IT6 to perform a proteome-wide analysis of interactors from yeast cells. A total of 183 proteins were identified at a 95% confidence level with complete iTRAQ data in all pulldown samples, and an average enrichment ratio was obtained to compare the relative levels of each interactor in MYC_6 or IT6 pulldowns relative to those in His_6-GFP (**Supplementary Table 3**). Once ribosomal proteins were excluded, the functional category among interactors most frequently enriched in the MYC_6 pulldown was related to protein folding (**Fig. 1e**), with a simulation-corrected P value of 0.00031. This set of proteins was not present among the enriched interactors of IT6, and no correlation was observed when comparing the cellular processes with specific interactors of MYC_6 and IT6. Notably, proteins belonging to the chaperone family were clearly enriched ($P = 7.3 \times 10^{-4}$) in MYC_6 compared to IT6 pulldowns (**Fig. 1f**), suggesting that overexpression of disordered proteins could compromise the efficiency of the molecular chaperone network by direct nonproductive interactions. Supporting this notion, the presence of MYC_6 clearly reduced ($P = 8.7 \times 10^{-6}$) the intracellular diffusion coefficient of GFP in yeast cells (**Fig. 1g**).

To enable the use of selected inntags in immunofluorescence and immunoprecipitation applications, we expressed and purified monomeric IT5 or dimeric IT6, and their corresponding fusions to GFP, from *Escherichia coli* cells (**Supplementary Fig. 4**) for mice immunization, and obtained highly specific monoclonal antibodies (**Fig. 2a**,**b**). Immunofluorescence analysis showed that IT6 was slightly superior to HA_6 and IT5, and all other tags

