



UNIVERSITAT DE
BARCELONA

**Development of a framework for data management,
harmonisation, visualisation, and analytics
of multi-omicsclinical trials data in the context
of autoimmune diseases**

Ivo Christopher Leist



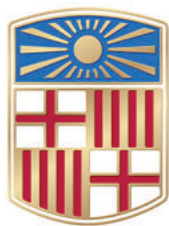
Aquesta tesi doctoral està subjecta a la llicència **Reconeixement- NoComercial – SenseObraDerivada 4.0. Espanya de Creative Commons.**

Esta tesis doctoral está sujeta a la licencia **Reconocimiento - NoComercial – SinObraDerivada 4.0. España de Creative Commons.**

This doctoral thesis is licensed under the **Creative Commons Attribution-NonCommercial-NoDerivs 4.0. Spain License.**

**Development of a framework for data management,
harmonisation, visualisation, and analytics of multi-omics
clinical trials data in the context of autoimmune diseases**

Ivo Christopher Leist



**UNIVERSITAT_{DE}
BARCELONA**

Doctoral Program in Biomedicine

Development of a framework for data management, harmonisation, visualisation, and analytics of multi-omics clinical trials data in the context of autoimmune diseases

Desenvolupament d'un marc per a la gestió, harmonització, visualització i anàlisi de dades multi-òmiques d'assaigs clínics en el context de les malalties autoimmunes

Ivo Christopher Leist

Thesis Director:

Dr. Ivo Glynne Gut

Biomedical Genomics Group

Centro Nacional de Análisis Genómico (CNAG)

Universitat de Barcelona (UB)

Tutor:

Dr. Modesto Orozco

Molecular Modelling and Bioinformatics Group

Institute for Research in Biomedicine (IRB Barcelona)

Universitat de Barcelona (UB)

Barcelona, 2025

**“Learning is the only thing the mind never exhausts,
never fears, and never regrets”**

— Leonardo da Vinci —

Acknowledgements

Professional context:

First and foremost, I would like to express my gratitude to Ivo Gut for recognising my potential in software engineering back then at the de.NBI and ELIXIR-DE plenary meeting in Berlin, and for giving me the opportunity to unleash it here at CNAG. Thank you, Ivo, for providing me with the resources, the freedom to explore various aspects of the software development lifecycle, and the support that enabled me to grow into a self-driven, software engineering-oriented scientist. On a more personal note, I genuinely appreciate the trust you have placed in me, that our projects advance successfully whether I'm working from the office, from home, or from Germany when visiting my girlfriend.

Secondly, I would also like to thank my thesis advisory committee members, Sergi Beltran, Cedric Notredame, Juan Ramón Gonzalez, and Modesto Orozco, for their insightful annual feedback, which greatly helped me frame my projects with a strong focus on software engineering within a clear biomedical context.

Next, my heartfelt thanks go to the members of the Data Platform and Tools Development team (known at CNAG as Team GPAP). I vividly recall the warm welcome I received from Davide Piscia and the entire team at my first remote stand-ups amidst the pandemic. This possibility to daily share my progress, challenges and get your input for subsequent steps has been the perfect environment to kickstart the development of my main project and, with it, my journey into software engineering. I am especially grateful to Alberto Corvò for his infinite patience in helping me to debug countless React JavaScript state management issues. Special thanks also to Sofia Chaves for her willingness to dedicate time to create icons and logos for my projects, as well as to review the user interfaces of the web applications I developed. Beyond the professional support, Sofia, I will also keep in my heart the moments of laughter when we were sharing the office space together with Ioana-Damaris Teoc and Marc Pauper, which made the sometimes challenging PhD journey much more bearable. In this regard, thank you Damaris for your consistently uplifting words, and Marc, I really enjoyed trying out together the latest Linux (command-line) tools for boosting

productivity. Last but not least within this team, I would like to thank Etienne Jodry, whose strong emphasis on good software engineering practices helped me to grow beyond my hackathon-experiences-infused minimum viable product mindset, towards a developer mentality that values code quality and maintainability.

Additionally, I would like to thank the members of the Biomedical Genomics group. I am especially grateful for the productive collaboration and companionship with Manuel Rueda, whose seniority and experience in the field have been invaluable in navigating the complexities of participating in a large European consortium and the process of publishing scientific articles. Manuel, thank you for continuously encouraging me to make the most of my PhD journey, and in particular for your support in the final stages of my thesis. Regarding the latter, I would also like to thank Gabriela Aguilera and Miranda Stobbe. Regarding the latter, I would also like to thank Gabriela Aguilera and Miranda Stobbe. And of course, I cannot forget to mention Sonal Rashmi and Davide Canevazzi, who have been my closest companions in the Biomedical Genomics group throughout this journey, sharing the ups and downs of the PhD experience, and always being there to lift each other up when needed.

The last research group at CNAG I would like to thank as a whole is Anna Esteve's team, which kindly introduced me to the world of RNA-Seq data analysis and provided valuable feedback on the "researcher-friendliness" of the 3TR data warehouse I developed during my PhD.

Another group of people I would like to acknowledge is the CNAG IT team, who patiently responded to my endless support tickets, whether it was about setting up a new production environment, or for resolving the various issues that arose along the way. In that regard, I am grateful to Gonzalo Vera, for always being approachable and willing to squeeze in my requests into his packed schedule, on many occasions, addressing them directly on the spot.

Next, I would like to express my sincere gratitude to all members of the 3TR consortium, which have been very welcoming from day one on. I especially enjoyed the large annual meetings, where I had the opportunity to meet many of you (Joanis

Parodis, Florian Tran, Guillermo Barturen, Matthias Antonin, Björn Tackenberg, Scott Jelinsky, apologies I cannot list all of you here!). Short interlude: Laura Jiménez and Manuel Rueda, thanks for the good companionship during these meetings. Among all 3TR members, I would especially like to thank Guillermo Barturen, whose feedback shaped the look and feel of the 3TR data warehouse, and Matthias Antonin, through whom I gained interesting insights into the field of multiple sclerosis research. Additionally, I would like to thank Charisios Triantafyllou who introduced me to signature analyses of single cell RNA-Seq data.

Still in a professional context, but on a more personal note, I would like to thank those whose support in the past has been pivotal to my career development. First, I want to thank my former boss at Roche Diabetes Care, Birgit Kräling. I am confident, Birgit, that the freedom and support you gave me during my internship/working student position to pursue my newfound passion for automation and software development played an important role in shaping the path that led me to where I am today.

Equally, I am deeply grateful to Rob Russell for giving me the opportunity to combine my emerging interest in software engineering with my long-standing passion for the biological sciences during my internship and master's thesis in his group at the BioQuant in Heidelberg. So thanks again, Rob, for your support and encouragement in my application for the PhD position at CNAG, despite the fact that you probably would have preferred that I would have done my PhD in your group. This gratitude also extends to Yvonne Lara, who, sadly, can no longer read these lines (RIP). I would also like to thank my former colleagues at the BioQuant, in particular Juan Carlos Gonzalez and Gurdeep Singh, from whom I learned the fundamentals of scientific programming, data analysis and machine learning. Yes, JC and Gurdeep, I still have the Roche trophy somewhere. Likewise, I am grateful to Jannik Buhr for introducing me to the world of R and the Tidyverse, which turned out to be an important skill for my PhD.

Finally in this section I would like to express my heartfelt thanks to Anna Solé Amat, whose warm and caring personality was truly a blessing. Thanks a lot, Anna, for always being incredibly helpful in guiding me through the bureaucratic maze of the PhD and your open ear for the ups and downs of my PhD journey even after CNAG's separation from CRG.

Personal context:

First and foremost, my deepest gratitude goes to my parents. Thanks mom and dad, I am so fortunate that you both are always so encouraging and supportive that I pursue my dreams, even when they take me far away from you. And as I learned during this time this does not stop you from visiting me for my birthdays. Thank you so much for the hours you spent on the German Autobahn, and the French/Spanish motorways to visit me in Barcelona.

A special shout-out goes to my grandmother Elisabeth: Oma, diese Zeilen in der bisher größten Errungenschaft meines Lebens sind für dich. Tausend Dank für die bedingungslose Liebe, die du mir bei jedem Besuch schenkst!

I am also incredibly thankful to my closest friends in Barcelona: Sabina, Alex, Tahnee, and Omer. As well as to those scattered across Europe/world. Thank you Nessi, Max and Jonas for never growing tired of hearing the latest updates from my PhD roller coaster. I truly appreciate your continued support and that you managed to visit me in Barcelona despite your busy lives. Speaking of visits, thanks members of Reise Gruppe R #Harter[Roche Evolution]Kern for making my 30th birthday celebration in Barcelona an unforgettable happening!

Another heartfelt thanks goes to all the flatmates who made me feel at home in Barcelona. Thank you, Niko, for never giving up on luring me out of my coding/thesis-writing cave to enjoy BCN life on the beach, skateboarding at Paral·lel/MACBA, or dancing the stress away at one of your DJ sessions/Casa Figari!

And finally, my deepest appreciation goes to you, Anni. It means the world to me that you have always been there for me, supporting all my decisions in the past years, well aware that these would take me away physically for extended periods of time. So I really appreciate that you have not held me back in Heidelberg, but rather encouraged me to take the leap for this Barcelona adventure. Thanks a LOT for being my anchor throughout this time. Especially for reminding me to take breaks, from the never-ending coding and writing to enjoy a well-deserved "Feierabend" with you or for a longer getaway. All these precious moments we actively spent together were essential

for my mental well-being and helped me to reopen the laptop with a fresh and re-energised mind. I know that I have really tested your patience, especially considering that this whole journey ended up taking much longer than we had imagined initially. Thanks again for understanding. I cannot wait to start the next chapter of my life together with you in Freiburg. Ich love te !

Thanks everyone for everything and apologies to anyone I potentially have missed to mention!

Abstract

(Multi-)omics technologies generate vast amounts of data that require proper organisation and storage for efficient analyses and interoperability. Historically research organisations managed data locally, but in today's large-scale international collaborative projects, this hinders efficient data exchange. A case in point, is the use of SFTP file server for data sharing purposes, which requires familiarity with specific software or command-line tools. Commercial cloud storage (e.g., Dropbox or Google Cloud) offers an alternative but demands trust in black-box solutions and their providers. Open-source alternatives exist, each with specific strengths and limitations.

In this thesis, I evaluated the applicability of these existing open-source data warehouse solutions for the 3TR project (3tr-imi.eu). It is a public-private partnership tackling seven immune-mediated diseases across 15 European countries involving 69 partners. As none of the tested software solutions met the needs of the 3TR project consortium I developed OmicsDM, a web-based platform that facilitates data storage, sharing, visualisation, and analysis. In 3TR, diseases are analysed both independently and jointly. Regarding the latter, the heterogeneity of the pheno-clinical data across studies poses a major challenge. To address this, I participated in the development of Convert-Pheno, a software toolkit for the interconversion of common data models for phenotypic data. Thus, laying the foundation for the envisioned cross-disease analysis. Next, the consortium needed a method to group patients by phenotypic resemblance rather than diagnostic labels alone. Thus, enabling researchers to construct phenotype-based cohorts whose analysis can reveal shared mechanisms or treatment responses across diseases To aid in the grouping we built Pheno-Ranker. It leverages semantic similarity to compare phenotypic data stored in GA4GH standards and beyond. For both Convert-Pheno and Pheno-Ranker I built intuitive web user interfaces, each with its own public playground to let users explore the capabilities of each of the tools.

To sum it up, the developments in this thesis are an important contribution to the improvement of the biomedical data management, harmonisation, visualisation and analytics infrastructure in Europe and beyond.

Resumen

Las tecnologías (multi)ómicas generan grandes cantidades de datos que requieren una organización y un almacenamiento adecuados para garantizar la eficiencia de los análisis y la interoperabilidad. Históricamente, los centros de investigación han sido los encargados de los datos a nivel local. Sin embargo, en los proyectos colaborativos internacionales a gran escala actuales, esto dificulta el intercambio eficiente de datos. Un ejemplo de ello es el uso del servidor de archivos SFTP para compartir datos, que requiere estar familiarizado con un software específico o con herramientas de línea de comandos. El almacenamiento comercial en la nube (por ejemplo, Dropbox o Google Cloud) ofrece una alternativa viable, pero exige confiar en soluciones de tipo “caja-negra” y en sus proveedores. Existen alternativas de código abierto, cada una con sus ventajas y limitaciones específicas.

En esta tesis, se evaluó la aplicabilidad de estas soluciones de almacenamiento de datos de código abierto existentes para el proyecto 3TR (3tr-imi.eu). El proyecto 3TR es una asociación público-privada que aborda siete enfermedades inmunomediadas en 15 países europeos y en la que participan 69 socios. Dado que ninguna de las soluciones de software probadas satisfacía las necesidades del consorcio del proyecto 3TR, se desarrolló OmicsDM, una plataforma web que facilita el almacenamiento, el intercambio, la visualización y el análisis de datos. En 3TR, las enfermedades se analizan tanto de forma independiente como conjunta. En cuanto a esto último, la heterogeneidad de los datos fenotípicos entre los distintos estudios supone un reto importante. Para abordarlo, participé en el desarrollo de Convert-Pheno, un conjunto de herramientas de software para la interconversión de modelos de datos comunes para datos fenotípicos. De este modo, se sentaron las bases para el análisis previsto entre enfermedades. Por otra parte, el consorcio necesitaba un método para agrupar a los pacientes por similitud fenotípica, en lugar de solo por etiquetas de diagnóstico. De este modo, los investigadores pueden construir cohortes basadas en fenotipos cuyo análisis puede revelar mecanismos compartidos o respuestas al tratamiento entre enfermedades. Para ayudar en este cometido, creamos Pheno-Ranker, éste aprovecha la similitud semántica para comparar los datos fenotípicos almacenados en los estándares GA4GH y otros. Tanto para Convert-Pheno como para Pheno-Ranker, se crearon interfaces de usuario web

intuitivas, cada una con su propio entorno público para que los usuarios pudieran explorar las capacidades de cada una de las herramientas.

En resumen, los avances de esta tesis son una importante contribución a la mejora de la infraestructura de gestión, armonización, visualización y análisis de datos biomédicos en Europa y más allá.

Palabras clave: Genómica, Bioinformática, Multi-ómica, Base de datos, Inmunología

Resum

Les tecnologies (multi)òmiques generen grans volums de dades que requereixen una organització i emmagatzematge adequats per tal de permetre anàlisis eficients i interoperabilitat. Històricament, els centres de recerca han gestionat les dades localment, però en els projectes col·laboratius internacionals a gran escala actuals, això dificulta l'intercanvi eficient de dades. Un exemple d'això és l'ús de servidors SFTP per compartir dades, que requereix coneixements específics de programari o eines de línia de comandos. L'emmagatzematge al núvol comercial (per exemple, Dropbox o Google Cloud) és una alternativa, però exigeix confiar en solucions opaques i els seus proveïdors. També existeixen alternatives de codi obert, cadascuna amb punts forts i limitacions específiques.

En aquesta tesi, es va avaluar l'aplicabilitat d'aquestes solucions de magatzems de dades de codi obert per al projecte 3TR (3tr-imi.eu). El consorci 3TR és una associació públic-privada que aborda set malalties immunomediades en 15 països europeus amb la participació de 69 socis. Com que cap de les solucions provades satisfacia les necessitats del consorci 3TR, es va desenvolupar OmicsDM, una plataforma web que facilita l'emmagatzematge, compartició, visualització i anàlisi de dades. Al projecte 3TR, les malalties s'analitzen tant de manera independent com conjunta. Pel que fa a aquesta última, l'heterogeneïtat de les dades fenotípiques i clíniques entre estudis representa un repte important. Per fer-hi front, vaig participar en el desenvolupament de Convert-Pheno, un conjunt d'eines per a la interconversió de models de dades comuns per a dades fenotípiques, assentant així les bases per a l'anàlisi entre malalties. Seguint aquesta línia, el consorci necessitava un mètode per agrupar pacients segons la semblança fenotípica i no només per les etiquetes diagnòstiques, permetent així als investigadors construir cohorts basats en el fenotip, la qual cosa pot revelar mecanismes compartits o respostes terapèutiques comunes entre malalties. Per facilitar aquest agrupament, vam desenvolupar Pheno-Ranker, una eina que utilitza la similitud semàntica per comparar dades fenotípiques emmagatzemades en formats GA4GH i altres. Per a Convert-Pheno i Pheno-Ranker, vaig desenvolupar interfícies web intuïtives amb espais de proves públics perquè els usuaris poguessin explorar-ne les funcionalitats.

En resum, els desenvolupaments d'aquesta tesi representen una contribució important a la millora de la infraestructura de gestió, harmonització, visualització i anàlisi de dades biomèdiques a Europa i més enllà.

Paraules clau: Genòmica, Bioinformàtica, Multi-òmica, Base de dades, Immunologia

Table of Contents

Acknowledgements.....	VII
Abstract.....	XIII
Resumen.....	XV
Resum.....	XVII
Abbreviations.....	XXI
Chapter 1: Introduction.....	23
1.1 The 3TR project and the connection to this thesis.....	25
1.1.1 Diseases investigated in the 3TR project.....	26
1.1.1.1 Chronic autoimmune diseases.....	27
1.1.1.2 Inflammatory bowel diseases.....	29
1.1.1.3 Respiratory diseases.....	30
1.1.1.4 Commonalities of chronic autoimmune, inflammatory bowel, and respiratory diseases.....	31
1.1.2 The potential of multi-omics analyses to find predictors for treatment response	32
1.1.2.1 Transcriptomics.....	32
1.1.2.1.1 Bulk RNA-Seq and its relevance in the study of diseases.....	33
1.1.2.1.2 Single-cell RNA sequencing and its increasing importance in the study of diseases.....	35
1.1.3 Data sources in the 3TR project and its storage.....	36
1.1.3.1 Legacy data.....	36
1.1.3.2 Newly produced data.....	37
1.1.4 The role of CNAG and myself in the 3TR project.....	37
1.2 The value of data warehouse solutions in biomedical research.....	38
1.2.1 Centralised vs. federated data warehouse architectures.....	39
1.3 Pheno-clinical data, crucial for future discoveries but difficult to handle.....	41
1.3.1 Definition of pheno-clinical data.....	41
1.3.2 The importance of pheno-clinical data.....	41
1.3.3 Pheno-clinical data collection and how to potentially overcome its challenges	42
1.3.3.1 REDCap and its role in the 3TR project.....	42
1.3.4 The importance of standardised pheno-clinical data storage.....	43
1.3.4.1 Global Alliance for Genomics and Health standards.....	44
1.3.4.1.1 Phenopackets.....	44
1.3.4.1.2 Beacon v2, Beacon Models and Beacon friendly format.....	44
1.4 Publications.....	46
Chapter 2: Objectives.....	47
Chapter 3: Materials and methods.....	49
3.1 3TR project data.....	51
3.1.1 Molecular data.....	51
3.1.2 Pheno-clinical data.....	53
3.2 Data warehouse requirements.....	55
3.3 Explored existing data warehouse solutions in the biomedical research community	55
3.3.1 tranSMART.....	56
3.3.2 Opal.....	57
3.3.3 MOLGENIS.....	57
3.4 Proposed solution: Development of a new data warehouse solution tailored to the 3TR project.....	58
3.4.1 Data storage.....	59
3.4.2 Client and server.....	60

3.4.3 Identity and access management.....	62
3.4.4 Architecture of OmicsDM.....	63
3.4.5 Interactive data visualisation.....	65
3.4.5.1 Usage of CZ CELL×GENE Annotate for single cell data visualisation....	65
3.4.5.2 CELL×GENE Annotate instances orchestration with ShinyProxy.....	65
3.4.6 Data analyses.....	67
3.4.6.1 Supported workflow management systems.....	67
3.4.6.2 Analyses scheduling and execution.....	70
3.4.6.3 Downstream analyses of RNA-Seq data.....	72
3.4.7 Deployment of OmicsDM.....	78
3.5 Standalone tools for pheno-clinical data exploration.....	79
3.5.1 Convert-Pheno-UI.....	79
3.5.2 Pheno-Ranker-UI.....	80
3.6 Life Science Login as identity provider.....	82
Chapter 4: Results.....	83
4.1 OmicsDM.....	85
4.1.1 Personas and user stories for the 3TR project.....	85
4.1.2 Applicability of existing data warehouse solutions to the 3TR project.....	86
4.1.3 Modules of OmicsDM.....	88
4.1.3.1 OmicsDM core module.....	89
4.1.3.2 OmisDM visualisation module.....	92
4.1.3.3 OmicsDM analysis module.....	95
4.1.4 Database schema of OmicsDM.....	100
4.2 Standalone tools for pheno-clinical data exploration.....	102
4.2.1 Convert-Pheno-UI.....	103
4.2.1.1 Additional features not included in the Convert-Pheno publication.....	113
4.2.2 Pheno-Ranker-UI.....	114
4.2.2.1 Additional Pheno-Ranker-UI features only briefly mentioned in the publication.....	137
4.3 OmicsDM in concert with Convert-Pheno and Pheno-Ranker.....	139
Chapter 5: Discussion.....	141
5.1 OmicsDM.....	143
5.1.1 OmicsDM as a future proof data warehouse solution for the 3TR project....	143
5.1.1.1 Decision for a centralised data warehouse architecture despite the trend towards federated solutions.....	147
5.1.2 OmicsDM vs existing data warehouse solutions.....	148
5.1.3 Applicability of OmicsDM to other consortia.....	150
5.2 Standalone tools for pheno-clinical data harmonisation & exploration.....	151
5.2.1 Convert-Pheno.....	151
5.2.2 Pheno-Ranker.....	152
Chapter 6: Conclusions.....	153
6.1 Overall.....	155
6.2 Development of OmicsDM, an integrative data management, analysis and visualisation platform for multi-omics data.....	155
6.3 Standalone tools for pheno-clinical data exploration.....	156
6.3.1 Development of a web UI for Convert-Pheno, a software toolkit for the interconversion of standard data models for phenotypic data.....	156
The UI enables users to:.....	156
6.3.1.1 Development of an R shiny UI for Pheno-Ranker, a toolkit for comparison of phenotypic data stored in GA4GH standards and beyond.....	156
Chapter 7: References.....	157

Abbreviations

Biology & Biomedical informatics	Computer science & General
AnnData: Annotated Data BFF: Beacon Friendly Format BPO: Biological Process Ontology CCO: Cellular Component Ontology CD: Crohn's Disease cDNA: Complementary DNA CNS: Central Nervous System COPD: Chronic Obstructive Pulmonary Disease DE: Differential Expressed DGE: Differential Gene Expression eCRF: Electronic Case Report Form EDC: Electronic Data Capture GA4GH: Global Alliance for Genomics and Health GMT: Gene Matrix Transposed file format GO: Gene Ontology GSEA: Gene Set Enrichment Analysis IBD: Inflammatory Bowel Disease IMI: Innovative Medicines Initiative KEGG: Kyoto Encyclopedia of Genes and Genomes LFC: Log2 Fold Change MDS: Multidimensional Scaling MFO: Molecular Function Ontology mRNA: Messenger RNA MS: Multiple Sclerosis ORA: Over-representation Analysis PXF: Phenotypic Exchange Format RA: Rheumatoid Arthritis REDCap: Research Electronic Data Capture RNA-Seq: RNA sequencing scRNA-Seq: Single-cell RNA Sequencing SLE: Systemic Lupus Erythematosus QC: Quality Control UC: Ulcerative Colitis	2FA: Two-factor authentication API: Application Programming Interface AuthN: Authentication AuthZ: Authorisation CLI: Command Line Interface CPU: Central Processing Unit CSV: Comma-separated Values CZ: Chan Zuckerberg DSL: Domain-specific Language GB: Gigabytes GDPR: General Data Protection Regulation HDF5: Hierarchical Data Format version 5 HTML: Hypertext Markup Language HTTP: Hypertext Transfer Protocol ID: Identifier JSON: JavaScript Object Notation OS: Operating System RAM: Random-access memory REST: Representational State Transfer SFTP: SSH File Transfer Protocol SQL: Structured Query Language SSH: Secure Shell SSO: Single Sign-On TSV: Tab-separated Values UI: User Interface URL: Uniform Resource Locator UUID: Universally Unique Identifier VM: Virtual Machine WebDAV: Web Distributed Authoring and Versioning

Chapter 1: Introduction

1.1 The 3TR project and the connection to this thesis

All the tools presented in this thesis were developed in the context of the 3TR project (3tr-imi.eu), which stands for Taxonomy, Treatment, Targets and Remission. 3TR is a seven-year public-private partnership funded by the European Commission through the Innovative Medicines Initiative (IMI). The investigation of seven autoimmune diseases in 15 European countries involving 69 partners makes 3TR the largest immunology project funded by the IMI to date (June 2025). Its primary objective is to consolidate data and patient information from clinical trials in seven diseases: Asthma, chronic obstructive pulmonary disease (COPD), Crohn's disease (CD), ulcerative colitis (UC), rheumatoid arthritis (RA), multiple sclerosis (MS), and systemic lupus erythematosus (SLE). In total, data of 50,000 patients will be collected from 50 clinical trials throughout the 3TR project. This data needs to be captured and placed in an appropriate framework. Thus, the first goal in my thesis was to set up a data warehouse solution for the 3TR project. Instead of merely storing data, our aim was to provide a platform enabling researchers to interact with the data through visualisation, analysis, and sharing of results with collaborators via a web interface. To this end, during this thesis I built a platform named OmicsDM, which aids the partners in their pursuit of 3TR's main goals. To these belong for instance the better understanding of the underlying molecular mechanisms of non-response to treatment. What sets 3TR apart from other projects is that on top of the individual disease analyses, the data can also be analysed across diseases, as treatments are the same or similar. In the end, all these insights combined will lead to a higher level of personalised therapy by predicting from which specific treatment a patient will benefit the most.

To achieve the envisioned cross-disease analyses, the data has to be stored in a structured way. Additionally, the patients' pheno-clinical ("phenotypic" + "clinical") data needs to be harmonised. Since this data comes in various formats, a way to convert them into a common format was needed. We decided to use standards from the Global Alliance for Genomics and Health (GA4GH), an international coalition committed to advancing FAIR (findable, accessible, interoperable, and reusable) data principles. However, GA4GH standards alone would not have been sufficient to enable downstream analyses across studies. In fact, a tool was needed to better understand (a) which patients are similar to each other and (b) which variables are the cause for a

potential dissimilarity. Thus, the tool should help to assess the quality of the data and to build cohorts for the downstream analyses.

To sum up, the 3TR project not only provided an ideal use case for the development of an interactive platform for data management, visualisation and analytics, but also gave me the opportunity to contribute to the development of tools for the harmonisation (Convert-Pheno) and exploration (Pheno-Ranker) of pheno-clinical data.

1.1.1 Diseases investigated in the 3TR project

The 3TR project is the continuation of previous research projects done by the partners involved in the project. It especially builds on the learnings of the IMI project PRECISESADS¹ which aimed to identify biomarkers for systemic autoimmune diseases (SADs). A key finding thereof was that individuals with different diseases such, as RA and systemic sclerosis, cluster together based on their molecular profiles. In fact, three distinct patterns were identified, named after the affected cell functions, namely "inflammatory", "lymphoid", and "interferon" [1]. The presence of these patterns is expected to be found in other diseases as well. Thus, the idea of the 3TR project was to cast an even wider net by including more autoimmune and inflammatory conditions. Table 1.1 shows the selected diseases, its allocation by the 3TR partners to one of the three categories (*i.e.*, chronic autoimmune, inflammatory bowel, and respiratory diseases), as well as the number of affected patients worldwide.

Table 1.1: Diseases investigated in the 3TR project, and the number of affected patients worldwide

Disease group	Disease name	Affected patients worldwide in million [reference]
Chronic autoimmune	Multiple sclerosis	~2.3 [2]
	Systemic lupus erythematosus	~3.4 [3]
	Rheumatoid arthritis	~17.6 [4]
Inflammatory bowel	Ulcerative colitis	~5.0 [5]
	Crohn's disease	~4.9 [6]
Respiratory	Asthma	~262.4 [7]
	Chronic obstructive pulmonary disease	~212.0 [8]

¹ imi.europa.eu/projects-results/project-factsheets/precisesads

One of the goals of 3TR was to test for the presence of the patterns identified in PRECISESADS in the newly collected data, to thereby verify or falsify the hypothesis that these disease groups share common molecular characteristics. Even though the diseases investigated in the 3TR project are not considered to be immediately life-threatening, such as cancer or heart disease, they still have a high societal burden. This is not only reflected by the large number of affected patients (Table 1.1) but also in the requirement of long-term expensive treatments. While inflammatory bowel and respiratory diseases have a localised impact, chronic autoimmune diseases are systemic, as they can affect multiple organs and tissues. Nevertheless, predominantly affected tissues or organs can also be identified in chronic autoimmune diseases (Figure 1.1).

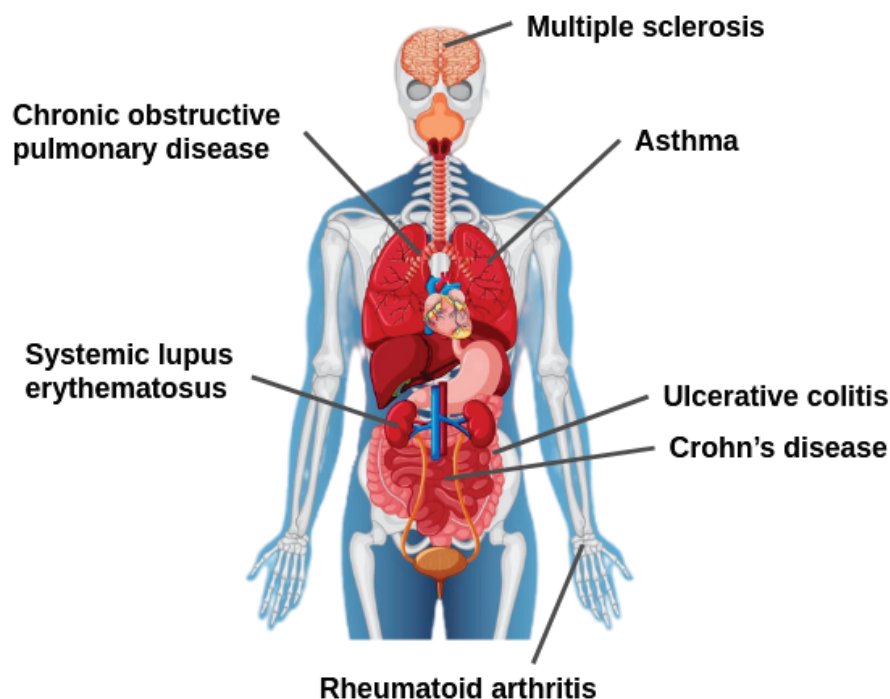


Figure 1.1: Tissues/organs predominantly affected by the diseases investigated in the 3TR project. Adapted from brgfx on Freepik.

1.1.1.1 Chronic autoimmune diseases

Autoimmune diseases include all conditions caused by the immune system mistakenly attacking the body's own cells. There are roughly 80 autoimmune diseases, of which 50% are considered rare [9]. An overview of the most common autoimmune diseases is shown in Figure 1.2, the diseases investigated in the 3TR projects are marked with boxes.

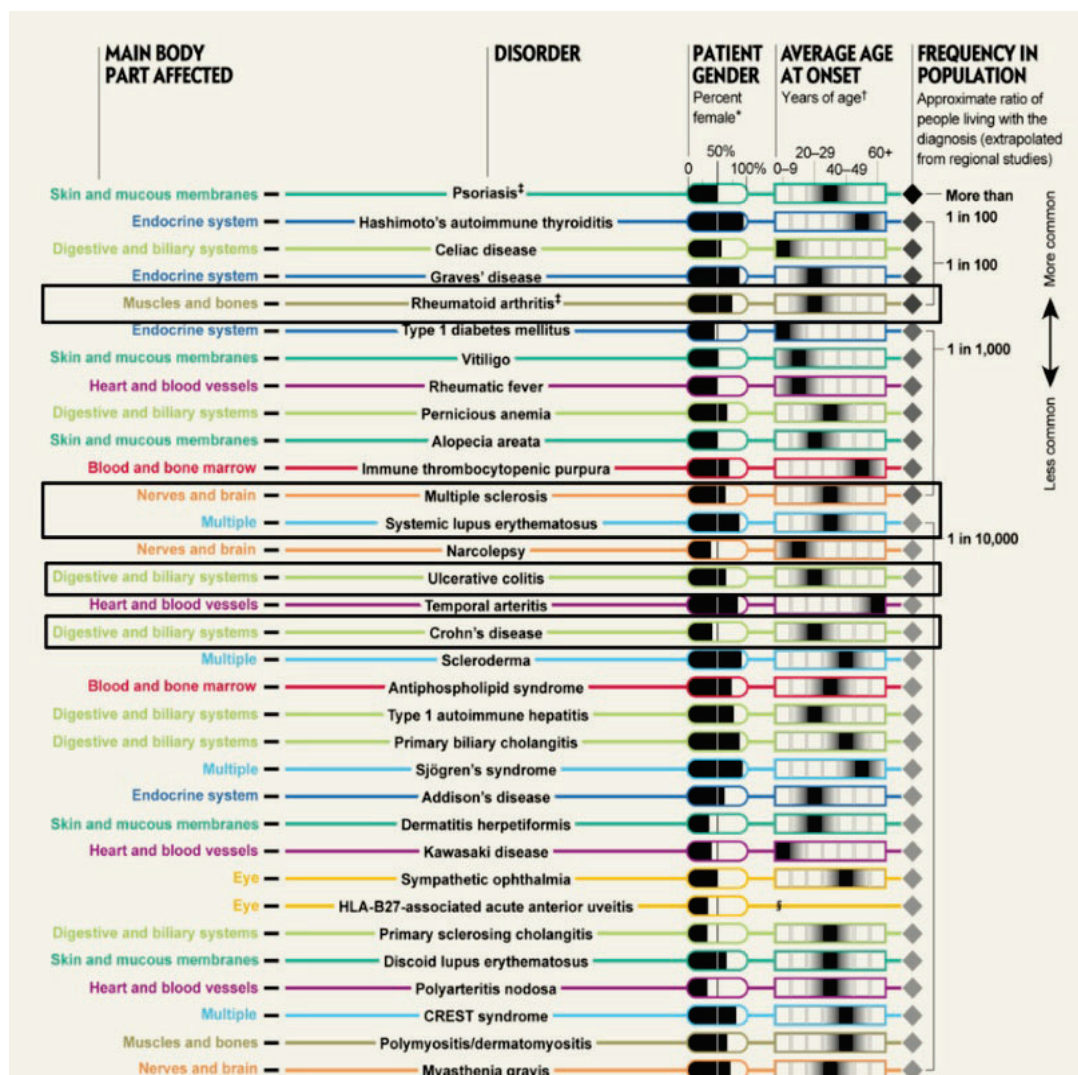


Figure 1.2: Overview of the most common autoimmune diseases. The figures shows the affected body parts, the male/female ratio (*depending on the underlying study biological or gender identity), the average age of onset and the frequency in the population († mean age of onset as published in 2012) of a specific autoimmune disease. Only diseases with a frequency of at least 1 in 10,000 are included. Diseases investigated in the 3TR project are marked with boxes. Adapted from scientificamerican.com/article/the-terrible-toll-of-76-autoimmune-diseases.

Not all autoimmune diseases are chronic. In fact, some, such as the Guillain-Barré Syndrome, are often only acute. Nevertheless, the majority of autoimmune diseases are chronic and currently not curable. Therefore, treatment primarily focuses on preventing rapid disease progression and managing symptoms. Unfortunately, experience shows that the existing treatments for diseases such as MS, RA, and SLE are not equally effective for all patients [10]. This is why these three diseases were selected for the 3TR project. While RA is best known for its effect on the joints, it is like SLE, a systemic disease (Figure 1.3). MS on the other hand predominantly affects the central nervous system (CNS) (Figure 1.4).

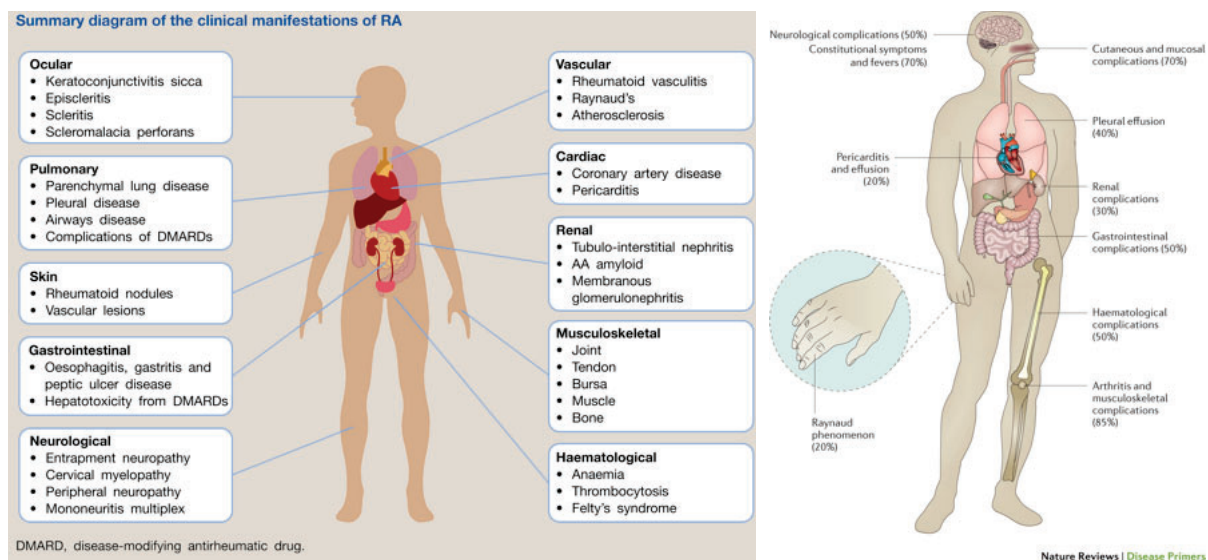


Figure 1.3: Affected tissues and organs in RA (left) and SLE (right). From [11], [12].

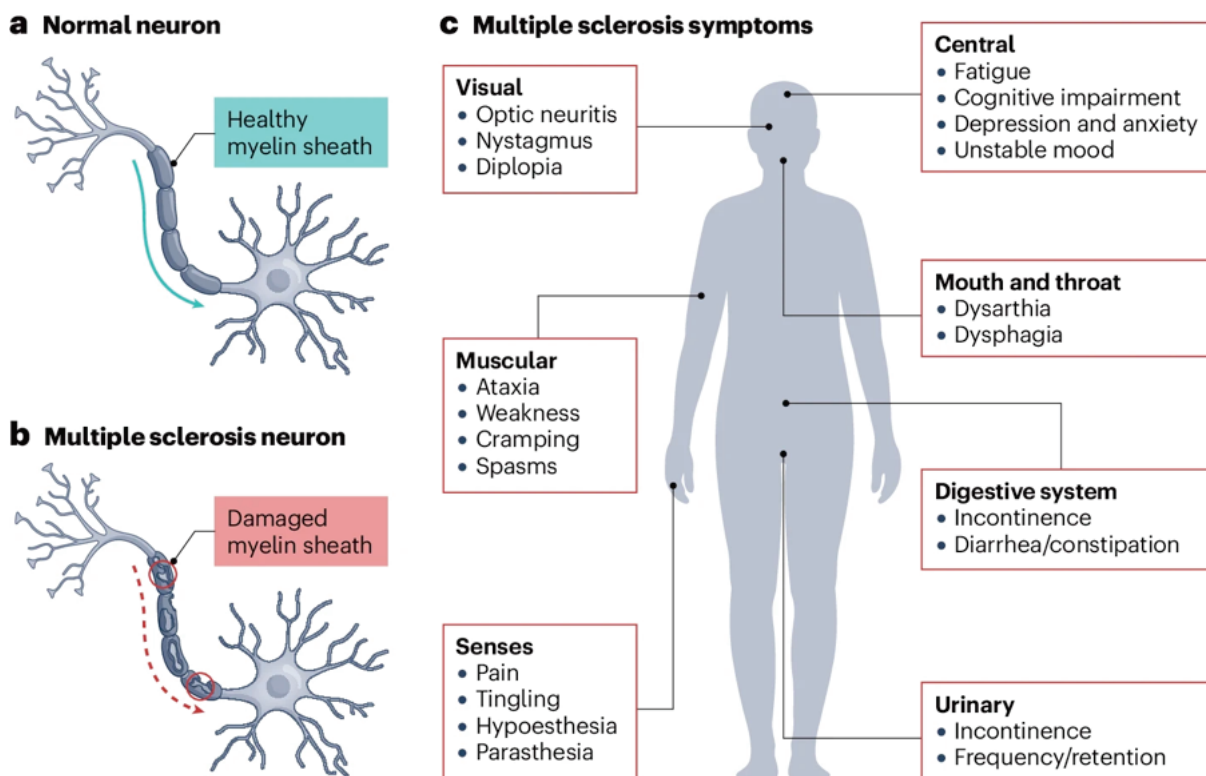


Figure 1.4: MS symptoms and its cause. The demyelination of the nerve fibres in the CNS. From [13].

1.1.1.2 Inflammatory bowel diseases

Inflammatory bowel diseases are a group of chronic inflammatory conditions of the gastrointestinal tract. The two most common forms of IBD are CD and UC. Both diseases are characterised by inflammation of the digestive tract, which can lead to symptoms such as abdominal pain, diarrhoea, and weight loss. An overview of the parts of the digestive system affected by IBD is shown in Figure 1.5.

The exact cause of IBD is unknown, but it is thought to be a combination of genetic [14], environmental [15], and immunological factors [16]. Unfortunately, IBD cannot be cured yet. Nevertheless, there are treatments available that can lower the burden of the disease and improve the patients' quality of life [17].

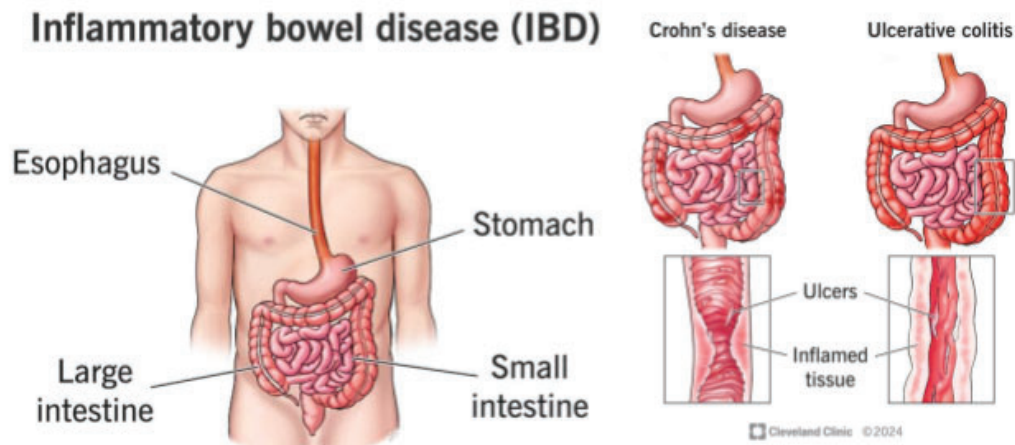


Figure 1.5: Overview how CD and UC affect the digestive system. From my.clevelandclinic.org/health/diseases/15587-inflammatory-bowel-disease-overview.

1.1.1.3 Respiratory diseases

Respiratory diseases encompass a wide range of conditions that affect the lungs and airways. Mainly caused by inhaled smoke, dust, microbes and other forms of air pollution [18]. Among the most common respiratory diseases [19] are the ones investigated in the 3TR project, namely asthma and COPD. How both diseases affect the respiratory system is illustrated in Figure 1.6.

While asthma usually is relatively mild, the progression of COPD can lead to severe breathing difficulties and even death. In fact, COPD is the fourth leading cause of death globally (hse.gov.uk/copd/aboutus.htm). In terms of treatment, neither asthma nor COPD can be cured. Nevertheless, the symptoms can be managed with medication and lifestyle changes. For both diseases an often-used frontline treatment is inhalation of β 2-agonists [20]. However, the usage of these drugs can have side effects, such as tachyarrhythmia and tremor [21]. In an attempt to find a treatment with less adverse effects, biologics, such as IL-13 and IL-5, are explored as an alternative option. While for asthma even for severe cases the treatment with biologics is already established [22], for COPD there are no biologics approved yet [23]. However, there are already promising results of on-going phase 3 clinical trials, such as BOREAS [24].

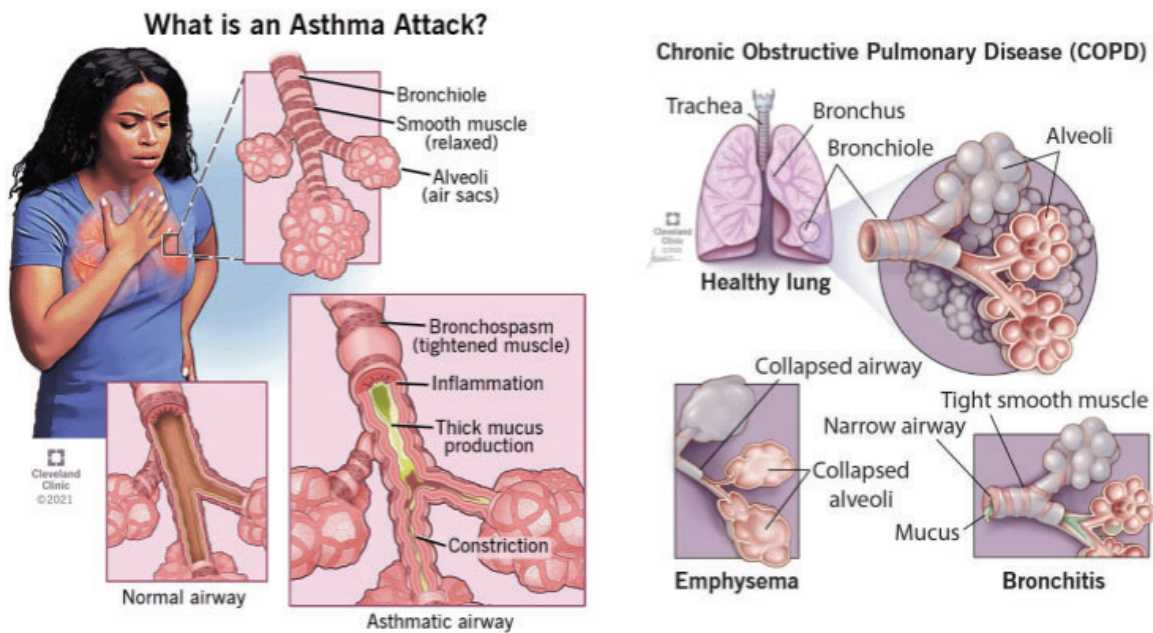


Figure 1.6: Overview how asthma (left) and COPD (right) affect the respiratory system. From my.clevelandclinic.org/health/diseases.

1.1.1.4 Commonalities of chronic autoimmune, inflammatory bowel, and respiratory diseases

The seven diseases targeted in the 3TR project have several commonalities justifying their investigation in a joint project. First, they are all chronic diseases that affect the immune system and can lead to inflammation and tissue damage. Additionally, all are characterised by periods of remission and relapse, where symptoms may worsen over time. Further, there may be a genetic predisposition to developing these diseases, as they tend to run in families [25], [26], [27], [28], [29], [30]. Another commonality is the alteration of immune pathways, such as the type I interferon pathway [31], [32], [33]. Additionally, some disease groups have similar clinical manifestations. For instance, not only in respiratory diseases is the lung affected but also often in chronic autoimmune diseases, such as SLE and RA [34]. Emerging evidence suggests that these disease groups, and their respective diseases, have molecular patterns in common. For instance, a comparative study of SLE, RA, and MS identified 23 shared susceptibility genes [35].

1.1.2 The potential of multi-omics analyses to find predictors for treatment response

While in the past, it was common to study diseases at one level, such as the genome (the whole genetic information of an organism), the advent of new and more cost-effective high-throughput technologies has made it possible to study diseases on multiple levels simultaneously. The relationship between these different levels (genome, transcriptome, proteome, and metabolome) is shown in Figure 1.7, whereby the suffix "-omics" indicates the comprehensive assessment of the entire set of molecules of one organism at a specific level.

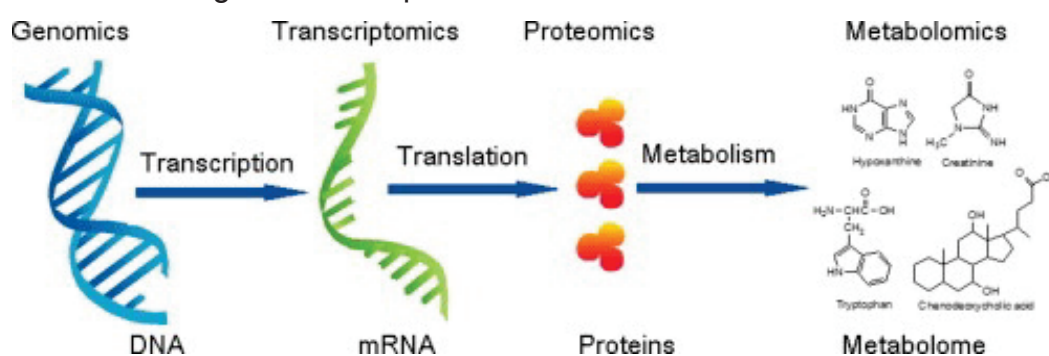


Figure 1.7: Relationship between genomics, transcriptomics, proteomics and metabolomics. From [36].

One of the aspirations of the 3TR project is to identify shared biological pathways across the diseases using multi-omics analyses [37], whereby the best possible outcome would be that changes in these pathways can be used as treatment response predictors. In fact, multi-omics analyses of prospective IBD cohorts are already showing promising results. In IBD patients the early shifts of gene expression and DNA methylation have been identified as predictors for response to anti-TNF treatment [38]

1.1.2.1 Transcriptomics

Transcriptomics is the study of the transcriptome, which is the complete set of RNA molecules in a cell or tissue (Figure 1.8). A highly effective method in that regard is RNA-Sequencing (RNA-Seq). It is a high-throughput technique for quantifying the transcriptome, which allows to obtain the gene expression profile in a sample by counting the number of reads (sequencing fragments) that map to each gene. Traditionally, gene expression is measured as an average across a sample that contains often multiple cell types. This method is known as bulk RNA-Seq. Nowadays there is also single-cell RNA-Seq (scRNA-Seq) available that allows insights into the gene expression profiles of individual cells (Figure 1.9).

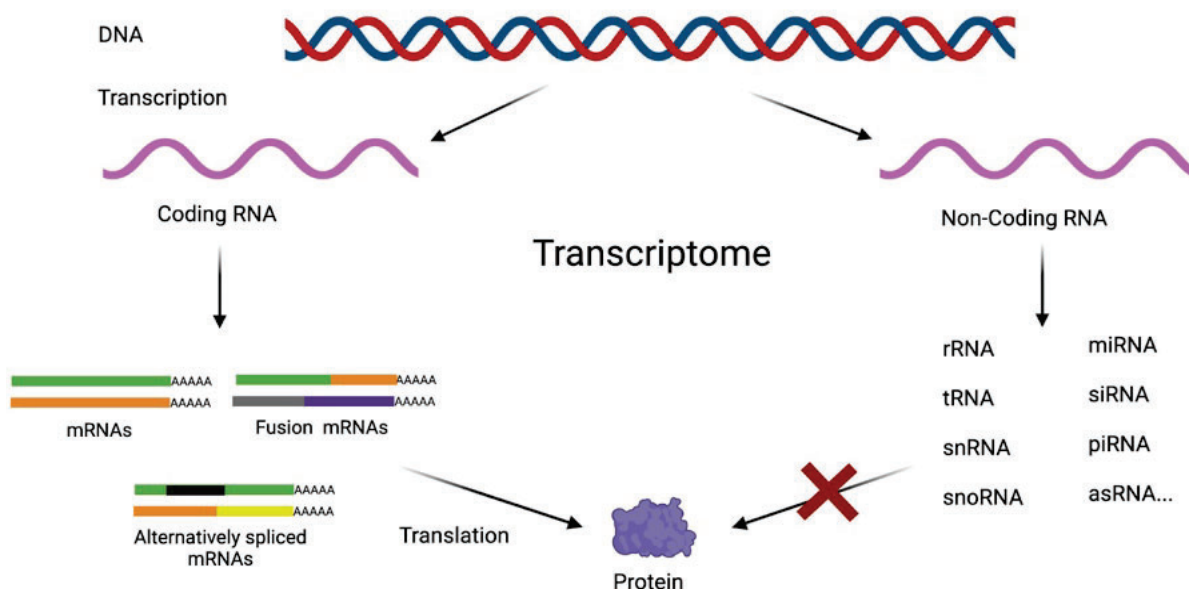


Figure 1.8: Schematic representation of the transcriptome and its components. DNA is transcribed into RNA, which can be classified into coding and non-coding RNAs. Coding RNAs include mRNAs, alternatively spliced mRNAs, and fusion mRNAs, which are eventually translated into proteins. Non-coding RNAs, such as rRNA, tRNAs do not code for proteins but play essential regulatory and structural roles in the cell. From bcgsc.ca/news/gsc-researchers-develop-new-method-single-cell-rna-sequencing.

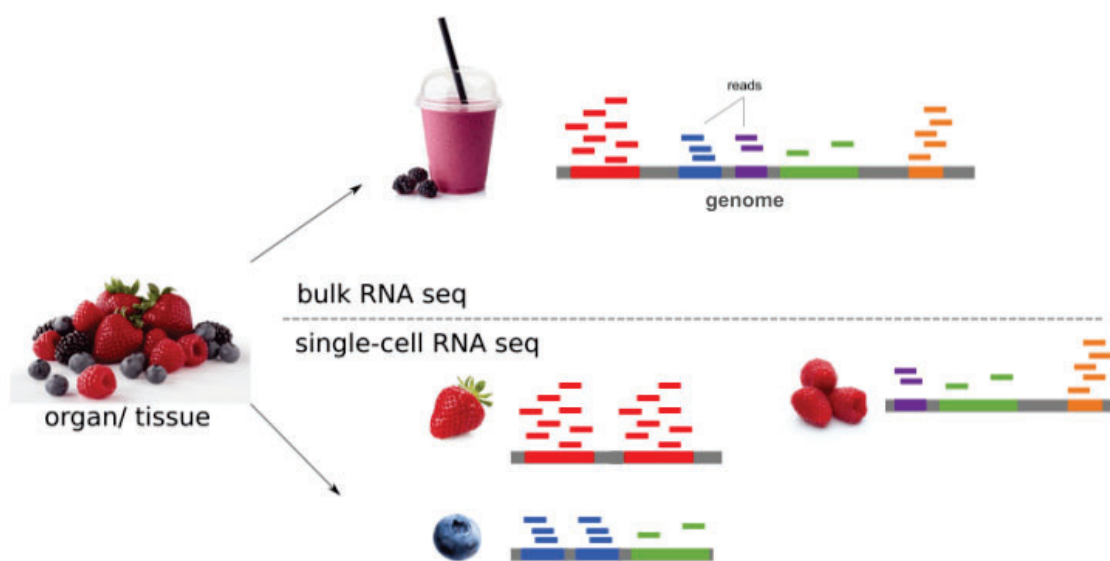


Figure 1.9: Schematic comparison of bulk RNA-Seq and scRNA-Seq. Bulk RNA-Seq is represented by a smoothie made from mixed berries, while scRNA-Seq is symbolised by individual berries. Next to the smoothie, respectively each berry type, is a section of the reference genome depicted and how the reads (coloured bars) are mapped to it. Adapted from [39].

1.1.2.1.1 Bulk RNA-Seq and its relevance in the study of diseases

Bulk RNA-Seq is a high-throughput technique used for quantification of the entire transcriptome of a sample. The process from the biological sample to the expression profile, depicted in Figure 1.10, is as follows: First, the messenger RNA (mRNA) is

isolated and then reverse transcribed into complementary DNA (cDNA). After the amplification of the cDNA the resulting library is sequenced. Finally, the reads are mapped to a reference genome to get the respective gene and its count, namely the expression level.

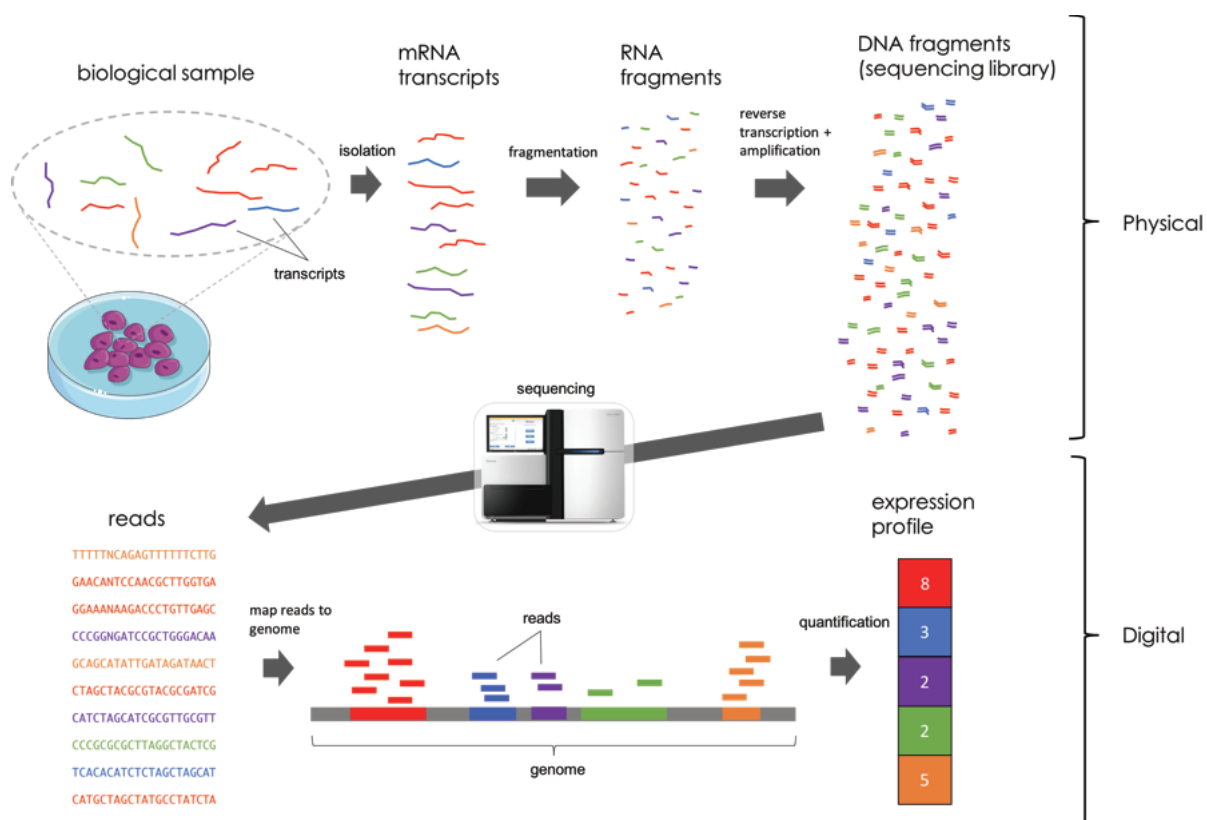


Figure 1.10: Overview of the RNA-Seq workflow, divided into physical and digital processes. The upper section illustrates the physical steps, including isolation, fragmentation, reverse transcription, and amplification. The sequencing step, positioned in the middle, serves as the transition between the physical and digital processes. The lower section represents the digital processes, such as read mapping to the reference genome and quantification of gene expression levels. Figure reproduced from [40] with permission from the author.

Even though in the study of diseases scRNA-Seq is increasingly used [41], it has not fully replaced bulk RNA-Seq. The latter remains relevant for several reasons. Foremost, bulk RNA-Seq is more cost-effective. Since it (a) does not require expensive equipment such as a flow cytometer for cell sorting, (b) the reagents are cheaper and (c) usually its data analysis requires less computing power. Additionally, bulk RNA-Seq is less time-consuming, as it does not require the isolation of individual cells and their subsequent barcoding as is needed for scRNA-Seq. Thus, for instance in an initial screening of a large number of samples in multiple conditions over several time points, bulk RNA-Seq is still the method of choice. Besides the financial aspect, researchers resort to bulk RNA-Seq due to its signal robustness. Since bulk RNA-Seq

tends to smooth out variability in gene expression across cells. Consequently, bulk RNA-Seq is commonly used to validate the results of scRNA-Seq studies by ensuring that the trends in gene expression identified at the single-cell level can also be observed at the tissue/organ level [42]. All the aforementioned reasons underline the continuing relevance of bulk RNA-Seq in the study of diseases. Consequently, bulk RNA-Seq is also used in the 3TR project.

1.1.2.1.2 Single-cell RNA sequencing and its increasing importance in the study of diseases

scRNA-Seq is a technique for sequencing of the RNA molecules in individual cells. In contrast to bulk RNA-Seq, scRNA-Seq provides information about the gene expression levels in each cell and thus can be used to identify different cell types and states within a sample. The actual sequencing process of scRNA-Seq is similar to bulk RNA-Seq. However, the process of isolating the cells and preparing the cDNA library is more complex. The whole process, depicted in Figure 1.11, starts with the isolation of the cells. After having passed the quality control (QC) in form of a cell viability test the individual cells are barcoded, so that the sequencing reads later can be assigned to the respective cell. These so-called GEMs (Gel Bead-In-Emulsions) are then collected and its containing cell lysed to enable the reverse transcription of the mRNA into cDNA. Finally, the oil is removed so the cDNA can be sequenced.

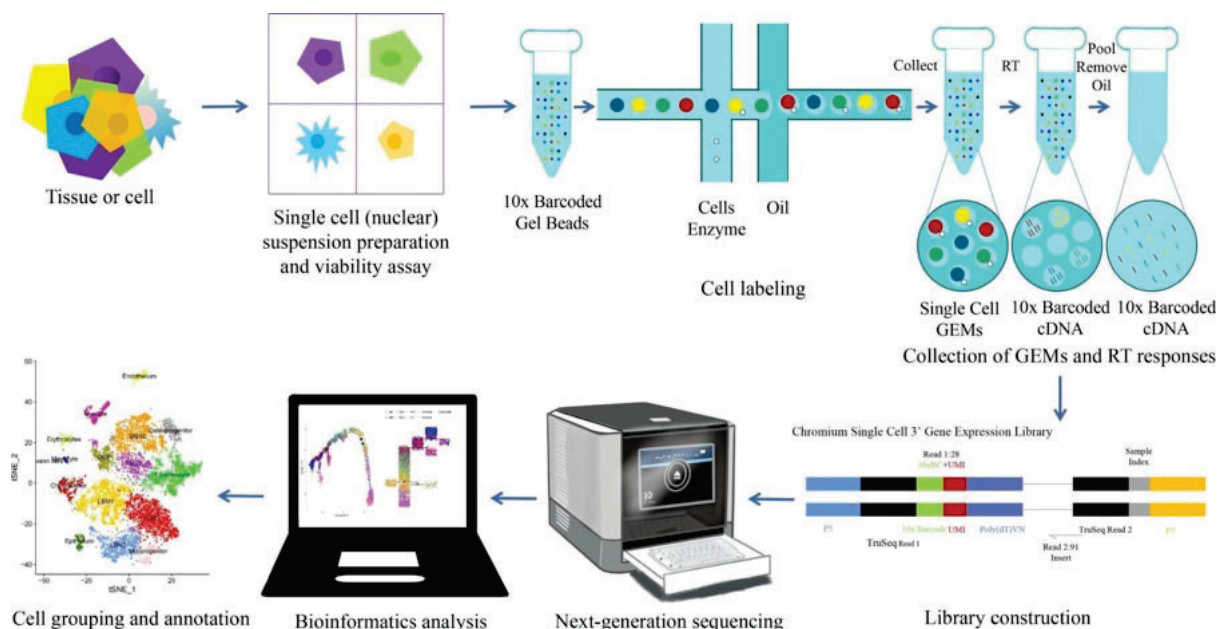


Figure 1.11: Overview of the scRNA-Seq workflow using 10x Genomics technology. From [43].

Soon after its introduction in 2009 [44], scRNA-Seq became a cornerstone in the study of diseases. Especially since it became clear that many diseases, such as cancer, autoimmune disorders, and neurodegenerative conditions, involve significant heterogeneity at the cellular level [45], [46], [47]. Notably, the heterogeneity in the gene expression level can be observed between different cell types and also within the same [29]. In fact, all these insights would not have been possible without scRNA-Seq. Therefore, scRNA-Seq also plays an important role in the 3TR project.

1.1.3 Data sources in the 3TR project and its storage

The data in the 3TR project can be divided into two categories, namely legacy data, collected in earlier clinical trials, and newly produced data from ongoing clinical trials during the project duration. For both cases, the 3TR data warehouse, developed by me during this thesis, is the central hub for storing and sharing the data.

1.1.3.1 Legacy data

In this thesis the term "legacy data" refers to data that has been collected in clinical trials before the start of the 3TR project. The inclusion criteria for the legacy data encompass that it is from patients diagnosed with one of the seven autoimmune diseases targeted in the 3TR project. Thus, functioning as comparative datasets to the newly produced data. At the time of the thesis compilation, data from the following clinical trials was uploaded to the 3TR data warehouse:

- ◆ Study of Mechanisms of Action of Omalizumab in Severe Asthma (SoMOSA)
- ◆ Biologics in Rheumatoid Arthritis Genetics and Genomics Study Syndicate (BRAGGSS)
- ◆ Rheumatoid Arthritis Medication Study (RAMS)
- ◆ Molecular Reclassification to Find Clinically Useful Biomarkers for Systemic Autoimmune Diseases (PRECISESADS)
- ◆ Study to Compare Oral PF-06651600, PF-06700841 and Placebo in Subjects With Moderate to Severe Ulcerative Colitis (VIBRATO)

A tabular summary of the stored datasets is shown in Table 1.2, where one dataset is always a combination of molecular files such as RNA-Seq counts and pheno-clinical data files. In both file types cases the data is stored in tabular data formats such as comma-separated values (CSV) tab-separated values (TSV) format.

Table 1.2: Uploaded legacy data (last update: June 2025)

Clinical study	SoMOSA	BRAGGSS	RAMS	PRECISESADS	VIBRATO
Owner	University of Southampton	University of Manchester		GENyO (FPS)	Pfizer
Disease	Asthma	RA		SLE, RA	UC
Molecular info	Transcriptome				Transcriptome & Proteome
Sample type	Blood				Blood & Tissue
Study type	Longitudinal/Observational				Longitudinal/ Trial
Medications	Omalizumab (Anti-IgE)	Adalimumab (Anti-TNF)	Methotrexate	Multiple	Brepocitinib (TYK2/JAK1)
Individuals	~150	~200	~200	~200	~500
Datasets	1	1	1	8	6

1.1.3.2 Newly produced data

Newly produced data refers to the data collected in the 50 clinical trials that are conducted during the 3TR project. The data is collected from patients diagnosed with one of the seven autoimmune diseases (Section 1.1.1) including both molecular data such as RNA-Seq data as well as pheno-clinical data. The latter was collected in various REDCap (Research Electronic Data Capture) [48] instances. For the role of REDCap in the 3TR project refer to Section 1.3.3.1.

1.1.4 The role of CNAG and myself in the 3TR project

CNAG is acting primarily as the data controller in the 3TR project. Consequently, my main responsibility as part of CNAG was to set up the 3TR data warehouse. Thus, we at CNAG must ensure that the data is stored and shared in a secure and compliant way. In fact, all the data, be it legacy data or newly produced data, is expected to be uploaded via a web interface to the 3TR data warehouse (OmicsDM). Once there, the data needs to be downloadable by the partners of the 3TR project or analysed in the data warehouse itself. See Figure 1.12 for a schematic overview. However, neither CNAG nor I were limited to the data controller role. In fact, CNAG also takes part in the RNA sequencing and the downstream analysis of the RNA-Seq data.

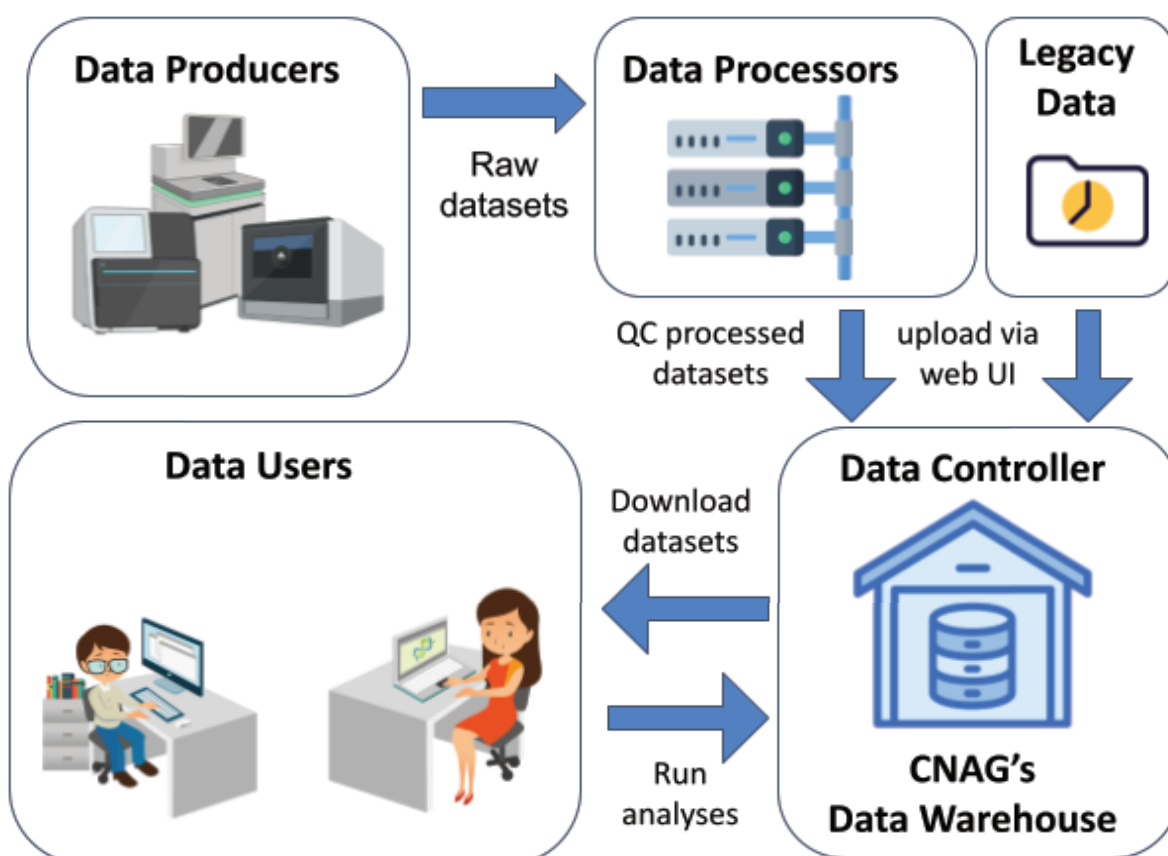


Figure 1.12: Data flow in the 3TR project. Raw datasets generated by data producers are sent to data processors for quality control (QC) and downstream analyses. The processed datasets as well as the legacy datasets are uploaded to the data controller (CNAG's data warehouse) via a web interface. From there, the data user can either download the datasets or run analyses directly in the data warehouse. Icon credits: Sequencing machine: Created in BioRender.com; Server: Freepik; Folder: iconsmind/Freepik; Data warehouse: pojok d/Freepik; Users: Andrawaag/Wikimedia Commons/Public Domain.

1.2 The value of data warehouse solutions in biomedical research

Especially in the context of large consortia projects such as 3TR, data warehouse solutions are essential. Since the partners in such projects could be spread across countries and institutions, shared collaborative platforms are needed. On these, the researchers can safely store, share, visualise, and finally analyse the collected data. Having a shared computational platform could foster collaboration between partners. Since the insights gained from an analysis could be made available to all involved partners instantly. Ideally, the results are shareable not only as a static report but also in an interactive manner inviting the partners to explore the data directly in the user interface. One field where this is considered to be especially beneficial is in the field of scRNA-Seq data analyses, as these datasets can nowadays easily reach the size of several gigabytes (GB) [49] making it challenging for the researchers to visualise them on their local machines.

1.2.1 Centralised vs. federated data warehouse architectures

There are two main architecture types for data warehouse solutions, namely centralised and federated. Their names already hint at the key difference between them. While in a centralised data warehouse all the data is stored in one (central) location, with the federated approach the data remains at its origin (Figure 1.13).

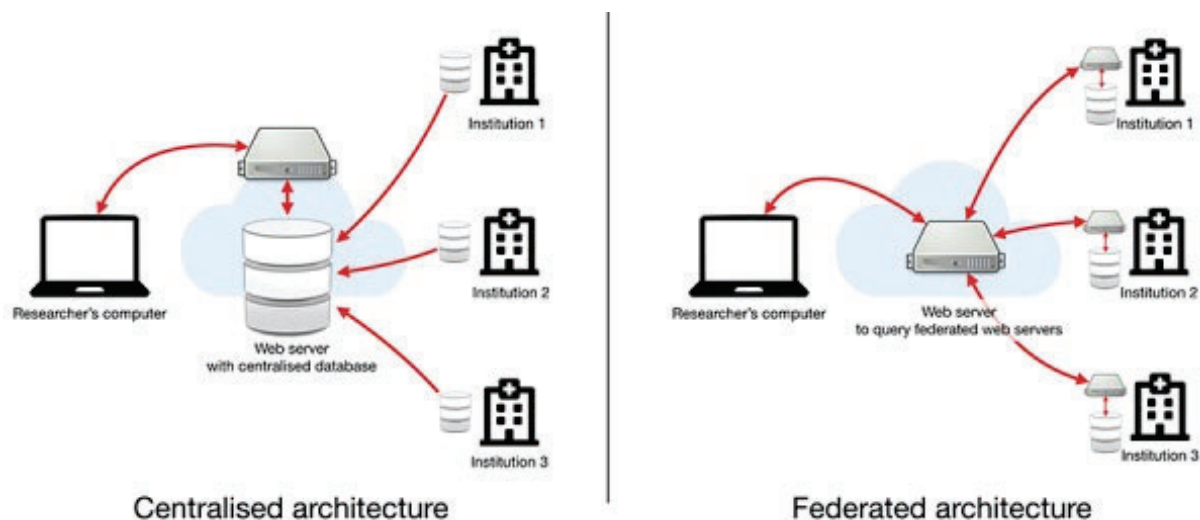


Figure 1.13: Comparison of centralised vs. federated data warehouse architecture. To make the data available to the researchers, in the centralised approach (left) the data from institutes needs to be transferred to one central database/location before it can be queried via a web server. This stands in contrast to the federated architecture (right) where the data remains in the institutes' databases and is accessed via a middleware layer, which needs to be set up at each institute. From [50].

Both architectures have their advantages and disadvantages. In Table 1.3 the two architectures are compared in the context of biomedical research ranging from (biomedical) data specific to technical and operational aspects. Bold fonts indicate where we see an advantage of one approach over the other. Points were extracted by literature research, not limited to scientific publications but also including insights from other sources such as the "DataSharingPlaybook"², created by the Innovative Health Initiative (IHI) and European Federation of Pharmaceutical Industries and Associations (EFPIA). Table 1.3 is not an exhaustive list but rather a selection of the most relevant aspects in the context of the 3TR project and thus for this thesis. As can be seen just by counting the occurrence of entries in bold font, there is no clear winner between a centralised and a federated architecture. In fact, this is also stressed by the "DataSharingPlaybook", as a clear recommendation for one of the two architectures is not expressed. Nevertheless, federated data warehouse solutions seem to gain more traction recently. Especially since raising privacy awareness and data protection

² ihi.europa.eu/sites/default/files/uploads/Documents/ProjectResources/IMI_IHI_DataSharingPlayBook_2024.pdf

regulations such as the General Data Protection Regulation (GDPR) poses additional legal difficulties potentially complicating envisioned analyses [51]. Consequently, keeping the data at its origin and executing the analyses on the data owner's side while only sending aggregated results to the researcher, circumvents these issues [52].

Table 1.3: Comparison of centralised vs. federated data warehouse architecture

(bold font indicates where we see an advantage of one approach over the other)

Category	Aspect	Centralised	Federated
Biomedical data	Integration	Centralised standardisation ensures data compatibility.	Potential incompatible data structures and formats.
	Privacy	Requires careful access controls, as all data is stored in one location.	Data owners retain full autonomy and responsibility.
Technical	Query performance	Fast, owing to single-location storage.	Potential query delays due to data fetching from external sources.
	Scalability	Can be challenging with large data volumes.	Easy incorporation of new data sources without migration.
	Data loss risk	Vulnerable to single point of failure [53].	Vulnerability would not affect all data sources.
Operational	Effort from the data providers	May face lengthy upload processes for large datasets.	Minimal upfront effort, as data remains in original locations.
	Maintenance	Centralised software updates.	Middleware layer adds complexity and requires technical expertise on the data provider side.
	Long-term support	Only one central database/web server needs to be maintained.	Requires the commitment from the data owners.

1.3 Pheno-clinical data, crucial for future discoveries but difficult to handle

1.3.1 Definition of pheno-clinical data

Pheno-clinical is a combination of the terms phenotypic and clinical data. Phenotypic data refers to the observable characteristics of an individual. These variables are often harmonised by using standardised vocabularies such as the Human Phenotype Ontology (HPO) [54]. Clinical data, on the other hand, relates to medical information collected during healthcare delivery, i.e., a clinical trial in the context of the 3TR project. These data could for instance include the patient's medications or results from medical tests. Its values can be also standardised by using vocabularies such as LOINC® (Logical Observation Identifiers Names and Codes) [55], a universal standard for identifying medical laboratory observations.

1.3.2 The importance of pheno-clinical data

Pheno-clinical data is crucial for advancing biomedical research and thus ultimately improving patient care. In fact, integrating pheno-clinical data with molecular data has led to many important breakthroughs such as being able to associate a genetic variant with a specific disease. Two important techniques in that context are Genome-Wide Association Studies (GWAS) [56] and Phenome-Wide Association Studies (PheWAS) [57]. Furthermore, pheno-clinical data is paramount for precision medicine. By assessing detailed phenotypic data in combination with molecular data, researchers can identify subgroups of patients that are more likely to respond to a particular treatment. This approach has the potential to revolutionise patient care by enabling the development of personalised treatment plans based on the individual patient's characteristics [58]. Moreover, pheno-clinical data facilitates the identification of new biomarkers for disease diagnosis, prognosis, and treatment response [59]. The latter is important in the context of the 3TR project since the main goal is to identify the molecular mechanisms of response and non-response to treatments in seven autoimmune diseases.

1.3.3 Pheno-clinical data collection and how to potentially overcome its challenges

The pheno-clinical data is typically collected via questionnaires conducted by the clinicians during the patients' visits. However, getting all the required information can be challenging for several reasons. First, the potential unwillingness of the patient to allow the sharing of the data for secondary use needs to be overcome [60]. Then, the time constraints of the clinicians need to be considered [61]. Further, the lack of digitalisation is known to be a reason for missing or inaccurate data [62]. Of course, the aforementioned challenges are not exhaustive but give a good overview of the main difficulties to be addressed. While the patients' concerns regarding the data sharing need to be discussed on a case-by-case basis the other challenges could be addressed by a wide adoption of electronic case report forms (eCRFs) [63]. To this end, over the years, more electronic data capture (EDC) software has been released. In fact, G2, a software review platform, lists over 100 EDC systems. Among these REDCap [48] is rated as one of the best free EDC applications (g2.com/categories/electronic-data-capture-edc). Consequently, in the 3TR project most eCRFs are generated via REDCap.

1.3.3.1 REDCap and its role in the 3TR project

REDCap is a secure web application for designing and managing online surveys and databases. It was developed at Vanderbilt University in 2004 as a collection tool for clinical researchers. It quickly gained traction in the biomedical community due to its ease of use and flexibility. An example of the REDCap interface is shown in Figure 1.14.

Another factor explaining its popularity is that REDCap is free to use for non-profit organisations. However, it is important to note that the software is not open source. So, to enable other organisations to extend the software's functionalities without the need to make the codebase public they formed a consortium in 2006 [64]. Another important aspect is REDCap's compliance with several regulations such as the Health Insurance Portability and Accountability Act (HIPAA) as well as GDPR. The latter is especially important in the context of the 3TR project since the patients are recruited across 15 European countries. However, it must be noted that there is not one REDCap instance for all the recruitment centres but multiple different ones. This is not only because of differing variables specific to the diseases but also due to more strict data sharing

regulations in some countries. This multitude of REDCap instances implied for the goals of the 3TR project and consequently for this thesis that a solution was needed to convert the data from the different REDCap instances into a common format before being able to do cross-study/disease analyses.

2. Smart Pump Patients/Meds

[Modify this instrument](#)

[VIDEO: Basic data entry \(16 min\)](#)

[Download PDF of](#)
- select PDF download option -

Re-assign this record to another Data Access Group? SiteH

[Editing existing Patient ID 1310-23](#)

Event Name: Med 03

Patient ID	1310-23
Pump type	☒ General infusion pump ☐ PCA ☐ Syringe reset
Primary/secondary	☒ Primary ☐ Secondary reset
IV fluids	Lactated Ringers
Drug	
Dose	
Rate	75
Current time	02-06-2013 15:00 Now M-D-Y H:M
Administration start time(current bag/syringe)	02-06-2013 09:41 Now M-D-Y H:M
Smartpump/IV infusing was used	☐ Yes ☒ No reset
Drug dictionary used	☐ Yes ☒ No ☐ Unknown reset
Is there a drug library for this?	☒ Yes ☐ No reset

Figure 1.14: Screenshot of the REDCap interface. Shown is the data entry for a patient treated with an infusion pump. From [65].

1.3.4 The importance of standardised pheno-clinical data storage

Even if the data is obtained in a structured way, be it with online surveys or questionnaires software such as REDCap, it is not granted that the data can be easily shared or analysed. Owing to the lack of an established data model in REDCap projects, some form of ETL (Extract, Transform, Load) processing must be performed before beginning the actual research. In that regard there are efforts underway in the biomedical community to standardise the storage of pheno-clinical data for secondary use. Two well-known initiatives working in that direction are the GA4GH [66] and the Observational Health Data Sciences and Informatics (OHDSI) [67].

1.3.4.1 Global Alliance for Genomics and Health standards

GA4GH is an international not-for-profit alliance of organisations that are working together to create standards and frameworks for the secure and responsible sharing of genomic and health-related data [68]. Two widely used and approved standards by the GA4GH steering committee are Phenopackets v2 [69] and Beacon v2 [70].

1.3.4.1.1 Phenopackets

Phenopackets is a data model for the structured storage and sharing of phenotypic data. Although the data schema is defined leveraging Protobuf, Phenopackets supports YAML and JavaScript Object Notation (JSON) as data exchange formats. While the initial version was tailored to the needs of the rare disease research community, the latest version (v2) is designed to also accommodate cancer and common diseases³, such as the autoimmune diseases targeted in the 3TR project. Thus, throughout this thesis, when Phenopackets is mentioned, I am referring to the Phenopackets v2 standard (Figure 1.15).

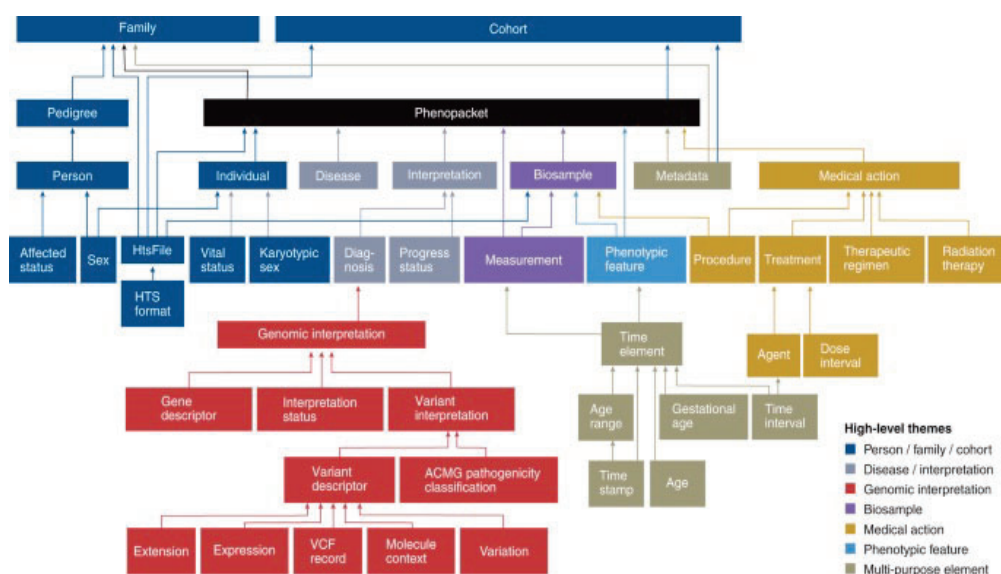


Figure 1.15: Overview of the Phenopackets v2 data model. Illustrating the hierarchical structure of the data model and the relationships between its elements. The colouring represents specific high-level themes, for example Genomic interpretation. Longform of the abbreviations: ACMG=American College of Medical Genetics; HTS=high-throughput screening; VCF=variant call format. From [71].

1.3.4.1.2 Beacon v2, Beacon Models and Beacon friendly format

In contrast to Phenopackets, Beacon v2 is an Application Programming Interface (API) specification for querying genomic and clinical data repositories (Figure 1.16). The API responses (in JSON) organise biological data according to the Beacon data models.

³ ga4gh.org/news_item/phenopackets-v2-expands-utility-to-provide-a-more-complete-medical-picture

Beacon Models are composed of seven entities. The relationships among them are shown in Figure 1.17. The only entity of interest for this thesis is *Individuals* (coloured in green in Figure 1.17), which contains pheno-clinical data for healthy controls and patients. In practical terms, many users generate the API response by querying preformatted data, structured according to Beacon v2 Models. This data originates from data exchange files called the Beacon Friendly Format (BFF), which are loaded as collections into a non-relational database, such as MongoDB [70].

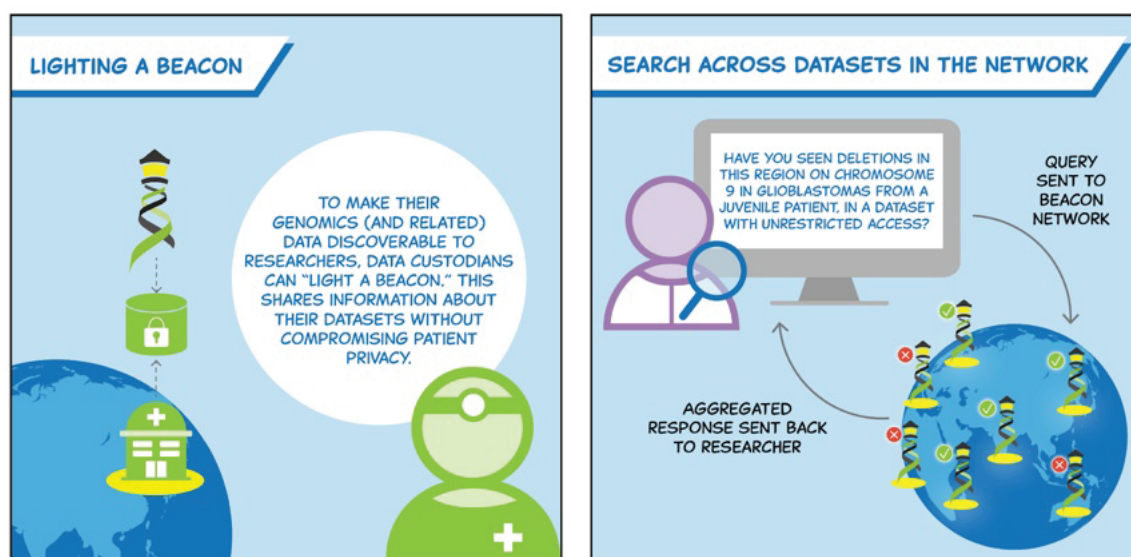


Figure 1.16: Illustration of the Beacon network process for genomic data discovery. On the left, the concept of "lighting a beacon", a metaphor for implementing the Beacon API is shown. This enables data custodians to make their genomic datasets discoverable to researchers while protecting patient privacy. The right panel demonstrates how researchers can query the Beacon network, i.e., all connected data repositories, to search for a patient with a specific allele or phenotype. The aggregated response, indicating one or multiple patients/datasets with the queried criteria (checkmark), or none (cross), and additional information such as the data access policy, is then sent back to the researcher. From ga4gh.org/product/beacon-api.

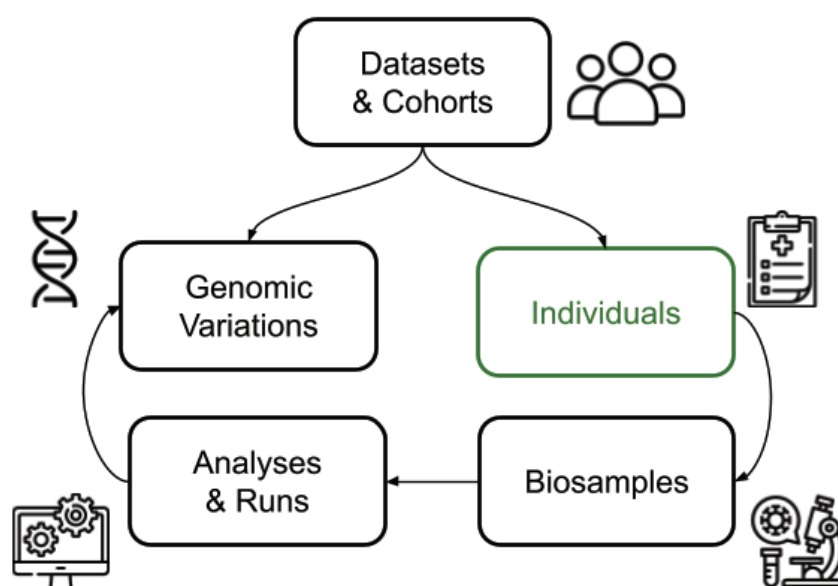


Figure 1.17: Diagram displaying the seven entities of the Beacon v2 Models. The arrows indicate the relationships between the entities. *Individuals* is coloured green to highlight its importance in this thesis. Adapted from [72].

1.4 Publications

Manuel Rueda, **Ivo C. Leist**, Ivo G. Gut; Convert-Pheno: A software toolkit for the interconversion of standard data models for phenotypic data. J Biomed Inform. 2023 Nov 28;104558. DOI: [10.1016/j.jbi.2023.104558](https://doi.org/10.1016/j.jbi.2023.104558). PMID: 38035971.

Ivo C. Leist, María Rivas-Torrubia, Marta E. Alarcón-Riquelme, Guillermo Barturen, PRECISESADS Clinical Consortium, Ivo G. Gut, Manuel Rueda; Pheno-Ranker: A Toolkit for Comparison of Phenotypic Data Stored in GA4GH Standards and Beyond. BMC Bioinformatics. DOI: [10.1186/s12859-024-05993-2](https://doi.org/10.1186/s12859-024-05993-2).

Chapter 2: Objectives

- a. Compare and test existing data warehouse solutions for biological data.
- b. Based on the results of Objective (a) either develop a new data warehouse tool or adapt an existing one to 3TR's needs.
- c. Allow the upload of large files to the data warehouse preferably directly from the web interface.
- d. Make the data warehouse accessible to partners within the 3TR consortium so they can upload and share their data.
- e. Explore and assess existing tools for single-cell data visualisation and integrate the most promising ones into the data warehouse.
- f. Allow analyses directly from the data warehouse.
- g. Assess existing tools which could aid the building of cohorts of patients across studies and diseases or develop new tools for this purpose. It would be ideal if the tool could be integrated into the data warehouse.
- h. Demonstrate the usefulness of the data warehouse in concert with externally or in-house developed auxiliary tools.

In this PhD project a data warehouse is set up for the 3TR project taking into account the successful model from the RD-Connect Genome-Phenome Analysis Platform (GPAP) [73]. Before starting the development of the data warehouse, I compared and tested existing data warehouse solutions for biological data (Objective a) to check if a solution was already available that could be adapted to the 3TR project. Because no suitable solution was found, I developed a new centralised data warehouse tool tailored to the needs of the 3TR project (Objective b). This involved the development of a user-friendly interface, as well as the provision of training and support to help users get the most out of the platform. The data warehouse allows the upload of large files directly from the web interface (Objective c). The latter is crucial since it is expected that not all the data uploaders will have the technical skills to upload data via an API, Web Distributed Authoring and Versioning (WebDAV) or similar.

The data warehouse is hosted at CNAG and is accessible to partners within the 3TR consortium so they can upload and share their data (Objective d). Notably, the

provision of fine-grained access is of utmost importance in the context of legacy data because depending on the data owner the data cannot be shared with all the partners. Moreover, it is possible for users to visualise specific data directly from the data warehouse (Objective e). Here the focus is on single-cell data visualisation since the growing size of single cell datasets makes it increasingly difficult to visualise them on a local machine. Another important building block of the data warehouse is an analysis module (Objective f) coming with a set of predefined pipelines for the analyses commonly run to save time and to avoid redundant processing steps for each analysis. Finally, the platform not only allows downloading of results but also its interactive exploration within the web interface. The initial use case is transcriptomic analyses because the first datasets produced by the 3TR consortium were RNA-Seq data.