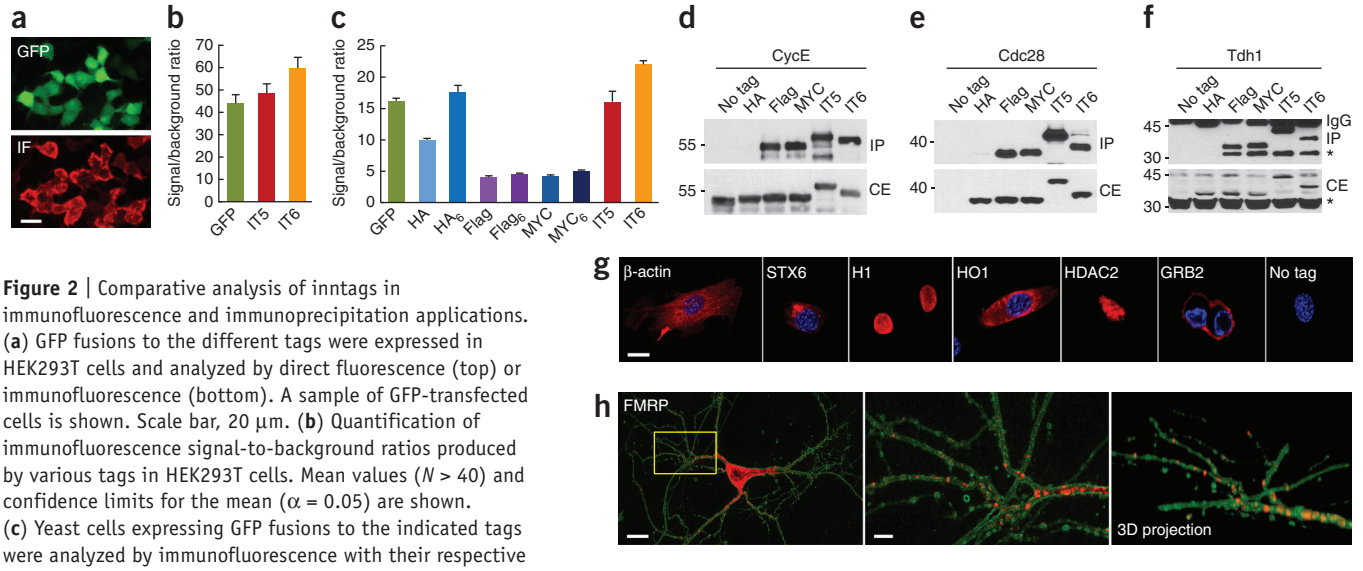


Figure 2 | Comparative analysis of inntags in immunofluorescence and immunoprecipitation applications. (**a**) GFP fusions to the different tags were expressed in HEK293T cells and analyzed by direct fluorescence (top) or immunofluorescence (bottom). A sample of GFP-transfected cells is shown. Scale bar, 20 µm. (**b**) Quantification of immunofluorescence signal-to-background ratios produced by various tags in HEK293T cells. Mean values ($N > 40$) and confidence limits for the mean ($\alpha = 0.05$) are shown. (**c**) Yeast cells expressing GFP fusions to the indicated tags were analyzed by immunofluorescence with their respective antibodies to obtain signal-to-background ratios. Mean values ($N = 100$) and confidence limits for the mean ($\alpha = 0.05$) are shown. (**d**,**e**) Total extracts (CEs) and immunoprecipitates (IPs) of HEK293T cells expressing cyclin E (**d**) and budding yeast cells expressing Cdc28 (**e**) fused to various tags, detected with antibodies to cyclin E (CycE) (**d**) and Cdc28 (**e**). (**f**) CEs and IPs of budding yeast cells expressing Tdh1 fused to the indicated tags, analyzed by immunoblotting with anti-Tdh1. Untagged Tdh1 is indicated by an asterisk. (**g**) Inntag IT6 fused to β–actin (cytoskeleton), STX6 (Golgi), HO1 (endoplasmic reticulum), histone H1 (nucleus), HDAC2 (nucleus) or GRB2 (plasma membrane) was expressed in NIH3T3 cells and analyzed by immunofluorescence (red) with an antibody to IT6. GRB2 images correspond to mitotic cells, where the cytoplasmic membrane is best seen. Hoechst-stained nuclei are also shown (blue). Scale bar, 10 µm. (**h**) Left, an inntag IT6 fusion to FMRP, which localizes to synapses, expressed in mouse hippocampal neurons and analyzed by immunofluorescence (red). A first-derivative image of cotransfected GFP to enhance somatic and neuritic limits is shown as reference. Scale bar, 25 µm. Middle and right, 5× amplification (middle) and 3D projection of the region outlined at left; scale bar, 5 µm. Comparable quantitative differences were obtained from a replicate experiment (**b**,**c**).

were very inefficient and produced low signal-to-background ratios (**Fig. 2c**). Inntags IT5 and IT6 were also very efficient for immunoprecipitation of human and budding yeast proteins (**Fig. 2d–f**) or fusions to GFP (**Supplementary Fig. 5a–c**). Inntags IT5 and IT6 displayed similar immunoprecipitation efficiencies to that of monomeric Flag and MYC tags, while monomeric HA was very ineffective.

Inntag fusions to paradigmatic proteins that localize to different cellular compartments in mouse fibroblasts (**Fig. 2g** and **Supplementary Fig. 6**) and hippocampal neurons (**Fig. 2h**) displayed the expected intracellular localization and, more importantly, we did not observe aggregates or a noticeable mislocalized fraction that would suggest anomalous effects or interactions.

Cyclin-dependent kinases (CDKs) exert multiple interactions (**Fig. 3a**) to perform a series of exquisitely regulated processes that drive the eukaryotic cell cycle[16], and modification of these interactions has a profound impact on their precise execution. None of the tags caused observable effects on levels or integrity of the CDK Cdc28 (**Supplementary Fig. 7a**). However, the Cdc28-$Flag_6$ fusion caused a dramatic change in both cell size and shape (**Fig. 3b**), indicating important alterations in the morphogenetic pathways involving Cdc28 in both G1 and G2 phases. In contrast, although the IT5 inntag has a similar length to that of the $Flag_6$ tag, it did not affect cell morphology and only produced a very mild effect in the mean critical size at the G1/S transition compared to $Flag_6$ (**Fig. 3c**). The HA_3 tag did not cause gross morphological defects, but it did induce a strong increase in the critical size at the G1/S transition (**Fig. 3c**). A comparable effect in the average critical size was produced by a fusion of Cdc28 to a much larger GFP tag (**Supplementary Fig. 8a**). By contrast, inntag IT6 did not cause significant ($P < 0.05$) changes in the critical size of budding yeast cells.