Additionally, new software toolkits are introduced which could aid the building of cohorts of patients across studies and diseases (Objective g). The challenges here are two-fold. First, the pheno-clinical data comes in various formats. Thus, 3TR needs a method to convert the pheno-clinical data of the different studies into one common format. And second a method/tool is required to compare the patients' pheno-clinical data within one study and across studies. This information is needed for creating cohorts of phenotypically similar patients to jointly analyse their molecular data. Lastly, it is assessed how the data warehouse and its auxiliary tools handle the submitted data, analyse it and if this done in a user-friendly way (Objective h).

Chapter 3: Materials and methods

3.1 3TR project data

The 3TR project data is categorised into two main types: molecular and pheno-clinical data. Molecular data encompasses information such as the transcriptome or proteome, while pheno-clinical data refers to the structured storage of phenotypic and clinical information. A formal definition of pheno-clinical data is provided in Section 1.3.1. Both the molecular data together with the pheno-clinical data are crucial for understanding the underlying mechanisms of the seven diseases targeted in the 3TR project (Section 1.1.1) and the potential development of new treatments.

3.1.1 Molecular data

In the 3TR data warehouse many different omics data types have been stored such as proteomics, epigenomics and transcriptomics. To the latter belongs bulk RNA-Seq (Section 1.1.2.1.1) of whole blood samples collected from patients participating in 3TR's prospective studies. The sequencing was performed at CNAG. The resulting data was processed by the 3TR partner GENyO centre at the Fundación Pública Andaluza Progreso y Salud (FPS) in Granada, Spain. The quality controlled results (gene expression count matrices) were uploaded in the form of tabular data files. In addition members of GENyO provided us with example scRNA-Seq data for testing the single-cell data visualisation module of OmicsDM. We asked them to provide the scRNA-Seq data in Annotated Data (AnnData) format due to its compatibility with single-cell visualisation tools such as Chan Zuckerberg (CZ) CELL×GENE Annotate [74], utilised within OmicsDM's visualisation module. The AnnData format was originally developed for the Scanpy (Single-cell analysis in Python) toolkit [75], but has since matured to a standalone Python package, called anndata [76]. The AnnData format is specifically designed to store both gene expression data and associated annotations. Table 3.1 summarises the attributes of AnnData objects with descriptions and examples, while Figure 3.1 provides a schematic representation.

Table 3.1: AnnData object attributes

Attribute	Description	Dimensionality	Example
X	Primary data matrix	Multidimensional	Gene expression counts (raw; normalised)
obs	Observations metadata	One-dimensional	Cell type
var	Variables metadata	One-dimensional	Gene name
obsm	Observation annotations	Multidimensional	PCA coordinates
varm	Variable annotations	Multidimensional	PCA loadings
obsp	Pairwise relationships between observations	Stack of 2D matrices	Cell-cell similarity
varp	Pairwise relationships between variables	Stack of 2D matrices	Gene-gene correlation
uns	Unstructured annotations for experimental or analysis parameters	Not applicable	Dataset information; cluster colours

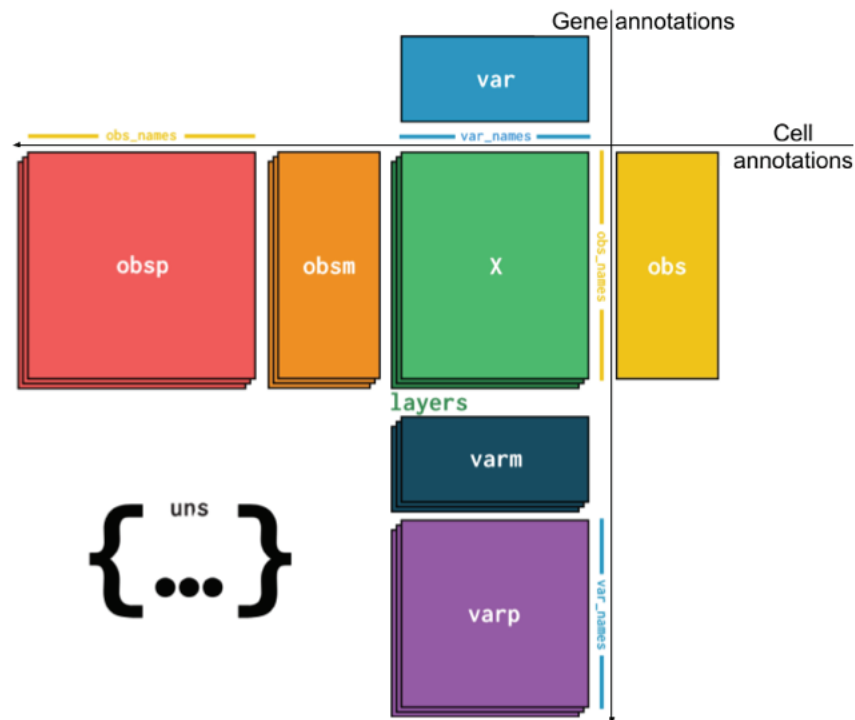


Figure 3.1: Schematic representation of the AnnData data structure. The matrix X stores the gene expression data, in which rows represent the cells and columns the genes. Extra metadata about the cells and genes is stored in obs, respectively var. Additional information is stored in their corresponding attributes ending with m, respectively p, whereby the suffix m indicates multidimensional annotations and p a collection of pairwise relationships. The multi-layered structure of AnnData represented by the matrices being stacked. For querying the annotation of a specific cell (obs) or genes (var) across all layers the attribute with the suffix _names is leveraged. Apart from the others is the uns attribute, which is neither in a matrix format nor directly associated to the cells or genes. Adapted from [76].

The AnnData object is commonly stored on-disk in the H5AD format, a file format optimised for storing single-cell gene expression data. Since it is based on HDF5 (Hierarchical Data Format version 5) it not only can be read by the anndata Python package but other HDF5 compatible tools as well. Alternatively, an AnnData object can also be written to the disk in Zarr format, which facilitates storage of large arrays in a compressed way. While both formats enable efficient access to the data and metadata of the experiments CZ CELL×GENE Annotate expects the data to be stored in the H5AD format. Thus, the 3TR partners are asked to provide the data in the H5AD format as well.

3.1.2 Pheno-clinical data

The pheno-clinical data was collected and stored by the 3TR data producers in REDCap instances (Section 1.3.3.1). To store the data in OmicsDM and use it for analyses several steps were required (Figure 3.2). First the data needed to be exported by the data producers in two different formats: (i) CSV files together with their corresponding data dictionaries and (ii) CSV files with labels, as explained in OmicsDM's online documentation: cnag-biomedical-informatics.github.io/omicsdm-documentation. The exported CSVs could not be utilised for any analyses right away. Experts of each of the diseases performed a quality control (QC) such as checking and addressing potential missing values in the CSV files with labels. In parallel, members of the Biomedical Genomics group at CNAG harmonised the CSV files into GA4GH standards such as PXF and BFF. To facilitate this process, we developed a software toolkit for the interconversion of standard data models for phenotypic data, named Convert-Pheno [72]. To use it, members of the Biomedical Genomics laboratory first had to manually create a mapping file based on the CSV file and their corresponding data dictionary for each REDCap instance. The mapping file together with the data dictionary- and the CSV file then served as input for Convert-Pheno. The resulting PXF and BFF files were uploaded to OmicsDM, to potentially enable future cross-disease analyses.

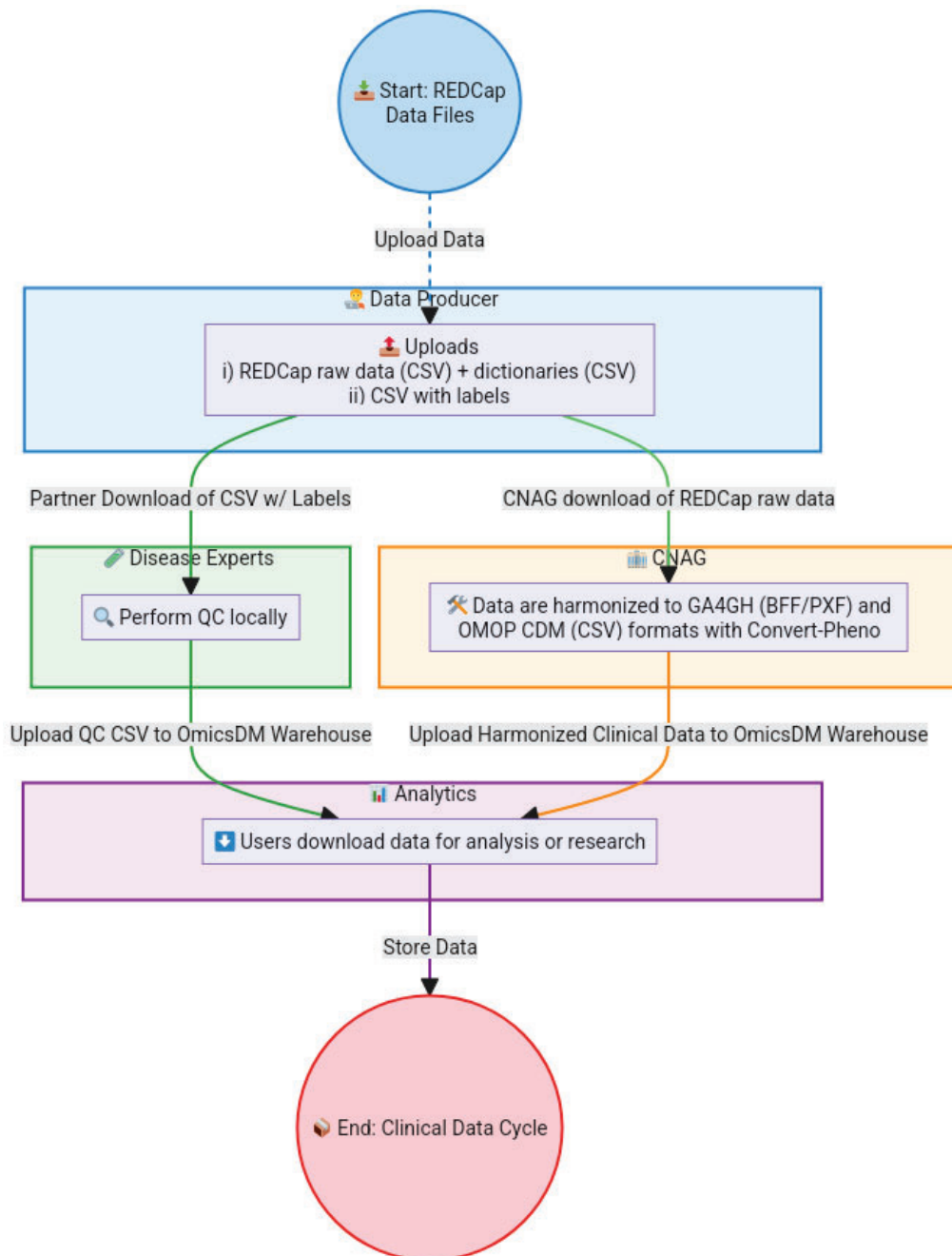


Figure 3.2: Clinical data processing and harmonisation workflow. It starts with REDCap data files, which are uploaded by the data producer in the form of (i) CSV files along its corresponding data dictionaries and (ii) as CSV files with fields labeled. The latter are downloaded by disease experts for QC, while raw data and dictionaries are downloaded by CNAG for harmonisation to GA4GH standards using Convert-Pheno. The resulting files are uploaded to OmicsDM, from which its users can download them for analytical purposes. The produced analyses results are then uploaded by the data analysts to OmicsDM to share their findings with the other partners. Adapted from cnag-biomedical-informatics.github.io/omicsdm-documentation.

3.2 Data warehouse requirements

The requirements for the data warehouse were collected from the 3TR partners as well from the scientists at CNAG in an iterative process. While the data warehouse should enable the structured storage of molecular and pheno-clinical data, its data schema should not be too rigid. In fact, a certain degree of flexibility is needed to accommodate evolving requirements, as for instance new data types such as imaging files. To quickly adapt to new demands the data schema should be configurable by a controlled vocabulary. From the users' perspective the data warehouse should come with a web-based interface enabling the 3TR partners to (a) up- and download data, (b) visualise molecular data such as scRNA-Seq data, and (c) perform basic analyses.

3.3 Explored existing data warehouse solutions in the biomedical research community

There are several data warehouse solutions available in the biomedical research community such as tranSMART [77], Opal [78], MOLGENIS [79], SODAR (System for Omics Data Access and Retrieval) [80], Terra (app.terra.bio). To avoid reinventing the wheel I tested several of these existing data warehouse solutions (Table 3.2) to check their compatibility with the 3TR project. tranSMART, Opal, and MOLGENIS were assessed most thoroughly as potential solutions for the 3TR project before the decision was made to develop the OmicsDM platform.

Table 3.2: Tested data warehouse solutions

Name	Focus	Assessed by	GitHub repository	Tested version
Opal	Biobanks	Self-hosting	obiba/opal	5.1.2
tranSMART	Not specified	Self-hosting	transmart/ transmartApp	v19-1
SODAR	Multi-omics	Self-hosting	bihealth/ sodar-server	v0.14.2
MOLGENIS EMX2	Not specified	Self-hosting	molgenis/ molgenis-emx2	v10.77.2
Terra	Not specified	Documentation	DataBiosphere/ terra-ui	May31, 2024

Name	Focus	Assessed by	GitHub repository	Tested version
CZ CELL×GENE Discover	scRNA-Seq	Self-hosting	chanzuckerberg/single-cell-data-portal	sha-8f31abd4
CellDepot	scRNA-Seq	Playground	interactivereport/CellDepot	v1.1
Single Cell Portal	scRNA-Seq	Playground	broadinstitute/single_cell_portal	0.2.2

3.3.1 tranSMART

tranSMART is an open-source data integration, sharing, and analysis platform for translational research. It was originally developed by Johnson & Johnson and later released as an open-source project [81]. Its development was continued by the tranSMART Foundation and later by the i2b2 tranSMART Foundation. The latter was a result of the merge with the informatics for Integrating Biology and the Bedside (ib2b2), a US-American centre for biomedical computing (i2b2.org/about). Over the years tranSMART has become a widely used platform in the biomedical research community with over 100 organisations ranging from academic institutions to pharmaceutical companies (i2b2transmart.org/home/our-history). Additionally, it has proven to be effective for many different diseases [82] and its users were largely satisfied with its usability [83]. tranSMART's architecture is shown in Figure 3.3 (left).

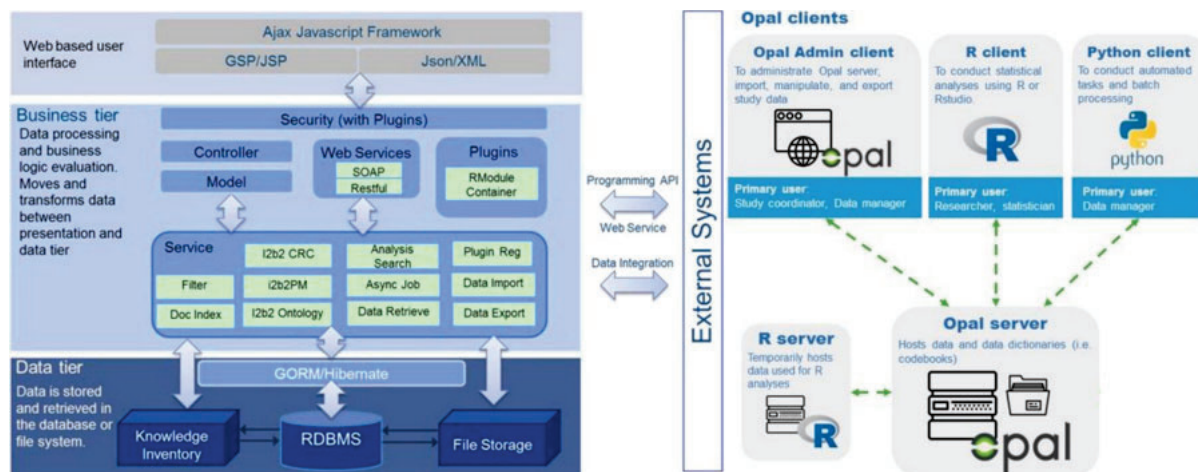


Figure 3.3: Architecture of the existing data warehouse solutions tranSMART (left) and Opal (right). Adapted from [77], [78].

3.3.2 Opal

Opal is a data warehouse solution that was originally developed as one part of the Open Source Software for BioBanks (OBiBa) suite during the Public Population Project in Genomics (P3G) project (p3gobservatory.org/scientificAction.do). It is designed to store and manage epidemiological data and has been used in many different studies and projects such as BioSHaRE [84], EUCAN-Connect (eucanconnect.eu) and the EU Child Cohort Network [85].

From the architectural point of view, Opal is a web-based application that consists of several components. In the centre is the Opal Server, the host and connection point for the data. Where the data can be either stored in the form of database entries or as a flat file. Further, it makes the data accessible to Rock, OBiBa's R server implementation. Opal users can connect to Opal depending on their role via three different clients. The data managers can administrate Opal, import, and export data either by the Opal Admin client, a web-based interface, or by the Python client. For the data analysts, there is an R client. All of Opal's building blocks are shown in Figure 3.3 (right).

3.3.3 MOLGENIS

MOLGENIS (Molecular Genetics Information System), in contrast to Opal and tranSMART, is not a standalone data warehouse tailored for specific use cases but rather a software generator for biomedical data warehouses. It was initially developed at the University Medical Center Groningen (UMCG) and the University of Amsterdam for molecular genetics data. However, it has since then found its use in many different scientific domains such as biobanking [86] and rare diseases research [87]. The main goal of MOLGENIS is to enable non-bioinformaticians to build their own data warehouses without the need for programming skills. In Figure 3.4 is shown how based on a domain-specific language (DSL) MOLGENIS is able to generate a web-based application.

Over the years the toolkit became more sophisticated and thus no longer everything is set up by modifying the FreeMarker templates. For instance, the menu can now be customised via drag and drop. However, the most important change is the introduction of the Entity Model Extensible (EMX) data model. Thus, the data model no longer needs to be defined in the DSL but can be generated from a Microsoft Excel or CSV

file. The data recently has been succeeded by the EMX2 data model [88] to further simplify the data model generation. To increase the visibility of this change the authors have renamed MOLGENIS data platform to MOLGENIS EMX2 (github.com/molgenis/molgenis-emx2) [89]. It is the newest version of the MOLGENIS data platform. As already indicated by the name, the most important change in this version is the switch to the EMX2 data model, as introduced in the previous section. On top of that it comes with more advanced UI components, data up-/download functionalities and a GraphQL API. Example applications, as for instance the usage of MOLGENIS EMX2 for the generation of a metadata catalogue in the LongITools (longitools.org) project, can be explored at emx2.dev.molgenis.org/apps/central/#.

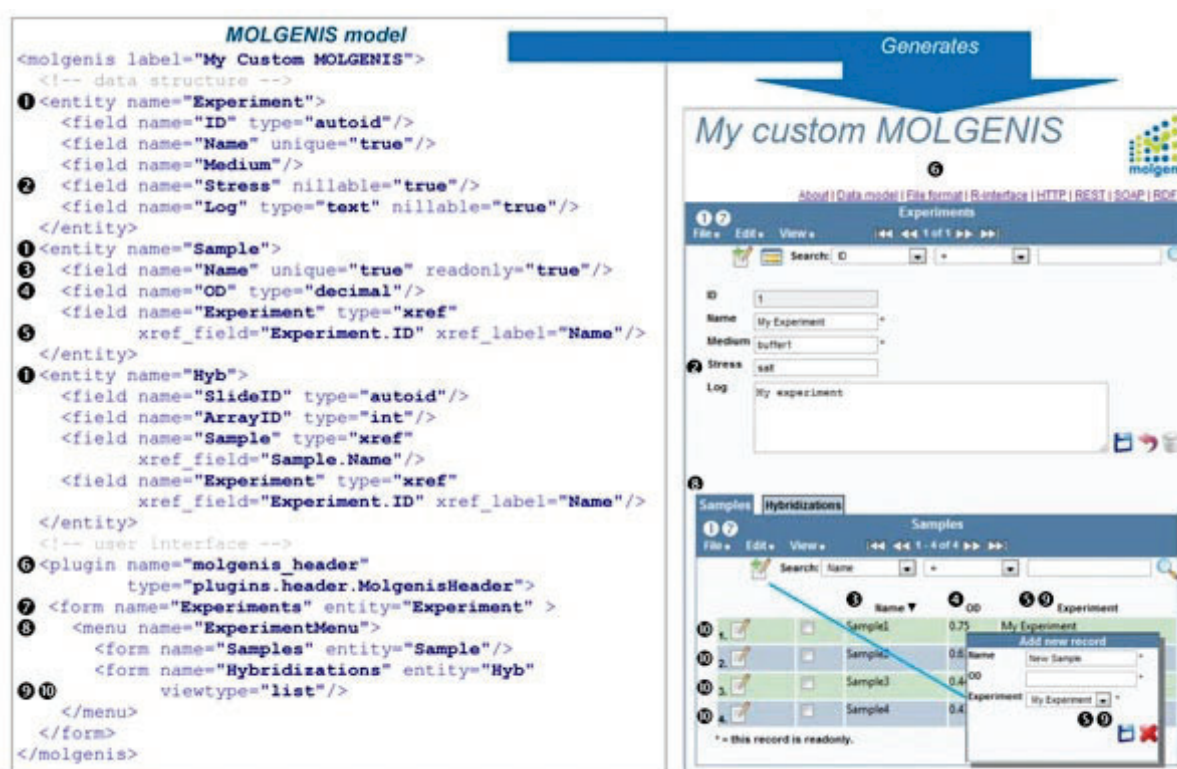


Figure 3.4: MOLGENIS from Domain Specific Language (DSL) to application. From [90].

3.4 Proposed solution: Development of a new data warehouse solution tailored to the 3TR project

As shown in the applicability testing of existing data warehouse solutions presented in Section 4.1.2, none fully met the requirements of the 3TR project. Therefore, a novel data warehouse solution tailored to the 3TR project, named OmicsDM, is proposed here. OmicsDM is a modular software suite to upload, share, visualise biological datasets and run analyses. The main goal of the data warehouse is to provide a secure

and scalable environment for the 3TR partners to store and share their data. Besides these core functionalities the data warehouse should also provide a user-friendly interface to run analyses or visualise single-cell data.

3.4.1 Data storage

Database

A database was needed to store various types of information about the uploaded files, such as the ownership or the experiments metadata and most importantly with which partner which files are shared. As our database management system we chose PostgreSQL ([postgresql.org](https://www.postgresql.org)). As implied by its name, in the past PostgreSQL was a purely relational Structured Query Language (SQL) database. However, with the introduction of the JSON and the JSONB column type PostgreSQL gained “not only SQL” (NoSQL) capabilities. The latter played a role in the decision for employing PostgreSQL in OmicsDM, since for instance the storage in key-value pairs could facilitate the potential utilisation of the developed data warehouse solution beyond the 3TR project.

S3 compatible storage

For the storage of the uploaded files in OmicsDM in a secure and scalable way, MinIO (min.io) was used. It is an open-source object storage server that is compatible with the Amazon S3 protocol, which utilises unique identifiers to store and retrieve data. To separate the data of different applications the files are stored in S3 buckets, accessible via a private or public Uniform Resource Locator (URL). The buckets' content can be accessed via keys, serving as unique identifiers for stored objects. Thus, S3 storage does not work like a traditional file system. To address the lack of hierarchy, it is common practice to simulate a folder structure by using forward slashes in the keys. Thus, a full URL to access an object in an S3 bucket could look as follows: https://s3.example.com/bucket_name/folder/subfolder/file.csv.

3.4.2 Client and server

The client of OmicsDM was implemented using React, a JavaScript library for building user interfaces. The React frontend consists of several views implemented with Material UI (mui.com), a library that provides a set of reusable components for building user interfaces. For the file uploads, Evaporate JS (github.com/TTLabs/EvaporateJS), a JavaScript library that is designed to simplify the process of uploading large files directly to an S3 compatible storage, is used. With this setup, the client only needs to interact with OmicsDM's underlying server to obtain a presigned URL for each file or chunk being uploaded, after which the client uploads the data directly to the S3 storage (Figure 3.5).

OmicsDM's server is powered by Flask, a lightweight general-purpose Python web framework. To allow the server to communicate with the client we decided to implement a Representational State Transfer (REST) API, which provides a standardised interface for exchanging JSON formatted data via Hypertext Transfer Protocol (HTTP) methods (Figure 3.6) in a stateless manner. The latter mentioned characteristic was especially important for our decision to adopt the REST architecture because not requiring to track session state greatly reduces potential server-side complexity. To simplify the API implementation we took advantage of the community maintained Flask extension Flask-RESTX (github.com/python-restx/flask-restx), which facilitates building REST APIs. As depicted in Figure 3.6 another noteworthy dependency in our server stack is SQLAlchemy (sqlalchemy.org), since its object-relational mapper (ORM) prevents SQL injections.

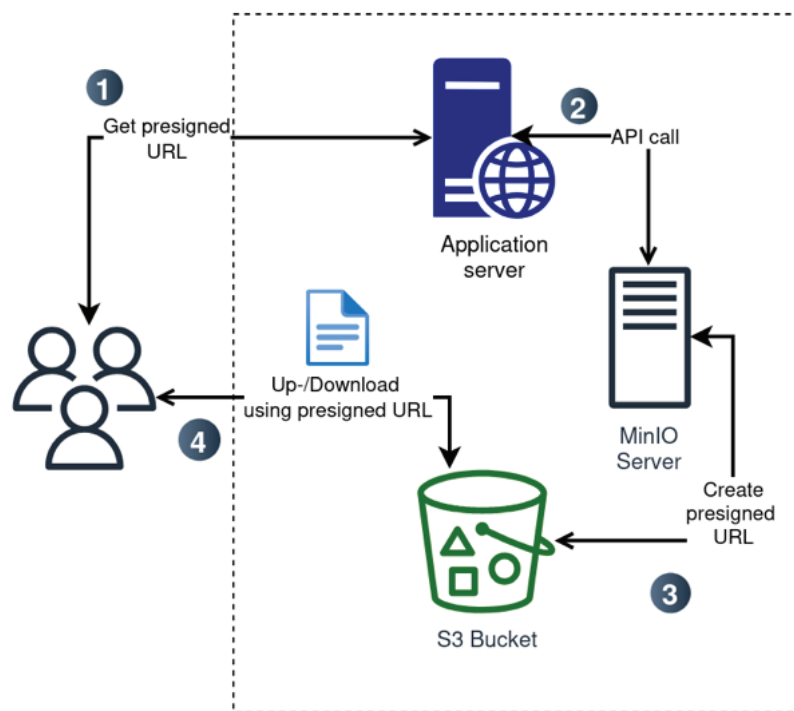


Figure 3.5: Schematic illustration of the file upload process in OmicsDM. (1) Users send a request to the application server to obtain a presigned URL for uploading or downloading files. (2) The application server requests the generation of a presigned URL for a specific file operation (upload or download) in the S3 bucket. (3) The MinIO server creates a presigned URL, which contains time-limited credentials and permissions to perform the requested file operation. This URL is sent back to the application server. (4) The application server provides the presigned URL to the user. The user can then use this URL to directly upload files to or download files from the S3 bucket. From cnag-biomedical-informatics.github.io/omicsdm-documentation.

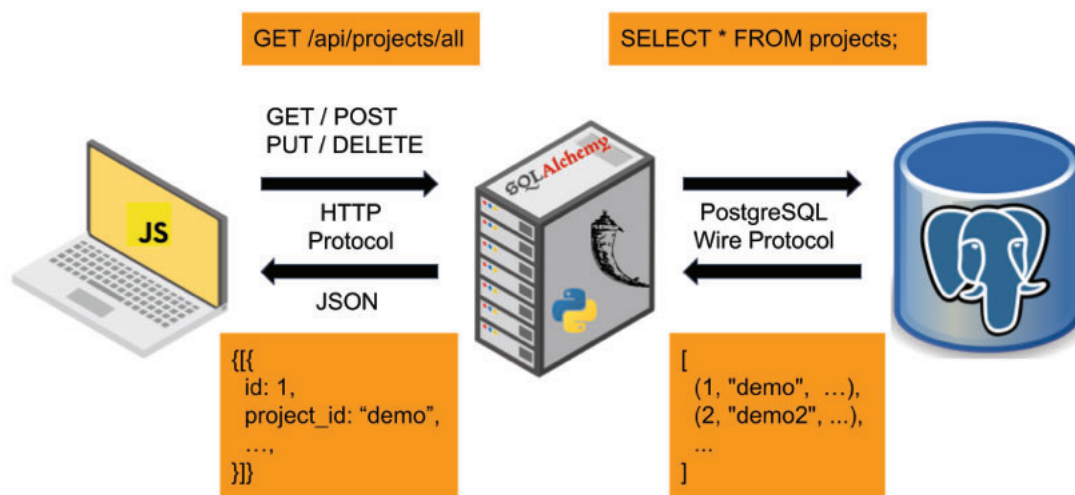


Figure 3.6: Schematic representation of a REST API utilising Flask and SQLAlchemy to communicate with a PostgreSQL database. The orange boxes illustrate an example request, respectively the responses exchanged at each stage of the communication flow.

3.4.3 Identity and access management

Since 3TR's data warehouse is not a public resource but should be only accessible within the project consortium the access to OmicsDM needs to be controlled. Thus, for each user we need to verify that the user is a member of the consortium. In the realm of application security these checks during the login process are called authentication (AuthN). However, only authenticating the user is not sufficient since not all the users should be allowed to for instance modify submitted metadata. This privilege should only be granted to the administrators of OmicsDM. Similarly not all uploaded files should be accessible to all partners but the owners should be able to decide to whom they grant access. Both required restrictions can be enforced by a role/group based authorisation (AuthZ) layer.

For both AuthN and AuthZ we chose Keycloak (keycloak.org), an open-source identity and access management solution that provides single sign-on (SSO) and multi-factor authentication capabilities. Keycloak supports a variety of protocols. The most important ones are summarised in Table 3.3. As apparent in Table 3.3 only OIDC, a protocol built upon the OAuth 2.0 framework, provides full AuthN and -Z, while providing adapters to JavaScript. Since the client was implemented in React JavaScript (Section 3.4.2) we decided to employ OpenID connect for identity and access management in OmicsDM. Figure 3.7 shows an exemplary authentication and authorisation flow.

Table 3.3: Protocols supported by Keycloak

Protocol	AuthN/AuthZ	2FA support	Token format	Existing adapters
Security Assertion Markup Language (SAML)	AuthN & limited AuthZ	Yes	SAML assertion (=signed XML)	mainly Java
Open Authorisation (OAuth)	AuthZ only	No	JSON Web Token (JWT) Opaque Token	Java, JavaScript, Node.js, etc.
OpenID Connect (OIDC)	AuthN & AuthZ	Yes	JWT (ID- & Access Token)	Java, JavaScript, Node.js, etc.

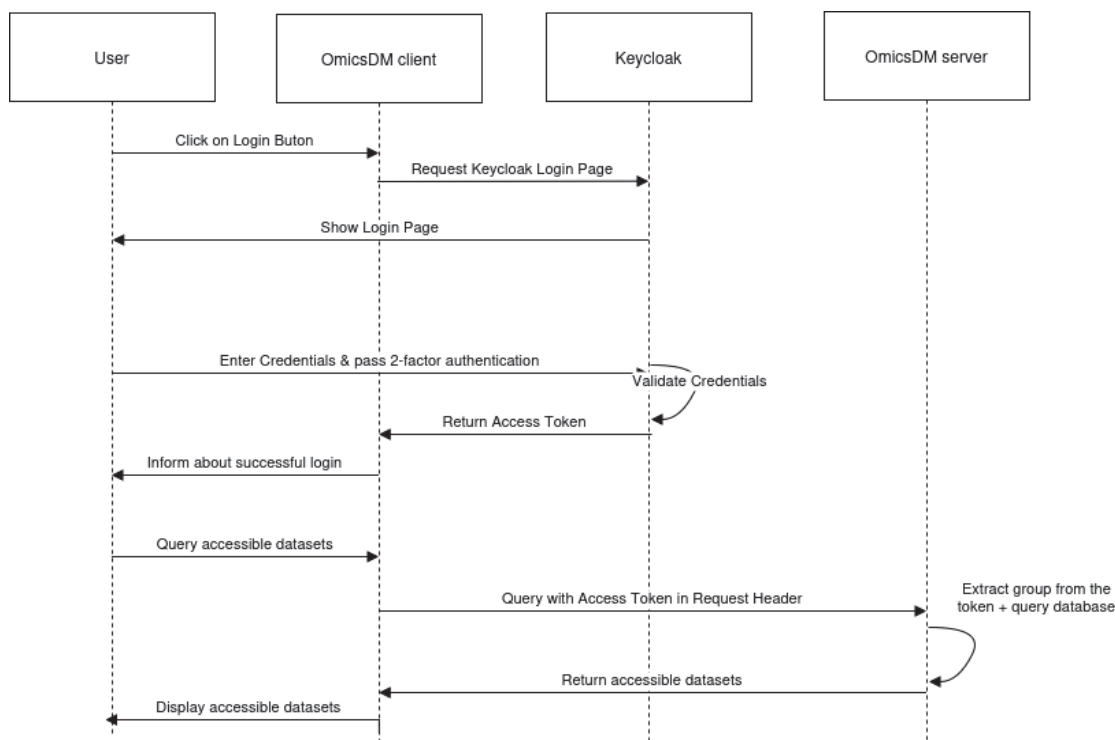


Figure 3.7: OmicsDM's authentication and authorisation flow utilising Keycloak and OpenID Connect. Upon navigation to the URL of OmicsDM's clients the partners were redirected to the Keycloak login page, where they have to enter their credentials. After successful authentication, the partners are redirected back to the OmicsDM client. Behind the scenes, the OmicsDM client receives an access token from Keycloak, which is used to authenticate the partners for the duration of their session. The authorisation is done based on the group information of the authenticated user. The group information is stored in the access token and used to grant/deny access to the uploaded data.

3.4.4 Architecture of OmicsDM

For easy deployment of OmicsDM's core building blocks (Section 3.4.1 - 3.4.4) it was decided to run each in a so-called container. These are standalone, lightweight, and executable packages that include everything needed to run an application such as the code, runtime, system packages, libraries etc. Containers are not only isolated from each other but most importantly from the underlying infrastructure, as depicted in Figure 3.8. Here, the most common containerisation technology Docker ([docker.com](https://www.docker.com)) was employed.

OmicsDM has a typical client-server (Section 3.4.2) architecture, in which we followed the commonly recommended scheme to run the client and the server in separate containers. Likewise, as depicted in Figure 3.9, the auxiliary services of OmicsDM, such as Keycloak, the PostgreSQL databases and the S3 storage are served from containers as well. This approach allows us to scale and deploy each of the services independently.

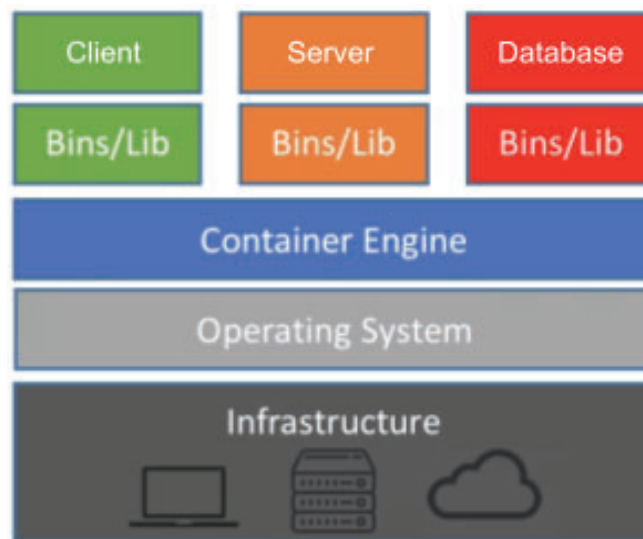


Figure 3.8: Container-based application architecture. Each element of OmicsDM runs in its own isolated container along with its specific binaries and libraries. These containers share a common container engine that operates on a single host operating system (OS), which in turn runs on the underlying physical or virtual infrastructure. Adapted from netapp.com/blog/containers-vs-vms.

OmicsDM has a typical client-server (Section 3.4.2) architecture, in which we followed the commonly recommended scheme to run the client and the server in separate containers. Likewise, as depicted in Figure 3.9, the auxiliary services of OmicsDM, such as Keycloak, the PostgreSQL databases and the S3 storage are served from containers as well. This approach allows us to scale and deploy each of the services independently. To ensure that all communications between the containers are secure and function as expected, a comprehensive testing strategy is employed, which includes (a) extensive backend test coverage through Pytest (docs.pytest.org) and (b) a robust end-to-end test suite using Cypress (cypress.io).

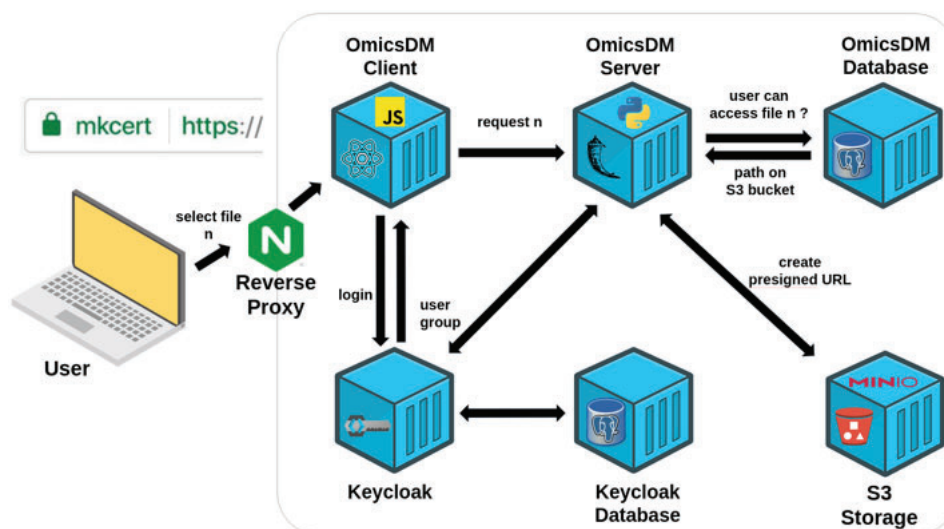


Figure 3.9: OmicsDM's core architecture. Illustrated is the communication in between the containers with the selection of a file as an exemplary user request.

3.4.5 Interactive data visualisation

3.4.5.1 Usage of CZ CELL×GENE Annotate for single cell data visualisation

After assessing the available tools for single cell data visualisation by literature review [91] and testing some tools myself I decided together with the 3TR partners and CNAG's Single Cell Genomics team, to use CZ CELL×GENE Annotate [74]. It is a web-based tool for the exploration of scRNA-Seq data, which not only is widely used in the single-cell community but also can be augmented with plugins. One of these plugins is utilised in OmicsDM's single-cell data visualisation module, namely VIP (Visualization In Plugin) [92] which provides additional analyses and plotting capabilities such as violin plots (Figure 3.10).

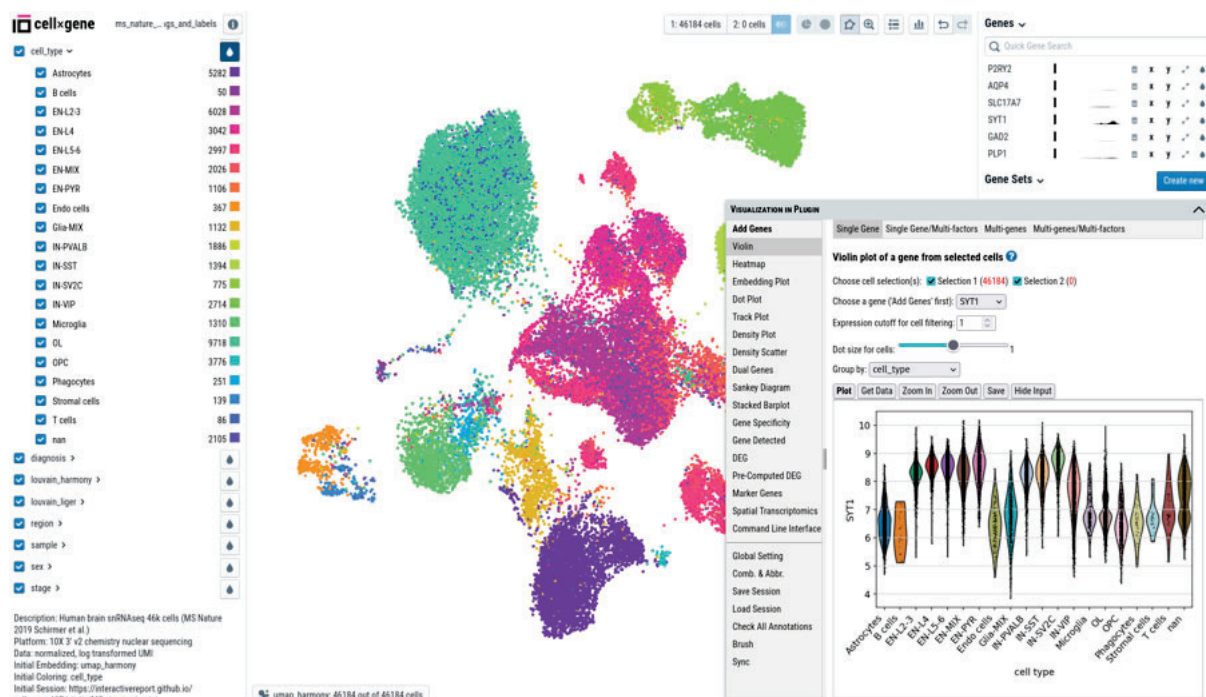


Figure 3.10: CZ CELL×GENE Annotate augmented with VIP (Visualization In Plugin). The main visualisation of the single-cell data is driven by CZ CELL×GENE Annotate featuring a scatter plot, here a UMAP (middle). The colours of the points can encode different types of information such as cell annotations (left) or gene expression levels (top right). In this figure colouring by cell type is chosen. These core functionalities can be extended by VIP, providing additional analyses such as violin plots (bottom right). Screenshot of cellxgenevip-ms.bxgenomics.com.

3.4.5.2 CELL×GENE Annotate instances orchestration with ShinyProxy

Having decided to use CZ CELL×GENE Annotate for the single-cell data visualisation module, one of its more recent software design choices had to be addressed first. While in previous versions it was possible to use one running instance of CZ CELL×GENE Annotate for all users, accessing different datasets, the developers decided to move away from this approach. Now instead, for each dataset a separate

instance of CZ CELL×GENE Annotate must be started. An instance in this context is a container, a standalone, executable package of software that includes everything needed to run a piece of software, here CZ CELL×GENE Annotate. Thus, on one machine multiple instances can be run, each in its own container. However, at that point the question was how to start/stop (orchestrate) these instances. To address this issue, ShinyProxy (shinyproxy.io) was employed. As implied by its name it was initially developed for hosting R Shiny applications, a web application framework for R, and making them publicly accessible. However, in the meantime it has been extended to support other web applications as well (Figure 3.11).

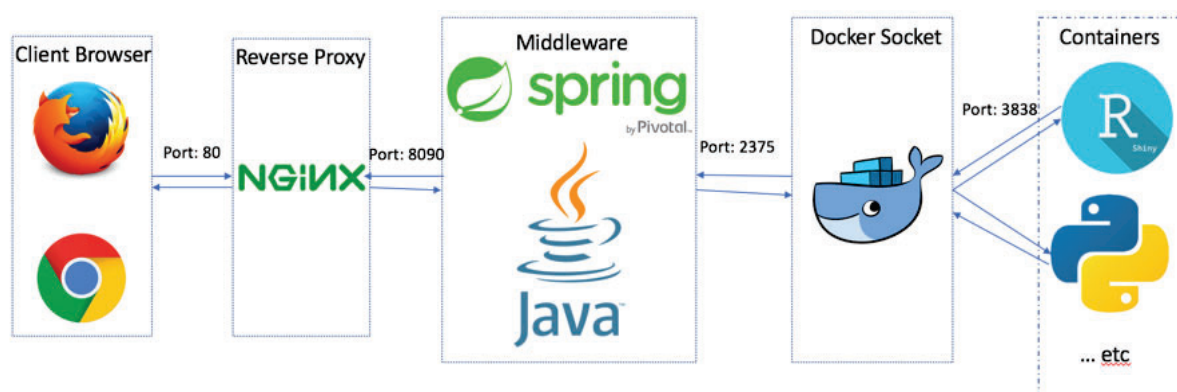


Figure 3.11: Conceptual overview of ShinyProxy. From lukesingham.com/shiny-containers-with-shinyproxy.

Next to the orchestration of the CZ CELL×GENE Annotate instances ShinyProxy also takes care of the user authentication and authorisation. In the context of OmicsDM ShinyProxy takes advantage of the SSO capabilities of Keycloak, respectively of the employed OIDC protocol (Section 3.4.3) by reusing the token already stored in the browser's local storage. Thus, when the link provided by OmicsDM to the started CZ CELL×GENE Annotate is opened in a new tab the user does not need to authenticate a second time. Instead, ShinyProxy leverages the access token stored in the browser's local storage. In Figure 3.12 the architecture enabling the orchestration of new CZ CELL×GENE Annotate instances from OmicsDM is depicted illustrating how a user can start a new instance. The automated shutdown of the instances after a certain time of inactivity is not handled by OmicsDM, but directly by ShinyProxy.

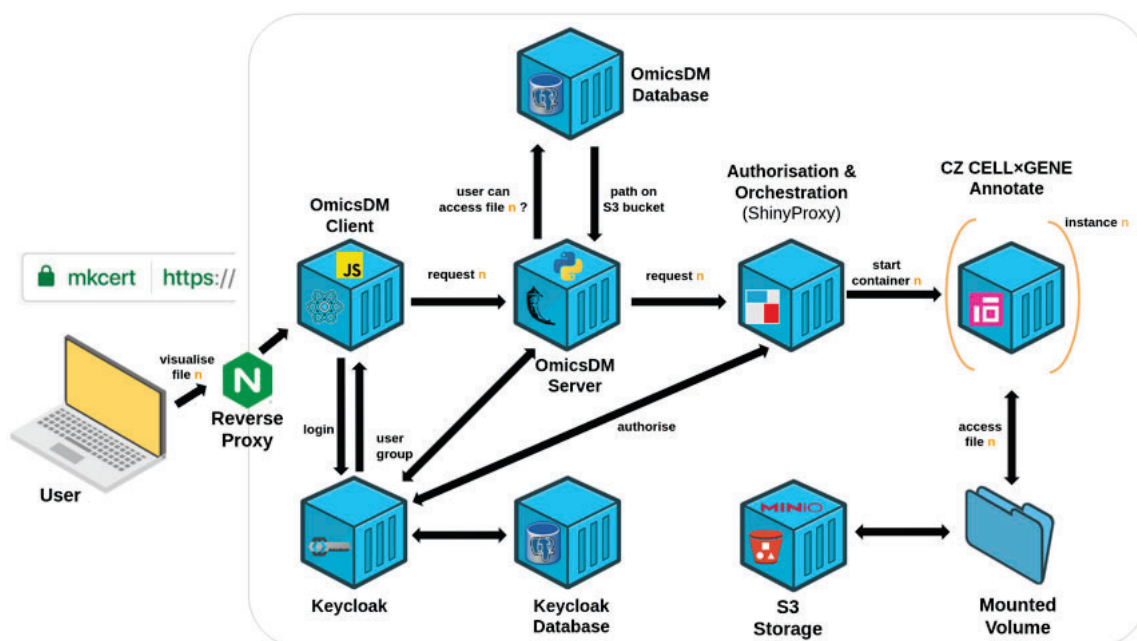


Figure 3.12: Architecture enabling the start of new CZ CELLxGENE Annotate instances. Illustrated is the communication in between the containers when a user requests to visualise a single cell dataset.

3.4.6 Data analyses

3.4.6.1 Supported workflow management systems

Biology is becoming increasingly data-intensive, with researchers having to process and analyse large amounts of data. While historically researchers were used to connect multiple tools via bash scripts, these are nowadays avoided in favour of workflow management systems. These software tools help to automate the execution of complex analysis pipelines by allowing researchers to define the steps of an analysis pipeline based on the expected inputs and outputs. The workflow management system takes care of executing the steps in the correct order, ensuring that the inputs and outputs are passed between the steps correctly. If necessary, the workflow management system can also handle errors by retrying failed steps. In most of the cases the final output of the workflow is a report summarising the results of the analysis (Figure 3.13).

One of the key concepts of a workflow management system is the unification of the definition of complex analysis pipelines [93]. This imposed structure is known to simplify pipeline construction [94], facilitate sharing, reusing of existing pipelines [95], and ultimately enhancing reproducibility [96]. Additionally, there is emerging evidence that using workflow management systems can also be beneficial in terms of execution time and memory efficiency [97].

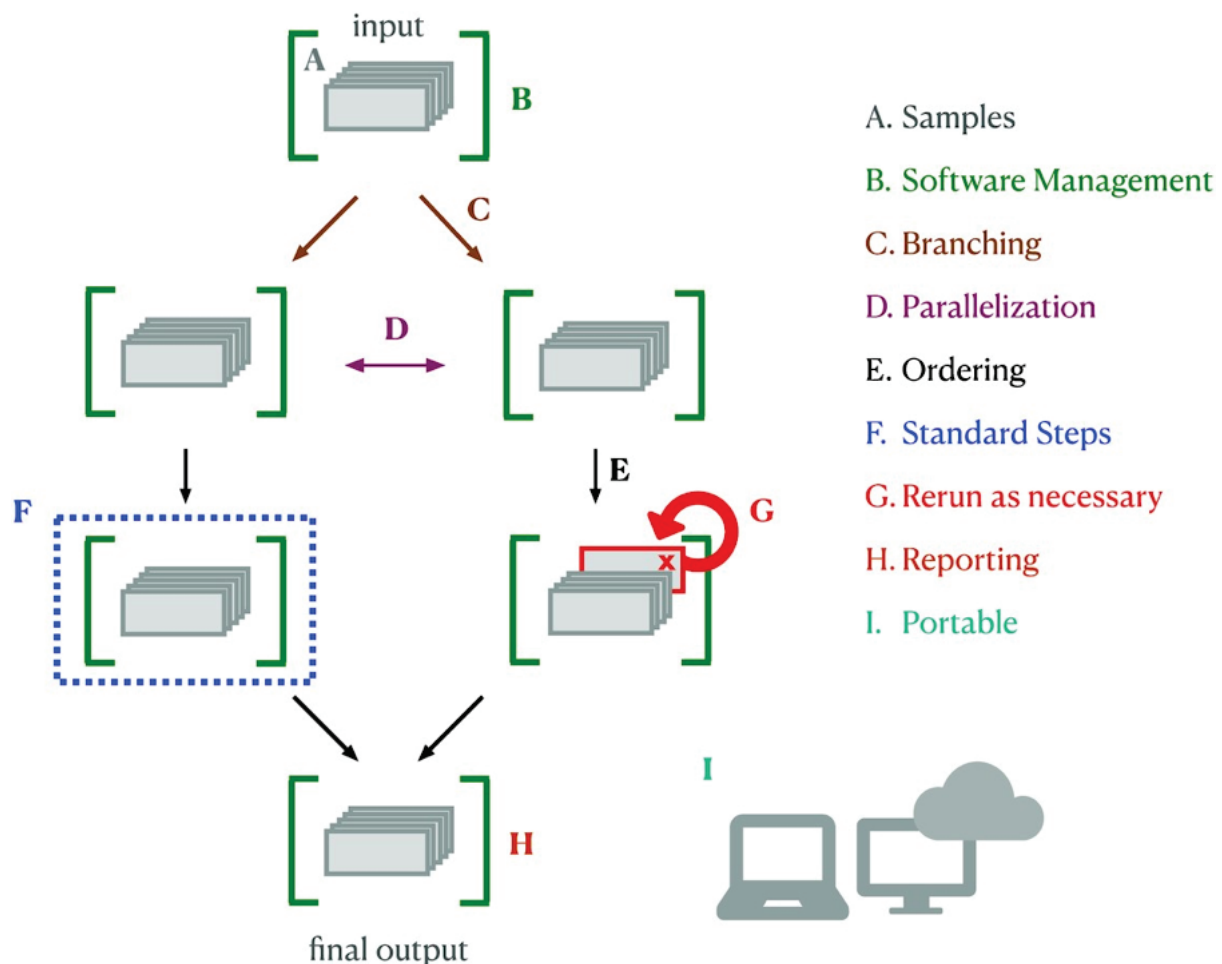


Figure 3.13: Workflow systems schematic working principle. From [98].

The development of workflow management systems is a highly competitive field. In fact, there is a constant emergence of new tools and improvements of existing ones. An idea over the vastness of the potential solutions to pick from can be gained by browsing through curated lists of workflow management systems, respectively pipeline tools on GitHub ([meirwah/awesome-workflow-engines](https://github.com/meirwah/awesome-workflow-engines), [pditommaso/awesome-pipeline](https://github.com/pditommaso/awesome-pipeline)). These two repositories alone already feature over 200 tools (last access June 2025). Consequently, choosing the right one for managing the analysis pipelines in OmicsDM could be seen as a challenge.

However, over the years at least the biomedical research community consolidated around four main workflow management systems: Snakemake [99], Nextflow [100], Common Workflow Language (CWL) and WDL (Workflow Description Language) [98]. The latter two are more focused on the description of the workflows and are not implementing the actual execution of the workflows. Since OmicsDM also needed the

execution part, this already narrowed down the choice to Nextflow and Snakemake. Between these two it was not possible to make a clear decision, since both seem to be more or less equally well adopted. Thus, OmicsDM is kept workflow agnostic, currently supporting both Nextflow and Snakemake pipelines.

Nextflow

Nextflow is a workflow management system that enables the development of complex and scalable workflows using a DSL. The Nextflow scripting language is an extension of Groovy, a superset of Java (Figure 3.14). Its building blocks are channels, processes, and operators. While channels are used to connect processes and to pass data between them, processes are the actual computational units that perform the analysis. Finally, operators can be used to manipulate channels to filter, merge, or split them to name a few examples. They are helping to bring the data to the processes in the right format and order. See all components in Figure 3.15.



Figure 3.14: Origin of Nextflow. From vibbits-nextflow-workshop.readthedocs.io/en/latest/nextflow

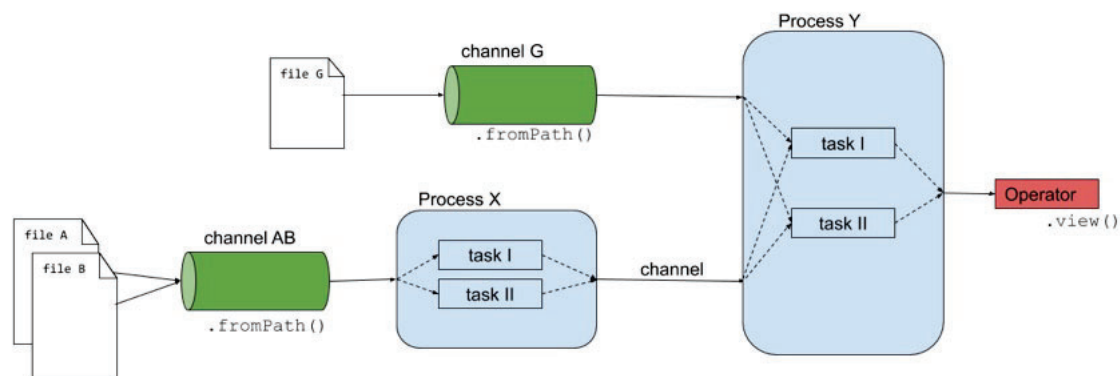


Figure 3.15: Nextflow workflow. From vibbits-nextflow-workshop.readthedocs.io/en/latest/nextflow.

Initially developed at the Centre for Genomic Regulation (CRG) in Barcelona, Spain (crg.eu/cedric_notredame). Nextflow's development is now driven by Seqera (seqera.io). Thus, it is continuously maintained and improved. The latest fundamental change was the removal of DSL1 in favour of DSL2. Making it easier to reuse workflows and to share them with the community. Consequently, the workflows in OmicsDM are implemented in Nextflow DSL2.

Snakemake

Snakemake is a workflow management system that enables reproducible and scalable data analyses. As hinted by *snake* in its name it is Python-based and inspired by the GNU *make* build system (Figure 3.16). The workflows are defined in a Snakefile, a Python script that specifies the rules for running the analysis pipelines. The rules define the input and output files of each step in the pipeline, as well as the commands to run the analysis. Snakemake automatically determines the order in which the rules should be executed based on the dependencies between the input and output files (Figure 3.17). In contrast to Nextflow, Snakemake's long-term support is not facilitated by having built a company on top of it but by an affiliation with NumFOCUS, a non-profit organisation that supports open-source scientific software.



Figure 3.16: Snakemake behind the name. From khoongweihao.medium.com/workflow-management-with-snakemake-an-introduction-with-python-62437e3e2952.



Figure 3.17: Snakemake workflow. From snakemake.readthedocs.io/en/v9.4.1/project_info/faq.html.

3.4.6.2 Analyses scheduling and execution

Since the backend of OmicsDM is implemented in a synchronous way, a queue system was needed to handle the analysis pipelines. The scheduling of an analysis is handled by Celery (celeryproject.org), an open-source task management tool that is written in Python. Celery's queue system is utilised as follows. First, the submitted analysis tasks, in OmicsDM called job, are passed to the queue by the backend of OmicsDM, where they are stored until they are picked up by the Celery workers. The workers are running in separate Docker containers and are connected to the queue via the Celery broker. For the Celery broker and its results backend, I followed the

recommendations of the Celery documentation (docs.celeryq.dev) which says to use RabbitMQ (rabbitmq.com) as the broker and Redis (redis.io) as the results backend. In addition to the containers used in the OmicsDM architecture (Figure 3.18), virtual machine (VM)s were also needed. A VM is a software that emulates a physical computer coming with its own OS to run applications, as depicted in Figure 3.19. The reason for having OmicsDM's services such as the backend, respectively celery workers 1 and 2 in their own VMs is for security and stability reasons. The idea was to ensure that if for instance due to a job consuming too much memory one of the workers crashes, the rest of the system is not affected. With this setup, the OmicsDM services continue to run without any interruption as well as the other Celery workers. Although this setup improves the general stability of the overall system, crashed VMs still need to be restarted manually to restore OmicsDM's full analytical capacity.

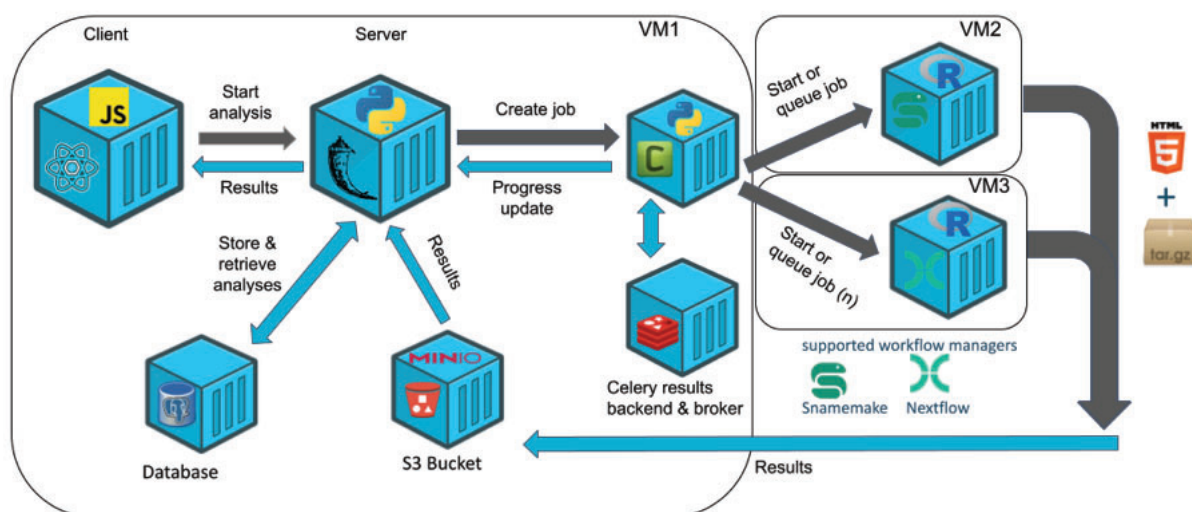


Figure 3.18: Schematic illustration of all the connected services enabling the running of analyses pipelines in OmicsDM. Authentication layer is not included for the sake of clarity.

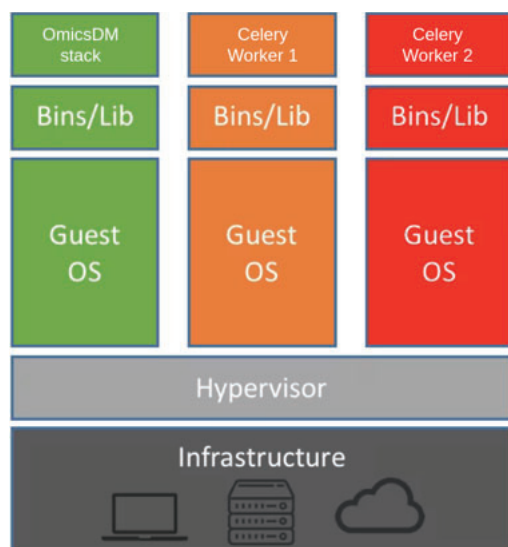


Figure 3.19: Virtual machines overview. In total there are three virtual VMs. One for OmicsDM's dockerised software stack (Client, server, databases etc.) and one for each celery worker. To run the docker containers each VM needed to have the specific binaries/libraries installed in their OS (here CentOS Linux 8). These VMs are managed by the hypervisor Proxmox Virtual Environment that runs on the underlying physical infrastructure. Adapted from netapp.com/blog/containers-vs-vms.

3.4.6.3 Downstream analyses of RNA-Seq data

The transcriptome, introduced in Section 1.1.2.1, is inherently dynamic since the abundance of specific RNA molecules, such as coding RNAs, fluctuate over time in response to internal rhythmicity [101] or external stimuli [102]. Changes due to the latter are of particular interest in the context of the 3TR project, since an external stimulus could be a treatment. The downstream analyses of RNA-Seq data are manifold. The four possible levels at which these analyses can be done as proposed by Li and Li [103] (Figure 3.20), together with the most common analysis performed are:

- Sample level: Transcriptome similarity
- Gene level: Differential gene expression
- Transcript level: Transcript isoform expression
- Exon level: Exon inclusion rates in alternative splicing

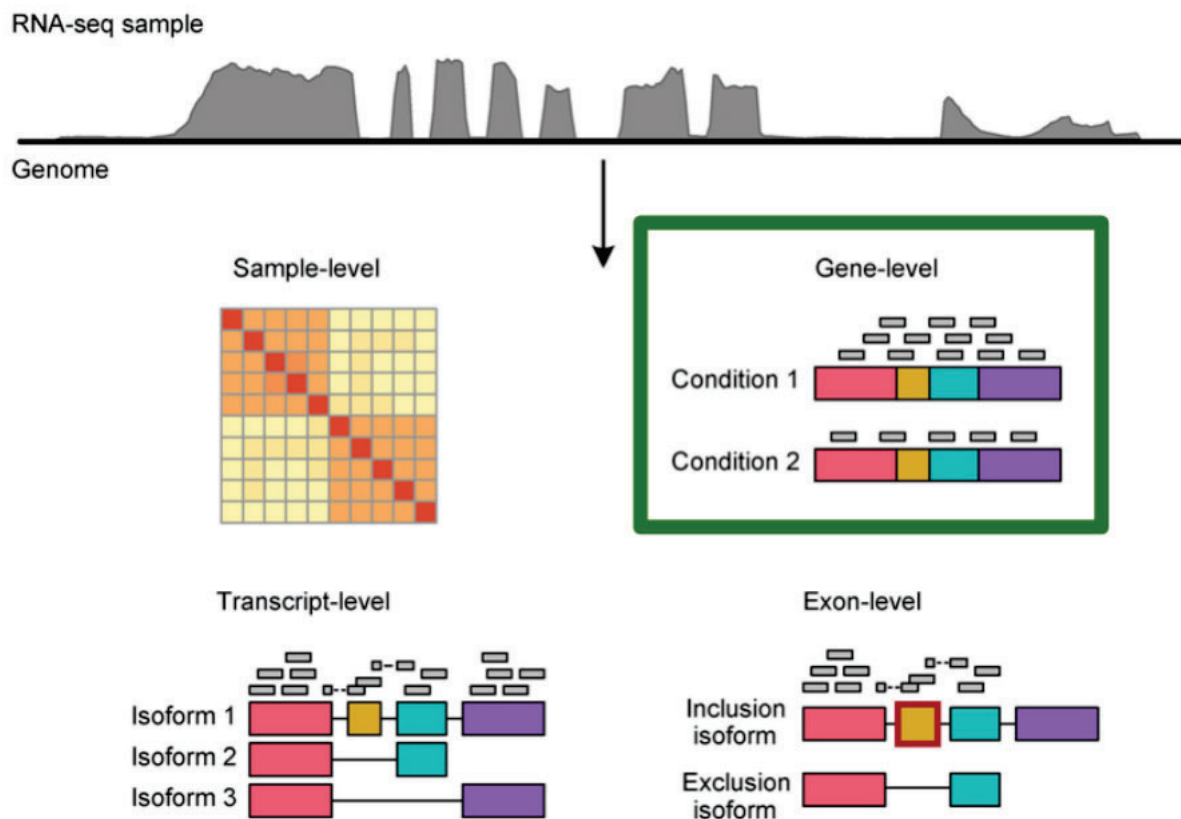


Figure 3.20: RNA-Seq data analyses levels. The analysis level of interest for this thesis is marked in green. From [103].

Here we will focus on the analyses on gene level. In these always two or more conditions are compared to each other using differential gene expression (DGE) analyses. In 3TR, the comparison is between the healthy controls vs. treated patients

and the transcriptome of each patient at different time points. The latter could for instance give insights into the role of specific genes in the treatment response. To identify the genes whose expression is significantly down- or upregulated, potentially due to a treatment, the mean of the gene expression across all samples and conditions (global mean) is compared against the mean of the gene expression in the specific condition (Figure 3.21). By analysing the transcriptome of patients, researchers can for instance identify genes that are differentially expressed between patients who respond to treatment and those who do not. However, the raw count data is not suitable for this kind of analysis, because of differing sequencing depth and other technical biases. Therefore, the counts are first normalised. Depending on the intended insights, different methods such as the z-score normalisation or log2 transformation, more specifically the log2 fold change (LFC), can be employed. The differences between the two methods are summarised in Table 3.4 In fact, both metrics are often used together: LFC highlights genes that are differentially expressed, while z-scores help visualise those differences across multiple samples in heatmaps. Figure 3.22 shows an example of the use of LFC and z-scores.

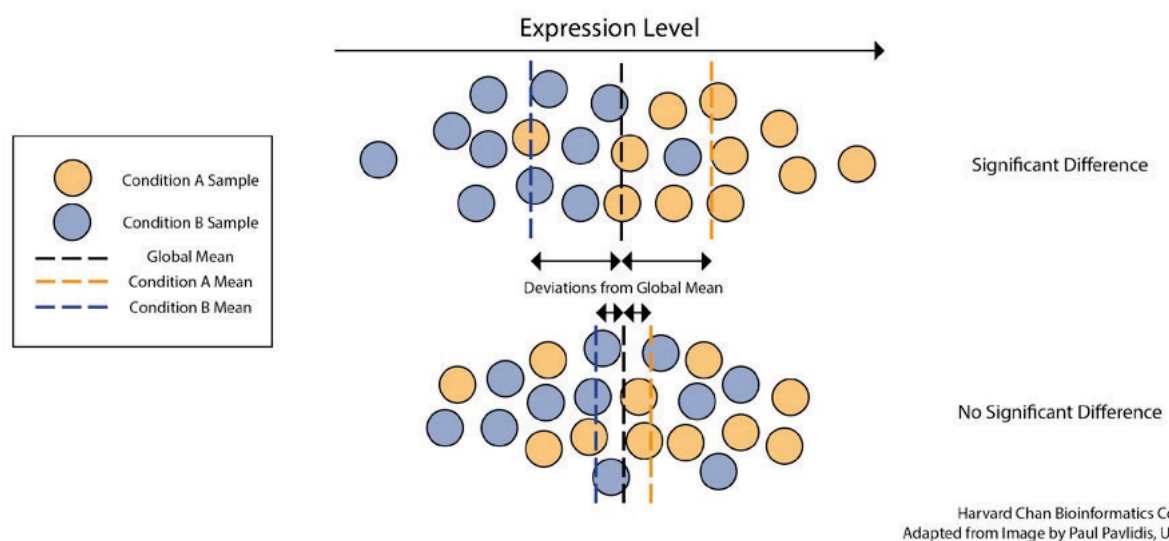


Figure 3.21: Expression levels of a selected gene in two conditions (A|B). Each dot represents the expression level (increasing from left to right) in one sample/patient, whereby the colour indicates the condition. The coloured vertical lines show the mean of the expression in each condition, respectively the mean of the expression across all samples and conditions (global mean). The deviation from the global mean indicates a potential differential expression of the gene, which is the case in the top panel but not in the bottom panel. From [104].

Table 3.4: LFC vs z-score

	LFC	z-score
Purpose	Differential expression between conditions	Within-gene variation across samples
Metric	Logarithmic ratio between two conditions	Standard deviations from the mean
Normalisation	Ratio-based	Mean-centred, variance-standardised
Best visualised by	Volcano plots	Heatmaps
Interpretation	Magnitude and direction of change	Position within a distribution

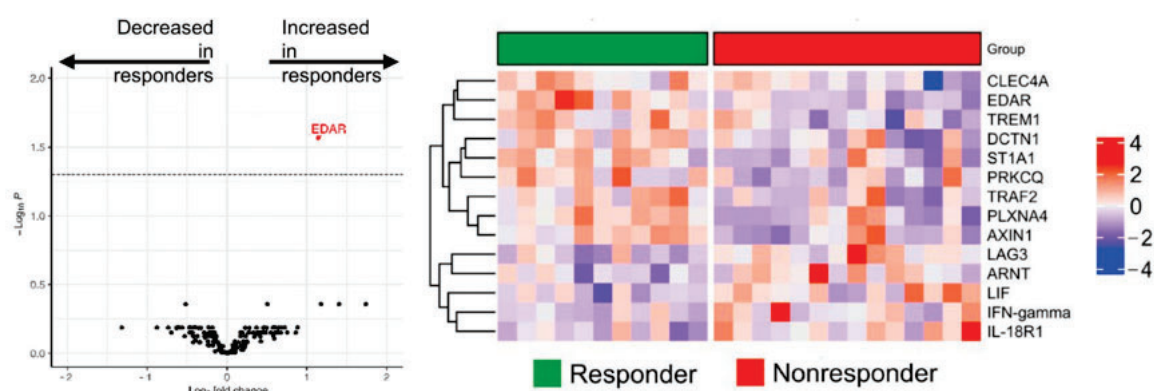


Figure 3.22: Example of a volcano plot and a heatmap to illustrate the use of LFC and z-scores for the identification of differentially expressed genes. Left: An example volcano plot showing the differential gene expression between responders and non-responders. The x-axis shows the LFC, while the y-axis shows the $-\log_{10}$ of the adjusted p-value. The red dots represent significantly upregulated genes. Right: An example heatmap depicting differential gene expression between responders and non-responders to treatment. The rows represent specific genes, while the columns represent the patients. On top is an annotation bar indicating if a patient is a responder (green) or a non-responder (red). The heatmap colour shows the level of expression based on z-scores, with red indicating upregulation and blue downregulation. The dendrogram shows the hierarchical clustering of genes based on expression similarity. Adapted from [105].

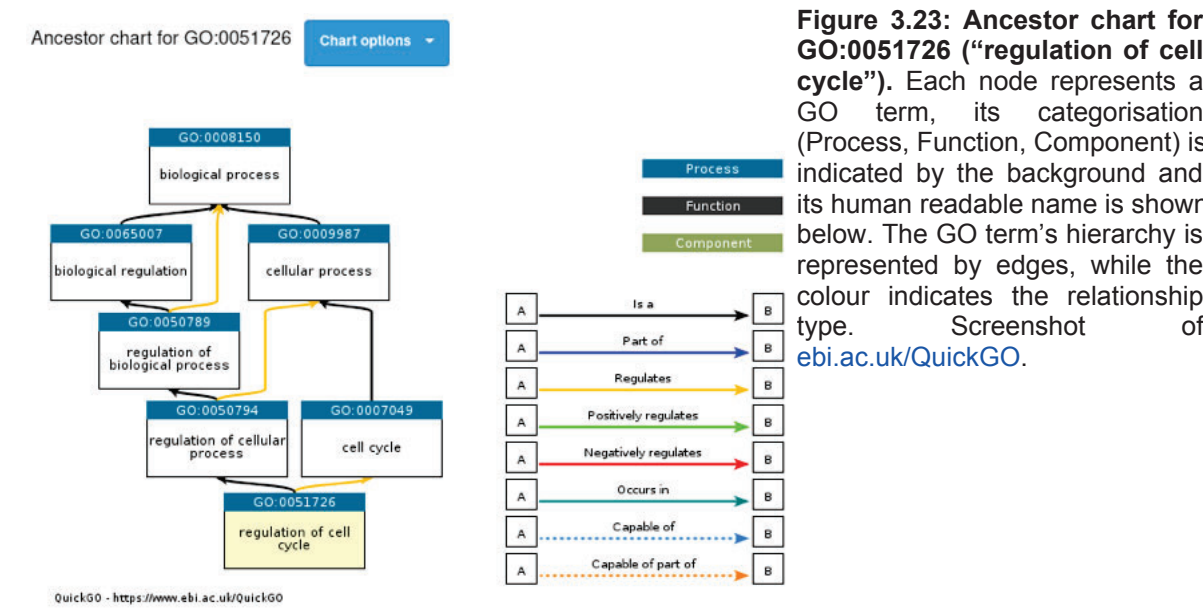
Functional enrichment analysis

After having identified differentially expressed genes (DEGs), a common next step is to perform a functional enrichment analysis. This approach aims to elucidate whether specific biological functions or pathways are statistically overrepresented among the DGEs relative to what would be expected by chance. It does so by comparing the list of DEGs to predefined genesets. The latter are collections of genes that share common biological characteristics, function or such a like. Researchers typically obtain these geneset from databases such as Gene Ontology (GO) [106], Kyoto Encyclopedia of Genes and Genomes (KEGG) [107], and Reactome [108] (Table 3.5). So is for

instance GO a widely used resource for gene function annotation. It is structured as a directed acyclic graph with three main branches: molecular functional ontology (MFO), biological process ontology (BPO), and cellular component ontology (CCO) Each branch contains terms that describe the function of a gene at different levels of specificity. For instance the term “cellular process” (GO:0009987) refers to any process occurring in a cell, while “regulation of cell cycle” (GO:0051726) is a more specific term that describes any process modulating the cell cycle. Figure 3.23 shows the relationship between these terms.

Table 3.5: Comparative summary of the most important gene to function/pathway annotation databases

	Gene Ontology (GO)	Kyoto Encyclopedia of Genes and Genomes (KEGG)	Reactome
Purpose	Classification of gene function	Pathway mapping and analysis	Pathway analysis and visualisation
Scope	Molecular functions, biological processes, cellular components	Pathways, disease mechanisms, drugs, chemical substances	Biological pathways and reactions
Structure	Hierarchical ontology	Hierarchical pathway database	Hierarchical pathway database
Output	Gene function annotations	Maps of metabolic and signalling pathways	Biological pathway diagrams
Example Use Case	Determining functional annotations for gene lists	Mapping genes to known pathways to understand diseases	Analysing biochemical pathways and reactions
Statistics (as of 2024-09-08)	40K GO terms; 7.9M annotations; 1.5M gene products; 5.4K species	574 Pathway maps; 15.6M pathway linked genes; 10.3K species	2.4K pathways; 11.1K annotated human protein gene products [109]; 91.7K reactions; 89 species



As easily imaginable, including for instance all the roughly 40,200 GO terms in a functional enrichment analysis could introduce statistical noise, respectively an inflated respectively an inflated number of false positives. To shrink the number of included terms one could pre-filter manually by e.g., removing too broad terms, such as “cellular process” (Figure 3.23). However, more advisable is to utilise by experts curated gene sets collections. One widely entrusted resource in that regard is the Molecular Signatures Database (MSigDB) [110]. Its gene sets collections can be retrieved either via the MSigDB API or the web UI (gsea-msigdb.org/gsea/msigdb). On the latter the gene sets are available in the form of a JSON bundle or the so-called Gene Matrix Transposed File Format (GMT). As this tab-delimited text file format is one of the expected inputs for all the functional enrichment analysis methods included in OmicsDM its structure is depicted in Figure 3.24 The actual methods utilising any of the aforementioned databases can be divided into three types: Overrepresentation- (ORA), ranking- or topology-based methods [111] as illustrated in Figure 3.25.

Each row represents one gene set →

	A	B	C	D
1	chr10q24	Cytogenetic band	PITX3	SPFH1
2	chr5q23	Cytogenetic band	ALDH7A1	IL13
3	chr8q24	Cytogenetic band	HAS2	LRRC14
4	chr16q24	Cytogenetic band	RPL13	GALNS
5	chr13q14	Cytogenetic band	AKAP11	ARL11
6	chr7p21	Cytogenetic band	ARL4A	SCIN
7	chr10q23	Cytogenetic band	SNCG	FER1L3
8	chr14q12	Cytogenetic band	C14orf125	FOXG1C
9	chr13q13	Cytogenetic band	ALG5	RFXAP
10	chr1p34	Cytogenetic band	JMJD2A	MRPS15
11	chr10q21	Cytogenetic band	MBL2	C10orf70

First column are gene set names. Duplicates are not allowed

Second column contains a brief description. Its optional

Figure 3.24: Explanatory screenshot of the Gene Matrix Transposed File Format. Adapted from docs.gsea-msigdb.org.

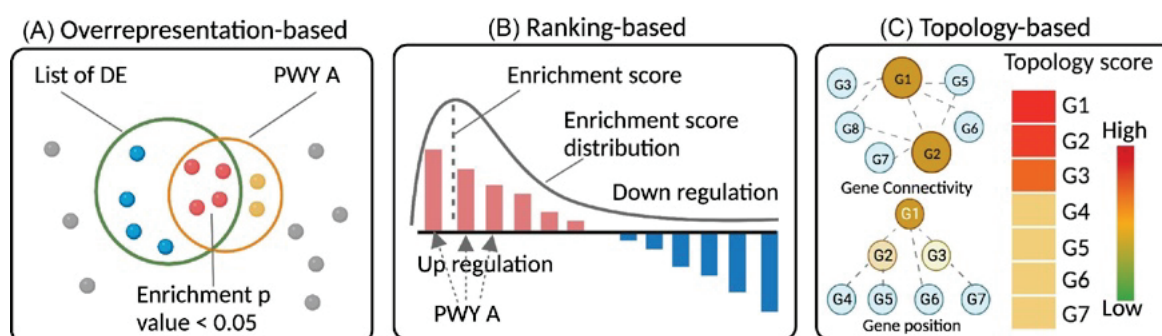


Figure 3.25: Types of functional enrichment analysis. (A) Overrepresentation-based methods identify pathways by assessing if a predefined set of genes annotated to a specific pathway (PWY A) are enriched (p -value < 0.05) in a list of differentially expressed (DE) genes. (B) Ranking-based methods start with sorting (ranking) genes based on certain criteria such as DE (up- or downregulation) followed by assessing if a gene set of a known pathway (PWY A) is overrepresented at the top or bottom of the ranked list. (C) Topology-based methods consider the pathway structure such as the interaction between genes (gene connectivity) in a pathway or the position of a gene in a pathway. PWY A represents an example pathway From [111].

Implementation of exemplary RNA-Seq data analysis pipelines

To showcase the analytical capabilities of OmicsDM, exemplary analysis pipelines, for both bulk- and single cell RNA-Seq data, were implemented. In the case of bulk RNA-Seq data the first step was to perform DGE with DESeq2 [112] was chosen, a widely used R package, specifically designed for the analysis of high-throughput RNA-Seq data. It uses a negative binomial distribution to model the count data and provides methods for normalisation, differential expression analysis, and visualisation. One of DESeq2's outputs are the LFC values.

Downstream of the DESeq2 analysis, respectively for the analysis of scRNA-Seq data the following functional enrichment analyses were implemented: (1) ORA-based GO analysis and (2) the ranking-based Gene set enrichment analyses (GSEA) [113]. For the GO analysis the R package gprofiler2 [114] was used, a programmatic R interface to the g:Profiler web server (biit.cs.ut.ee/gprofiler).

Finally, for the ranking-based GSEA the following methods were implemented: z-scoring, Gene set variation analysis (GSVA) [115] and fgsea. The latter is according to its developers not only 100 times faster than the traditional GSEA method, but also able to detect more statistically significant pathways [116]. Note, in the scRNA-Seq context before performing any of the aforementioned analyses, such as z-scoring, the gene expression matrix first needed to be extracted using Scanpy (introduced in Section 3.1.1). This Python package was also utilised to incorporate the functional enrichment results, such as the z-scoring, into the scRNA-Seq dataset. The underlying Python and R scripts are available in the OmicsDM repository (github.com/CNAG-Biomedical-Informatics/omicsdm). To prevent potential clashes of the required dependencies, respectively package versions, each pipeline step (individual analysis) was containerised.

Each of the exemplary analyses were wrapped in a Snakefile and also as a Nextflow script. The steps in both are as follows:

1. Download the necessary input files from the S3 storage
2. Preprocess the input data (optional, depends on the selected analysis)
3. Run the analysis
4. Integrate the functional enrichment analysis results into the single cell dataset (optional)
5. Embed the results in a self-contained HTML report utilising R markdown

The initial scripts to perform DESeq2, fgsea and gprofiler2 were implemented in R by Anna Esteve Codina and her team at CNAG. The exemplary scRNA-Seq pipeline has been developed in collaboration with members from GENyO. Note, that for the GSVA analysis the R package escape (Easy single cell analysis platform for enrichment) [117] was employed, since it facilitates GSEAs, such as GSVA in the single cell context. My main contribution was the aforementioned wrapping of each pipeline step inside a Snakefile, respectively their containerisation.

3.4.7 Deployment of OmicsDM

The developed data warehouse solution consists of a client and a server (Section 3.4.2) to make sure that each part on its own, respectively the client and server in concert, work as expected proper testing was required (Section 3.4.4). The automated testing, the build and deploy of the server, respectively the client, were performed with Jenkins (jenkins.io), an open-source automation server. All the so deployed containers were monitored with Portainer (portainer.io), which provides a web interface to manage containers. To securely manage credentials and configuration files, Gitea (gitea.io), an open-source self-hosted Git service with a web-based interface, was used. Credentials and configurations were securely stored in private Gitea repositories and accessed through its API during Jenkins pipeline executions. The latest version of OmicsDM is available to the partners of the 3TR project at <https://3tr.gpap.cnag.eu/portal>.

For development purposes and to gather feedback from CNAG members not involved in the 3TR project we also deployed an OmicsDM playground. The source code,

including deployment constructions is available here: github.com/CNAG-Biomedical-Informatics/omicsdm. Its necessary services (containers) such as the Celery workers (Section 3.4.6.2) were deployed on VMs.

The VMs were provided by CNAG's IT department. Nginx (nginx.com) was used as a reverse proxy to make the playgrounds accessible on CNAG's intranet. Table 3.6 lists the VM specifications, such as the used number of central processing unit (CPU) cores and Random-access memory (RAM), for the OmicsDM playground deployment.

Table 3.6: VM specifications for OmicsDM playground deployment

VM	Purpose	CPU (cores)	RAM (GB)	Storage (GB)	OS
1	Playground for OmicsDM	6	64	100 GB	CentOS Linux 8
2	Celery worker	6	32	100 GB	CentOS Linux 8
3	Celery worker	6	32	100 GB	CentOS Linux 8

3.5 Standalone tools for pheno-clinical data exploration

3.5.1 Convert-Pheno-UI

For the UI of Convert-Pheno a similar tech stack as for OmicsDM was employed. Thus, the server is also powered by Python Flask and communicates via SQLAlchemy with a PostgreSQL database. One noteworthy dependency is the inclusion of the Docker software development kit (SDK) for Python Docker. This helps users to get started with running the UI locally, since in the UI's first initiation the server will pull the docker image of the Convert-Pheno command line tool from Docker Hub (hub.docker.com/r/manuelrueda/convert-pheno) and start the respective container. The latter is utilised via the Python docker SDK for each conversion requested by the user. Thus, eliminating the potential difficulties users might face when installing Convert-Pheno from the Comprehensive Perl Archive Network (CPAN).

Similarly to OmicsDM, Convert-Pheno's client is as well implemented with JavaScript React. However, here for the file upload the open-source library Filepond (pqina.nl/filepond) was utilised, since in this context no upload to an S3 bucket was necessary.

The source code for both the server and the client is open-source and available here: github.com/CNAG-Biomedical-Informatics/convert-pheno-ui

For interested users who do not want to run anything on their personal computer (PC), there is a public playground available: <https://convert-pheno.cnag.cat>. Its architecture deployable with Docker-compose is depicted in Figure 3.26. Note, the relatively low system requirements. Our playground runs currently on one VM with one CPU core and 4 GB of RAM.

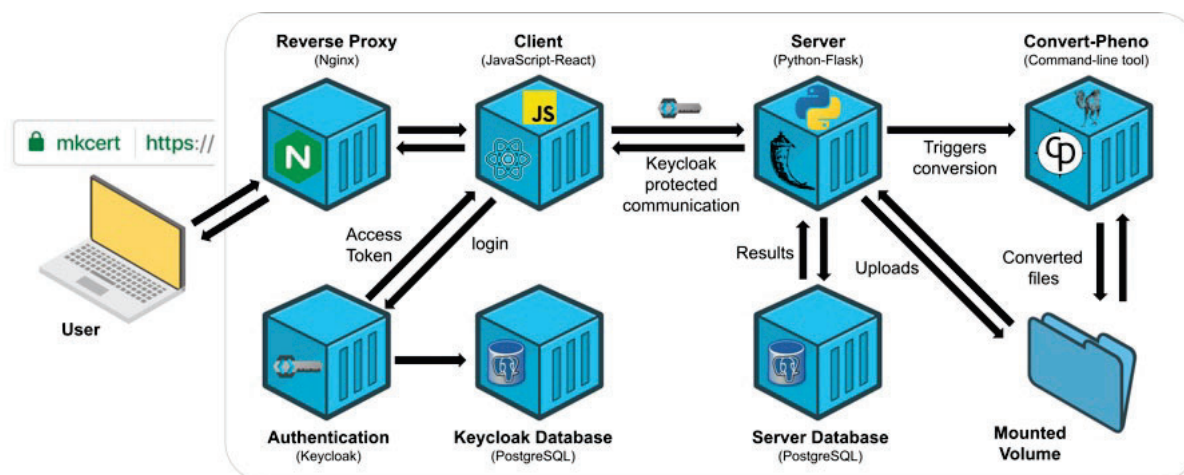


Figure 3.26: Architecture of Convert-Pheno-UI's playground. It is a client-server architecture with a nginx reverse proxy exposing the client to the outside. Keycloak is used for user authentication and management. The server and Convert-Pheno's command line interface (CLI) are in individual Docker containers with a shared mounted volume, allowing for file access and storage of conversion results in the database. Clients can request results for display or download. From [72].

3.5.2 Pheno-Ranker-UI

The UI of Pheno-Ranker is implemented in R using the shiny package, a web application framework. Moreover, the gridlayout package (github.com/rstudio/gridlayout) is utilised, since it facilitates building of dashboard-style layouts by leveraging CSS grid. Interactive visualisations are provided by plotly, interactiveComplexHeatmap [118] and by cyjShiny [119], a Shiny widget based for network visualisation using cytoscape.js. To modularise the code and to make the R shiny application production ready the golem package (thinkr-open.github.io/golem) is used. As a database either SQLite or PostgreSQL is employed, depending on the environment the application is run in. The SQLite database is used when the application is run locally. However, it has to be noted that SQLite allows only one write operation at a time, which can lead to concurrency issues when multiple users are

using the application at the same time (sqlite.org/whentouse.html). Consequently, PostgreSQL is utilised in the production environment to facilitate concurrent access. The Pheno-Ranker-UI is distributed and open-sourced as an R package (github.com/CNAG-Biomedical-Informatics/pheno-ranker-ui). However, since it is not possible to pack a Perl module into an R package, the Pheno-Ranker module (metacpan.org/dist/Pheno-Ranker) has to be installed separately. Thus, I also provide the Pheno-Ranker-UI, in a containerised version, which already includes the Pheno-Ranker module on Docker Hub (hub.docker.com/r/leistivo/pheno-ranker-ui).

Since pheno-clinical data is considered highly sensitive data one Pheno-Ranker-UI is only to be used by one user at a time. Thus, for each user a new Pheno-Ranker-UI Docker container is started by a self-hosted ShinyProxy instance. See Figure 3.27 for an overview of the architecture. Moreover, CNAG's ShinyProxy instance is secured with Keycloak, so each user can only access their own Pheno-Ranker-UI instances. The Keycloak universally unique identifier (UUID) is used to query the database for past Pheno-Ranker runs and also acts as the key to access any file on the mounted volume. Therefore, it is guaranteed that each user can only access their own data. Another security measure is that the respective Pheno-Ranker-UI container is automatically stopped/removed when the user closes the browser or tab. Pheno-Ranker-UI instances can be started by navigating to <https://pheno-ranker.cnag.eu>. Since with a high number of to be compared patients especially the plotting with R can be quite computationally intensive, we allocated to the virtual machine running the playground 12 CPU cores and 80 GB RAM.

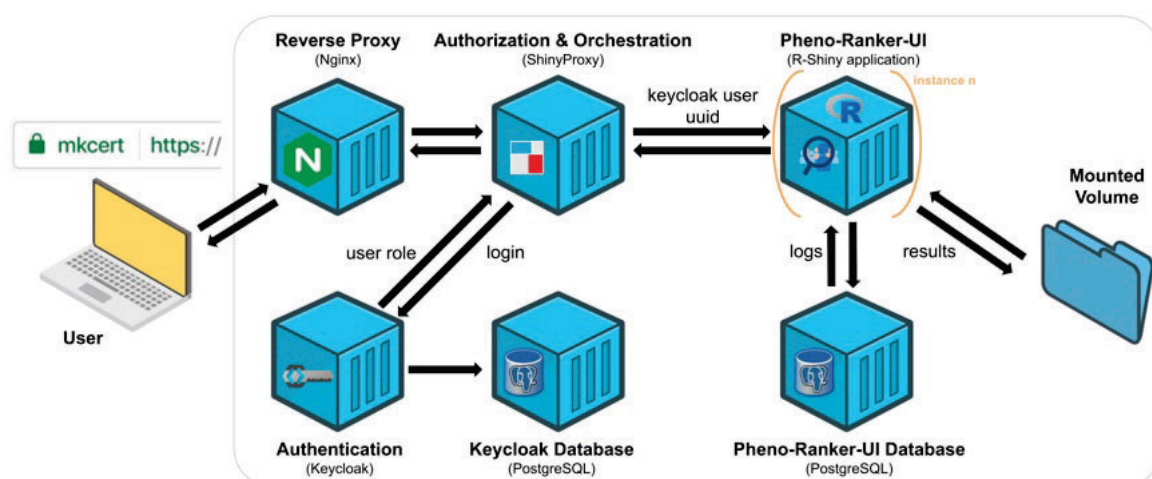


Figure 3.27: Architecture of Pheno-Ranker-UI's playground. ShinyProxy is used for authorisation and orchestration of the Pheno-Ranker instances. Keycloak is used for user authentication and management. Each started instance is mounted to a shared volume and connected to a database, allowing for file access and storage of the pheno-ranker results and retrieval of past runs. From [120].

3.6 Life Science Login as identity provider

In order to allow users to explore the full potential of Convert-Pheno-UI and Pheno-Ranker-UI we wanted to give our users the option to try these two UIs in a secure fashion with their own data. While it would be fairly simple to just write the output of each user session into a different folder this would have meant that the users would not be able to revisit the results of past runs. Another approach could have been to allow users to create their own users in our Keycloak instances. However, this would have required us to store the user's personal data such as email, name, etc. at our premises, which posed privacy concerns, especially in the light of GDPR. To avoid this, we instead opted for a federated login approach. In the field of life science, a well perceived provider in this regard is LS Login (Life Science Login) since it allows users to login via their Open Researcher and Contributor ID (ORCID). Consequently, we incorporated LS Login in our Keycloak authentication process. Thus, in doing so we not only offer users the convenience of signing in via ORCID but also are fully GDPR compliant since the only information we store from our users is the non retraceable Keycloak UUID.

Chapter 4: Results

4.1 OmicsDM

4.1.1 Personas and user stories for the 3TR project

To better understand the needs of the future users of the data warehouse needed in the 3TR project, I created three personas (Table 4.1). These personas are based on the interviews I conducted with the stakeholders of the 3TR project and the CNAG employees. The subdivision of the data owner into two different personas aims to address the particularities of the 3TR project. In it there are two different data sources, namely clinical trials conducted within the 3TR project and legacy projects. Data originating from the latter might require additional security measures, especially when the data is provided by an industry partner. To get a clear picture of the needs of the stakeholders of the 3TR before starting the development of OmicsDM, user stories were created (Table 4.2 & Table 4.3). Notably, initially for the uploaders of legacy data, the same user stories apply as for the uploaders of data produced during 3TR. However, to reflect potential data sharing restrictions of legacy data two additional user stories has been added, which differ between the two data uploader personas (Table 4.2 last row).