Molecular chaperones use energy derived from ATP hydrolysis to transmit conformational changes to client proteins, thus facilitating their proper folding and promoting assembly or disassembly of high-order protein complexes[17]. Ydj1 is the most abundant J-domain chaperone in budding yeast and, besides acting in proteome homeostasis, has a key role in releasing the G1 cyclin Cln3 from the endoplasmic reticulum to trigger cell-cycle entry[18]. As Ydj1 contains a dimerization domain at the C terminus (**Fig. 3d**) that is essential for its function, we used C-terminal fusions at the endogenous *YDJ1* locus (**Supplementary Fig. 7b**) to perform a comparative analysis of tags on Ydj1 function. Similarly to Cdc28, fusion of Ydj1 to the $Flag_6$ tag produced the most severe effects, which could be observed even under normal conditions of growth at 30 °C (**Fig. 3e**). Both $Flag_6$ and HA_3 tags produced a clear reduction in growth at high temperature, where Ydj1 function becomes particularly essential. By contrast, fusion of Ydj1 to inntags IT5 and IT6 produced only very moderate effects on growth at 37 °C (**Fig. 3e**). Ydj1 is also important for coordinating growth and cell cycle entry to set the critical size[18,19] and, consistently with their effects on growth rate, both $Flag_6$ and HA_3 tags caused a marked increase in cell size (**Fig. 3f**), similar to that produced by a large GFP tag (**Supplementary Fig. 8b**). In contrast, neither IT5 nor IT6 produced an obvious effect on the critical size.

In summary, monomeric HA, Flag and MYC tags are inefficient compared to inntags in critical applications such as immunoprecipitation and immunofluorescence, and multimeric versions