Table 4.1: Personas and their requirements for the data warehouse

Persona	Needs
Uploader of newly produced data	Upload data produced in 3TR, which should be shared with all partners immediately
Uploader of legacy data	Upload data from legacy projects, which should be initially private until I decide what to share with whom
Bioinformatician	Download/visualise/analyse molecular & pheno-clinical data

Table 4.2: User stories for data uploaders of during 3TR produced- and legacy data

As an uploader of	
data produced during 3TR, I want:	legacy data, I want:
A data storage with a high level of security, so I can upload my data without any concerns.	
To upload my pheno-clinical & molecular data, so it can be of use in the 3TR project.	
All my uploads to be versioned, so the analyses can be reproduced.	
To automatically share my data with other partners, so it can be visualised and analysed by other partners.	An extra layer of security, so I (i) can decide with which partners I want to share the uploaded data and (ii) attach a data usage policy file

Table 4.3: User stories for bioinformaticians

As a bioinformatician, I want to:
Download the data, so I can analyse it.
Interactively explore data, especially scRNA-Seq output, so I can better understand it before downloading it.
Analyse the data, so I can generate insights.
Share my analyses results with my partners, so we can discuss them.

4.1.2 Applicability of existing data warehouse solutions to the 3TR project

The applicability of existing data warehouse solutions for the 3TR project was evaluated with regards to the user stories and the needs of the stakeholders of the 3TR project. An initial assessment was based on the documentation and public demo instances of each of the tools. The results are summarised in Table 4.4, which shows that none of the existing data warehouse solutions fulfils all the needs of the 3TR project. Most notably, none comes with automatic file versioning. Moreover, while except for Opal all the examined solutions provide connectivity to third-party visualisation tools, only SODAR and Terra support interactive single cell data visualisation. After this initial assessment, the potential applicability of tranSMART, Opal and MOLGENIS has been examined in more depth by installing and running these tools locally, to see if any of these could be adapted so it would fulfil all requirements.

Table 4.4: Applicability of existing data warehouse solutions for the 3TR project

Features	tranSMART	Opal	MOLGENIS EMX2	SODAR	Terra
Two-factor authentication (2FA)	No	Yes	Yes	No	Yes
File upload via the UI	Yes	Yes	Yes	No	Yes
Automatic file versioning	No	No	No	No	No
Run Analyses	Yes	Yes	Yes	Yes	Yes
Interactive data visualisation	Yes	No	Yes	Yes	Yes
Interactive Single cell data visualisation	No	No	No	Yes	Yes

tranSMART

tranSMART looked promising at first, as for example the private company *The Hyve* (thehyve.nl), a european data and informatics provider for life science organisations has built glowing bear (github.com/thehyve/glowing-bear), a commercial software for clinical data selection and data analysis, based on tranSMART . However, tranSMART seems to have been discontinued. In fact, the last update was in November 2022. Likewise, the latest containerised version of tranSMART v.19.1 (github.com/tranSMART-Foundation/transmart-docker) appears to be no longer fully functional. As shown in Figure 4.1 (a) the file upload functionality was not working. Thus, tranSMART was disregarded for the 3TR project.

Opal

OBiBa's Opal software stack was the one which was explored in more depth. In fact, at the onset of the 3TR project it was planned to use Opal for data upload/sharing and then building on top of it. Thus, it was deployed on CNAG's infrastructure for testing purposes. However, quickly realised that its flexibility in terms of data uploading, was a disadvantage for the 3TR project. Stakeholders wanted a more controlled environment where the data upload process is more guided. Meaning its allowance of nested folder structure was confusing for the data uploaders.

However, the critical limitation was the absence of an accurate indication of the file upload progress. While there is a fine red line extending from left to right in the very top visible after having selected a file to be uploaded (Figure 4.1 b) its initial progress speed does not seem to correlate with the file size. In fact, for larger files (tested with 10 GB) the bar is progressing as quickly as for smaller files (e.g., 25 MB). The only apparent difference is that when uploading a large file, the bar does not seem to reach the end. Thus, for the data uploader it is (i) impossible to estimate how long the upload will take and (ii) to know if the upload is still ongoing or got stuck. Since one of 3TR's requirements was to be able to upload large files of more than 10 GB Opal was ultimately disregarded as well.

MOLGENIS EMX2

In contrast to Opal MOLGENIS EMX2 provides accurate feedback about the file upload progress (Figure 4.1 c), but it seems not to support the upload of large files. The file upload process seems to fail silently when trying to upload a 10 GB file. Thus, MOLGENIS EMX2 was also not further considered for the 3TR project.

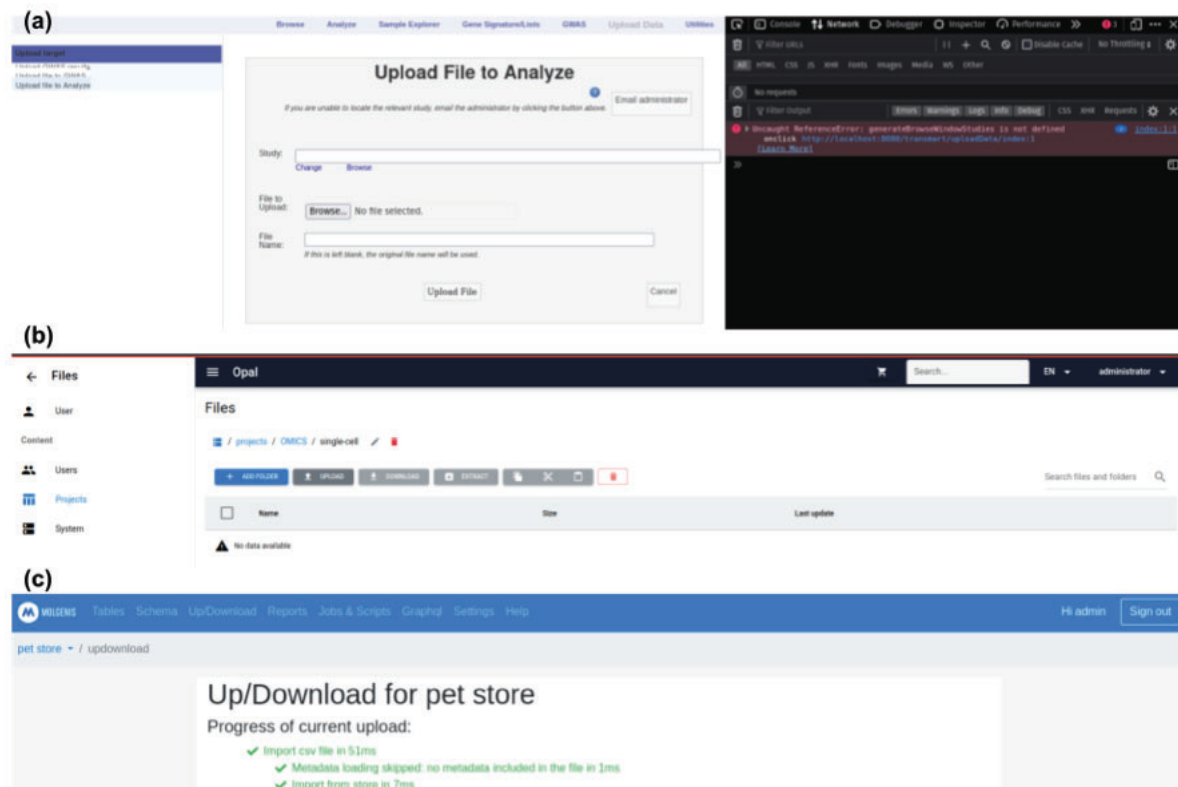


Figure 4.1: Screenshots when trying to upload a 10 GB file to selected data warehouse solutions. (a) Self hosted transSMART. (b) Public Opal demo (opal-demo.obiba.org). (c) Self hosted MOLGENIS EMX2.

4.1.3 Modules of OmicsDM

As none of the existing data warehouse solutions fulfilled the criteria for the 3TR project, I developed a new solution, called OmicsDM. This web application, allows users to upload, share, analyse, and visualise molecular and pheno-clinical data. It consists of three modules:

- ◆ Core: Data submission, sharing and download
- ◆ Visualisation: Single cell data visualisation using CZ CELL×GENE Annotate
- ◆ Analysis: RNA-Seq data analysis using DESeq2, fgsea, and g:Profiler2

At the time of compilation of the thesis, the core module was already up and running for the 3TR project. The analysis and visualisation modules were both available on the playground version of OmicsDM.

The usage flow is as follows. First of all, the user has to login with their password and a 2FA token. Then, the user can upload their data to the data warehouse's S3 bucket (introduced in Section 3.4.1). Once the data is uploaded the user can share the data with other partners, if desired. In the case of single cell data the user can further explore the data using CZ CELL×GENE Annotate. In the case of RNA-Seq data, there is also the possibility to run analyses. Due to OmicsDM's modular architecture other types of analyses can easily be plugged into the system as well. A schematic overview of the connected services and its underlying technicalities is shown in Figure 4.2.

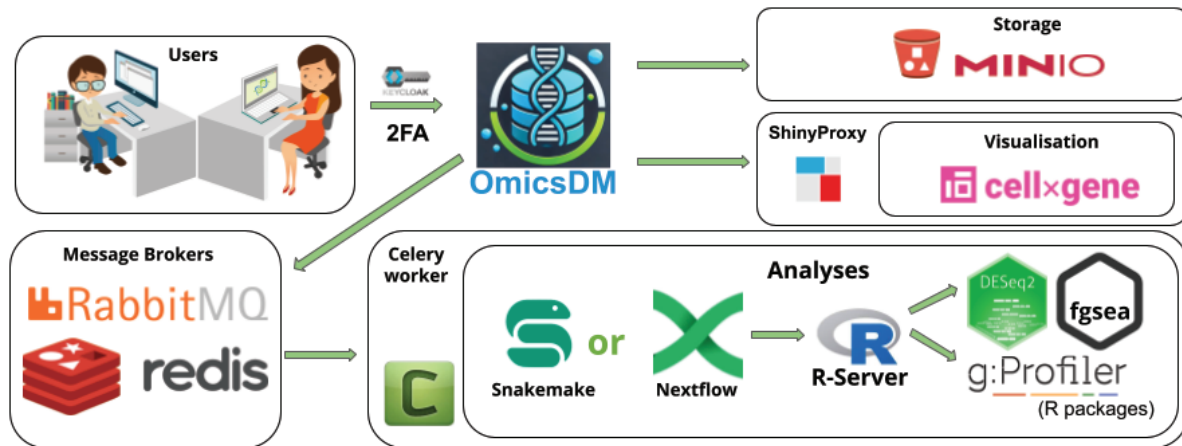


Figure 4.2: Overview of the connected services in OmicsDM and its underlying technologies. After authentication with 2FA users can (i) upload data to the Minio S3 bucket, (ii) visualise single cell data in CZ CELL×GENE Annotate, which instances are orchestrated with ShinyProxy, (iii) run Snakemake or Nextflow analysis pipelines, scheduled with Python Celery. RabbitMQ and Redis serve as message brokers to coordinate task execution.

4.1.3.1 OmicsDM core module

The core functionalities of OmicsDM are (i) uploading of molecular and pheno-clinical data along with queryable metadata to the 3TR data warehouse, (ii) group based data sharing and (iii) presigned URL (introduced in Section 3.4.2) mediated downloading of files. Figure 4.3 shows the respective UI elements. To not end up with a complex nested folder structure and to be able to easily filter the uploaded files, we decided to have a three-step data upload process (Figure 4.3 a-c). Note, to ensure consistent naming of the projects and compliance to the 3TR consortium rules such that all newly produced data needs to be accessible to all partners, we decided to only give OmicsDM administrators the right to create projects (Figure 4.4). As soon as the project is created, the first step for the dataset owners is to create the dataset by (i) filling in the table with dataset-specific metadata, such as the sample type and (ii) uploading the corresponding pheno-clinical data (Figure 4.3 b). Once this is

completed, the second step is to upload the molecular data (Figure 4.3 c). In the case of legacy data, there is a third step, namely the option to provide data access rights for selected datasets, along its associated files, to specific consortium partners (Figure 4.3 d). Finally, the users can browse through all the accessible files in a filterable HTML table (Figure 4.3 e). Clicking the download files button overlays another table revealing the associated presigned URLs with the following pathname structure: <dataset_owner_keycloak_group>/<dataset_id>/<filename>. For security reasons we decided to not give users the right to modify any filled in data. Instead if any correction of the submitted metadata would be desired the users need to ask the OmicsDM administrators. The views to carry out these edits (Figure 4.5) can be accessed, when logged in as admin, via OmicsDM's sidebar (Figure 4.4 bottom left).

a) Select Project (Projects are created by the admins)

b) Create Dataset

Create datasets

Upload a template as csv (tab separated) to prefill the table:

[Browse...](#) No file selected.

[DOWNLOAD TEMPLATE](#)

+1 +10 +100 VALIDATE SUBMIT RELOAD INSTRUCTIONS

Dataset ID*	Name	Disease*	Treatment*	Molecular Information*	Sample Type*	Data Type*	Value Type*
test	test	ASTHMA	drug1	info1	tissue	microarray	integer

Platform* Genome Assembly* Annotation* Samples Count* Features Count* Features ID* Healthy Controls Included*

illumina hg38 annotation1 100 100 1 true

Contact* Tags Visibility Data Usage Policy File Clinical File

ny@cnag.eu drug1 visible to all [Browse...](#) data_policy.pdf [Browse...](#) 3TRBatch1Clinical_LABELS_2024-02-14_1551.csv

File data_policy.pdf Progress 0 % File 3TRBatch1Clinical_LABELS_2024-02-14_1551.csv Progress 0 %

c) Submit Files

Dataset ID*	File*	Comments
test	Browse... COUNTS_genes_TEST_134_135_summed.csv File COUNTS_genes_TEST_134_135_summed.csv Progress 0 %	
test2	Browse... 2 files selected. File COUNTS_genes_TEST_134_135_summed.csv Progress 0 % File info_TEST_134_135.csv Progress 0 %	

d) Share Datasets

SHARE/UNSHARE SELECTION SHOW ASSOCIATED FILES

1 of 2 selected

Filter by Dataset ID test Filter by Project ID proj1

Dataset ID	Project ID
test	proj1
cnag	proj2

Share/Unshare Selected Datasets

Selection of your own Datasets to apply the action: share with groups

Changes will be applied to all files associated to the selected Datasets

ALL

Datasets ID	Current Visibility	Current Group
test	private	3tr

NOT NOW UPDATE TABLE

e) Download Files

DOWNLOAD FILES DELETE FILES

1 of 2 selected

Filter by Project ID Filter by Dataset ID Filter by File Name Filter by Shared with

Project ID	Dataset ID	File Name	Shared with
3tr	test	info_TEST_134_135.csv	ALL GROUPS
3tr	test	COUNTS_genes_TEST_134_135	ALL GROUPS

Figure 4.3: UI elements of OmicsDM's core module. From the top left to the bottom right: (a) project selection, (b) dataset creation, (c) file upload, (d) dataset sharing and (c) file download.

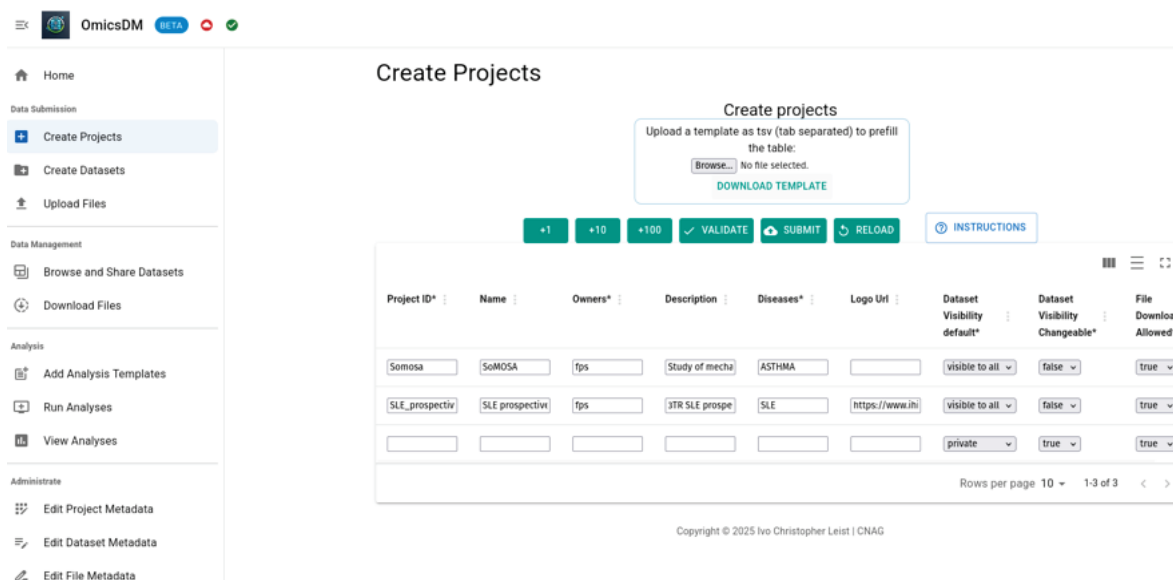


Figure 4.4: Screenshot of the project creation in OmicsDM. Note that this view is only accessible when logged in with an administrator account.

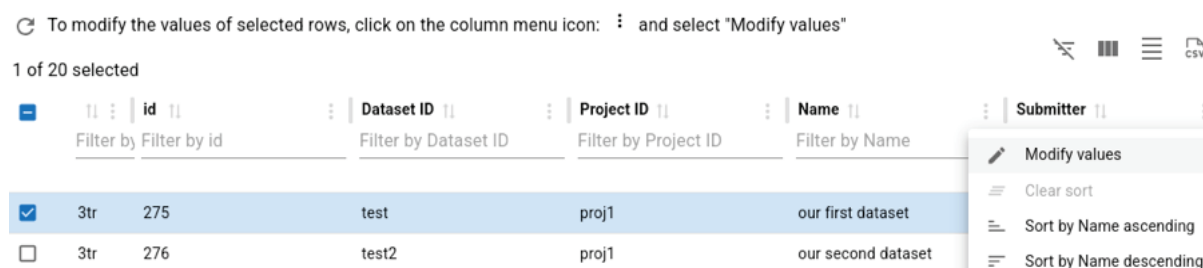


Figure 4.5: Administrator view to modify metadata values. The table is filterable by the fields in the top row. Here no filters are applied. By selecting the checkboxes in the first column the administrator has the option to bulk edit the values in a specific column. The modify values option is integrated in the column menu, accessible by the “.”.

Feasibility experiments of large file upload via a web UI

Uploading large files via the UI is a common problem in web application development. To address this issue, I conducted several experiments to find the best way to upload large files to OmicsDM (Figure 4.6). The experiments showed that as expected trying to upload a large file via the web interface without chunking led to the browser crashing. Most likely due to potential memory limits of the browser/the client machine. Thus, uploading a whole file at once is not a feasible solution. So the next experiment was to chunk the file and upload it in smaller parts. However, this seems to only have moved the bottleneck from the client to the Python Flask server. As now the server was crashing due to the high number of requests, especially when multiple users tried to upload large files at the same time. Thus the last and successful experiment was to skip the Python Flask server and directly upload the chunks to the S3 bucket (introduced in Section 3.4.1). This way the server was not involved in the upload

process, but only in creating the presigned URLs for the chunks. Thus, for the large file upload in OmicsDM the last approach was chosen. The actual implementation is described in (Section 3.4.2).

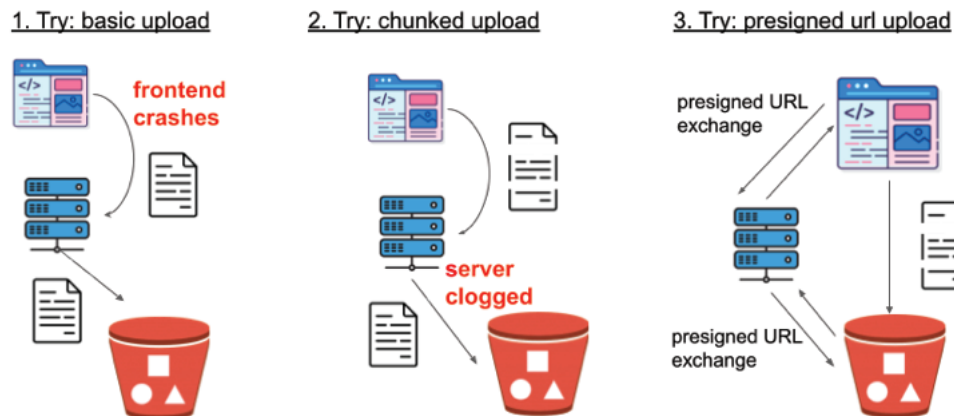


Figure 4.6: Experiments on how to upload large files to OmicsDM.

4.1.3.2 OmisDM visualisation module

At the moment of writing the thesis the visualisation module is focused on single cell data, but other types of data visualisations can easily be plugged in. Before starting the development of the single cell data visualisation module the most popular single cell data visualisation tools regarding their applicability to the 3TR project were assessed. Tools worth to be included in the evaluation were selected based on search engine results and literature research. Of the latter, the most valuable source was a review article by Cakir *et al.* 2020 [91]. In it 13 different single cell data visualisation tools were compared based on various criteria. Ranging from ease of use, supported file formats, platform to maintenance status (Figure 4.7).

	ASAP	Bbrowser	cellxgene	Granatum	iSEE	Loom viewer	Loupe Cell Browser	SCope	scSVA	scVI	Single Cell Explorer	SPRING	UCSC Cell Browser
Web Sharing	✓		✓		✓	✓		✓	✓		✓	✓	✓
Interactivity	✓	✓	✓		✓		✓	✓	✓		✓	✓	✓
Docker	✓		✓		✓			✓	✓				
Cloud Support			✓	✓				✓	✓			✓	
SaaS	✓			✓				✓			✓	✓	✓
Loom	✓				✓	✓		✓	✓	✓	✓		✓
h5ad		✓	✓						✓	✓	✓		✓
SCE					✓								
Seurat		✓									✓		✓
csv/txt	✓	✓		✓					✓	✓	✓	✓	✓
Platform	Java/ R	Desktop	Python	R	R	Python	Desktop	Python	R	Python	Python	Python	Python
Last updated*	>3 m	<1 m	<1 m	>12 m	<1 m	>6 m	>3 m	<1 m	>12 m	<1 m	>6 m	>12 m	<1 m
Number of contributors**	2	?	19	3	7	5	?	5	2	24	1	3	6
Number of active developers**	0	?	6	0	3	0	?	2	0	4	0	0	2
Reviewed Version	1.0	2.2.13	0.11.0	0.0.0.900	1.4.0	0.32.4	3.1.1	1.7.2	0.2.0	0.5.0	1.0.0	1.6.0	0.5.49

Figure 4.7: Tabular comparison of single-cell data visualisation tools. Annotated with arrows pointing to (a) the most relevant criteria for the decision-making process of which tool to integrate in OmicsDM and (b) the tools being selected for future evaluation. Among the latter, the colour green indicates the tools which are considered for a heads-up comparison, namely cellxgene (renamed since 2022 to CZ CELL×GENE Annotate) and UCSC Cell Browser. Adapted from [91].

After the intersection of their findings and the needs of the 3TR project such as web sharing and support of the h5ad file format the following tools remained: cellxgene [74], scSVA [121], Single Cell Explorer [122], and USCS Cell Browser [123]. Of these scSVA and Single Cell Explorer were ruled out quickly since these are no longer maintained^{4,5}. Thus, the final decision was between cellxgene and USCS Cell Browser, which both are actively maintained. In fact, cellxgene even matured to an ecosystem of tools, reflected by its renaming to CZ CELL×GENE Annotate in 2022. Overall, it is more feature-rich, while providing a more modern look and feel in comparison to USCS Cell Browser (Figure 4.8). Another distinguishing feature of CZ Cell×GENE Annotate is the support of the selection of two groups of cells (Figure 4.8 CZ Cell×GENE Annotate top middle) for differential gene expression calculation. Thus, CZ

4 github.com/klarman-cell-observatory/scSVA

5 github.com/d-feng/scExplorer

CELL×GENE Annotate is not only a static viewer as USCS Cell Browser, but also allows the user to perform some basic analyses.

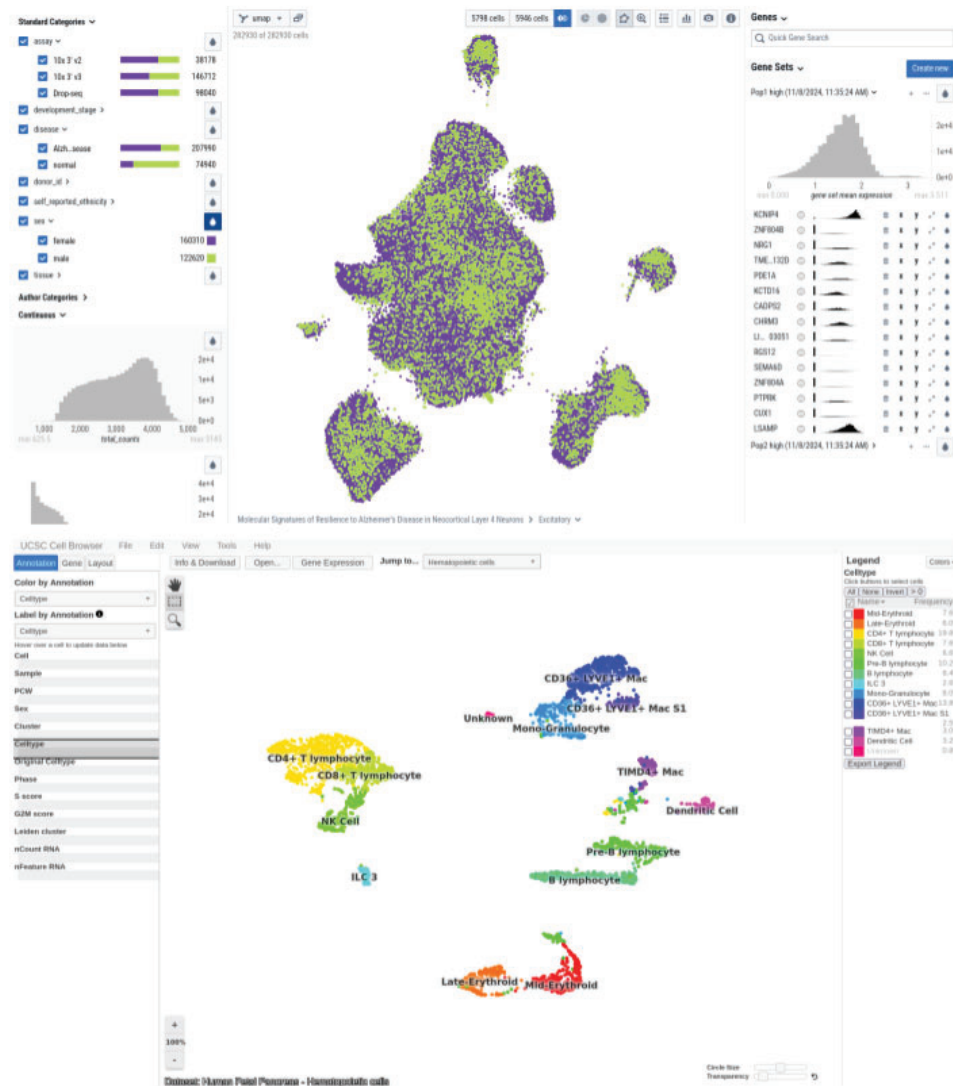


Figure 4.8: Comparison of the single-cell data visualisation tools: CZ CELL×GENE Annotate (top) and UCSC Cell Browser (bottom). Note the feature richness of CZ CELL×GENE Annotate, in combination with a more streamlined UI in comparison to USCS Cell Browser. Screenshots taken from the respective hosted services: cellxgene.cziscience.com and cells.ucsc.edu.

Moreover, CZ CELL×GENE Annotate is well adopted by various research organisations, as apparent by the existence of various community extensions⁶. One noteworthy example is the Visualisation in Plugin (VIP) [92], which enables the generation of eighteen common quality control and analytical plots directly in CZ CELL×GENE Annotate. This extensibility potential in concert with the aforementioned aspects lead to the final decision to integrate CZ CELL×GENE Annotate into OmicsDM.

⁶ cellxgene.cziscience.com/docs/05__Annotate and Analyze Your Data/5_7__Community Extensions

Users interested in exploring single cell data directly in OmicsDM can do so by clicking the CZ CELL×GENE Annotate logo in the files overview table (Figure 4.9). In the so triggered execution flow, first it will be ensured that the user is authorised to access the selected file. Second, ShinyProxy will launch a CZ CELL×GENE Annotate container (as explained in Section 3.4.5.2) and finally, the user gets redirected to that instance. All custom annotations and gene sets created by the users are stored in an S3 bucket. Consequently, each user can always continue their exploration from where they left off.

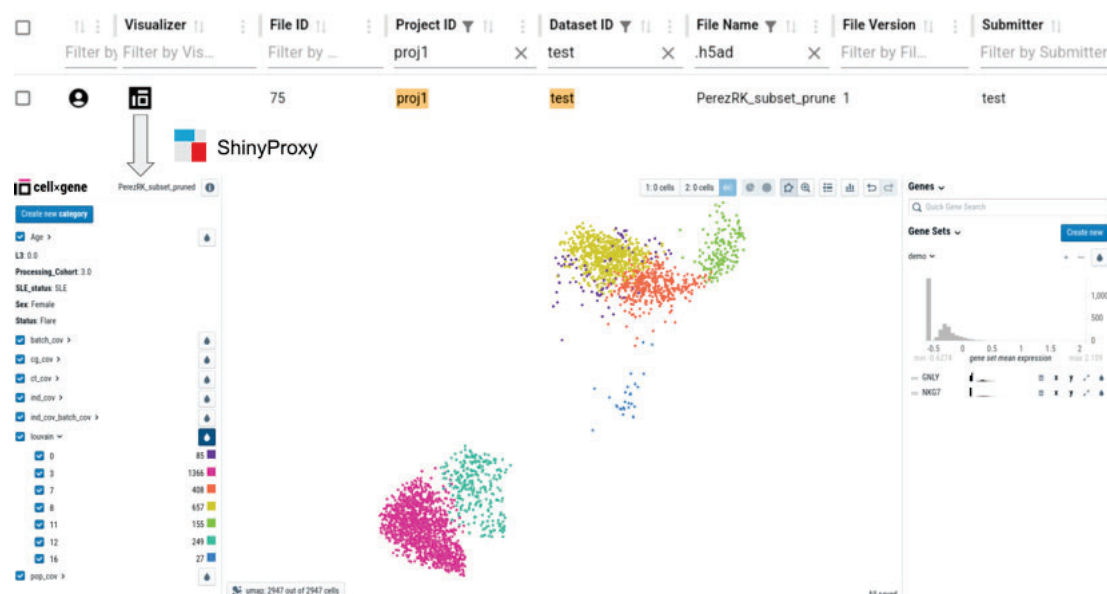


Figure 4.9: Screenshots of OmicsDM's single cell data visualisation module. From top to bottom: (top) uploaded file view (bottom) CZ CELL×GENE Annotate.

4.1.3.3 OmicsDM analysis module

The analysis module of OmicsDM enables users to analyse the data they have uploaded to the data warehouse. It is designed to be workflow manager agnostic allowing the use of different workflow managers such as Snakemake or Nextflow. Job scheduling is handled by Celery, and analysis results are stored in the data warehouse for future reference. It is currently tailored to an RNA-Seq analysis use case. Table 4.5 shows the supported data analyses for bulk- and scRNA-Seq data, respectively the expected input data (refer back to Section 3.4.6.3 for the structure of the GMT file format). Note, that the module's architecture allows to run any kind of analysis script be it for instance in Python, R, or Perl, as long it is provided in a containerised form.

Table 4.5: Supported analyses in the data analysis module of OmicsDM.

Analysis	Expected input data	Utilised package
Differential gene expression (DGE)	bulk RNA-Seq	R DESeq2 [112]
GO enrichment analysis	DGE output + optional a GMT file	R g:Profiler2 [114]
Gene set enrichment analysis	DGE output + GMT file	R fgsea [116]
Gene set variation analysis (GSVA)	DGE output/scRNA-Seq + GMT file	R escape [117]
Average z-scores per geneset	DGE output/scRNA-Seq + GMT file	Python Scanpy [75]

These analyses can be configured intuitively via a web interface in a three-step process. First, the files (file versions) to be included in the analyses have to be selected (Figure 4.10). Second, the user can choose which of the supported analyses (Table 4.5) should be included in the analysis pipeline. Per selected analysis (Figure 4.11 a left) the inbuilt JSON editor (Figure 4.11 a right) will be prepopulated with the possible parameters to be set for each analysis. One important default parameter is “bases_on”, allowing the user to pass analysis results downstream to another analysis. As it is the case in the exemplary pipeline shown in Figure 4.11. Here the DESeq2 results are the expected input for the follow up analysis z-scoring. Note, it is also possible to reuse existing output if it is for instance desired to just run another follow up analysis. Another interesting parameter is “integrate_output_into_sc_dataset”. Setting it to “true” allows the user to later on explore the corresponding results in CZ CELL×GENE Annotate.

Alternatively to the described stepwise prepopulation, there is the possibility to load the analysis configuration of a former analysis (Figure 4.11 - Tab Finished analyses). Either way, the whole configuration file can be directly modified within the UI and only if the automatic validation is passed the user can submit the analysis pipeline.

File ID	Owner	Project ID	Dataset ID	File	Version
73	3tr	proj1	test	COUNTS_genes_TEST_134	1
74	3tr	proj1	test	info_TEST_134_135.csv	1
76	3tr	proj1	test	modules.gmt	1
75	3tr	proj1	test	PerezRK_subset_pruned.h5	1

Figure 4.10: Screenshot of the file selection in OmicsDM's data analysis module.

1 Select Files 2 Configure Pipeline 3 Submit Analysis

As a starting point please use the prefilled templates or take a finished analysis

TEMPLATES FINISHED ANALYSES

RESET

1. Select an analysis type

bulk RNA-seq scRNA-seq

2. Select a base analysis

deseq2 USE PREVIOUS RESULTS

3. Select one or multiple follow ups

getgo fgsea

gsva z-scoring

```

1 {
2   "deseq2": {
3     "analysis": "deseq2",
4     "files": {
5       "counts": "COUNTS_genes_TEST_134_135_summed.csv",
6       "info": "info_TEST_134_135.csv"
7     },
8     "options": {
9       "bases_on": null,
10      "contrast": {
11        "TEST_135_100_vs_TEST_134_500":
12          "group,TEST_135_100,TEST_134_500",
13        "TEST_135_100_vs_TEST_135_500":
14          "group,TEST_135_100,TEST_135_500",
15        "TEST_135_500_vs_TEST_134_500":
16          "group,TEST_135_500,TEST_134_500"
17      },
18      "control": "TEST_134_500",
19      "covariates": {},
20      "group": "GROUP",
21      "integrate_output_into_sc_dataset": false,
22      "nod": "-group",
23      "onlypca": "FALSE",
24      "plot": "TRUE",
25      "plot_atr": {
26        "de_genes_n": 50,
27        "heatmap_ann": "group",
28        "pca": "group"
29      },
30      "project": "TEST_134_135"
31    },
32    "report": "deseq2"
33  },
34  "z-scoring": {
35    "analysis": "zscoring",
36    "files": {
37      "gnt": "c2.cp.kegg_medicus.v2024.1.Hs.symbols.gnt",
38      "h5ad": "PerezRK_subset_pruned.h5ad"
39    },
40    "options": {
41      "bases_on": "deseq2",
42      "integrate_output_into_sc_dataset": true
43    },
44    "report": "zscoring"
45  }
46 }

```

Figure 4.11: Screenshot of the analysis pipeline configuration in OmicsDM's data analysis module.

Depending on the current load of OmicsDM the pipeline will be executed directly or scheduled. In both cases there is no need to keep the browser tab open. The user can just copy the URL to check the status later. The analysis status, respectively results

page is divided in three parts. The top shows a flow diagram of the pipeline (Figure 4.12 a). In the middle is an expandable section revealing the used files, as well as the employed configuration. Finally, the tabbed view below displays, while an analysis is still running, its execution logs. Once an analysis is completed a “downloads results” button will appear. Pressing it triggers an overlay window showing the presigned URLs of the individual analyses. Following these links will start the download of a zip archive file containing the generated files and the metadata of the corresponding analysis. Among these files there is also a self-contained HTML, comprising: (i) an overview of the created files, (ii) an embedded PDF viewer showing the created plots, (iii) the session info stating the used R/Python packages (versions). These reports are also explorable within the aforementioned tabbed view (Figure 4.12 b).

While pure bulk RNA-Seq analyses are limited to a static report, analyses which include single cell data can also leverage the interactive capabilities of OmicsDM’s visualisation module. Thus, allowing the users to explore for instance z-scoring results directly in CZ CELL×GENE Annotate (Figure 4.13). To enable that functionality the previously described parameter “integrate_output_into_sc_dataset” needs to be set to “true” in the analysis configuration. If that is the case the CZ CELL×GENE Annotate icon will be visible in the top right of the results page (Figure 4.12 b). Clicking it will start an instance (as described in Section 4.1.3.2).

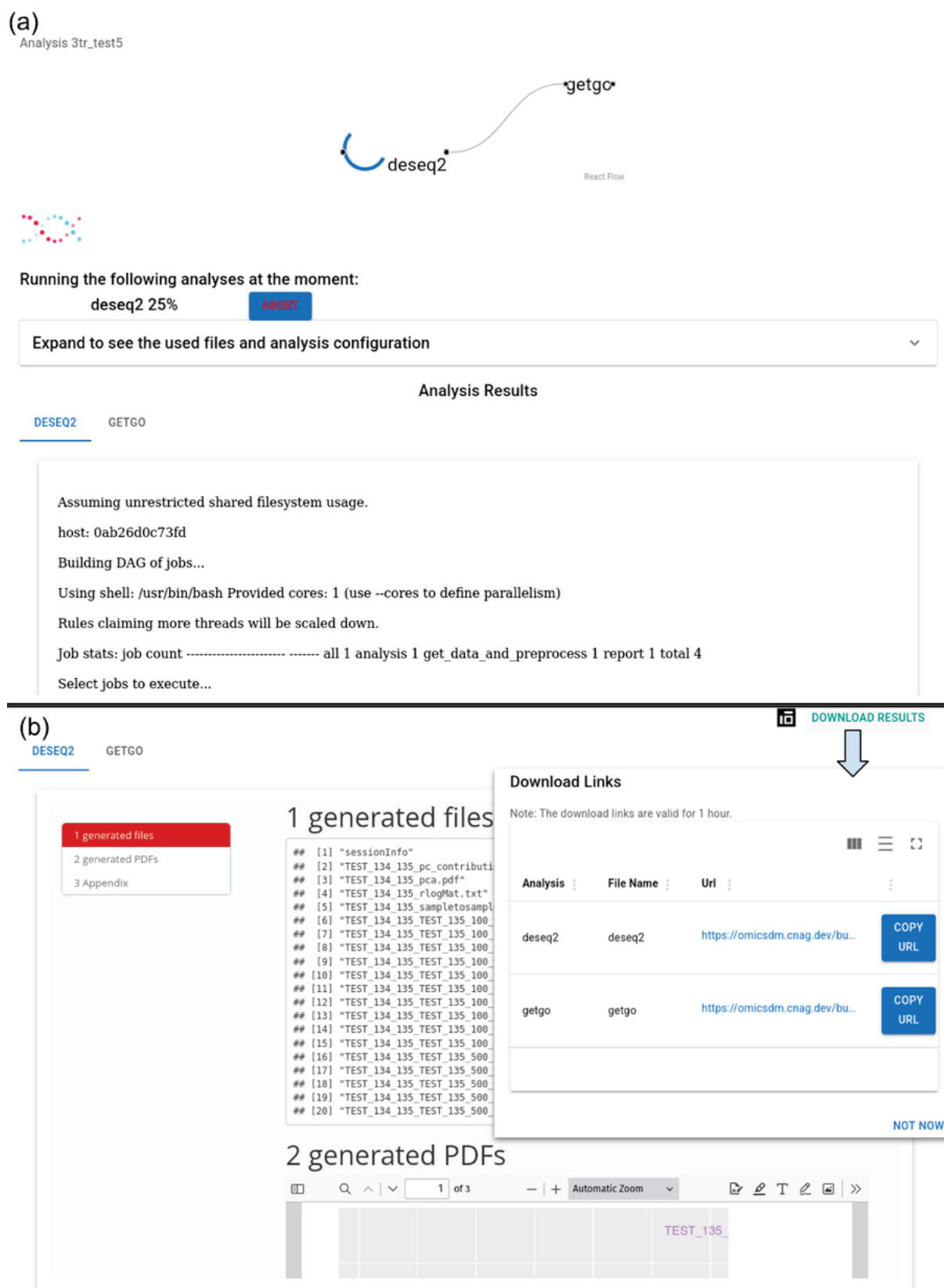


Figure 4.12: Excerpts of the analysis pipeline status/results view in OmicsDM's data analysis module. (a) One analysis within the pipeline just started. (b) Pipeline completed.

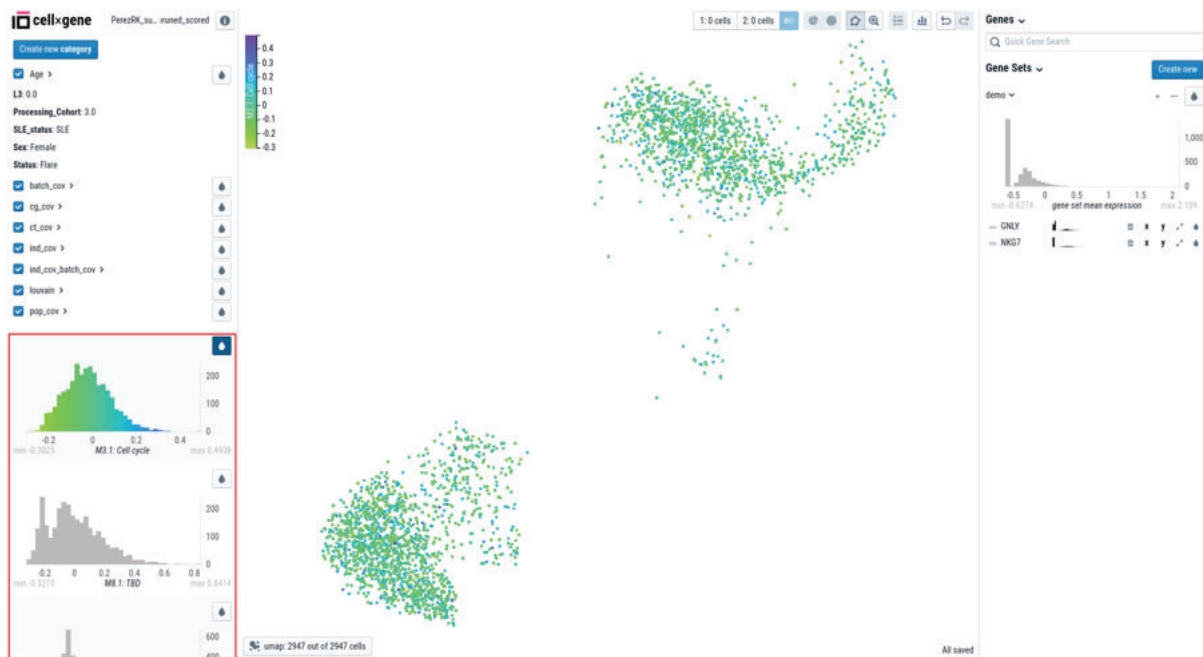


Figure 4.13: scRNA-Seq dataset with incorporated average z-scoring per module. The red box shows the average z-score distribution per module.

4.1.4 Database schema of OmicsDM

The database schema (Figure 4.14) sitting behind all the functionalities of OmicsDM contains four main entities: projects, datasets, files, and analyses. Each dataset and its files, belongs exclusively to one project, but can be involved in multiple analyses. In the analysis entity there is an association to files as well as datasets, which allows to run an analysis on a specific dataset (collection of files) without including all the files of the respective dataset. In the main entities there is the attribute `extra_cols`. This attribute was introduced to store additional metadata in a flexible way, allowing the deployment of OmicsDM for different use cases without having to change the database schema. The attributes defined in `extra_cols` relevant for the 3TR project are shown in Table 4.6.

Table 4.6: 3TR project specific entity attributes (abc:character; 1 2 3 4:integer; ✓:boolean; 🔗:URL)

Entity	Extra attributes (<i>extra_cols</i>)
Projects	description abc, diseases abc, logo_url 🔗, dataset_visibility_default abc, dataset_visibility_changeable ✓, file_dl_allowed ✓
Analyses	None
Files	comments abc
Datasets	disease abc, treatment abc, molecular_info abc, sample_type abc, data_type 1 2 3 4, value_type abc, platform abc, genome_assembly abc, genome_annotation abc, samples_count 1 2 3 4, features_count 1 2 3 4, features_ids abc, healthy_controls_included ✓, additional_info abc, contact abc, tags abc, pheno-clinical_data_file 🔗, extra_policies_file 🔗

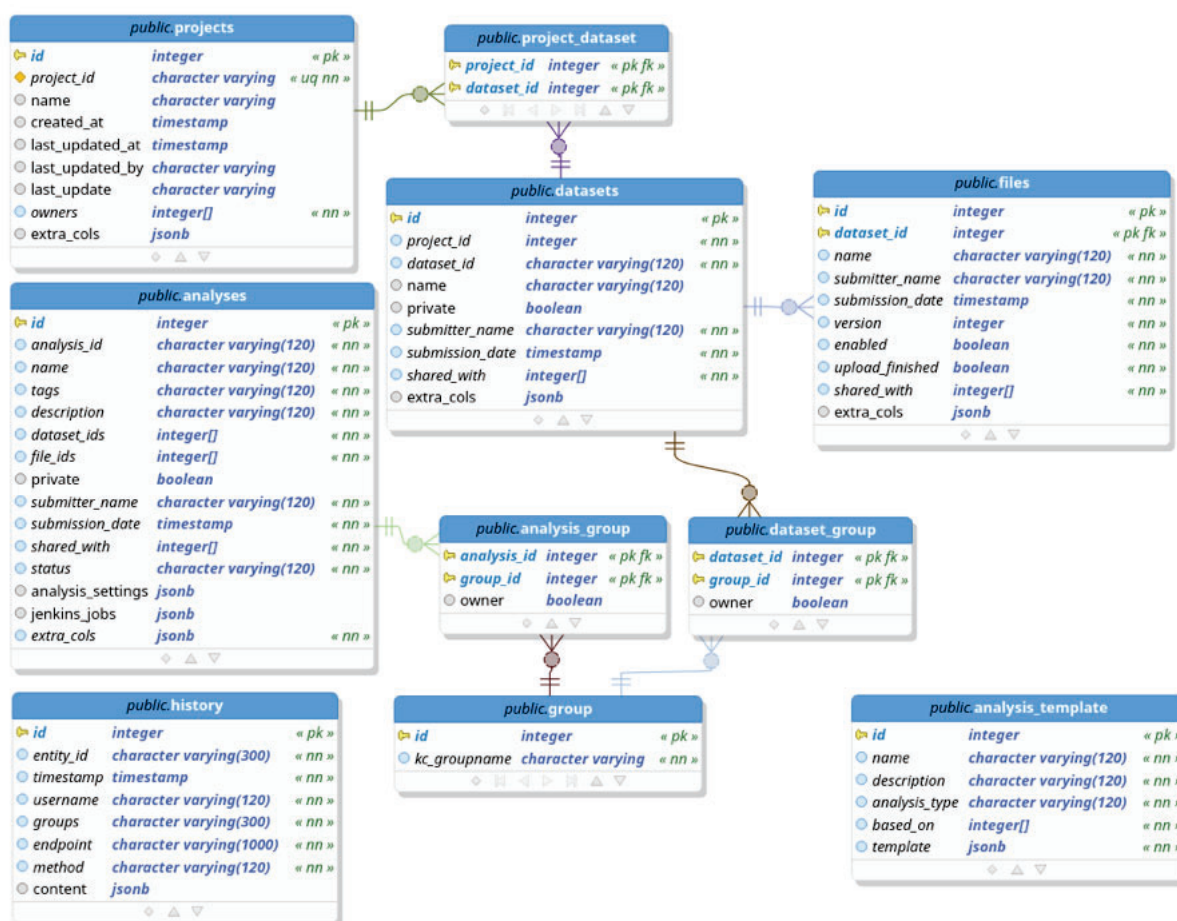


Figure 4.14: Database schema of OmicsDM. Each box shows an entity (table), with its attributes (columns) and their required data types such as integer. The role or constraints of an attribute are shown by the icon in front of the attribute name, respectively behind it in green (key[pk] = primary key, fk = foreign key, diamond[uq] = unique, blue dot [nn] = not null). Relations between the entities are shown by lines between the boxes, whereby the type of the relation is indicated by the symbols (crow's foot notation). Here Mandatory-One to Optional-Many.

One of the core functionalities of OmicsDM, namely the sharing of files is leveraged by the *group* entity. This entity contains the groups defined in the employed identity and access management system Keycloak (Section 3.4.3). In the 3TR context a group is a collection of users belonging to the same institute or company. The *groups* table gets populated by the first action of a user after logging in by extracting the group information from the Keycloak token. The id assigned to the group name is subsequently utilised across multiple entities. Foremost, it is used to define which groups own the by the administrator created projects (Section 4.1.3.1) and thus can create datasets in it, and upload files to its datasets. Further, the group id is also used in the entities *dataset_groups* and *analyses_groups* so the system knows which groups own a dataset, respectively an analysis. With regards to sharing datasets, files and analyses the group is utilised in the attributes *shared_with* of the respective

entities. However, it has to be noted that shared with on the file level is immutable. Meaning, file sharing can only be done on the dataset level. This design decision was made to prevent the user from sharing individual files since in biology a single file is often not meaningful without the context of the other files. As for instance an RNA-Seq experiment cannot be analysed without the corresponding pheno-clinical data. Finally, the group entity is also used in the history entity, where all actions performed in OmicsDM are stored. While the history entity is not directly needed for the core functionalities of OmicsDM it can be valuable for auditing purposes. Thus, it was made immutable. Meaning that once an action is stored in the history table it can no longer be changed or deleted. Allowing the 3TR partners to get potentially audit relevant information such as who has downloaded which of their files.

Finally, the schema also shows one seemingly unconnected entity, namely `analysis_template`. The latter plays a role in the Analysis module of OmicsDM, precisely the running of analyses from the web interface (Section 4.1.3.3), since it stores the default configuration of the analysis pipelines. All the aforementioned entities were designed to be as flexible as possible. These design decisions were taken with the hope that these will foster the adoption of OmicsDM for different use cases without the developers having to change the database schema.

4.2 Standalone tools for pheno-clinical data exploration

I actively participated in the development of two tools that facilitate the exploration of pheno-clinical data in the context of the 3TR project: Convert-Pheno, a software toolkit for the interconversion of standard data models for phenotypic data and Pheno-Ranker, which is specifically designed for semantic similarity analysis of phenotypic and clinical data. The core of both tools are Perl modules that are run through command-line interfaces, developed by laboratory member Manuel Rueda. In both endeavours my main contribution was not limited to the development of the web UI but also included the design of the software architecture underlying the public playground of each tool. To the corresponding publications I contributed various figures such as UI screenshots and software architecture overviews.

4.2.1 Convert-Pheno-UI

The Convert-Pheno-UI is a web interface for the Convert-Pheno tool. It allows users to convert pheno-clinical data from different formats to GA4GH standards for phenotypic data exchange formats such as BFF or PXF (introduced in Section 1.3.4.1). The user can upload their data in different formats such as REDCap or OMOP-CDM and select the desired output format (4.15 top). The data will then be converted, and the user can download the converted data in the desired format. Additionally, the user can explore the converted data by filtering the data represented in HTML tables (Figure 4.15 bottom).

1)

Input Format: ☒ REDCap, ☐ BFF (Beacon v2), ☐ PXF (Phenopackets v2), ☐ OMOP-CDM, ☐ CDISC-CDM, ☐ OpenEHR (coming soon)

Output Format(s): ☒ BFF (Beacon v2), ☒ PXF (Phenopackets v2), ☐ OMOP-CDM (coming soon)

Conversion finished Job ID: 1681323254

Input Format: RedCap

Output Format(s): BEACON V2, PHENOPACKETS V2 (DOWNLOAD ALL)

2)

HIDE QUERY BUILDER

Below are the table columns headers drag and drop them into one of the colored fields below and then enter the values to filter the table

Columns: diseases, ethnicity, exposures, id, interventionsOrProcedures, measures, phenotypicFeatures, redcap_event_name, sex, study_id, treatments

Drag inclusion criteria here: ethnicity: Caucasian, treatments: Budesonide

Drag exclusion criteria here: sex: Female, interventionsOrProcedures: Fistula Repair

APPLY FILTER

SHOW HIDE TABLE COLUMNS

#	id	study_id	redcap_ev...	diseases	sex	ethnicity	interventi...	exposures	measures	treatments	phenotypi...
1	107:week_...	107	week_0_ar...	Inflammat...	Male (NCIT...	Caucasian ...	Colon Res...	Alcohol.S...	Leukocyte...	Budesonid...	Asthma.N...
2	107:week_...	107	week_14_...	Inflammat...	Male (NCIT...		NA_NA		Leukocyte...		NA

Figure 4.15: Convert-Pheno-UI screenshots of the input (1) and output (2) stages. From [72].

The UI has been open sourced (github.com/CNAG-Biomedical-Informatics/convert-pheno-ui) and published together with Convert-Pheno's CLI in the Journal of Biomedical Informatics. A playground version is available at: <https://convert-pheno.cnag.cat>. The article: "Convert-Pheno: A software toolkit for the interconversion

of standard data models for phenotypic data” can be found online doi.org/10.1016/j.jbi.2023.104558 or below.



Original Research

Convert-Pheno: A software toolkit for the interconversion of standard data models for phenotypic data

Manuel Rueda^{a,b,*}, Ivo C. Leist^{a,b}, Ivo G. Gut^{a,b}

^a Centro Nacional de Análisis Genómico, C/Baldiri Reixac 4, 08028 Barcelona, Spain

^b Universitat de Barcelona (UB), Barcelona, Spain



ABSTRACT

Efficient sharing and integration of phenotypic data is crucial for advancing biomedical research and enhancing patient outcomes in precision medicine and public health. To achieve this, the health data community has developed standards to promote the harmonization of variable names and values. However, the use of diverse standards across different research centers can hinder progress. Here we present Convert-Pheno, an open-source software toolkit that enables the interconversion of common data models for phenotypic data such as Beacon v2 Models, CDISC-ODM, OMOP-CDM, Phenopackets v2, and REDCap. Along with the software, we have created a detailed documentation that includes information on deployment and installation.

1. Introduction

Structured storage of phenotypic and clinical data (“pheno-clinical”), which encompasses the organized collection of information pertaining to observable characteristics and medical aspects of individuals, plays a vital role in advancing biomedical research and enhancing patient outcomes [1]. However, pheno-clinical data are often scattered across various sources such as Excel files, PDFs, text files, databases, REDCap projects, Electronic Health Records (EHRs), etc., resulting in a fragmented information landscape [2]. Furthermore, many of these sources use proprietary formats, requiring users to export or dump data to gain full access [3]. Often, each research center must develop a unique approach to achieve its goals within specific constraints, resulting in a lack of standardization that makes cross-center studies challenging. For instance, one center may use the variable name “sex” and values “M” and “F” to define biological sex, while another may use “sexo” with values “Hombre” and “Mujer”, leading to errors and inconsistencies that increase with the complexity of the data. Moreover, human error and typos may occur without standardized vocabulary enforcement.

Enforcing health data standards is a practical approach to address the issue of inconsistent phenotypic data storage [1,4–9]. These standards provide guidelines or specifications for collecting, processing, transmitting, and maintaining health-related information in a consistent and interoperable manner. Typically, a standard includes a data model defined by a schema that organizes and relates data elements [10]. Note that the terms “data model” and “standard” are often used interchangeably due to overlapping definitions. The scientific community

has developed several standard data models for clinical information, such as OMOP-CDM [11], HL7-FHIR [12] for healthcare interoperability, DICOM [13] for medical imaging, and CDISC [14] for clinical research. Here we focus on text-based data models of clinical description and leave aside those for imaging data.

Standards can be viewed as a “higher” level of data harmonization, whereas ontologies play a critical role at a “lower” level by providing the vocabulary for expressing values within the data model. In particular, ontologies provide a formalized conceptualization of knowledge that specifies the concepts and relationships within a particular domain [15]. Examples of health data ontologies include Gene Ontology (GO) for molecular functions [16], Human Phenotype Ontology (HPO) for phenotypic abnormalities [17], Logical Observation Identifiers Names and Codes (LOINC) for identifying medical laboratory observations [18], and the National Cancer Institute Thesaurus (NCIt) for a general-purpose biomedical coding (<https://ncit.nci.nih.gov>). In summary, standardizing pheno-clinical data is essential to ensure consistency and accuracy, facilitating data sharing and integration across different research groups and platforms. This accelerates scientific progress and maximizes the potential impact of research findings.

Although standardizing data is a positive step forward, various research groups use different standards or coding schemes, creating barriers to data integration and sharing [1,6]. In addition, the complexity of the data models makes it challenging for non-experts to perform one-to-one mapping of variables. To address this challenge, we have developed Convert-Pheno, an open-source software that facilitates the interconversion of common data models for phenotypic data, without altering the

* Corresponding author.

E-mail address: manuel.rueda@cnag.eu (M. Rueda).

<https://doi.org/10.1016/j.jbi.2023.104558>

Received 11 July 2023; Received in revised form 24 November 2023; Accepted 27 November 2023

Available online 29 November 2023

1532-0464/© 2023 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

underlying ontologies, except to fill in missing terms in required fields. It accepts various input formats, including Beacon v2 Models [19], CDISC-ODM [14,19], OMOP-CDM (<https://www.ohdsi.org>), Phenopackets v2 [21], and REDCap projects [22]. By mapping all input data to the Beacon v2 Models [19] as a pivotal target model, Convert-Pheno can generate both Beacon v2 and Phenopackets v2 as output formats, ensuring the retention of the ontological integrity. This method represents a substantial advance in fostering compatibility and interoperability among various data systems, with plans to facilitate support for additional formats in future releases.

2. Materials and methods

2.1. Software implementation

2.1.1. Module

The core of Convert-Pheno is a Perl module that can also be used in Python programs through a pre-built Python binding (see Sup. Fig. 1 and documentation at <https://cnag-biomedical-informatics.github.io/convert-pheno/implementation>). The Perl Module is available through CPAN (<https://metacpan.org/pod/Convert::Pheno>).

2.1.2. Command line interface

Inspired by the *convert* utility from ImageMagick [23] and Open Babel [24], the command-line interface (CLI) was developed to handle text files, such as CSV or JSON/YAML. The CLI provides a variety of options to facilitate file conversions for the user (see more information at <https://github.com/CNAG-Biomedical-Informatics/convert-pheno>).

2.1.3. REST API

A lightweight REST API (REpresentational State Transfer Application Programming Interface) was designed to enable real-time (“on-the-fly”) web data conversions. Web data conversions are useful when used as a microservice within other REST API environments, such as the Beacon v2 API [25]. Within the Convert-Pheno GitHub repository, we provide two versions for the REST API, one written in Perl (that uses the Mojolicious web framework) and another implemented in Python (that uses the FastAPI web framework). The definition of the API is according to the OpenAPI 3.0.2 specification. The API validates the incoming data against the specification and an error message is generated if the JSON is not compliant. More details about this process can be found in the accompanying documentation (<https://cnag-biomedical-informatics.github.io/convert-pheno/use-as-an-api>).

2.1.4. Web application user interface

We also offer a web application user interface that provides an intuitive way to convert pheno-clinical datasets and generate searchable table outputs (see Fig. 1). The backend of the web application was implemented using Flask, a Python web framework, and the frontend was implemented using React, a JavaScript library for building user interfaces (See Supporting Text 1 and Supporting Fig. 2 for more details on the implementation).

2.1.5. Architecture and deployment

The user can choose to download Convert-Pheno as a standalone tool or together with the web application user interface. The standalone version can be installed directly from CPAN, from source or

containerized (see instructions at <https://cnag-biomedical-informatics.github.io/convert-pheno/download-and-installation>). To deploy Convert-Pheno along with the web application, we offer a Docker Compose file that is available at <https://github.com/CNAG-Biomedical-Informatics/convert-pheno-ui>.

2.1.6. Documentation

Additionally, we created an extensive documentation (built with *mkdocs* and deployed as GitHub pages at <https://cnag-biomedical-informatics.github.io/convert-pheno>) that provides technical details and supplements this paper. The documentation contains a link to Google Colab where users can interactively test the CLI and display results as tables without the need of performing any local installation of the software. As suggested by a referee, a backup copy of the Google Colab was stored within the GitHub repository at <https://github.com/CNAG-Biomedical-Informatics/convert-pheno/tree/main/nb>. This ensures continued availability and access to the Colab environment. The documentation also provides links to the deployment instructions.

2.2. Accepted formats

2.2.1. Inputs

2.2.1.1. Beacon v2 Models. Beacon v2 is an API specification for genomic and pheno-clinical data discovery and sharing across organizations and platforms [19]. It consists of two components, the *Framework*, and the *Models*. The *Framework* defines the format for the requests and responses, whereas the *Models* (a.k.a., default schemas) define the structure of the biological data responses. Here, we focus on the *Models*, which comprise seven entities (a.k.a., entry types) defined using JSON Schema 2020-12 format (see Fig. 2). Out of the seven, we will be concentrating on the *individuals* entity, because it is the one containing pheno-clinical data for patients and healthy controls. Other entities like *datasets* or *biosamples* contain useful metadata, yet typically have fewer nested properties, making them comparatively easier to map.

There are two approaches for generating Beacon v2 responses to queries to the *individuals* entry type: real-time mapping within a Beacon v2 API (e.g., data are read from relational database and immediately serialized to JSON format) or using pre-converted data. Convert-Pheno supports both approaches, with the “pre-converted data” option providing more control over shared contents [25]. The pre-converted data, known as Beacon Friendly Format (BFF), consists of seven JSON files, including the *individuals* entity. In this work, we primarily focus on converting the *individuals* entity, which includes the terms *diseases*, *ethnicity*, *exposures*, *geographicOrigin*, *id*, *info*, *interventionsOrProcedures*, *karyotypicSex*, *measures*, *pedigrees*, *phenotypicFeatures*, *sex* and *treatments*. More details about these terms can be found in the Beacon v2 documentation (<https://docs.genomebeacons.org/schemas-md/individuals-defaultSchema>).

2.2.1.2. Phenopackets v2. Phenopackets v2 is a standardized format that enables the structured and computable exchange of pheno-clinic (and genomic) data, facilitating interoperability and collaboration in biomedical research and clinical practice [21,26]. Phenopackets v2 employs the top-level element *Phenopacket* to describe phenotypic data for a person, which is semantically related to the *individuals* entity in Beacon v2. Phenopackets organizes data hierarchically and employs

biomedical ontologies to define terms. It incorporates standardized metadata, terminologies, and codes for phenotypic features and their relationships, along with information on provenance and data quality metrics. Phenopackets schema supports the FAIR (findable, accessible, interoperable, and reusable) principles [27] and is defined using Google's protocol buffers framework (protobuf). Nevertheless, PXF (Phenotype Exchange Format) files can be serialized in either JSON or YAML formats, both accepted by Convert-Pheno as an input and output. The Phenopackets v2 publications [21,26] and documentation (<https://phenopacket-schema.readthedocs.io>) provide examples of how data are structured in PXF files.

2.2.1.3. OMOP-CDM. OMOP-CDM (Observational Medical Outcomes Partnership Common Data Model) is an open community data standard designed to harmonize the structure and content of observational data. It uses the Observational Health Data Sciences and Informatics (OHDSI) standardized vocabulary to define biomedical terms, bringing uniformity across its various clinical domains. OMOP-CDM is based on tables and can be implemented using various relational database management systems, making it database-agnostic. OMOP-CDM offers standardized SQL scripts that define data structures and relationships, along with tools and utilities for managing the database (available at <https://www.ohdsi.org/software-tools>). PostgreSQL is a commonly used option for implementing this model. One can access OMOP-CDM databases programmatically using SQL queries or by utilizing the open-source WebAPI service (<http://webapidoc.ohdsi.org>) developed by the OHDSI community. Convert-Pheno can convert data from file-based sources such as PostgreSQL exports or any other relational database in CSV format, as well as perform real-time conversions from WebAPI data or SQL queries (upon conversion to the accepted JSON format). Convert-Pheno currently supports OMOP-CDM versions 5.3 and 5.4, and we are prepared to support version 6 once we are able to test the code with version 6 projects. The tables used during the mapping are CONCEPT, CONDITION_OCCURRENCE, DRUG_EXPOSURE, MEASUREMENT, PERSON, PROCEDURE_OCCURRENCE, OBSERVATION and VISIT_OCCURRENCE.

2.2.1.4. REDCap. REDCap (Research Electronic Data Capture) is a secure web-based software platform designed to support data management and research studies [22,28]. REDCap provides a customizable interface for creating and managing electronic case report forms (eCRFs) and surveys, combined with tools for importing/exporting data, managing study participants, quality control checks, etc. REDCap is widely used in academic, clinical, and translational research studies across different domains, and is known for its ease of use, flexibility, and robust security features. REDCap projects are inherently "free format", meaning the project creator has the flexibility to determine the identifiers for variables, data dictionaries, and other elements. Due to this flexibility, it can be challenging to create a solution that can handle the vast array of possibilities in REDCap projects. Despite this, we were able to successfully convert data from REDCap project exports to both BFF and PXF by using a mapping file. These pilot conversions were achieved as part of the 3TR project (<https://3tr-imi.eu>) and will be further described in the results section.

2.2.1.5. CDISC-ODM. CDISC-ODM (Clinical Data Interchange Standards Consortium - Operational Data Model) is a standard format for representing clinical trial data in a structured and machine-readable

manner. The ODM defines a set of XML-based schemas for organizing and exchanging data elements, facilitating data integration from different sources and platforms, and supporting interoperability across the clinical research lifecycle (<https://www.cdisc.org/standards/data-exchange/odm>). CDISC-ODM is widely adopted in the pharmaceutical and biotech industries, as well as by academic institutions and government agencies [20]. Convert-Pheno supports CDISC-ODM version 1.3.2, and the conversion process is analogous to that of REDCap projects.

2.2.2. Data mapping

2.2.2.1. Conversion to Beacon v2 Models. Internally, the software converts all data models to the *individuals* entity from Beacon v2 Models [19,25], which serves as the central model (see Fig. 3). The choice of Beacon v2 as the target model was influenced by its utilization of JSON Schema, accommodation of additional properties, compatibility with Beacon v2 API, expandability to other entities, and overlap with Phenopackets v2. This flexibility offered by Beacon v2 schemas enables us to handle a wider range of phenotypic data, enhancing the utility and applicability of our tool.

2.2.2.2. Schema mapping. When starting a new conversion between two data models, the first step is to map variables between the two data schemas. In the early stages of development, we explored the possibility of employing configuration files to guide the mapping process as an alternative to hardcoded solutions. However, JSON data structures' complexity, mainly due to nesting, made this approach impractical for most scenarios, except for REDCap and CDISC-ODM data, which are mapped to Beacon v2 Models via configuration files. As of the current writing, the mapping tables have been manually created and can be accessed in the [Supplementary Data](#) (Mapping Tables file), as well as in the online documentation. The tables serve as a guide for implementers when developing the source code for Convert-Pheno. Each format conversion is implemented as a dedicated Perl submodule.

2.2.2.3. Lossless or lossy conversion. When converting data from one data standard to another, it is important to consider the possibility of losing information due to differences in schema and field mapping. To mitigate this, we aimed for a lossless conversion by incorporating non-mappable variables as *additionalProperties* within the Beacon v2 Models schema. This allows users to access the original variables and their values through database queries, especially when using non-relational databases like MongoDB. During the conversion process, handling variables that cannot be directly mapped can result in one of two scenarios:

a) Unmappable variables

Often, the input data model has variables that do not directly map to the target but are still useful to retain in the output format. If the target format allows for extra properties in a given term (as BFF does), these original variables are stored under the 'info' property (or '.' + 'property name'). This commonly happens in conversions from OMOP-CDM to BFF. See an example below:

```

"interventionsOrProcedures" : [
  {
    "_info" : {
      "PROCEDURE_OCCURRENCE" : {
        "OMOP_columns" : {
          "modifier_concept_id" : 0,
          "modifier_source_value" : null,
          "person_id" : 2,
          "procedure_concept_id" : 4163872,
          "procedure_date" : "1955-10-22",
          "procedure_datetime" : "1955-10-22 00:00:00",
          "procedure_occurrence_id" : 6,
          "procedure_source_concept_id" : 4163872,
          "procedure_source_value" : 399208008,
          "procedure_type_concept_id" : 38000275,
          "provider_id" : "\\N",
          "quantity" : "\\N",
          "visit_detail_id" : 0,
          "visit_occurrence_id" : 103
        }
      }
    },
    "ageAtProcedure" : {
      "age" : {
        "iso8601duration" : "35Y"
      }
    },
    "dateOfProcedure" : "1955-10-22",
    "procedureCode" : {
      "id" : "SNOMED:399208008",
      "label" : "Plain chest X-ray"
    }
  }
]

```

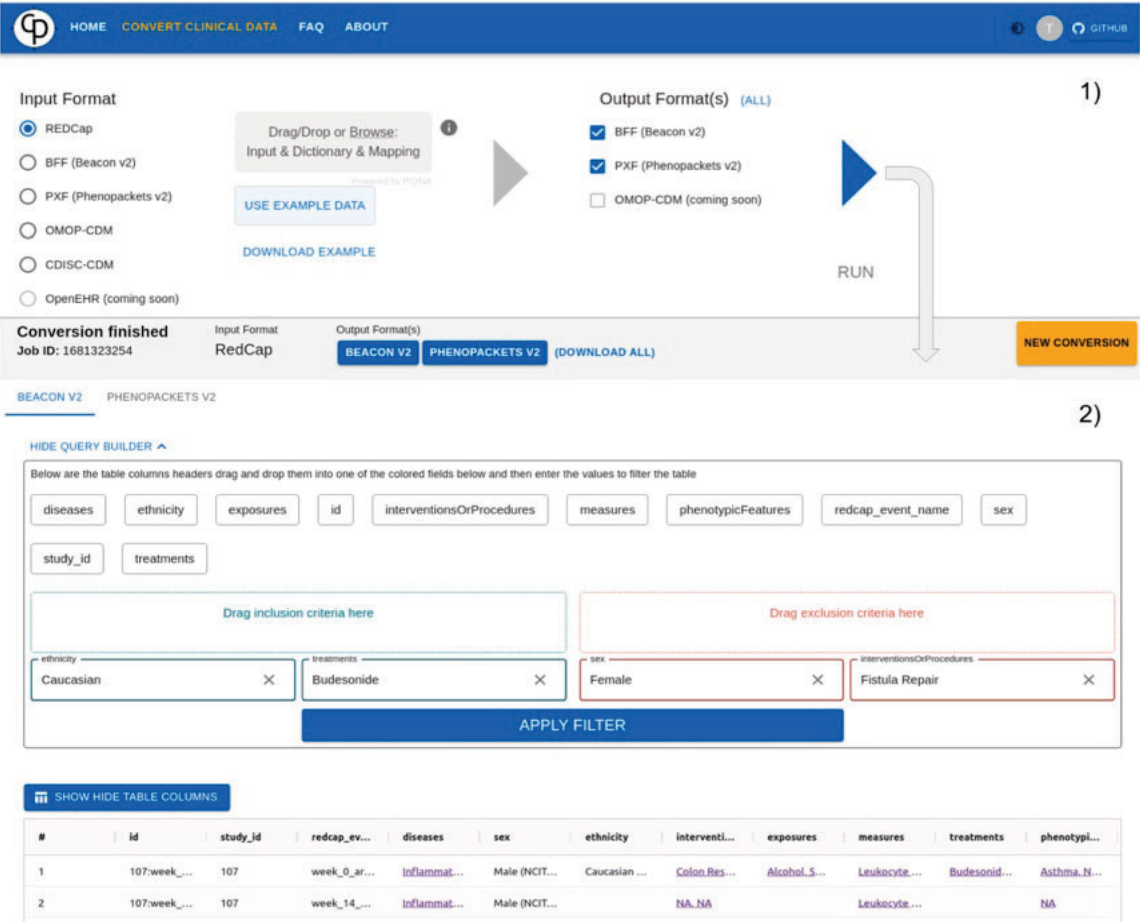


Fig. 1. Screenshots of the Convert-Pheno web application interface for both the input (1) and output (2) stages.

search by setting the command line option `-search` to `mixed` (default is `exact`). With this setting, Convert-Pheno will perform an exact search first and if unsuccessful it will perform a full-text search followed by a string-based comparison according to Dice-Sorensen (default) or cosine (optional) methods. The threshold for this similarity is very conservative (0.8) but is changeable by the user. For instance, when using the NCIt ontology and searching for the phrase “Exercise pain management” with exact search, the label “Exercise Pain Management” (case insensitive) will be the only one matched. Conversely, searching for the phrase “Brain Hemorrhage” will not yield any results with exact search, but using a mixed search (with the default parameters) will yield “Intra-ventricular Brain Hemorrhage”. More detailed information on the effect of threshold and other requirements are extended in the online documentation.

Presently, we do not support API calls for REDCap and CDISC-ODM formats due to the complexity of their input data (see Table 1).

2.2.3. Outputs

Convert-Pheno can output data in either the Beacon v2 Models or the Phenopackets v2 formats, which we discussed earlier in the ‘Input’ section. The conversion to Phenopackets v2 format involves applying the ‘bff2pxf’ step to data formatted according to Beacon v2 Models

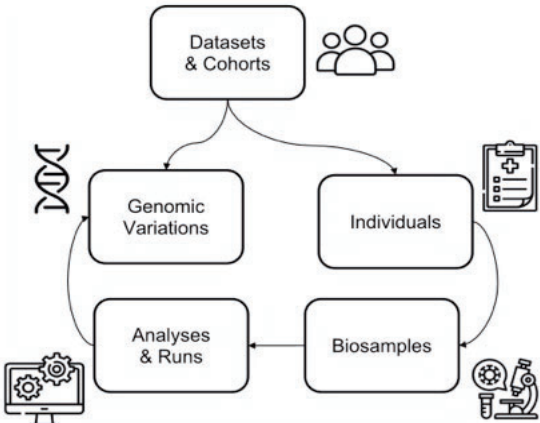


Fig. 2. Diagram displaying the seven entities of the Beacon v2 Models.

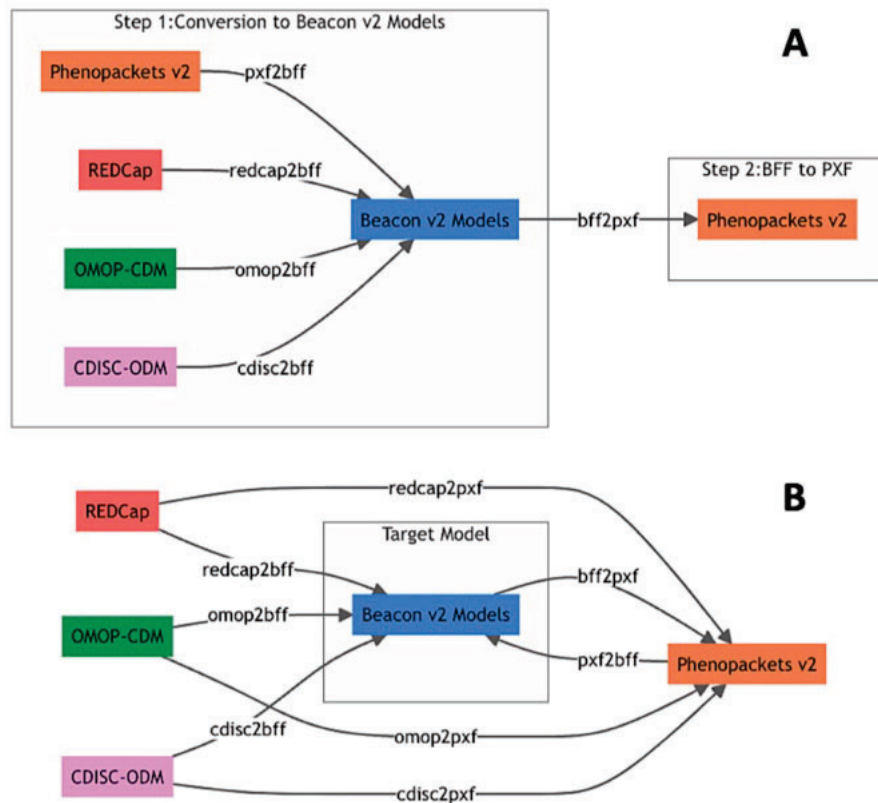


Fig. 3. Diagrams showing Convert-Pheno's: A) Steps in data conversion, B) All possible interconversions between phenotypic data models.

Table 1

Accepted functionalities depending on the type of input data.

Input	CLI	UI	Module	API
Beacon v2 Models	YES	YES	YES	YES
CDISC-ODM	YES	YES	YES	NO
Phenopackets v2	YES	YES	YES	YES
OMOP-CDM	YES	YES	YES	YES
REDCap	YES	YES	YES	NO

(refer to Fig. 3). We chose these formats deliberately because they can hold both genomic variations and phenotypic data. Both standards play a central role in mapping exercises due to their structured and compact data schemas, which are designed to encapsulate a diverse range of data relevant to bio-medical-genomics research. Their role is not to replace or entirely encompass FHIR and other EHR data models; rather, they facilitate streamlined data representation in genomic and biomedical research environments, thereby fostering effective data sharing and integration initiatives.

To ensure the integrity and validity of the output data, we employed validation tools during the development stage. Specifically, we utilized the BFF-validator [25] for validating Beacon Friendly Format (BFF) data and Phenopacket-tools [30] for validating Phenopacket Exchange Format (PXF) data. These validators were instrumental in ensuring that the converted data adheres to the respective schemas and standards. By incorporating non-mapped variables and implementing rigorous

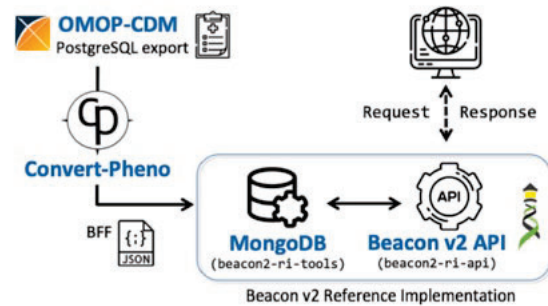


Fig. 4. Integration of Convert-Pheno with the Beacon v2 Reference Implementation in the context of an OMOP-CDM to Beacon v2 conversion.

validation processes, we aimed to mitigate information loss and ensure the accuracy and fidelity of the converted data.

An important aspect of Convert-Pheno is its inclusion of test data for all formats, generously supplied by our partners, which is included in the installation and on which we have thoroughly tested our code. In the future, we plan to support additional formats, as we have already received requests from the community. These requests include the addition of other input formats, such as OpenEHR, as well as the implementation of OMOP-CDM as an output format. To this end, we have created a tentative roadmap that can be accessed at

<https://cnag-biomedical-informatics.github.io/convert-pheno/future-plans>.

3. Results

3.1. Use case I: REDCap to Beacon v2 Models

The first use case involves converting REDCap projects into Beacon v2 Models, which was developed in the context of the 3TR project. The 3TR project (<https://3tr-imi.eu>) is a seven-year initiative funded by the European Commission through the Innovative Medicines Initiative. One of our contributions to the 3TR project is to consolidate molecular and clinical trials data related to seven autoimmune diseases: asthma, chronic obstructive pulmonary disease, Crohn's disease, ulcerative colitis, rheumatoid arthritis, multiple sclerosis, and systemic lupus erythematosus.

At the outset of the project, it was decided that most centers would store their clinical data in REDCap projects. To post-process the data, the REDCap projects would be exported to CSV files along with their corresponding dictionaries. As mentioned previously, Convert-Pheno offers various modes of operation, but for this purpose, we utilized the command-line version as it can handle text files. To perform the conversion, we require the following three files:

- REDCap export (CSV)
- REDCap data dictionary (CSV)
- Configuration (mapping) file (YAML)

The starting point for each center is to log into their REDCap system and export the data to CSV format. After exporting the data, the user must also download the REDCap dictionary in CSV format. This can be done within REDCap by navigating to 'Project Setup/Data Dictionary/Download the current'. When using REDCap projects as an input format, Convert-Pheno requires a mapping file to link REDCap project variables to a format that it can process. The mapping file, in YAML format, connects variables to the terms required by the *individuals* entity in the Beacon v2 models. The mapping file is defined using JSON Schema 2020-12, which allows for the validation of user files to identify any potential errors. In addition to variable mapping, we perform "data augmentation" by the inclusion of ontologies, as both BFF and PFX require them to define phenotypic values (see Methods and Supporting Table 1).

3.2. Use case II: OMOP-CDM conversion to Beacon v2

This section summarizes the steps involved in converting an OMOP-CDM export to Beacon v2 Models, as performed in the context of Spain's personalized medicine program IMPaCT (Precision Medicine Infrastructure associated with Science and Technology). IMPaCT's mission is to establish the foundations to facilitate the effective deployment of Precision Medicine in the National Health System, ensuring scientific/technical quality, equity and efficiency in the use of available scientific resources to respond to the needs of citizens (<https://impact-data.bsc.es/en/about/impact>). IMPaCT-Data is part of IMPaCT, which focuses on integrating and analyzing clinical, molecular, and genetic data for coordinated secondary use with other IMPaCT programs (<https://impact-data.bsc.es/en/>). One of IMPaCT-Data's objectives is to create a methodology to convert OMOP-CDM data into Beacon v2 format.

As described in the input section under 2.2.1.3, OMOP-CDM data are represented in a tabular format and implemented as a SQL instance (e.g., PostgreSQL). Two approaches can be used for the conversion process: i) real-time (on-the-fly) conversion through SQL queries or API calls, or ii) database export. Several factors can influence the selection of an approach for data conversion, such as resource availability, security concerns, and the intended purpose of the converted data (e.g., data

discovery vs. data analytics). In this regard, a disadvantage of working at the text file level is that newly added data will not be accessible until a new export is created. Here, we will describe how to convert data from an export of PostgreSQL instance in either .SQL or .CSV format.

OMOP-CDM databases typically contain millions of rows of data for thousands of patients, and performing periodic exports provides administrators with greater control over what data is shared and the ability to filter out unnecessary fields. To handle large files, the software has two modes of operation. The default mode is called `-no stream`, in which some operations are performed in RAM memory. However, this can become unfeasible for very large files, so we provide the `-stream` mode, in which the files are processed line-by-line, allowing for efficient processing without becoming limited by RAM memory. Converting data to OMOP-CDM is usually less complex than converting to REDCap since it doesn't require the addition of ontologies (see Supporting Table 1) as the required vocabulary is already present in the OMOP-CDM standardized vocabulary. Ontologies are defined as *concept_id* within the table *CONCEPT* in OMOP-CDM. Convert-Pheno is capable of transforming both full and partial exports of records from OMOP-CDM databases. A full export includes all the necessary ontologies in the *CONCEPT* table, eliminating the need to search for additional databases. However, in a partial export, if certain vocabularies are missing, Convert-Pheno searches the included OHDSI local database to find the required ontologies (see Supporting Table 1). The flag `-ohdsi-db` is required to enable this search.

For development purposes, we used the synthetic EUNOMIA dataset (<https://ohdsi.github.io/Eunomia/>). This demonstration dataset was loaded into a PostgreSQL instance and exported to a .SQL file by our partners at IMIM (Institut Hospital del Mar d'Investigacions Mèdiques). Convert-Pheno was then deployed locally at IMIM for testing, and refinements were made to accommodate features that were absent in the EUNOMIA dataset. We aim to provide a more extensive explanation with more details in a future communication, specific to this conversion.

3.3. Use case III: Beacon v2 Models to Phenopackets v2 via REST API

As a final example, we will demonstrate a "real-time" conversion from Beacon v2 Models to Phenopackets using the REST API. The API accepts a JSON payload that is sent to the endpoint `/api`. Currently, the API requires only two properties: "method" and "data". In this case, "method" is set to "bff2pfx" and "data" contains an object of the *individuals* entity (Beacon v2 Models) in JSON format. On-the-fly data transformations are useful in cases where a web application has data in one format (e.g., Beacon v2 Models) but needs to display it in another format (e.g., Phenopackets v2). The API response can be used to display the converted data on a different web application, making it a valuable feature of Convert-Pheno. For more detailed information, please refer to our online documentation (<https://cnag-biomedical-informatics.github.io/convert-pheno/use-as-an-api>).

4. Discussion

The standardization of phenotypic data is essential for advancing biomedical research and improving patient outcomes. It promotes data sharing and integration across research groups and platforms, accelerating scientific progress and maximizing the impact of research findings. Therefore, the development and adoption of standardized phenotypic data collection and sharing protocols are critical for advancing precision medicine and improving public health. In this sense, many research centers are adopting health data standards such as OMOP-CDM or Phenopackets v2 to represent their phenotypic and clinical data, which is a positive step forward. However, comparing or sharing data among scientists who use different standards still presents a challenge, especially when data from different studies or centers need to be compared.

From an academic perspective, there have been previous attempts to perform unidirectional conversion (from one standard to another) of

health data standards. Many of which are reported as formal scientific publications [31–45], while others are available on websites (e.g., OMOP on FHIR: <https://omoponfhir.org>) or GitHub repositories (e.g., <https://github.com/MeasureAuthoringTool/health-data-standards>, <https://github.com/microsoft/FHIR-Converter>). The industry has also made some efforts in data conversion, but their focus is primarily on transforming clinical data rather than interconverting standards. Examples of companies in this field include Biomeris (<https://www.biomeris.it>), Bitac (<https://www.bitac.com/en/>), BMD (<https://www.bmd-software.com>), Databricks (<https://www.databricks.com>) and Veratech (<https://www.veratech.es>). Additionally, Data Connect is a solution proposed for the discovery and search of biomedical data developed by the Discovery Work Stream of the GA4GH (<https://github.com/ga4gh-discovery/data-connect>). Data Connect provides a comprehensive solution for sharing and discovering genomic and clinical data across different repositories and platforms, which can be used in conjunction with Beacon v2 or Phenopackets v2.

None of the cited examples involve Beacon v2 or Phenopackets v2 conversion, as both standards are relatively new, and to date, many efforts towards interoperability have been led by members of their respective ecosystems, including ourselves. For instance, we participated in the BioHackathon-Europe 2021, where we attempted to convert OMOP-CDM to Beacon v2 (<https://github.com/elixir-luxembourg/BH2021-beacon-2.x-omop>), and some of our partners are currently working on an OMOP-CDM to PXF exporter (<https://github.com/phenopackets/omop-exporter>) [46]. Finally, the Beacon v2 Reference Implementation (B2RI) [25] includes utilities to facilitate the conversion of raw phenotypic data to BFF format.

Convert-Pheno was developed to standardize and harmonize phenotypic data stored in various sources within the 3TR project. Our initial focus was on the REDCap format, and we later expanded it to include additional formats based on the requirements of our scientific projects and collaborations. To the best of our knowledge, Convert-Pheno is the first tool to attempt multidirectional conversions of health data models. We recognize that numerous health data standards exist and interconverting them can be a formidable task. Our approach is to focus on one task at a time and work towards addressing the needs of our projects, as well as taking requests from the community. A limitation of our study lies in the absence of a formal evaluation of the mappings themselves. Our evaluation has predominantly relied on our familiarity with the Beacon v2 and Phenopackets v2 standards, coupled with input and feedback from our collaborators. While we collectively possess expertise in specific domains, we welcome suggestions and contributions from the wider community to improve the quality of our mappings.

Implemented as a toolkit, Convert-Pheno enables users to perform both “on-the-fly” conversions (where possible) and text file-based conversions. The file-level conversion offers the advantage of performing the data conversion once (per export), after which the file can be used for multiple purposes. For example, the converted BFF file becomes compatible with the Beacon v2 federated data discovery system. To facilitate this, we recommend using the B2RI [25], which offers tools for loading the data into a MongoDB database that can be directly queried using the Beacon v2 API (as illustrated in Fig. 4). Additionally, the converted BFF or PXF files can be deposited in repositories such as the EGA [47]. Furthermore, the converted files can be directly shared with collaborators, promoting data transparency and reproducibility, as it allows researchers to compare and validate results across different studies and datasets (semantic interoperability). Lastly, the converted files require minimal handling to be compatible with artificial intelligence (A.I.) algorithms, enabling powerful data analytics [48,49].

5. Conclusion

We have developed Convert-Pheno, an open-source toolkit that allows for an efficient inter-conversion among commonly used standards

for phenotypic data. The software can function as a command-line utility for file exports or as a REST API for real-time conversions. The command-line interface includes a web app interface that allows for user-friendly interactive operations and data search, displayed as HTML tables. Currently, the software accepts as input Phenopackets v2, Beacon v2 Models, CDISC-ODM, REDCap and OMOP-CDM, and as an output Beacon v2 Models and Phenopackets v2 (both being GA4GH standards that also can store genomic variations data). The software has been tested with synthetic and real datasets, for which we present three use cases, the first being a REDCap project conversion in the context of the 3TR project, the second an OMOP-CDM to Beacon v2 conversion in the context of Spain’s personalized program IMPaCT, and the third an API-based conversion from Beacon v2 to Phenopackets v2. The software includes a comprehensive web-based documentation (including a Google Colab interactive notebook) and synthetic (or anonymized) data that can be used as a reference. Our goal is to continue expanding the range of supported formats in the future. In addition to promoting data transparency and reproducibility, converting data to standards such as Phenopackets v2 or Beacon v2 models enables semantic interoperability. This, in turn, can enhance data discovery and potentially lead to better predictions via machine learning tools, as it allows researchers to compare and validate results across different studies and datasets. We hope that this will contribute to the advancement of genomic research and ultimately improve patient care.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We would like to express our gratitude to Ioannis Parodis (and his team) and Florian Tran from the 3TR project for providing us with access to anonymized data in REDCap and CDISC-ODM formats. Additionally, we thank Juan Manuel Ramirez and Miguel Ángel Mayer for their assistance in creating the EUNOMIA synthetic dataset for OMOP-CDM and for testing our code locally at IMIM. Our thanks extend to the BSC and EGA teams for their invaluable feedback, and to Davide Piscia and his team (especially Sofia Chaves) for their assistance with the web application. We would also like to thank the anonymous referees who reviewed this paper, their insights and comments significantly contributed to enhancing the quality of this manuscript.

Funding

This work was carried out in the course of the 3TR project with funding from the Innovative Medicines Initiative 2 Joint Undertaking (JU) under grant agreement No 831434. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and EFPIA (Disclaimer: Content of this publication reflects only the author’s view and the JU is not responsible for any use that may be made of the information it contains). Institutional support was from the Spanish Instituto de Salud Carlos III, Fondo de Investigaciones Sanitarias and cofunded with ERDF funds (PI19/01772). We acknowledge the institutional support of the Spanish Ministry of Science and Innovation through the Instituto de Salud Carlos III and the 2014–2020 Smart Growth Operating Program, to the EMBL partnership and institutional co-financing with the European Regional Development Fund (MINECO/FEDER, BIO2015-71792-P). We also acknowledge the support from the Generalitat de Catalunya through the Departament de Salut and the Departament d’Empresa i Coneixement.

Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.jbi.2023.104558>.