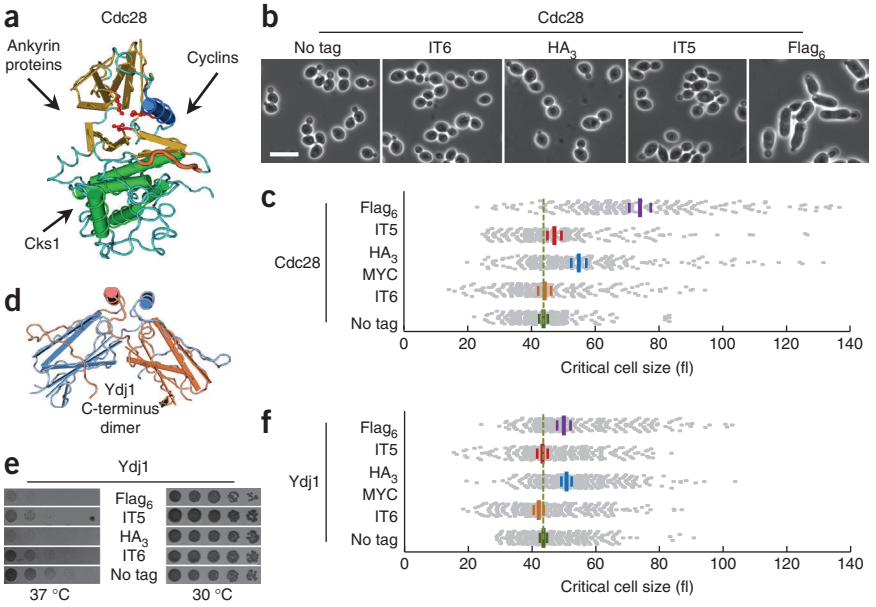


Figure 3 | Functional innocuity tests of inntags on proteins with strong interaction requirements. (**a**) Structural representation of Cdc28 based on the PDB structure of Cdk2 (PDB 1W98) showing major interaction interfaces of CDKs with cyclins, Cks1-type subunits and ankyrin-containing proteins. The cyclin-interacting PSTAIRE helix (blue), the T-loop (orange) and key catalytic residues (red) are also indicated. (**b**) Bright-field images of budding yeast cells expressing endogenous levels of Cdc28 fused to IT5, IT6, HA_3 or $Flag_6$. A control strain (no tag) is shown as reference. Scale bar, 10 µm. (**c**) Individual critical volumes at budding of cells as in **b** are plotted ($N > 200$). Mean values (thick vertical lines) and confidence limits ($\alpha = 0.05$, thin vertical lines) for the mean are also shown. (**d**) PDB structure of the C-terminal Ydj1 dimer (PDB 1XA0). (**e**) Serial dilutions of budding yeast cells expressing endogenous levels of Ydj1 fused to IT5, IT6, HA_3 or $Flag_6$ were plated and incubated for growth at the indicated temperatures. A control strain (no tag) is shown as reference. (**f**) Individual critical volumes at budding of cells as in **e** are plotted ($N > 250$). Mean values (thick vertical lines) and confidence limits ($\alpha = 0.05$, thin vertical lines) for the mean are also shown. Comparable quantitative differences were obtained from a replicate experiment (**c**,**f**).

of these tags cause strong deleterious effects in key cell-fitness parameters and have profound functional interference effects on a CDK and a chaperone, proteins with strong interaction requirements. Given their structural and functional innocuity and their suitability for *in situ* analysis, we expect that inntags will prove to be valuable tools for studying physical and functional interactions of proteins, particularly when single-molecule[20] or high-throughput approaches[21] are intended.

METHODS

Methods and any associated references are available in the online version of the paper.

Accession codes. Addgene: Inntag sequences in plasmid vector have been deposited under accession numbers 66869, 66870 and 66871. Leibniz Institute DSMZ: hybridoma data have been deposited under accession numbers ACC3234, ACC3242 and ACC3236.

Note: Any Supplementary Information and Source Data files are available in the online version of the paper.

ACKNOWLEDGMENTS

We thank A. Cornadó, E. Rebollo and S. Dillon (Harvard University) for technical assistance, M. Azor (AntibodyBcn) for administrative support and C. Mann (Centre de Génétique Moléculaire) for the anti-Cdc28 antibody. This work was funded by the Ministry of Economy and Competitiveness of Spain (INNPACTO IPT-010000-2010-19), Consolider-Ingenio 2010 (CSD2007-15), the Instituto Nacional de Bioinformática (INB) and the European Union (FEDER) and received financial support from Antibody BCN and Immunostep. M.O. acknowledges support from Institució Catalana de Recerca i Estudis Avançats.

AUTHOR CONTRIBUTIONS

M.V.G., G.Y. and R.O. performed the experiments. L.C. carried out the bioinformatics screen. L.T. and J.C. obtained the monoclonal antibodies. A.I., M.J. and R.J. directed monoclonal antibody selection and preparation, and J.L.G. and M.O. directed the bioinformatics screen. C.G. and M.A. directed the wet lab experiments. C.G., M.O. and M.A. conceived and designed the experiments, and M.A. wrote the manuscript.

COMPETING FINANCIAL INTERESTS

The authors declare competing financial interests: details are available in the online version of the paper.