References

- [1] B.H. de Mello, S.J. Rigo, C.A. da Costa, R. da Rosa Righi, B. Donida, M.R. Bez, L. C. Schunke, Semantic interoperability in health records standards: a systematic literature review, *Health Technol.* 12 (2022) 255–272, <https://doi.org/10.1007/s12553-022-00639-w>.
- [2] K.S. Lytle, B.L. Westra, L. Whittenburg, M. Adams, M. Akre, S. Ali, M. Furukawa, S. Hartleben, M. Hook, S.G. Johnson, T. (Tess) Settergren, M. Thibodeaux, Information Models Offer Value to Standardize Electronic Health Record Flowsheet Data: A Fall Prevention Exemplar, *J. Nurs. Scholarsh.* 53 (2021) 306–314, <https://doi.org/10.1111/jnu.12646>.
- [3] P. Berens, M.S. Ayhan, Proprietary data formats block health research, *Nature*. 565 (2019) 429, <https://doi.org/10.1038/d41586-019-00231-9>.
- [4] L. Marco-Ruiz, J.G. Bellika, Semantic interoperability in clinical decision support systems: a systematic review, *Stud. Health Technol. Inform.* 216 (2015) 958.
- [5] D. Wang, M. Peleg, S.W. Tu, A.A. Boxwala, R.A. Greenes, V.L. Patel, E.H. Shortliffe, Representation primitives, process models and patient data in computer-interpretable clinical practice guidelines: a literature review of guideline representation models, *Int. J. Med. Inf.* 68 (2002) 59–70, [https://doi.org/10.1016/s1386-5056\(02\)00065-5](https://doi.org/10.1016/s1386-5056(02)00065-5).
- [6] T.N. Arvanitis, Semantic interoperability in healthcare, *Stud. Health Technol. Inform.* 202 (2014) 5–8.
- [7] H.L. Rehm, A.J.H. Page, L. Smith, J.B. Adams, G. Alterovitz, L.J. Babb, M. P. Barkley, M. Baudis, M.J.S. Beauvais, T. Beck, J.S. Beckmann, S. Beltran, D. Bernick, A. Bernier, J.K. Bonfield, T.F. Boughtwood, G. Bourque, S.R. Bowers, A. J. Brookes, M. Brudno, M.H. Brush, D. Bujold, T. Burdett, O.J. Buske, M.N. Cabili, D.L. Cameron, R.J. Carroll, E. Casas-Silva, D. Chakravarty, B.P. Chaudhari, S. H. Chen, J.M. Cherry, J. Chung, M. Cline, H.L. Clissold, R.M. Cook-Deegan, M. Courtot, F. Cunningham, M. Cupak, R.M. Davies, D. Denisko, M.J. Doerr, L. I. Dolman, E.S. Dove, L.J. Dursi, S.O.M. Dyke, J.A. Eddy, K. Eilbeck, K.P. Ellrott, S. Fairley, K.A. Fakhro, H.V. Firth, M.S. Fitzsimons, M. Fiume, P. Flicek, I.M. Fore, M.A. Freeberg, R.R. Freimuth, L.A. Fromont, J. Fuerth, C.L. Gaff, W. Gan, E. M. Ghanaim, D. Glazer, R.C. Green, M. Griffith, O.L. Griffith, R.L. Grossman, T. Groza, J.M.G. Auvi, R. Guigó, D. Gupta, M.A. Haendel, A. Hamosh, D.P. Hansen, R.K. Hart, D.M. Hartley, D. Haussler, R.M. Hendricks-Sturup, C.W.L. Ho, A. E. Hobb, M.M. Hoffman, O.M. Hofmann, P. Holub, J.S. Hsu, J.-P. Hubaux, S. E. Hunt, A. Husami, J.O. Jacobsen, S.S. Jamuar, E.L. Jones, J. Jeanson, A. Jené, A. L. Johns, Y. Joly, S.J.M. Jones, A. Kanitz, K. Kato, T.M. Keane, K. Kekesi-Lafrance, J. Kelleher, G. Kerry, S.-S. Khor, B.M. Knoppers, M.A. Konopko, K. Kosaki, M. Kuba, J. Lawson, R. Leinonen, S. Li, M.F. Lin, M. Linden, X. Liu, I. Uđara Liyanage, J. Lopez, A.M. Lucassen, M. Lukowski, A.L. Mann, J. Marshall, M. Mattioni, A. Metke-Jimenez, A. Middleton, R.J. Milne, F. Molnár-Gábor, N. Mulder, M. C. Munoz-Torres, R. Nag, H. Nakagawa, J. Nasir, A. Navarro, T.H. Nelson, A. Niewielska, A. Nisselle, J. Niu, T.H. Nyronen, B.D. O'Connor, S. Oesterle, S. Ogishima, V.O. Wang, L.A.D. Paglione, E. Palumbo, H.E. Parkinson, A. Philippakis, A.D. Pizarro, A. Pric, J. Rambla, A. Rendon, R.A. Rider, P. N. Robinson, K.W. Rodmer, L.L. Rodriguez, A.F. Rubin, M. Rueda, G.A. Rushton, R.S. Ryan, G.I. Saunders, H. Schuilenburg, T. Schwede, S. Scollen, A. Senf, N. C. Sheffield, N. Skantharajah, A.V. Smith, H.J. Sofia, D. Spalding, A.B. Spurdle, Z. Stark, International policies and standards for data sharing across genomic research and healthcare, *Cell, Genomics*. 1 (2021), 100029, <https://doi.org/10.1016/j.xgen.2021.100029>.
- [8] R.H. Dolin, L. Alschuler, C. Beebe, P.V. Biron, S.L. Boyer, D. Essin, E. Kimber, T. Lincoln, J.E. Mattison, The HL7 clinical document architecture, *J. Am. Med. Inform. Assoc. JAMIA*. 8 (2001) 552–569, <https://doi.org/10.1136/jamia.2001.0080552>.
- [9] A. Rossi Mori, F. Consorti, E. Galeazzi, Standards to support development of terminological systems for healthcare telematics, *Methods Inf. Med.* 37 (1998) 551–563.
- [10] Aspden P, Corrigan JM, Wolcott J, et al, in: Health Care Data Stand., Patient Safety: Achieving a New Standard for Care. Washington (DC): National Academies Press (US); Institute of Medicine (US) Committee on Data Standards for Patient Safety, 2004. <https://www.ncbi.nlm.nih.gov/books/NBK216088>.
- [11] E.A. Voss, R. Makadia, A. Matcho, Q. Ma, C. Knoll, M. Schuemie, F.J. DeFalco, A. Londhe, V. Zhu, P.B. Ryan, Feasibility and utility of applications of the common data model to multiple, disparate observational health databases, *J. Am. Med. Inform. Assoc. JAMIA*. 22 (2015) 553–564, <https://doi.org/10.1093/jamia/ocu023>.
- [12] S.N. Duda, N. Kennedy, D. Conway, A.C. Cheng, V. Nguyen, T. Zayas-Cabán, P. A. Harris, HL7 FHIR-based tools and initiatives to support clinical research: a scoping review, *J. Am. Med. Inform. Assoc. JAMIA*. 29 (2022) 1642–1653, <https://doi.org/10.1093/jamia/ocac105>.
- [13] DICOM reference guide, *Health Devices*. 30 (2001) 5–30.
- [14] T. Souza, R. Kush, J.P. Evans, Global clinical data interchange standards are here!, *Drug Discov. Today*. 12 (2007) 174–181, <https://doi.org/10.1016/j.drudis.2006.12.012>.
- [15] J.B.L. Bard, S.Y. Rhee, Ontologies in biology: design, applications and future challenges, *Nat. Rev. Genet.* 5 (2004) 213–222, <https://doi.org/10.1038/nrg1295>.
- [16] Gene Ontology Consortium, Gene Ontology Consortium: going forward, *Nucleic Acids Res.* 43 (2015) D1049–1056, <https://doi.org/10.1093/nar/gku1179>.
- [17] S. Kohler, N.A. Vasilevsky, M. Engelstad, E. Foster, J. McMurry, S. Ayne, G. Baynam, S.M. Bello, C.F. Boerkoel, K.M. Boycott, M. Brudno, O.J. Buske, P.F. Chinnery, V. Cipriani, L.E. Connell, H.J. Dawkins, L.E. DeMare, A.D. Devereau, B.B. de Vries, H.V. Firth, K. Freson, D. Greene, A. Hamosh, I. Helbig, C. Hum, J.A. Jahn, R. James, R. Krause, F. Laulederkind SJ, H. Lochmuller, G.J. Lyon, S. Ogishima, A. Olry, W.H. Ouwehand, N. Pontikos, A. Rath, F. Schaefer, R.H. Scott, M. Segal, P.I. Sergouniotis, R. Sever, C.L. Smith, V. Straub, R. Thompson, C. Turner, E. Turro, M. W. Veltman, T. Vulliamy, J. Yu, J. von Ziegenweid, A. Zankl, S. Zuchner, T. Zemojtel, J.O. Jacobsen, T. Groza, D. Smedley, C.J. Mungall, M. Haendel, P.N. Robinson, The Human Phenotype Ontology in 2017, *Nucleic Acids Res.* 45 (2017) D865–D876, <https://doi.org/10.1093/nar/gkw1039>.
- [18] A.W. Forrey, C.J. McDonald, G. DeMoor, S.M. Huff, D. Leavelle, D. Leland, T. Fiers, L. Charles, B. Griffin, F. Stalling, A. Tullis, K. Hutchins, J. Baenziger, Logical observation identifier names and codes (LOINC) database: a public use set of codes and names for electronic reporting of clinical laboratory test results, *Clin. Chem.* 42 (1996) 81–90.
- [19] J. Rambla, M. Baudis, R. Ariosa, T. Beck, L.A. Fromont, A. Navarro, R. Paloots, M. Rueda, G. Saunders, B. Singh, J.D. Spalding, J. Törnroos, C. Vassallo, C.D. Veal, A.J. Brookes, Beacon v2 and Beacon Networks: a “lingua franca” for federated data discovery in biomedical genomics, and beyond, *Hum. Mutat.* 43 (2022) 9, <https://doi.org/10.1002/humu.24369>.
- [20] S. Hume, J. Aerts, S. Sarnikar, V. Huser, Current applications and future directions for the CDISC Operational Data Model standard: A methodological review, *J. Biomed. Inform.* 60 (2016) 352–362, <https://doi.org/10.1016/j.jbi.2016.02.016>.
- [21] J.O.B. Jacobsen, M. Baudis, G.S. Baynam, J.S. Beckmann, S. Beltran, O.J. Buske, T. J. Callahan, C.G. Chute, M. Courtot, D. Danis, O. Elemento, A. Essenwanger, R.R. Freimuth, M.A. Gargano, T. Groza, A. Hamosh, N.L. Harris, R. Kaliyaperumal, K.C. K. Lloyd, A. Khalifa, P.M. Krawitz, S. Köhler, B.J. Laraway, H. Lehtväisäliho, L. Matalonga, J.A. McMurry, A. Metke-Jimenez, C.J. Mungall, M.C. Munoz-Torres, S. Ogishima, A. Papakonstantinou, D. Piscià, N. Pontikos, N. Queralt-Rosinach, M. Roos, J. Sass, P.N. Schofield, D. Seelow, A. Siapos, D. Smedley, L.D. Smith, R. Steinhaus, J.C. Sundaramurthi, E.M. Swietlik, S. Thun, N.A. Vasilevsky, A.H. Wagner, J.L. Warner, C. Weiland, GAGH Phenopacket Modeling Consortium, M.A. Haendel, P.N. Robinson, The GA4GH Phenopacket schema defines a computable representation of clinical data, *Nat. Biotechnol.* 40 (2022) 817–820, <https://doi.org/10.1038/s41587-022-01357-4>.
- [22] P.A. Harris, R. Taylor, R. Thielke, J. Payne, N. Gonzalez, J.G. Conde, Research electronic data capture (REDCap)—a metadata-driven methodology and workflow process for providing translational research informatics support, *J. Biomed. Inform.* 42 (2009) 377–381, <https://doi.org/10.1016/j.jbi.2008.08.010>.
- [23] ImageMagick Studio LLC., (2023). <https://imagemagick.org>.
- [24] N.M. O’Boyle, M. Banck, C.A. James, C. Morley, T. Vandermeersch, G. R. Hutchison, Open Babel: An open chemical toolbox, *J. Cheminformatics*. 3 (2011) 33, <https://doi.org/10.1186/1758-2946-3-33>.
- [25] M. Rueda, R. Ariosa, M. Moldes, J. Rambla, Beacon v2 Reference Implementation: a toolkit to enable federated sharing of genomic and phenotypic data, *Bioinform. Oxf. Engl.* 38 (2022) 4656–4657, <https://doi.org/10.1093/bioinformatics/btac568>.
- [26] M.S. Ladewig, J.O.B. Jacobsen, A.H. Wagner, D. Danis, B. El Kassaby, M. Gargano, T. Groza, M. Baudis, R. Steinhaus, D. Seelow, N.E. Bechrakis, C.J. Mungall, P. N. Schofield, O. Elemento, L. Smith, J.A. McMurry, M. Munoz-Torres, M. A. Haendel, P.N. Robinson, GA4GH Phenopackets: A Practical Introduction, *Adv. Genet.* 4 (2023) 2200016, <https://doi.org/10.1002/ggn2.20200016>.
- [27] M.D. Wilkinson, M. Dumontier, I.J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L.B. da Silva Santos, P.E. Bourne, J. Bouwman, A.J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C.T. Evelo, R. Finkers, A. Gonzalez-Beltran, A.J.G. Gray, P. Groth, C. Goble, J.S. Grethe, J. Heringa, P.A.C. ‘t Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, J.S. Lusher, M.E. Martone, A. Mons, A.L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M.A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao, B. Mons, The FAIR Guiding Principles for scientific data management and stewardship, *Sci. Data*. 3 (2016) 160018, <https://doi.org/10.1038/sdata.2016.18>.
- [28] P.A. Harris, R. Taylor, B.L. Minor, V. Elliott, M. Fernandez, L. O’Neal, L. McLeod, G. Delacqua, F. Delacqua, J. Kirby, S.N. Duda, REDCap Consortium, The REDCap consortium: Building an international community of software platform partners, *J. Biomed. Inform.* 95 (2019), 103208, <https://doi.org/10.1016/j.jbi.2019.103208>.
- [29] J.S. Amberger, C.A. Bocchini, F. Schiettecatte, A.F. Scott, A. Hamosh, OMIM.org: Online Mendelian Inheritance in Man (OMIM®), an online catalog of human genes and genetic disorders, D789-798, *Nucleic Acids Res.* 43 (2015), <https://doi.org/10.1093/nar/gku1205>.
- [30] D. Danis, J.O.B. Jacobsen, A.H. Wagner, T. Groza, M.A. Beckwith, L. Rekerle, L. C. Carmody, J. Reese, H. Hegde, M.S. Ladewig, B. Seitz, M. Munoz-Torres, N. L. Harris, J. Rambla, M. Baudis, C.J. Mungall, M.A. Haendel, P.N. Robinson, Phenopacket-tools: Building and validating GA4GH Phenopackets, *PLoS One*. 18 (2023) e0285433.
- [31] Y. Yu, N. Zong, A. Wen, S. Liu, D.J. Stone, D. Knaack, A.M. Chamberlain, E. Pfaff, D. Gabriel, C.G. Chute, N. Shah, G. Jiang, Developing an FTL tool for converting the PCORnet CDM into the OMOP CDM to facilitate the COVID-19 data integration, *J. Biomed. Inform.* 127 (2022), 104002, <https://doi.org/10.1016/j.jbi.2022.104002>.

- [32] A.-K. Kock-Schoppenhauer, H. Ulrich, S. Wagen-Zink, P. Duhm-Harbeck, J. Ingenerf, P. Neuhaus, M. Dugas, P. Bruland, Compatibility between metadata standards: import pipeline of CDISC ODM to the Samplify.MDR, *Stud. Health Technol. Inform.* 247 (2018) 221–225.
- [33] O. Gologorskaya, L. Yuan, Meyeon Park, R. Hink, D. Huang, Vijaykumar Rayanker, N. Lee, H. Bijli, Govardian Giri, A. Shetty, M. Pletcher, EPIC EMR to OMOP CDM to Research Data Mart: An Unmaintained Road or a Highway?, (2017). <https://doi.org/10.13140/RG.2.2.19755.64801>.
- [34] M.J. Gurley, J. Warner, Y. Bushmanova, F. Wehbe, REDCap2OMOP: A platform for ETling REDCap projects into the OMOP CDM, in: 2021. https://www.ohdsi.org/wp-content/uploads/2021/09/65_poster-REDCap2OMOP.pdf.
- [35] S. Gessner, M. Storck, S. Hegselmann, M. Dugas, I. Soto-Rey, Automated Transformation of CDISC ODM to OpenClinica, *Stud. Health Technol. Inform.* 243 (2017) 95–99.
- [36] E. Girani, M. Gabetta, A. Alloni, M. Stuppia, L. Sacchi, N. Barbarini, Automatic Data Transfer from OMOP-CDM to REDCap: A Semantically-Enriched Framework, in: J. Delgado, A. Benis, P. de Toledo, P. Gallos, M. Giacomini, A. Martínez-García, D. Salvi (Eds.), *Stud. Health Technol. Inform.*, IOS Press, 2021. <https://doi.org/10.3233/SHTI210805>.
- [37] K. Yamamoto, K. Ota, I. Akiya, A. Shintani, A pragmatic method for transforming clinical research data from the research electronic data capture “REDCap” to Clinical Data Interchange Standards Consortium (CDISC) Study Data Tabulation Model (SDTM): Development and evaluation of REDCap2SDTM, *J. Biomed. Inform.* 70 (2017) 65–76, <https://doi.org/10.1016/j.jbi.2017.05.003>.
- [38] K. Yamamoto, Introduction to Research Electronic Data Capture (REDCap) and REDCap2SDTM, a Conversion Tool to Facilitate Clinical Research Data Sharing, *Brain Nerve Shinkai Kenkyu No Shinpo*. 69 (2017) 848–855. <https://doi.org/10.11477/mf.1416200830>.
- [39] T. Oda, S.-W. Chiu, T. Yamaguchi, Semi-automated Conversion of Clinical Trial Legacy Data into CDISC SDTM Standards Format Using Supervised Machine Learning, *Methods, Inf. Med.* 60 (2021) 49–61, <https://doi.org/10.1055/s-0041-1731388>.
- [40] M. Dugas, ODM2CDA and CDA2ODM: tools to convert documentation forms between EDC and EHR systems, *BMC Med. Inform. Decis. Mak.* 15 (2015) 40, <https://doi.org/10.1186/s12911-015-0163-5>.
- [41] I. Soto-Rey, P. Neuhaus, P. Bruland, S. Gefner, J. Varghese, S. Hegselmann, T. Brix, M. Dugas, M. Storck, Standardising the Development of ODM Converters: The ODMToolBox, *Stud. Health Technol. Inform.* 247 (2018) 231–235.
- [42] I. Soto-Rey, S. Gefner, M. Dugas, Operational Data Model Conversion to ResearchKit, *Stud. Health Technol. Inform.* 245 (2017) 225–229.
- [43] J. Doods, P. Neuhaus, M. Dugas, Converting ODM Metadata to FHIR Questionnaire Resources, *Stud. Health Technol. Inform.* 228 (2016) 456–460.
- [44] V. Papez, M. Moinat, S. Payralbe, F.W. Asselbergs, R.T. Lumbers, H. Hemingway, R. Dobson, S. Denaxas, Transforming and evaluating electronic health record disease phenotyping algorithms using the OMOP common data model: a case study in heart failure, *JAMIA Open*. 4 (2021) ooab001, <https://doi.org/10.1093/jamiaopen/ooab001>.
- [45] V. Papez, M. Moinat, E.A. Voss, S. Bazakou, A. Van Winzum, A. Peviani, S. Payralbe, M. Kallfelz, F.W. Asselbergs, D. Prieto-Alhambra, R.J.B. Dobson, S. Denaxas, Transforming and evaluating the UK Biobank to the OMOP Common Data Model for COVID-19 research and beyond, *J. Am. Med. Inform. Assoc. JAMIA*. 30 (2022) 103–111, <https://doi.org/10.1093/jamia/ocac203>.
- [46] N. Queral-Rosinach, P.A. Moreno, T. Callahan, G. Delussu, C. Fraboulet, J. Jacobsen, L.J. Castro, R. Kaliyaperumal, M. Kulmanov, P. Robinson, V. Satagopam, A. Siapas, V. Touré, D. Welter, Mapping OHDSI OMOP Common Data Model and GA4GH Phenopackets for COVID-19 disease epidemics and analytics, *BioHackrXiv* (2022). <https://doi.org/10.37044/oxf.io/ep3xh>.
- [47] M.A. Freeberg, L.A. Fromont, T. D'Altri, A.F. Romero, J.I. Ciges, A. Jene, G. Kerry, M. Moldes, R. Ariosa, S. Bahena, D. Barrowdale, M.C. Barbero, D. Fernandez-Orth, C. Garcia-Linares, E. Garcia-Rios, F. Haziza, B. Juhasz, O.M. Llobet, G. Milla, A. Mohan, M. Rueda, A. Sankar, D. Shaju, A. Shimpi, B. Singh, C. Thomas, S. de la Torre, U. Uyan, C. Vasallo, P. Flicek, R. Guigo, A. Navarro, H. Parkinson, T. Keane, J. Rambla, The European Genome-phenome Archive in 2021, D980–D987, *Nucleic Acids Res.* 50 (2022), <https://doi.org/10.1093/nar/gkab1059>.
- [48] A. Kline, H. Wang, Y. Li, S. Dennis, M. Hutch, Z. Xu, F. Wang, F. Cheng, Y. Luo, Multimodal machine learning in precision health: A scoping review, *NPJ Digit. Med.* 5 (2022) 171, <https://doi.org/10.1038/s41746-022-00712-8>.
- [49] K.B. Johnson, W.-Q. Wei, D. Weeraratne, M.E. Frisse, K. Misulis, K. Rhee, J. Zhao, J.L. Snowden, Precision Medicine, AI, and the Future of Personalized Health Care, *Clin. Transl. Sci.* 14 (2021) 86–93, <https://doi.org/10.1111/cts.12884>.

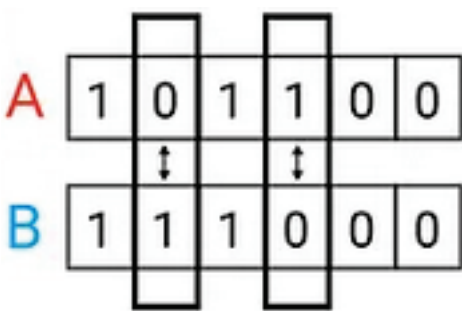
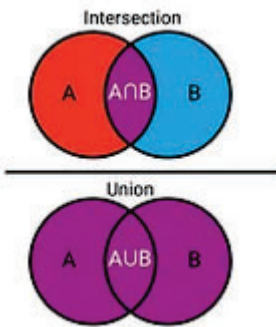
4.2.1.1 Additional features not included in the Convert-Pheno publication

At the time point of the manuscript submission Convert-Pheno and its UI only supported the conversion of REDCap exports and OMOP-CDM table to GA4GH standards, However, now tabular data files, such as TSV files, are also supported as input formats. Similarly to the REDCap data, the user needs to provide a mapping file which maps the columns of the tabular data file to the GA4GH standard fields. The mapping file must be in YAML format and can be uploaded directly in the UI.

4.2.2 Pheno-Ranker-UI

The Pheno-Ranker-UI is a web interface for Pheno-Ranker, a toolkit for comparison of phenotypic data stored in GA4GH standards and beyond. It allows users to perform semantic similarity analysis of pheno-clinical data using Hamming distance or Jaccard Index (Table 4.7). The users can choose between two ranking modes: cohort and patient. While cohort mode offers an all-vs-all cross comparison of individuals, patient mode helps with a one-to-many comparison to elucidate which individuals in the cohort(s) are the most dissimilar/similar to a selected patient. For all these comparisons, first the pheno-clinical data gets flattened and one-hot-encoded. The obtained binary representations of the individuals' pheno-clinical information is compared using Hamming distance and Jaccard Index as explained in Table 4.7.

Table 4.7: Similarity metrics employed in Pheno-Ranker

	Hamming distance	Jaccard Index
Graphical representation (from medium.com/data-science/9-distance-measures-in-data-science-918109d069fa)		
Explanation	Counts the number of dissimilar terms.	Measures the similarity between two sets by dividing the size of the intersection by the size of the union of the sets.
Calculation speed	Very fast, it scales linearly with the number of features ($O(N)$).	In general slower, $O(N)$ for dense data but $O(N \log N)$ for sparse data with 1000s of variables.
When best to use	No or few missing values ($< 30\%$), More would inflate similarity and mask true differences.	$> 30\%$ missing values, since it ignores shared missing values.

The main purpose of these utilities is to enable the user to explore the potential of the UI without requiring the user to upload their own data. One example is the view I have built on top of Pheno-Ranker's *bff-pxf-simulator*, which generates synthetic Pheno-Ranker input data either in the form of PXF or BFF files based on the default or

[illegible]

These Pheno-Ranker input views of both modes come with a sidebar that allows the users first to set a custom patient identifier (ID) prefix, to quickly grasp the patients' cohort in the ranking output when comparing patients across multiple cohorts. Second, to adjust the weight of specific terms in the input data and third allow the inclusion or

exclusion of top-level terms by drag and drop (Figure 4.16 (c) top right). The two latter options can be useful when the user wants to investigate the impact of specific phenoclinical information on the similarity score.

After the ranking process is finished, the main outputs are a multidimensional scaling (MDS) plot and a heatmap. While in both visualisations the Hamming distance is used as the similarity measure, the MDS plot relies solely on the distance matrix to position samples in a two-dimensional space, whereas the heatmap combines that distance matrix with complete-linkage hierarchical clustering to group similar samples. (Figure 4.16 (c), bottom). Note that the insights obtained from the heatmap differ between the used modes. While in the cohort mode with only one cohort it provides an overview over the cohort's heterogeneity, in the cohort mode with two or more cohorts it gives insights into the differences between the cohorts. In the patient mode the target is indicated by bold text or specific colouring. This way the user can easily see how similar, respectively dissimilar the target is to the patients in the reference cohort(s). In addition to the output of the MDS plot and the heatmap the patient mode comes with three extra tables. One showing the binary representation of each patient and another one displaying the ranking of the patients in the reference cohort compared to the query patient (Figure 4.16 (d) top). The latter can be sorted by either one of the available similarity metrics (Hamming distance or Jaccard Index) and their respective p-values/z-scores. In these two tables the user can then select a patient of interest which renders a third table giving the user the possibility to see in which phenoclinic information the patient differs from the reference cohort (Figure 4.16 (d) bottom).

The advantages of the UI over using Pheno-Ranker's CLI are with regards to the output handling that the users would not need to (i) plot the distance matrices to get any insights or (ii) have to parse the patient mode's JSON to get explore in detail the differences between two specific individuals. In addition the UI also simplifies the running of the tool, since it can be argued that especially setting inclusion or exclusion criteria via drag and drop instead via text is much more convenient. Similarly, being able to access past runs in the UI's history sidebar (Figure 4.16 (d) right; shown in its collapsed state) can be perceived to be more comfortable than going through the hierarchy of folders to find a specific plot.

Finally the UI, also comes with UI exclusive utilities to get example data. One utility worth pointing out utilises the cohorts within the Phenopackets corpus of the Monarch Initiative. It is the first web UI-based method that allows users to select specific cohorts and merge the corresponding individuals' PXF files into a single file.

Pheno-Ranker-UI has been open sourced (github.com/CNAG-Biomedical-Informatics/pheno-ranker-ui) and published together with Pheno-Ranker's CLI in the Journal BMC Bioinformatics. A playground version is available at: <https://pheno-ranker.cnag.eu>. The article "Pheno-Ranker: A Toolkit for Comparison of Phenotypic Data Stored in GA4GH Standards and Beyond" can be found online doi.org/10.1016/j.jbi.2023.104558 or below:

Leist et al. *BMC Bioinformatics* (2024) 25:373
<https://doi.org/10.1186/s12859-024-05993-2>

BMC Bioinformatics

SOFTWARE

Open Access

Pheno-Ranker: a toolkit for comparison of phenotypic data stored in GA4GH standards and beyond

Ivo C. Leist^{1,2}, María Rivas-Torrubia³, Marta E. Alarcón-Riquelme^{3,4}, Guillermo Barturen^{3,5,6}, PRECISEADS Clinical Consortium⁷, Ivo G. Gut^{1,2} and Manuel Rueda^{1,2*}

*Correspondence:
manuel.rueda@cnag.eu

¹ Centro Nacional de Análisis Genómico, C/Baldiri Reixac 4, 08028 Barcelona, Spain

² Universitat de Barcelona (UB), Barcelona, Spain

³ Pfizer–University of Granada–Junta de Andalucía Centre for Genomics and Oncological Research, Granada, Spain

⁴ Institute of Environmental Medicine, Karolinska Institute, Stockholm, Sweden

⁵ Department of Genetics, Faculty of Science, University of Granada, 18071 Granada, Spain

⁶ Bioinformatics Laboratory, Centro de Investigación Biomédica, Biotechnology Institute, PTS, Avda del Conocimiento S/N, 18100 Granada, Spain

Abstract

Background: Phenotypic data comparison is essential for disease association studies, patient stratification, and genotype–phenotype correlation analysis. To support these efforts, the Global Alliance for Genomics and Health (GA4GH) established Phenopackets v2 and Beacon v2 standards for storing, sharing, and discovering genomic and phenotypic data. These standards provide a consistent framework for organizing biological data, simplifying their transformation into computer-friendly formats. However, matching participants using GA4GH-based formats remains challenging, as current methods are not fully compatible, limiting their effectiveness.

Results: Here, we introduce Pheno-Ranker, an open-source software toolkit for individual-level comparison of phenotypic data. As input, it accepts JSON/YAML data exchange formats from Beacon v2 and Phenopackets v2 data models, as well as any data structure encoded in JSON, YAML, or CSV formats. Internally, the hierarchical data structure is flattened to one dimension and then transformed through one-hot encoding. This allows for efficient pairwise (all-to-all) comparisons within cohorts or for matching of a patient's profile in cohorts. Users have the flexibility to refine their comparisons by including or excluding terms, applying weights to variables, and obtaining statistical significance through Z-scores and *p*-values. The output consists of text files, which can be further analyzed using unsupervised learning techniques, such as clustering or multidimensional scaling (MDS), and with graph analytics. Pheno-Ranker's performance has been validated with simulated and synthetic data, showing its accuracy, robustness, and efficiency across various health data scenarios. A real data use case from the PRECISEADS study highlights its practical utility in clinical research.

Conclusions: Pheno-Ranker is a user-friendly, lightweight software for semantic similarity analysis of phenotypic data in Beacon v2 and Phenopackets v2 formats, extendable to other data types. It enables the comparison of a wide range of variables beyond HPO or OMIM terms while preserving full context. The software is designed as a command-line tool with additional utilities for CSV import, data simulation, summary statistics plotting, and QR code generation. For interactive analysis, it also includes a web-based user interface built with R Shiny. Links to the online documentation,



© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

including a Google Colab tutorial, and the tool's source code are available on the project home page: <https://github.com/CNAG-Biomedical-Informatics/pheno-ranker>.

Keywords: GA4GH, Phenopacket v2, Beacon v2, Health data model, Semantic similarity, Genomics

Background

Comparing phenotypic and clinical (“pheno-clinical”) data is essential in both research and clinical settings, facilitating informed decisions across a wide range of diseases [1, 2]. For instance, the use of similarity matching contributes to the accurate diagnosis of diseases by aligning patient profiles with comparable cases [3–5]. Similarity matching also has been used in the development of human disease networks, grouping diseases based on common traits to deepen our understanding of their origins [6]. These analytical methods are key in propelling medical research forward and improving patient care, offering essential insights into a variety of medical conditions.

Standardization of phenotypic data is crucial for enabling meaningful patient matching in clinical research [7]. Historically, researchers faced challenges in working with phenotypic data due to the lack of a unified format and consistent nomenclature across different research centers. This variation in data representation hindered data sharing, integration, and effective collaboration. To overcome these challenges, health data standards have been developed to promote the harmonization of variable names and values [8]. These standards provide a common framework for organizing and structuring data, allowing researchers to use standardized vocabularies to define the values for variables [9]. By using standardized terms, researchers can ensure that phenotypic data are described consistently, enabling more accurate and reliable similarity matching and analysis across different datasets and research centers [10, 11]. This standardization enhances data interoperability, facilitates data discovery, and promotes collaborative research efforts in the field of phenotypic analysis.

The use of standardized vocabularies has paved the way for performing matching based on terms (or semantic) similarity [12, 13]. In this regard, many methods have emerged to compare pheno-clinical data encoded with SNOMED CT [14], OMIM [15], ICD-10 [16] or HPO (Human Phenotype Ontology) [17] terms [4, 10, 18–36]. The widespread use of HPO in comparison methods is not coincidental, stemming from its foundation as a standardized vocabulary that structures data into a directed acyclic graph (DAG). This design allows each term to have a unique identifier and a set of relationships with other terms, facilitating the detailed representation of phenotypic abnormalities in human diseases and the hierarchical nature of disease terms and their interconnections. While HPO matching provides valuable insights, it is limited to phenotypic features. As a result, researchers often include genes, proteins or variants to expand the scope of their search [5, 37–49].

The GA4GH [50] has recently approved the Phenopacket v2 Schema [51], a standard for sharing and interoperability of pheno-clinical and genomic data, and Beacon v2 [52], which not only includes models for biological data but also provides an API specification for federated data discovery. Both standards are gaining popularity among research centers, for their structured and compact data schemas, which promote effective data sharing and integration initiatives [50]. A peculiarity of Beacon v2 and Phenopacket v2

schemas is that they do not strictly prescribe ontologies. Instead, they recommend certain ontologies for use within them [53]. For example, NCIt [54] or HPO terms can be used for describing phenotypic features, and diseases can be described using OMIM or ICD-10 terminologies. Currently, to the best of our knowledge, no tool is available that can directly perform comparisons of individuals in deeply nested data such as those present in Phenopackets v2 and Beacon v2 data models.

Here, we present Pheno-Ranker, an open-source toolkit designed for semantic similarity analysis of pheno-clinical data. It specifically supports GA4GH standards Phenopackets v2 and Beacon v2, accepting their JSON/YAML data exchange formats as input. This allows Pheno-Ranker to compare individuals within cohorts or to find the best matches of a patient profile in cohorts. While it primarily processes health-related data, Pheno-Ranker accepts as input any data serialized in JSON, YAML, and CSV formats. Researchers can include or exclude terms and define their own weightings for variables, empowering them to focus on relevant traits and prioritize specific comparisons. The output from Pheno-Ranker is text files that can be further analyzed (with scripts included in the tool's GitHub repository [55] and the online documentation [56]) using unsupervised learning techniques such as clustering [57] and multidimensional scaling (MDS) [58], or through graph analytics [59–61]. This flexibility makes Pheno-Ranker a powerful tool not only in healthcare contexts but in any domain requiring data comparison. The software includes comprehensive online documentation (including a Google Colab notebook) and a web user interface built as an R Shiny app [62].

The remainder of this manuscript is structured as follows: the 'Implementation' section explains the components, data formats, and the algorithm of Pheno-Ranker. The 'Results and Discussion' section evaluates Pheno-Ranker's performance on simulated and real-world datasets, and discusses its utility and limitations. The 'Conclusion' summarizes the key findings and potential future applications.

Implementation

Components

The core of Pheno-Ranker is a Perl module designed to be executed via a command-line interface (CLI) script (see online documentation [56]). In addition to the CLI, we developed a Web App User Interface (UI) based on R Shiny [63] for interactive analyses, available at <https://pheno-ranker.cnag.eu> (see full architecture in Additional file 1: Fig. SF1). During the development of the software, we encountered several critical issues, which are addressed in Additional file 2: Tab. ST1. Installation instructions for both the web application and CLI are provided in the 'Download and Installation' section of the article, as well as in the online documentation.

Input files

Pheno-Ranker natively processes JSON and YAML text files. The two formats supported out-of-the-box are:

- i. The *individuals* entity of Beacon v2 Models, encompassing the (top) terms *diseases*, *ethnicity*, *exposures*, *geographicOrigin*, *id*, *interventionsOrProcedures*, *kary-*

otipicSex, *measures*, *pedigrees*, *phenotypicFeatures*, *sex* and *treatments*, serialized into the data exchange file known as Beacon Friendly Format (BFF) [64, 65].

- ii. The *phenopacket* top-element of the Phenopacket v2 schema that encompasses the terms *id*, *subject*, *phenotypicFeatures*, *measurements*, *biosamples*, *interpretations*, *diseases*, *medicalActions*, *files* and *metaData*, serialized into Phenotype Exchange Format (PXF) files [66].

Both file formats have a similar data structure in which biomedical data are hierarchically structured (i.e., tree-like) under the terms. Besides these formats, the software accepts any other JSON/YAML data, regardless of being biomedically related or not (movies, books, etc.) if they are accompanied by a configuration file. The documentation [56] has examples of how to use Pheno-Ranker with various types of data.

Algorithm

The Pheno-Ranker computational algorithm works as follows (see also Additional file 1: Fig. SF2a):

1. Each object containing one individual (loaded from PXF or BFF files) for the reference cohort(s) is “flattened” into a one-dimensional hash data structure (i.e., associative array or lookup table) and the variables are initialized with weights of one. For terms that consist of arrays of objects (e.g., *phenotypicFeatures*), the element indices are replaced with the CURIE-style identifiers (“id”) from the *required* ontology class (see Additional file 1: Fig. SF2b). We used an ad-hoc filtering (that can be changed with a configuration file) to filter out variables that do not provide any value to the similarity.
2. We generate a global hash for the reference cohort(s) by utilizing the unique variables. The size of the hash depends on the number of variables present in the cohort. The algorithm is optimized to handle many variables, even exceeding 10,000 (e.g., when considering genomic variation data). To address any potential limitations, the algorithm allows selecting a subset of *N* random variables from the total available (with the flag `--max-number-var`).
3. We use the global hash to convert categorical data into numerical form through one-hot encoding. For each individual in the reference cohort(s), we create a binary string (also referred in the text as “binary vector” or simply as “vector”) reflecting the variables in the global hash. The characters within this vector coincide with the global hash’s sorted keys, marking a ‘1’ for each variable present in an individual’s data and a ‘0’ for absent variables. The length of the vector corresponds to the number of keys in the global hash, ensuring a uniform representation of each individual’s data in line with the global hash’s size.
4. When working with a target patient’s data from a JSON file, it is flattened using the same method as described in step one. We then calculate the patient’s binary vector using the global hash derived from the cohort, omitting any variables unique to the patient. This approach of excluding patient-specific variables makes it easier to search within unrelated databases that contain pre-computed data.
5. Compute metrics using the binary strings.

The software allows the user to include or exclude specific terms to make the comparison more precise. For instance, users can include terms like *sex*, *ethnicity*, and *measures* or exclude terms such as *id* from the comparison. The full set of terms can be found in the online documentation. Finally, if the files contain HPO terms, it is possible to include ascendant terms in the comparisons.

Metrics and output

Pheno-Ranker operates in two modes: *cohort* mode and *patient* mode, with the metrics depending on the mode selected.

In *cohort mode*, Pheno-Ranker computes pairwise comparisons between individuals within the cohort(s). The default metric is the Hamming distance [67] which assesses dissimilarity by counting the differing positions between two binary strings (a distance of 0 indicates identical variables between two individuals). Additionally, the software can calculate Jaccard indexes [68] to measure similarity. The Jaccard index, more effective with incomplete data, measures the ratio of intersection over the union (a value of 1 indicates identical variables between two individuals). The resulting output is a symmetrical square matrix from all $N \times N$ comparisons (N =number of individuals). During development, we tested several other metrics and found minimal impact on the overall results (see Additional file 1: Fig. SF3). Additionally, since the binary vectors can be exported as text files, users have the flexibility to conduct further analysis using alternative metrics or tools if desired.

In *patient* mode, the goal is to compare patient data with all individuals in the reference cohort(s) through pairwise comparisons. By default, the output is a table listing the most similar individuals, sorted by Hamming distance, but the sorting method can be configured to use the Jaccard index instead. In this mode, assessing the statistical significance of match results is crucial as it enables users to make well-informed decisions. To facilitate this, we provide standardized values for both metrics in the form of Z-scores, along with their corresponding *p*-values (see extended information at Additional file 3).

All intermediate files generated by Pheno-Ranker (including associative arrays, and data on variable coverage in the cohort(s)) can be exported using the `-export` option. In patient mode, the option `-align` exports “alignment” files. The software enables users to assign specific weights to any variable in the vector, with the weights adding repetitions to the characters in the binary digit string for efficient comparison without affecting computational speed (e.g., weight of 5 will create a string of ‘11111’ for that variable if present and ‘00000’ if absent) [69]. With regards to execution time, Additional file 1: Fig. SF9 provides detailed information for datasets of varying sizes.

Included utilities

Pheno-Ranker comes with four additional utilities: (i) *bff-pxf-plot*, a script to create images with summary statistics from BFF/PXF files (see Additional file 1: Fig. SF7), (ii) *bff-pxf-simulator*, a script that generates simulated BFF/PXF files using randomized terms, (iii) *csv2pheno-ranker*, a script for converting any CSV file into a format compatible with Pheno-Ranker, facilitating the use of publicly available datasets (such as those found at Kaggle [70]), and (vi) scripts for encoding and decoding Pheno-Ranker data into QR (Quick Response) codes [71]. Here, the content for the code is the binary vector (see

Implementation/Algorithm section and Additional file 1: Fig. SF8). As an estimate, the capacity for version 40-L QR code is 2,953 bytes, which theoretically enables the storage of up to 7,089 variables (uncompressed) per individual [72]. More information on these utilities can be found in Additional file 2: Table ST2 and the online documentation [56].

Datasets and experimental setup

To assess the precision and efficiency of Pheno-Ranker, we used three datasets from different sources:

- I. *Simulated datasets*: To generate the simulated data, we used the included utility *bff-pxf-simulator*. We tested Pheno-Ranker with both Beacon v2 Models and Phenopackets v2 with their respective data exchange formats (BFF and PXF). For clarity, we will show results using BFF, though the findings are applicable to PXF as well.
- II. *Synthetic dataset*: The data are part of the 'CINECA_synthetic_cohort_EUROPE_UK1', which comprises 2,504 samples with genetic data derived from the 1000 Genomes Project's phase 3 and the Geuvadis study [73]. Accompanying these genetic data are 76 synthetic phenotypic attributes (many of them incomplete) available through the UK Biobank [74]. The phenotypic data were "augmented" with ontology terms and structured as the *individuals* entity of the Beacon v2 Models. Finally, the data were serialized into BFF as described elsewhere [65]. The location of the file is provided in the 'Availability of Data and Materials' section.
- III. *Use Case—PRECISESADS dataset*: As part of the 3TR (Taxonomy, Treatment, Targets, and Remission) project, funded by the European Commission through the Innovative Medicines Initiative, we obtained access to clinical data generated by the PRECISESADS Clinical Consortium (Additional file 4: Part A). The PRECISESADS cohort comprises 617 healthy controls and patients with seven systemic autoimmune diseases (SADs), including 99 cases of mixed connective tissue disease (MCTD), 106 cases of primary antiphospholipid syndrome (PAPS), 385 cases of primary Sjögren's syndrome (pSjS), 376 cases of rheumatoid arthritis (RA), 469 cases of systemic lupus erythematosus (SLE), 402 cases of systemic sclerosis (SSc), and 166 cases of undifferentiated connective tissue disease (UCTD). The raw clinical data for these projects were stored in CSV format and comprised 92 variables, encompassing comorbidities, treatments, phenotypic characteristics of the different diseases, and recruitment information. The CSV file was converted to a format suitable to Pheno-Ranker with the included utility *csv2pheno-ranker*.

On these datasets we employed the metrics explained on the section 'Metrics and output' to measure the (dis)similarity among individuals. In *cohort* mode, the resulting symmetrical square matrix is represented through heatmaps with clusters (see examples at Fig. 3) using R software [57]. To the distance matrices, we applied multidimensional scaling (MDS), a type of non-linear dimensionality reduction technique, to visualize the similarity levels among individual cases within the dataset [58, 75] (R scripts are available at [76]). In the MDS plots, the X and Y axes denote the two most significant dimensions that encapsulate most of the variation in the distances among individuals,

thereby offering a spatial depiction of the pairwise dissimilarities (see examples at Fig. 5 and Additional file 1: Fig. SF5d). Regarding timing, creating the 2D distance matrices was notably efficient, with all tasks completing in under one second on a single core of an Intel i9 Workstation. Note that tables and figures for the simulated and synthetic datasets can be reproduced using the commands available in the share directory of the GitHub repository [76].

Results and discussion

User flowchart

Figure 1 outlines a typical user workflow, divided into: (i) preparation, (ii) ranking, (iii) plotting, and (iv) other features. Steps (i) to (iii) are common to both the CLI and Web App UI, while (iv) utilizes the CLI utilities. Additional details, including versions, programming languages, documentation links, and brief descriptions, are provided in Additional file 2: Tab. ST2.

User interface

We provide a web application (playground accessible at [62]) that offers a user-friendly interface, integrating Pheno-Ranker's various command-line functionalities. A shared demo account is available for testing. For users working with sensitive data, secure access is provided through ORCID iD login. Returning users can quickly access the two ranking modes—cohort and patient—while new users can follow the interactive decision tree for guidance (see Fig. 2a).

Users can: (i) upload their own data in BFF, PXF, or CSV format leveraging the *csv-2pheno-ranker* tool, which converts it into a Pheno-Ranker-compatible format, (ii) select simulated data obtained from the built-in *bff-pxf-simulator* (see Fig. 2b), (iii) use example data from a Phenopackets corpus [77] (see Fig. 2c), and, (iv) employ data directly queried from Beacon v2 APIs (record granularity) [52]. The UI output will vary depending on whether it is in cohort mode or in patient mode. In both modes, the software generates a variety of results such as heatmaps (with clusters) [78], dimensionality reduction plots [58, 75, 79] and graph-based plots [60, 80, 81] (see Fig. 2c). To the patient mode exclusive is the output of HTML tables (Fig. 2d) such as an alignment resembling that in BLAST [82]. A settings bar allows users to select datasets, define patient ID prefixes, and configure advanced options such as weighting variables and excluding specific terms via a drag-and-drop interface. Each utility and ranking mode includes a history sidebar for users to revisit, rename, or delete past runs, which are stored for at least 30 days.