1. Shine, J., Fettes, I., Lan, N.C.Y., Roberts, J.L. & Baxter, J.D. *Nature* **285**, 456–463 (1980).
2. Waugh, D.S. *Trends Biotechnol.* **23**, 316–320 (2005).
3. Stadler, C. *et al. Nat. Methods* **10**, 315–323 (2013).
4. Hoffmann, C. *et al. Nat. Methods* **2**, 171–176 (2005).
5. Tompa, P. *Trends Biochem. Sci.* **37**, 509–516 (2012).
6. Woestenenk, E.A., Hammarstrom, M., van den Berg, S., Hard, T. & Berglund, H. *J. Struct. Funct. Genomics* **5**, 217–229 (2004).
7. Schembri, L. *et al. Nat. Methods* **4**, 107–108 (2007).
8. Chant, A., Kraemer-Pecore, C.M., Watkin, R. & Kneale, G.G. *Protein Expr. Purif.* **39**, 152–159 (2005).
9. Liu, X. *et al. Proc. Natl. Acad. Sci. USA* **94**, 10669–10674 (1997).
10. Song, J. & Markley, J.L. *Protein Pept. Lett.* **14**, 265–268 (2007).
11. Landgraf, D., Okumus, B., Chien, P., Baker, T.A. & Paulsson, J. *Nat. Methods* **9**, 480–482 (2012).
12. Kopito, R.R. *Trends Cell Biol.* **10**, 524–530 (2000).
13. Vavouri, T., Semple, J.I., Garcia-Verdugo, R. & Lehner, B. *Cell* **138**, 198–208 (2009).
14. Sopko, R. *et al. Mol. Cell* **21**, 319–330 (2006).
15. Cook, M. & Tyers, M. *Curr. Opin. Biotechnol.* **18**, 341–350 (2007).
16. Morgan, D.O. *The Cell Cycle: Principles of Control.* (New Science Press, 2007).
17. Kim, Y.E., Hipp, M.S., Bracher, A., Hayer-Hartl, M. & Hartl, F.U. *Annu. Rev. Biochem.* **82**, 323–355 (2013).
18. Vergés, E., Colomina, N., Garí, E., Gallego, C. & Aldea, M. *Mol. Cell* **26**, 649–662 (2007).
19. Ferrezuelo, F. *et al. Nat. Commun.* **3**, 1012 (2012).
20. Fredriksson, S. *et al. Nat. Methods* **4**, 327–329 (2007).
21. Zhang, X. *et al. Nat. Methods* **5**, 163–165 (2008).

ONLINE METHODS

Bioinformatics screening for inntag candidates. The aim of our initial bioinformatics screen was to obtain a list of protein domains suitable to prepare fusion proteins. Owing to the difficulty of predicting antigenicity from pure theoretical approaches, our strategy took as starting point more than 2,000 protein families reported in VarDB[22] and epitopes reported in IEDB[23]. Protein structures containing the 63,424 epitopes listed in IEDB were obtained using BLASTP with standard setting against the PDB database for protein structures[24]. Finally, the list was extended to include possible homologs using the corresponding PFAM[25] domains. The final list included 3,500 antigenic proteins covering 1,383 PFAM domains. The initial list was further checked for the following properties. First, antigenicity was analyzed by BCIPRED[26]. The method uses a combination of sequence-based estimations including hydrophilicity, flexibility, accessibility, propensity of β-turns, antigenic propensity and polarity. Domains considered antigenic by at least two of the above criteria were accepted. Second, solvent accessibility, distance to epitope of N or C terminals and solvent accessibility of epitope, globular index, were calculated from the 3D structure using VMD[27]. These properties were used to discard domains where the predicted epitope was not accessible, binding to epitope would collide with the fused protein or terminal structures that could preclude a viable fusion construct. Third, functional characterization was obtained from Gene Ontology[28] and PFAM[25] annotations. Discarded terms affected to the following categories: Membrane related and solubility, activity, binding, polymeric state, and protein binding promiscuity.

The screening led to a short list of 226 domains (**Supplementary Table 1**) that was manually curated to further discard specific activities or properties that were not annotated in the above databases. These structures were further scrutinized to remove those annotated as having any enzymatic activity and those that bind small cofactors or other proteins already used in molecular interaction assays. We also rejected domains that have the ability to form stable homopolymers or show known regulatory sequences for degradation or localization. Finally, we excluded entries belonging to vertebrate organisms.

Disorder propensity was determined at the single–amino acid level with the aid of PONDR-FIT, a meta-predictor software that combines several individual disorder prediction algorithms[29].

Mammalian cell culture, expression vectors and cell analysis. NIH3T3 and HEK293T cells were obtained from ATCC and maintained for fewer than seven passages in DMEM containing glutamine, antibiotics and 10% FCS. Cells were periodically tested for mycoplasma infection (Minerva BioLabs). Hippocampal neurons were obtained from mouse hippocampi and cultured in supplemented Neurobasal medium as described[30]. Synthetic DNA encoding inntags was subcloned in a plasmid derived from pEGFP-N1 (Clontech) where some of the polylinker sequences had been replaced with a T7 promoter, ribosome binding sequences for gene expression in *E. coli* and mammalian cells, and a His_6 peptide for affinity purification. Inntag DNAs can be obtained from Addgene. Cells were transfected with Lipofectamine 2000 (Invitrogen) and analyzed 24 h after transfection. Expression level and aggregation index of GFP fusions in HEK293T cells were analyzed by epifluorescence microscopy with the aid of ImageJ.

Yeast strains, growth conditions and cell measurements. Yeast parental strain CML128 and methods used for chromosomal gene transplacement have been described[31]. Yeast cells were grown under exponential conditions for 7–8 generations in SC medium[32] with 2% glucose at 30 °C unless stated otherwise. Cell volume at bud emergence was determined from bright-field images with the aid of the ImageJ plugin BudJ[19] (http://www.ibmb.csic.es/home/maldea). Cell volume distributions were obtained with the aid of a Coulter Counter ZB as described[33].

Inntag purification. Expression of His_6-tagged inntags and GFP fusions was induced in *E. coli* BL21 (DE3) and cells were induced with 0.5 mM IPTG and grown at 25 °C for 18 h at 220 r.p.m. Expressed proteins were affinity purified under native conditions on Ni-NTA (Qiagen) as directed by the manufacturer.

iTRAQ analysis of the differential interactome of MYC_6 and IT6. The iTRAQ analysis was done at the BIDMC Proteomics Core Center (Harvard University) essentially as described[34]. Ni-NTA beads with either His_6-GFP, His_6-GFP-MYC_6 or His_6-GFP-IT6 were used in interaction assays with yeast cell extracts (CML128) as described[35]. High-salt eluates were digested with trypsin, labeled with 8 isobaric tags, pooled and separated by 2D liquid chromatography into 15 fractions, being each fraction analyzed in an 8-plex run using an AB/Sciex 4800 MALDI-TOF/TOF mass spectrometer. GO analysis was done by GO Term Finder (http://www.yeastgenome.org).

Monoclonal antibody production. Female BALB/c mice were immunized intraperitoneally with 75 µg of each inntag, emulsified in adjuvant (Stimune Adjuvant; Prionics). 30 and 60 d later, mice were injected with 75 µg of each antigen in adjuvant. 4 d before the fusion procedure, each mouse received an intravenous injection of the same antigen in PBS 20 mM phosphate. Splenocytes were fused with the Sp2/0-Ag14 mouse myeloma cell line[36]. Hybridoma clones were selected by conventional indirect ELISA, and cells from positive wells were cloned by limiting dilution. Relevant hybridomas have been deposited at the Leibniz Institute DSMZ. Monoclonal antibodies were purified from supernatants by affinity chromatography on protein G (HiTrap Protein G Sepharose High Performance; GE Healthcare).

Western blotting and immunoprecipitation analysis. Western blot analysis[31] was performed with antibodies to GFP (mouse clones 7.1 and 13.1, Roche), HA (mouse clone 12CA5, Roche), Flag (mouse clone M2, Sigma), MYC (mouse clone 9E10, Sigma), IT5 (mouse clone R19/8-11/18, AntibodyBcn), IT6 (mouse clone R19/4-11/15, AntibodyBcn), cyclin E1 (mouse clone EP435E, human specific, Millipore), Cdc28 (rabbit polyclonal, a gift from C. Mann), Ydj1 (mouse clone 1G10.H8, Abnova) and tubulin (mouse clone B-5-1-2, Sigma). Antibodies were used at a final concentration of 2 ng/ml. Solubility of inntag-GFP fusions in HEK293T cells was assessed by centrifugation (at 20,800 × *g* for 15 min) of cell extracts prepared in lysis buffer (50 mM Tris-HCl (pH 8), 150 mM NaCl, 2 mM DTT and protease and phosphatase inhibitors). Immunoprecipitation of yeast cell extracts was performed with equal amounts of yeast cells (100 OD_{600}) and 1 µg of the corresponding monoclonal antibodies as described[35].

Immunofluorescence. HEK293T or NIH3T3 cells were quickly washed in PBS and fixed in 4% paraformaldehyde and 4% sucrose for 30 min at room temperature. Fixed cells were permeabilized with 0.1% Triton X-100 in PBS for 5 min at 4 °C and blocked with 1% BSA in PBS. Blocking of hippocampal neurons was done with 5% NGS in PBS. Yeast cells were fixed and spheroplasted as described[35]. The aforementioned primary antibodies were detected with goat Alexa Fluor 568–labeled anti-mouse secondary antibodies (Molecular Probes) in blocking solution. Anti-GFP (mouse clones 7.1 and 13.1, Roche, 10 ng/ml) was used as reference for signal and background analysis. Fluorescence loss in photobleaching (FLIP) was performed in a Zeiss LSM780 confocal microscope. A small circular region of the cytoplasm (3.6 μm^2) was repetitively photobleached at full laser power for 3 s at a time and, between the bleaching periods, the cell was imaged with low intensity light to record fluorescence loss. For quantitative analysis, background intensity was subtracted, and intensities outside the photobleached area were measured over time and normalized using a transfected nonbleached cell. Fluorescence correlation spectroscopy (FCS) was performed essentially as described[37], and correlation data were fitted with the aid of QuickFit 3 software (http://www.dkfz.de/Macromol/quickfit/) assuming an anomalous mode of diffusion.

22. Hayes, C.N. *et al. Bioinformatics* **24**, 2564–2565 (2008).
23. Vita, R. *et al. Nucleic Acids Res.* **38**, D854–D862 (2010).
24. Bernstein, F.C. *et al. J. Mol. Biol.* **112**, 535–542 (1977).
25. Punta, M. *et al. Nucleic Acids Res.* **40**, D290–D301 (2012).
26. Saha, S. & Raghava, G.P.S. in *Artificial Immune Systems* (eds. Nicosia, G. *et al.*) 197–204 (Springer, 2004).
27. Humphrey, W., Dalke, A. & Schulten, K. *J. Mol. Graph.* **14**, 33–38 (1996).
28. The Gene Ontology Consortium. *Nucleic Acids Res.* **41**, D530–D535 (2013).
29. Xue, B. *et al. Biochim. Biophys. Acta* **1804**, 996–1010 (2010).
30. Pedraza, N. *et al. J. Neurosci.* **34**, 13988–13997 (2014).
31. Gallego, C., Garí, E., Colomina, N., Herrero, E. & Aldea, M. *EMBO J.* **16**, 7196–7206 (1997).
32. Sambrook, J. & Russell, D.W. *Molecular cloning: A laboratory manual*. CSHL Press, Cold Spring Harbor, NY, (2001).
33. Yahya, G., Parisi, E., Flores, A., Gallego, C. & Aldea, M. *Mol. Cell* **53**, 115–126 (2014).
34. Afkarian, M. *et al. Mol. Cell. Proteomics* **9**, 2195–2204 (2010).
35. Wang, H., Garí, E., Vergés, E., Gallego, C. & Aldea, M. *EMBO J.* **23**, 180–190 (2004).
36. Llanes, D., Nogal, M., Prados, F. & Viñuela, E. *Hybridoma* **10**, 757–762 (1992).
37. Slaughter, B.D., Schwartz, J.W. & Li, R. *Proc. Natl. Acad. Sci. USA* **104**, 20320–20325 (2007).

doi:10.1038/nmeth.3556