Simulated datasets

Our initial experiment aimed to validate Pheno-Ranker's capability to handle the variability of information present in BFF files (JSON data structures). We created 10 and 100 completely random individuals by selecting 10 *diseases*, *exposures*, *phenotypicFeatures*, *procedures* and *treatments* (shuffled from pools of 100). Apart from the previous 50 properties, each individual was randomly assigned a biological *sex* (male or female) and a *ethnicity*. As shown in Additional File 1: Fig. SF5a and SF5c, which depict heatmaps of pairwise Hamming distances between all individuals in the cohort, the diagonal elements represent perfect matches (self-matches). The

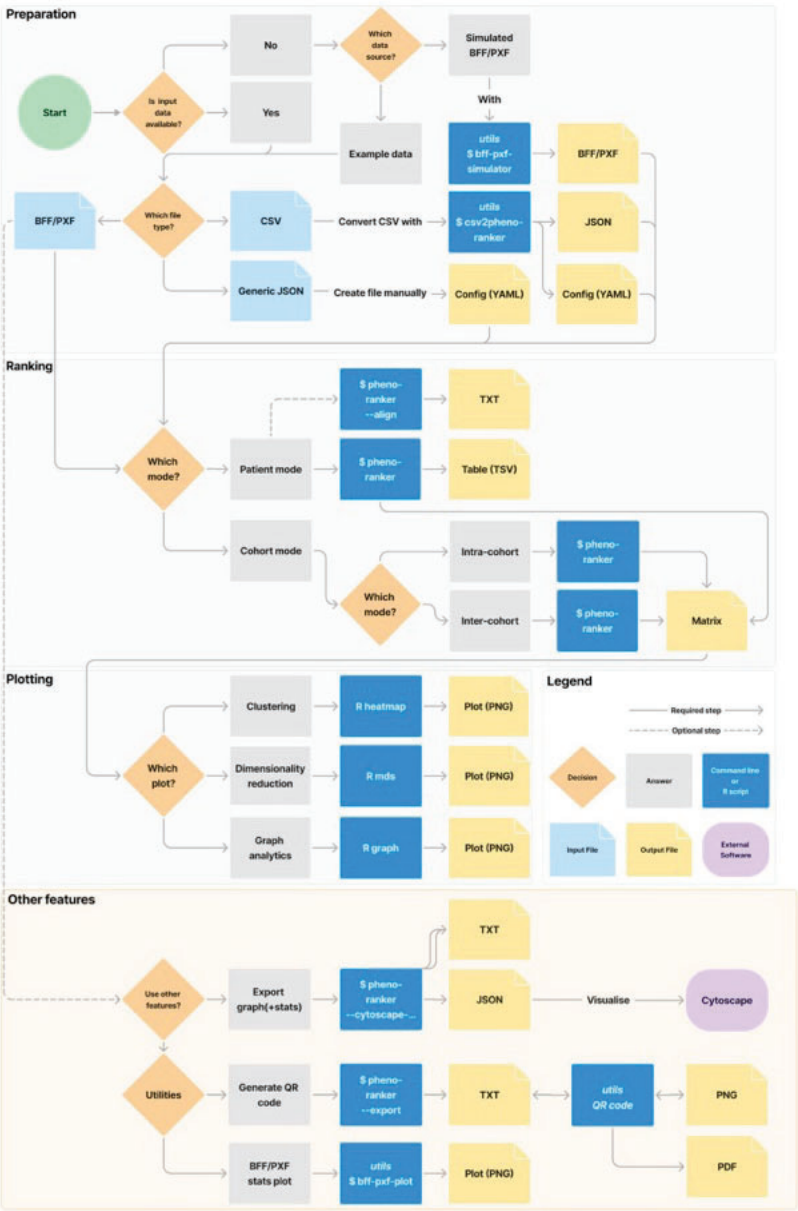


Fig. 1 Flowchart of the available options in the Pheno-Ranker toolkit. 'Preparation,' 'Ranking,' and 'Plotting' can be performed using both the CLI and UI, while 'Other Features' are exclusive to the CLI. For more information, refer to the online documentation and Supporting Table 2

non-diagonal elements show distances ranging from 82 to 104, with a mean of 93.9 and a standard deviation of 4.3 for 10 individuals (see Additional File 1: Fig. SF4b), and from 74 to 106, with a mean of 94.2 and a standard deviation of 4.3 for 100 individuals (see Additional File 1: Fig. SF4d). These values are representative of a lower

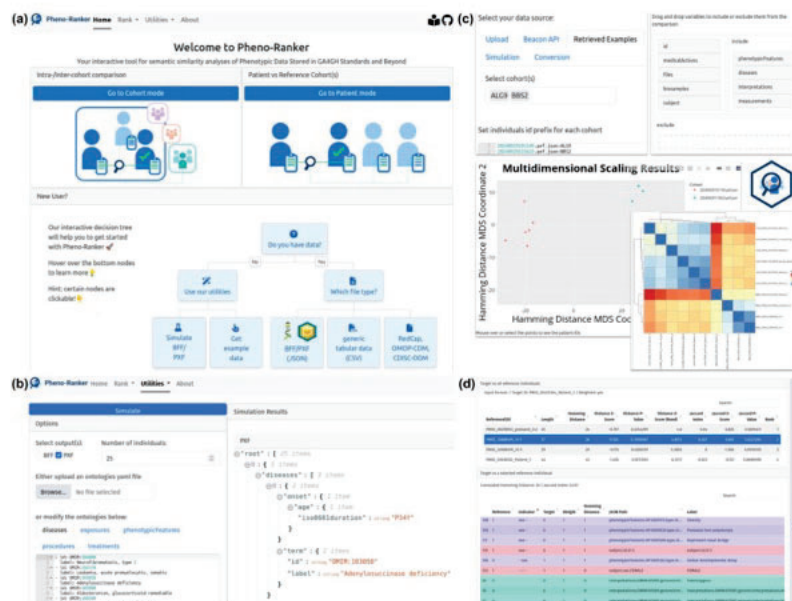


Fig. 2 Screenshots of Pheno-Ranker User Interface. **a** Landing page, where returning users can access ranking modes, while new users can follow the interactive decision tree. **b** Utility for simulating synthetic patient data, customizable with user-defined ontologies. **c** Cohort mode settings and output: Individuals can be uploaded or retrieved from Pheno-Ranker-UI's utilities. Top-level terms (e.g., *phenotypicFeatures*, *diseases*, *interpretations*,...) can be selected via drag-and-drop. Results are displayed as a heatmap or multidimensional scaling matrix scatter plot. These elements are also available in *patient* mode. **d** Tabular output of patient mode, showing differences between two individuals. The top table serves as a selector for the bottom table, which provides a one-to-one comparison of the target and reference patient (highlighted in blue). Rows are colored by top-level terms for clarity

bound similarity (or similarity for a random model) using these number of variables. We used 10 and 100 individuals to show that the behavior is consistent, regardless of the number of individuals.

The second experiment was designed to show the power of Pheno-Ranker for recognizing patterns in a cohort. For this purpose, we created 100 individuals, but this time, each individual had exactly 1 *phenotypicFeatures*, 1 *diseases*, and 1 *treatments* that were randomly selected from pools of 2 (note that the features, diseases and treatments chosen are irrelevant, they just serve for demonstration purposes), along with *sex* and *ethnicity*. When we only included *phenotypicFeatures*, the heatmap depicted two clusters, consisting of the two features from the pool (see Fig. 3a). When we repeat the analysis but this time including *sex*, the 2 clusters became 4 as the two *phenotypicFeatures* were spread equally among the females and males (see Fig. 3b). Continuing with this, we then included *diseases* on the analysis, and we got 8 clusters (see Fig. 3c). Finally, we included the *treatments* and we obtained 16 clusters (see Fig. 3d).

The third experiment aimed to demonstrate the effectiveness of assigning weights to variables. We simulated a cohort of 100 individuals, each with 2 *phenotypicMeasures*, 2 *diseases*, and 2 *treatments*, chosen from pools of 5, along with *sex* and *ethnicity*. In

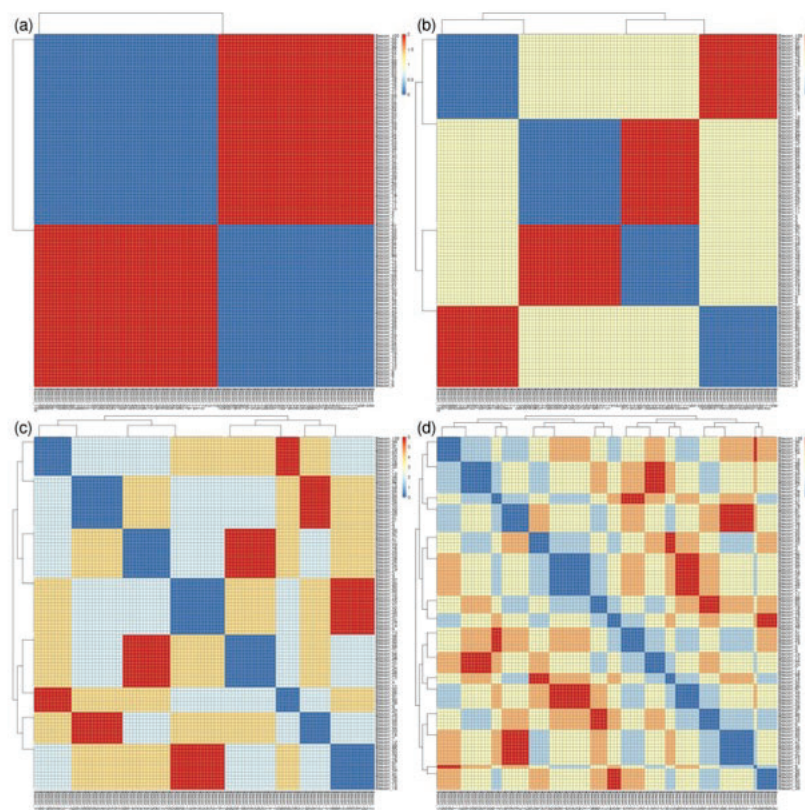


Fig. 3 Heatmaps and clusters generated using Pheno-Ranker from simulated Beacon v2 data (*individuals* entity), based on pairwise Hamming distances between individuals. The dataset comprises 100 individuals, each with 1 *phenotypicFeatures*, 1 *disease*, and 1 *treatment*, randomly selected from a pool of two, along with their *sex* and *ethnicity*. **a** The heatmap shows two clusters when only phenotypic features are considered, representing the two distinct features in the pool. **b** By including sex, the clusters increase to four, distributing the two phenotypic features across both sexes. **c** Adding diseases to the analysis results in eight clusters. **d** Finally, incorporating treatments leads to a total of 16 clusters. Figure results can be reproduced by using the commands at [76]

Additional file 1: Fig. SF5a we display the results when only including *phenotypicFeatures*, *diseases* and *treatments*. In Additional file 1: Fig. SF5b we include a weights file to the calculation, giving the variable 'Caucasian' a weight of 10. In Additional file 1: Fig. SF5c we also added the variable 'RxNorm:1,000,000' (Tribenzor) with a weight of 5. Finally, in Additional file 1: Fig. SF5d we display multidimensional data scaling on data from Additional file 1: Fig. SF5c.

The fourth experiment aimed to assess the ability to stratify patients in situations where data are incomplete or missing, a common issue in real-world datasets. We created two small cohorts of 10 individuals each to ensure labels remained visible in the plots. The first cohort consisted of individuals with a single disease (disease A), while the second cohort included individuals with two diseases, selected from a pool of five (A, B, C, D, and E). This approach generated $C(5,2) = 10$ unique combinations of two diseases, with disease A pairing with B, C, D, or E to form combinations like AB, AC, AD, and

AE. Repetitions of features were not allowed. For a sufficiently large cohort, 40% of individuals in the second cohort were expected to carry disease A. In our sample, only ‘Beacon_7’ (female) and ‘Beacon_10’ (male) from the second cohort had disease A. These two individuals clustered with the first cohort, as shown in Additional File 1: Fig. SF6a, where only *diseases* term was considered. As expected, when the term *sex* was included, the number of clusters doubled (Additional File 1: Fig. SF6b). Recalculating using the Jaccard similarity index yielded the same results (Additional File 1: Fig. SF6c and SF6d).

The previous results demonstrate Pheno-Ranker’s effectiveness in *cohort mode* for classification. Additionally, its *patient mode* allows for comparing a single patient’s data against one or more cohorts, providing various metrics to evaluate the significance of these matches. To evaluate the robustness, and continuing with simulated data, we created a patient consisting of exactly 2 *phenotypicFeatures* sampled from a pool of 25. The reference cohort, comprising 1000 individuals, was created similarly. This setup yields a match probability of ~ 3 individuals, calculated as $1000 / C(25,2)$. In Table 1 we display the results for the first 10 matches. In this experiment, when including only *phenotypicFeatures* in the calculation, our patient got an exact match (distance = 0) with 3 individuals from the simulated reference cohort (Beacon_154, Beacon_444 and Beacon_697). As expected, the distances ranged from 0 (perfect match) to 4 (differences on each position of the 4-character vector) with a mean of 3.85 and a standard deviation of 0.36. To build on this experiment, we generated a new patient and a reference cohort of 1000 individuals, each with 3 *phenotypicFeatures*, 3 *diseases*, and 3 *treatments* chosen from pools of 5. The results, shown in Table 2, revealed an exact match for the new patient with two individuals in the cohort (Beacon_121 and Beacon_49), sharing all variables (9 out of 15 possible), resulting in a Z-score of -3.38 . This run highlights the Z-score metric’s stringency, as matches with a distance equal to 2 (which still have 8 matches) were not deemed significant. It is important to note that the Z-score’s significance is expected to be more pronounced with real data, which typically does not have perfect matches.

Synthetic dataset

Next, we used the synthetic data from the ‘CINECA_synthetic_cohort_EUROPE_UK1’ dataset (see Implementation section, Dataset II), which consists of 2504 individuals. The

Table 1 Descriptors from the comparison between a given simulated patient and a cohort of 1000 individuals with 2 *phenotypicFeatures* sampled from a pool of 25 features. The patient was created with the same conditions. Table results can be reproduced by using the commands at [76]

Reference(ID)	Length	Hamming-distance	Distance-Z-score	Distance-P-value	Distance-Z-score(rand)	Jaccard-index	Jaccard-Z-score	Jaccard-P-value
Beacon_154	2	0	-5.093	0.0000002	-1.4142	1.000	7.447	0.0000000
Beacon_444	2	0	-5.093	0.0000002	-1.4142	1.000	7.447	0.0000000
Beacon_697	2	0	-5.093	0.0000002	-1.4142	1.000	7.447	0.0000000
Beacon_385	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630
Beacon_253	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630
Beacon_192	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630
Beacon_977	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630
Beacon_975	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630
Beacon_898	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630
Beacon_886	3	2	-2.343	0.0095640	0.5774	0.333	2.219	0.1113630

Table 2 Descriptors from the comparison between a given simulated patient and a cohort of 1000 individuals with 3 *phenotypicFeatures*, 3 *diseases* and 3 *treatments* from pools of 5. The patient was created with the same conditions. Table results can be reproduced by using the commands at [76]

Reference(ID)	Length	Hamming-distance	Distance-Z-score	Distance-P-value	Distance-Z-score(RAND)	Jaccard-index	Jaccard-Z-score	Jaccard-P-value
Beacon_121	9	0	-3.381	0.0003608	-3.0000	1.000	4.442	0.0002886
Beacon_49	9	0	-3.381	0.0003608	-3.0000	1.000	4.442	0.0002886
Beacon_821	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_56	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_666	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_74	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_829	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_343	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_337	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852
Beacon_365	10	2	-2.433	0.0074969	-1.8974	0.800	2.847	0.0323852

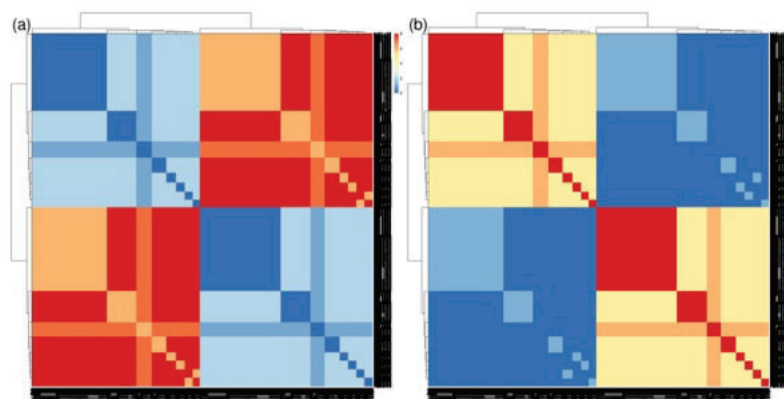


Fig. 4 Heatmaps and clusters generated using Pheno-Ranker from synthetic Beacon v2 data (individuals entity), based on pairwise Hamming distances between individuals. The synthetic dataset 'CINECA_synthetic_cohort_EUROPE_UK1' includes 2504 individuals, with terms *sex* and *ethnicity* incorporated, and weights of 3 assigned to *sex*. **a** Displays the heatmap based on Hamming distance, reflecting dissimilarity among individuals. **b** Shows the heatmap according to the Jaccard index, highlighting similarities. Figure results can be reproduced by using the commands at [76]

purpose of this test was to check whether Pheno-Ranker was able to handle a dataset that is regularly used to debug Beacon v2 API deployments [65]. We performed calculations in *cohort* mode, including the terms *sex* and *ethnicity* and we gave a weight of 3 to the term *sex* that included variables 'sex.id.NCIT:C20197' and 'sex.id.NCIT:C16576'. In Fig. 4a we display the heatmap according to Hamming distance (dissimilarity) and Fig. 4b according to Jaccard index (similarity). Note that both metrics yield similar yet complementary results.

Use case—PRECISESADS dataset

In the previous sections, Pheno-Ranker has undergone extensive testing with simulated and synthetic datasets. However, it is important to acknowledge that such datasets may not fully encompass the variability present in real-world data, as they often contain a

consistent number of variables, unlike the more complex and variable data typically encountered in clinical trials. Therefore, we conducted an analysis on the PRECISESADS clinical records to demonstrate the utility of the tool with a real-world dataset. The analyses were performed by incorporating all available information from the PRECISESADS cohort, with non-informative values (e.g., unknown, missing) replaced by 'NA' (i.e., Not Available).

With the *cohort* mode, we aimed to identify hidden potential confounders and/or clinical features that might differentiate between patients. As anticipated, most clinical features and comorbidities were absent in healthy controls, consistent with the definition of healthy control recruitment, resulting in a distinct cluster separate from the patients (Fig. 5a). SADs are known to share phenotypic commonalities, reflected in a single cluster, but with a more dispersed distribution than that of healthy controls. Notably, two diseases, SSc and SLE, stood out from the other five, distributed in the upper and lower parts of the SADs clusters (Fig. 5a).

To further investigate the characteristics distinguishing SADs patients, *cohort* mode was rerun excluding the healthy controls, and ANOVA analyses were conducted between MDS dimensions and each included variable (see Additional file 5 and R script file [76]). The primary variable associated with both MDS dimensions was the recruitment center.

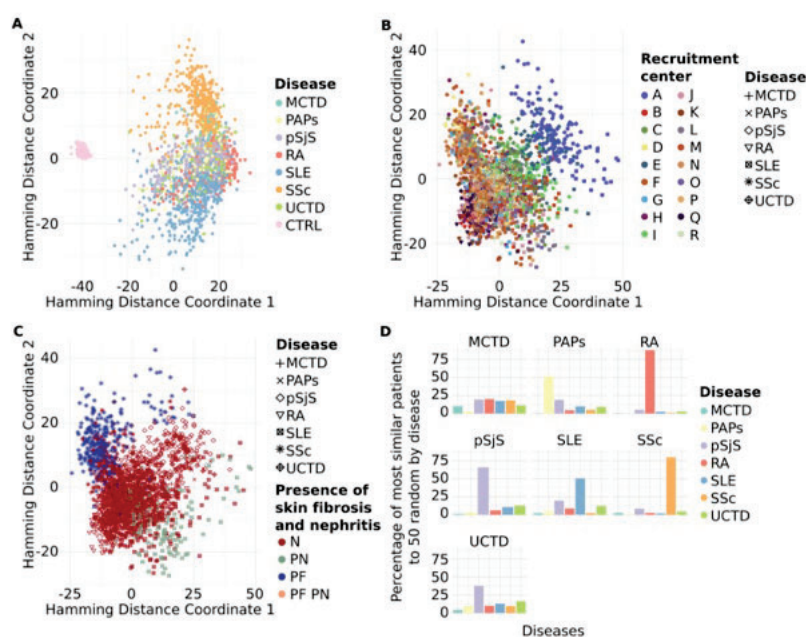


Fig. 5 Pheno-Ranker's analysis of the PRECISESADS dataset involves 7 disease-specific cohorts and 1 control cohort. Figures **a–c** showcase plots generated through multidimensional data scaling (MDS) applied to the distance matrix from an inter-cohort (cohort mode) analysis. Specifically, **a** demonstrates the MDS of the 8 cohorts, including the control, with each cohort colored differently; **b** presents the MDS plot of the 7 diseases, differentiated by research center and symbolized for each disease; and **c** shows the MDS plot of the 7 diseases, highlighted by the presence of skin fibrosis and nephritis. Figure **d** illustrates the Pheno-Ranker's performance on patient mode, revealing the percentage of the most similar patients (from all diseases) to 50 randomly selected patients for each disease

This was reflected in the position of center 'A' in the SADs cluster during dimensional reduction analysis (Fig. 5b). A detailed examination of the variables revealed that center 'A' exhibited the highest proportion of missing values (data not shown), potentially explaining the observed difference. Subsequently, the next two variables most strongly associated with the MDS dimensions were the presence of skin fibrosis, a hallmark of SSc [83], and the presence of nephritis, a major and quite specific clinical complication in SLE patients [84]. These findings elucidated the separation of these two diseases from the rest of the SADs (Fig. 5c).

Finally, we applied Pheno-Ranker in *patient* mode to evaluate its patient-matching capabilities. Figure 5d displays the percentage of patients most similar to 50 randomly chosen individuals from each disease group. Notably, Pheno-Ranker reliably identified similarities across various diseases. Yet, it faced difficulties with Mixed Connective Tissue Disease (MCTD) and Undifferentiated Connective Tissue Disease (UCTD) due to the shared phenotypic features with other conditions.

Features, capabilities, and limitations

Coding schemes like HPO or OMIM terms have greatly enhanced genomic and phenotypic data analysis. Their use boosts semantic similarity analysis, making it easier to identify related terms and concepts, thus enriching our data understanding. To compare phenotypic data encoded in these schemes, various methods have emerged, including text-based comparisons and analyses of term relationships from ontology structures [4, 10, 18–36].

Here, we present Pheno-Ranker, an innovative approach that extends beyond the constraints of tools dependent on pre-selected ontologies. We want to clarify that Pheno-Ranker is not intended to outperform these types of tools. If a researcher wants to compare individuals based on HPO terms (or others such as OMIM, SNOMED CT, etc.), Pheno-Ranker's only strength lies in its ability to directly process BFF or PXF data exchange formats, eliminating the need for data conversions. Rather, its uniqueness lies in the versatility and detail it offers; it is compatible with any standardized terminology leading to more nuanced and flexible outcomes. Pheno-Ranker handles data in JSON/YAML formats, first flattening the data to 1D lookup tables and then converting them into binary digit strings. This conversion conserves data context and streamlines efficient similarity matching, also ensuring compatibility with machine learning algorithms. To our knowledge, Pheno-Ranker is the first tool to directly perform similarity calculations on two GA4GH standards, making it particularly suitable for patient matching (match-making) and stratification in GA4GH-compliant data repositories. The software can be configured to work with other health data standards, like OpenEHR [85], as shown in the example at the online documentation [56]. Additionally, in conjunction with our Convert-Pheno tool [86], it supports data from other clinical data models such as the Observational Medical Outcomes Partnership Common Data Model (OMOP CDM), ensuring compatibility with various health data ecosystems.

Pheno-Ranker includes a suite of utilities, significantly broadening its use beyond YAML or JSON hierarchical data formats. Firstly, *bff-pxf-plot* is a tool developed to create summary statistics plots as PNG images for BFF and PXF formats (see Additional file 1: Fig. SF7 and online documentation [56]). Secondly, *bff-pxf-simulator* is designed

to generate simulated data in BFF/PXF exchange formats, addressing the scarcity of synthetic or simulated datasets for Beacon v2 and Phenopackets v2 in the literature [87, 88]. Our simulated data are versatile, serving purposes ranging from technical, like testing Beacon v2 implementations [65], to scientific, such as creating “control” cases for outlier detection. Thirdly, the *csv2pheno-ranker* utility enables converting CSV files into a format compatible with Pheno-Ranker. This expands the scope of analysis, for example, enabling comparison of samples in a multi-sample VCF based on their “genomic fingerprint”, as shown in our documentation [56]. Fourthly, we offer a set of utilities for encoding/decoding the binary vector into QR codes (PNG images) plus generating PDF reports (see Additional file 1: Fig. SF8 and examples at the documentation [56]). This concept aligns with the health passports [89] that store patient information in barcodes (i.e., those used to encode in COVID-19 vaccination state). Pheno-Ranker efficiently condenses data by directly encoding binary vectors into QR codes, enhancing the capacity for storage of variables within the code. Although currently a proof of concept, our solution could be applied to health data transfer, especially with the implementation of proper security measures, such as data encryption [8]. Other futuristic applications could be patient enrolment in clinical trials or “wireless” similarity matching via a Pheno-Ranker mobile app.

The simplicity of our algorithm makes it suitable for personalized implementations, allowing users to store the one-hot encoded data in a database for efficient comparisons using for instance SQL functions or vector databases. The Pheno-Ranker algorithm could easily be implemented in a federated network, such as Beacon v2 networks (see explanation at Additional file 6). This approach holds promise for collaborative research, such as hospital networks, aiming to find similar patients across different institutions.

One limitation of Pheno-Ranker is its handling of continuous data, such as age or numerical measurements. For age data, both the Beacon v2 and Phenopackets v2 schemas allow the use of ranges instead of specific ages, which enhances privacy. For other continuous variables, we recommend using ranges or pre-processing them into bins/categories whenever possible. Cautious handling of continuous data is essential, depending on the research goals and the nature of the data.

Another limitation of our method is its lack of terminology-matching capability, meaning that variables describing identical concepts across different standardized vocabularies are not recognized as matches. This limitation arises not from a flaw in our approach, but rather from the lack of prescribed ontologies within the Beacon v2 and Phenopacket v2 schemas. Additionally, our method currently supports only exact matches, excluding the possibility of fuzzy searches. Despite these challenges, the rapidly advancing field of ontology mapping, coupled with the ongoing progress in natural language processing (NLP), holds promise for overcoming these limitations soon.

Pheno-Ranker has been rigorously tested using a variety of simulated and synthetic datasets. However, these datasets may not encompass the full variability found in real-world data, like that from clinical trials. To address this, we also analyzed data from the PRECISESADS project, which enabled us to stratify diseases by their unique characteristics, identify outliers, and match patients with the same disease across the entire dataset.

It is important to note, though, that in case-control comparisons, the number of variables between groups can skew similarity measures. The Hamming distance

may be artificially higher for cases with more variables, requiring normalization or a focus on shared variables. The Jaccard index, while less impacted, can underestimate similarity in cases with many additional variables. Addressing this bias involves selecting key variables, considering data imputation for missing values, and performing sensitivity analyses. Expert input is vital for including/excluding relevant variables and validating results. Pheno-Ranker's ability to tailor analysis to key traits demonstrates its capacity for delivering valuable insights from complex datasets.

Data security and privacy are crucial when deploying Pheno-Ranker. Each center is responsible for maintaining the confidentiality, integrity, and availability of their data, and must implement appropriate safeguards to protect sensitive information.

At CNAG, we are preparing to assess Pheno-Ranker's effectiveness with PXF data from the Genome-Phenome Analysis Platform [90], and exploring its capabilities in federated analysis contexts (refer to the Additional file 6). In the broader scope of Pheno-Ranker's capabilities, its applicability is not just confined to genomic and phenotypic data. It can be effectively adapted for use with diverse JSON or CSV datasets, provided an appropriate configuration file is used. This feature significantly widens the range of potential applications, enabling researchers to investigate semantic similarities in various fields.

In conclusion, Pheno-Ranker is a versatile and powerful tool for individual-level comparison of pheno-clinical data. By uncovering hidden patterns and relationships, Pheno-Ranker has the potential to advance personalized medicine, disease classification, and scientific discovery. We look forward to further developments and real-world applications of Pheno-Ranker, leveraging its capabilities to drive progress in genomics and beyond.

Conclusions

We introduce Pheno-Ranker, an open-source software for phenotypic data analysis applicable in research and clinical settings. Compatible with Beacon v2 and Phenopackets v2 data models, it processes data in JSON, YAML, or CSV formats across any domain. Pheno-Ranker offers two operational modes: cohort comparison and individual matching to the closest profiles within cohorts. The output consists of structured text files, which can be further analyzed using included R scripts for unsupervised learning techniques such as clustering, multidimensional scaling and graph analytics, broadening their applicability across various analytical tasks.

Pheno-Ranker includes a user-friendly R Shiny web application and a command-line interface, catering to different user preferences. Multiple installation options, including a containerized version, ensure seamless integration into existing workflows, while online documentation and an interactive Google Colab notebook offer guidance. The software also comes equipped with a suite of utilities that enable generation of simulated data, data visualization through plotting, and creation of QR codes, enhancing its practical utility. Pheno-Ranker enhances research and clinical workflows by enabling comparison of phenotypic data, helping researchers identify significant patterns, match patients, construct cohorts, and explore genotype-phenotype relationships.

Availability and requirements

- *Project name:* Pheno-Ranker
- *Project home page:* <https://github.com/CNAG-Biomedical-Informatics/pheno-ranker>
- *Operating system (s):* Linux, MacOS
- *Programming languages:* Perl, Python, R, JavaScript
- *Other requirements:* Docker
- *License:* Artistic License 2 (CLI) and GNU GPL v3 (UI)
- *Any restrictions to use by non-academics:* None

Download and installation

Pheno-Ranker can be installed locally on Linux or macOS. For the CLI, we offer five installation options, including from GitHub source, containerized options (Docker), and CPAN, depending on whether all utilities are needed (see Additional file 2: Table ST2 and <https://cnag-biomedical-informatics.github.io/pheno-ranker/download-and-installation>). The R Shiny web application can be easily deployed using Docker for a streamlined setup.

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12859-024-05993-2>.

Additional file1 (PDF 3430 KB)
 Additional file2 (PDF 142 KB)
 Additional file3 (PDF 62 KB)
 Additional file4 (PDF 145 KB)
 Additional file5 (XLSX 32 KB)
 Additional file6 (PDF 224 KB)

Acknowledgements

We would like to thank the members of the Biomedical Genomics laboratory for their valuable feedback, Davide Piscia and his team at CNAG for their help with the web application (especially Sofia Chaves for the logo design, help with Figure 1 and usability suggestions), and Gonzalo Vera for his assistance in setting up the R Shiny App. We would also like to thank the two anonymous referees for their valuable feedback and insights. ChatGPT 3.5/4 by OpenAI was used to improve readability and language of some of the paragraphs.

Author contributions

ICL implemented Pheno-Ranker's web app UI, interpreted data and revised the manuscript. MRT and GB performed analysis and interpretation of PRECISEADS data and revised the manuscript. MAR obtained funding and revised the manuscript. PCC generated PRECISEADS dataset. IGG obtained funding and revised the manuscript. MR contributed to the design of the work, developed Pheno-Ranker CLI, interpreted the data and wrote the manuscript. All authors read and approved the final manuscript.

Funding

This work was carried out in the course of the 3TR project with funding from the Innovative Medicines Initiative 2 Joint Undertaking (JU) under grant agreement No 831434. The JU receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA (Disclaimer: Content of this publication reflects only the author's view and the JU is not responsible for any use that may be made of the information it contains). Institutional support was from the Spanish Instituto de Salud Carlos III, Fondo de Investigaciones Sanitarias and cofunded with ERDF funds (PI19/01772). We acknowledge the institutional support of the Spanish Ministry of Science and Innovation through the Instituto de Salud Carlos III and the 2014–2020 Smart Growth Operating Program, and institutional co-financing with the European Regional Development Fund (MINECO/FEDER, BIO2015-71792-P). We also acknowledge the support from the Generalitat de Catalunya through the Departament de Salut and the Departament d'Empresa i Coneixement. This work has received support from the EU/EFPIA Innovative Medicines Initiative Joint Undertaking (PRECISEADS, grant n. 115565) including in-kind contributions from the EFPIA members involved.

Availability of data and materials

All simulated data presented in this paper can be replicated using the utility *bff-pxf-simulator*, which is accessible via CPAN distribution, containerized versions, GitHub repository at https://github.com/CNAG-Biomedical-Informatics/pheno-ranker/tree/main/utills/bff_pxf_simulator, and through the web app user interface (<https://pheno-ranker.cnag.eu>). Synthetic data for the CINECA cohort is available at https://github.com/EGA-archive/eacon2-ri-tools/blob/main/CINECA_synthetic_cohort_EUROPE_UK1/bff. PRECISEADS clinical dataset is hosted by ELIXIR Luxembourg (URL: <https://elixir-luxembourg.org>). The dataset is available upon request, and the access procedure is described on the data landing page ([doi.org/https://doi.org/10.17881/th9v-xt85](https://doi.org/10.17881/th9v-xt85)).

Declarations

Ethics approval and consent to participate

The Ethical Review Boards of the 18 participating institutions approved the protocol of the cross-sectional study (see Additional File 4: Part B for the names of the ethics committees). In addition, the boards of the 6 sites involved approved the inception study protocol. The studies adhered to the standards set by the International Conference on Harmonization and Good Clinical Practice (ICH-GCP), and to the ethical principles that have their origin in the Declaration of Helsinki (2013). All study participants provided written informed consent prior to their enrolment in the PRECISEADS project. The protection of the confidentiality of records that could identify the included subjects is ensured as defined by the EU Directive 2001/20/EC and the applicable national and international requirements relating to data protection in each participating country. The CS study is registered with number NCT02890121, and the inception study with number NCT02890134 in ClinicalTrials.gov.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interest.

Received: 28 March 2024 Accepted: 19 November 2024

Published online: 04 December 2024

References

- Yurkovich JT, Evans SJ, Rappaport N, Boore JL, Lovejoy JC, Price ND, et al. The transition from genomics to phenomics in personalized population health. *Nat Rev Genet.* 2023. <https://doi.org/10.1038/s41576-023-00674-x>.
- Houle D, Govindaraju DR, Omholt S. Phenomics: the next challenge. *Nat Rev Genet.* 2010;11:855–66.
- Jacobsen JOB, Kelly C, Cipriani V, Research Consortium GE, Mungall CJ, Reese J, et al. Phenotype-driven approaches to enhance variant prioritization and diagnosis of rare disease. *Hum Mutat.* 2022; 43: 1071–81.
- Lagorce D, Lebreton E, Matalonga L, Hongnat O, Chahdil M, Piscia D, et al. Phenotypic similarity-based approach for variant prioritization for unsolved rare disease: a preliminary methodological report. *Eur J Hum Genet.* 2023. <https://doi.org/10.1038/s41431-023-01486-7>.
- Buske OJ, Girdea M, Dumitriu S, Gallinger B, Hartley T, Trang H, et al. PhenomeCentral: a portal for phenotypic and genotypic matchmaking of patients with rare genetic diseases. *Hum Mutat.* 2015;36:931–40.
- Chen J, Xu H, Jegga A, Zhang K, White PS, Zhang G. Novel phenotype-disease matching tool for rare genetic diseases. *Genet Med.* 2019;21:339–46.
- Parimbelli E, Marini S, Sacchi L, Bellazzi R. Patient similarity for precision medicine: a systematic review. *J Biomed Inform.* 2018;83:87–96.
- Stellmach C, Muzoora MR, Thun S. Digitalization of health data: interoperability of the proposed european health data space. *Stud Health Technol Inform.* 2022;298:132–6.
- Bard JBL, Rhee SY. Ontologies in biology: design, applications and future challenges. *Nat Rev Genet.* 2004;5:213–22.
- Pesquita C, Faria D, Falcão AO, Lord P, Couto FM. Semantic similarity in biomedical ontologies. *PLoS Comput Biol.* 2009;5:e1000443.
- Bernabé CH, Queralt-Rosinach N, Silva Souza VE, da Silva B, Santos LO, Mons B, Jacobsen A, et al. The use of foundational ontologies in biomedical research. *J Biomed Semant.* 2023;14:21.
- Pedersen T, Pakhomov SVS, Patwardhan S, Chute CG. Measures of semantic similarity and relatedness in the biomedical domain. *J Biomed Inform.* 2007;40:288–99.
- Gan M, Dou X, Jiang R. From ontology to semantic similarity: calculation of ontology-based semantic similarity. *Sci World J.* 2013;2013:793091.
- Lee D, Cornet R, Lau F, de Keizer N. A survey of SNOMED CT implementations. *J Biomed Inform.* 2013;46:87–96.
- Amberger JS, Bocchini CA, Schiettecatte F, Scott AF, Hamosh A. OMIM.org: Online Mendelian Inheritance in Man (OMIM®), an online catalog of human genes and genetic disorders. *Nucleic Acids Res.* 2015;43:D789–798.
- Organization WH. ICD-10 : international statistical classification of diseases and related health problems : tenth revision. Spanish version, 1st edition published by PAHO as Publicación Científica 544, 2004.
- Robinson PN, Mundlos S. The human phenotype ontology. *Clin Genet.* 2010;77:525–34.
- Girardi D, Wartner S, Halmerbauer G, Ehrenmüller M, Kosorus H, Dreiseitl S. Using concept hierarchies to improve calculation of patient similarity. *J Biomed Inform.* 2016;63:66–73.
- Tashkandi A, Wiese L, Wiese L. Efficient in-database patient similarity analysis for personalized medical decision support systems. *Big Data Res.* 2018;13:52–64.

20. Gong X, Jiang J, Duan Z, Lu H. A new method to measure the semantic similarity from query phenotypic abnormalities to diseases based on the human phenotype ontology. *BMC Bioinformatics*. 2018;19(Suppl 4):162.
21. Bauer S, Köhler S, Schulz MH, Robinson PN. Bayesian ontology querying for accurate and noise-tolerant semantic searches. *Bioinformatics*. 2012;28:2502–8.
22. Shen F, Peng S, Fan Y, Wen A, Liu S, Wang Y, et al. HPO2Vec+: Leveraging heterogeneous knowledge resources to enrich node embeddings for the human phenotype ontology. *J Biomed Inform*. 2019;96:103246.
23. Xue H, Peng J, Shang X. Predicting disease-related phenotypes using an integrated phenotype similarity measurement based on HPO. *BMC Syst Biol*. 2019;13(Suppl 2):34.
24. Lancaster O, Beck T, Atlan D, Swertz M, Thangavelu D, Veal C, et al. Cafe Variome: general-purpose software for making genotype-phenotype data discoverable in restricted or open access contexts. *Hum Mutat*. 2015;36:957–64.
25. Masino AJ, Dechene ET, Dulik MC, Wilkens A, Spinner NB, Krantz ID, et al. Clinical phenotype-based gene prioritization: an initial study using semantic similarity and the human phenotype ontology. *BMC Bioinform*. 2014;15:248.
26. Köhler S, Schulz MH, Krawitz P, Bauer S, Dolken S, Ott CE, et al. Clinical diagnostics in human genetics with semantic similarity searches in ontologies. *Am J Hum Genet*. 2009;85:457–64.
27. Yang J, Shu L, Duan H, Li H. A robust phenotype-driven likelihood ratio analysis approach assisting interpretable clinical diagnosis of rare diseases. *J Biomed Inform*. 2023;142:104372.
28. Glueck M, Hamilton P, Chevalier F, Breslav S, Khan A, Wigdor D, et al. PhenoBlocks: phenotype comparison visualizations. *IEEE Trans Visual Comput Graphics*. 2016;22:101–10.
29. Glueck M, Gvozdkik A, Chevalier F, Khan A, Brudno M, Wigdor D. PhenoStacks: cross-sectional cohort phenotype comparison visualizations. *IEEE Trans Visual Comput Graph*. 2017;23:191–200.
30. Peng J, Li Q, Shang X. Investigations on factors influencing HPO-based semantic similarity calculation. *J Biomed Semant*. 2017;8(Suppl 1):34.
31. Peng J, Xue H, Hui W, Lu J, Chen B, Jiang Q, et al. An online tool for measuring and visualizing phenotype similarities using HPO. *BMC Genomics*. 2018;19(Suppl 6):571.
32. Köhler S. Improved ontology-based similarity calculations using a study-wise annotation model. *Database*. 2018;2018:bay026.
33. Groza T, Wu H, Dinger ME, Danis D, Hilton C, Bagley A, et al. Term-BLAST-like alignment tool for concept recognition in noisy clinical texts. *Bioinformatics*. 2023;39:btad716.
34. Melton GB, Parsons S, Morrison FP, Rothschild AS, Markatou M, Hripscak G. Inter-patient distance metrics using SNOMED CT defining relationships. *J Biomed Inform*. 2006;39:697–705.
35. Hier DB, Kopel J, Brint SU, Wunsch DC, Olbricht GR, Azizi S, et al. Evaluation of standard and semantically-augmented distance metrics for neurology patients. *BMC Med Inform Decis Mak*. 2020;20:203.
36. Batet M, Sánchez D, Valls A. An ontology-based measure to compute semantic similarity in biomedicine. *J Biomed Inform*. 2011;44:118–25.
37. Şenay K, Abdelhakim M, Uludag M, Althagafi A, Alghamdi M, Hoehndorf R. Starvar: symptom-based tool for automatic ranking of variants using evidence from literature and genomes. *BMC Bioinformatics*. 2023;24:294.
38. Jagadeesh KA, Birgmeier J, Guturu H, Deisseroth CA, Wenger AM, Bernstein JA, et al. Phrank measures phenotype sets similarity to greatly improve Mendelian diagnostic disease prioritization. *Genet Med*. 2019;21:464–70.
39. Huang D, Jiang J, Zhao T, Wu S, Li P, Lyu Y, et al. diseaseGPS: auxiliary diagnostic system for genetic disorders based on genotype and phenotype. *Bioinformatics*. 2023;39:btad517.
40. Sanjak J, Binder J, Yadaw AS, Zhu Q, Mathé EA. Clustering rare diseases within an ontology-enriched knowledge graph. *J Am Med Inform Assoc*. 2023;19:ocad186.
41. Dursun C, Shimoyama N, Shimoyama M, Schläppi M, Bozdag S. Phenogeneranker: A tool for gene prioritization using complete multiplex heterogeneous networks. In: *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*. 2019. pp. 279–88.
42. Gao J, Liu L, Yao S, Huang X, Mamitsuka H, Zhu S. HPOAnnotator: improving large-scale prediction of HPO annotations by low-rank approximation with HPO semantic similarities and multiple PPI networks. *BMC Med Genomics*. 2019;12(Suppl 10):187.
43. Köhler S, Øien NC, Buske OJ, Groza T, Jacobsen JOB, McNamara C, et al. Encoding clinical data with the human phenotype ontology for computational differential diagnostics. *Curr Protoc Hum Genet*. 2019;103:e92.
44. Smedley D, Jacobsen JOB, Jäger M, Köhler S, Holtgrewe M, Schubach M, et al. Next-generation diagnostics and disease-gene discovery with the Exomiser. *Nat Protoc*. 2015;10:2004–15.
45. Yang H, Robinson PN, Wang K. Phenolyzer: phenotype-based prioritization of candidate genes for human diseases. *Nat Methods*. 2015;12:841–3.
46. Hoehndorf R, Schofield PN, Gkoutos GV. PhenomeNET: a whole-phenome approach to disease gene discovery. *Nucleic Acids Res*. 2011;39:e119.
47. Girdea M, Dumitriu S, Fiume M, Bowdin S, Boycott KM, Chénier S, et al. PhenoTips: patient phenotyping software for clinical and research use. *Hum Mutat*. 2013;34:1057–65.
48. Greene D, BioResource NIHR, Richardson S, Turro E. Phenotype similarity regression for identifying the genetic determinants of rare diseases. *Am J Hum Genet*. 2016;98:490–9.
49. Deng Y, Gao L, Wang B, Guo X. HPOSim: an R package for phenotypic similarity measure and enrichment analysis based on the human phenotype ontology. *PLoS ONE*. 2015;10:e0115692.
50. Rehm HL, Page AJH, Smith L, Adams JB, Alterovitz G, Babb LJ, et al. GA4GH: International policies and standards for data sharing across genomic research and healthcare. *Cell Genom*. 2021;1:100029.
51. Jacobsen JOB, Baudis M, Baynam GS, Beckmann JS, Beltran S, Buske OJ, et al. The GA4GH Phenopacket schema defines a computable representation of clinical data. *Nat Biotechnol*. 2022;40:817–20.
52. Rambla J, Baudis M, Ariosa R, Beck T, Fromont LA, Navarro A, et al. Beacon v2 and Beacon Networks: a “lingua franca” for federated data discovery in biomedical genomics, and beyond. *Hum Mutat*. 2022;43:9.
53. Martínez-Romero M, Jonquet C, O’Connor MJ, Graybeal J, Pazos A, Musen MA. NCBO ontology recommender 2.0: an enhanced approach for biomedical ontology recommendation. *J Biomed Semant*. 2017;8:21.

54. Golbeck J, Fragoso G, Hartel F, Hendler J, Oberthaler J, Parsia B. The National cancer institute's Thésaurus and ontology. *J Web Semant.* 2003;1:75–80.
55. Pheno-Ranker code repository. <https://github.com/CNAG-Biomedical-Informatics/pheno-ranker>. Accessed 24 Sep 2024.
56. Pheno-Ranker documentation. <https://cnag-biomedical-informatics.github.io/pheno-ranker>. Accessed 24 Sep 2024.
57. Kolde R. pheatmap: Pretty Heatmaps. 2019.
58. Wickham H. ggplot2: Elegant graphics for data analysis. New York: Springer-Verlag; 2016.
59. Epskamp S, Cramer AOJ, Waldorp LJ, Schmittmann VD, Borsboom D. qgraph: Network visualizations of relationships in psychometric data. *J Stat Softw.* 2012;48:1–18.
60. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.* 2003;13:2498–504.
61. Csardi G, Nepusz T. The igraph software package for complex network research. *InterJournal.* 2006;Complex Systems:1695.
62. Pheno-Ranker Web App. <https://pheno-ranker.cnag.eu>. Accessed 24 Sep 2024.
63. Chang W, Cheng J, Allaire JJ, Sievert C, Schloerke B, Xie Y, et al. shiny: Web Application Framework for R. 2024.
64. Beacon v2 Documentation. <https://docs.genomebeacons.org>. Accessed 24 Sep 2024.
65. Rueda M, Ariosa R, Moldes M, Rambla J. Beacon v2 Reference Implementation: a toolkit to enable federated sharing of genomic and phenotypic data. *Bioinformatics.* 2022;38:4656–7.
66. Phenopacket v2 Schema documentation. <https://phenopacket-schema.readthedocs.io>. Accessed 24 Sep 2024.
67. Hamming RW. Error detecting and error correcting codes. *Bell Syst Tech J.* 1950;29:147–60.
68. Chung NC, Miasojedow B, Startek M, Gambin A. Jaccard/Tanimoto similarity test and estimation methods for biological presence-absence data. *BMC Bioinformatics.* 2019;20(Suppl 15):644.
69. Fan B, Kong Q, Yuan X, Wang Z, Pan C. Learning weighted Hamming distance for binary descriptors. In: *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26–31, 2013.* IEEE; 2013. p. 2395–9.
70. Kaggle Datasets. <https://www.kaggle.com/datasets>. Accessed 24 Sep 2024.
71. Song C, Kakuta Y, Negoro K, Moroi R, Masamune A, Sasaki E, et al. Collection of patient-generated health data with a mobile application and transfer to hospital information system via QR codes. *Comput Methods Progr Biomed Update.* 2023;3:100099.
72. Wikipedia contributors. QR code—Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=QR_code&oldid=1188867736. 2023.
73. Genomes Project C, Auton A, Brooks LD, Durbin RM, Garrison EP, Kang HM, et al. A global reference for human genetic variation. *Nature.* 2015;526:68–74.
74. Sudlow C, Gallacher J, Allen N, Beral V, Burton P, Danesh J, et al. UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Med.* 2015;12:e1001779.
75. Slowikowski K. ggrepel: automatically position non-overlapping text labels with "ggplot2." 2023.
76. Pheno-Ranker Shared Files. <https://github.com/CNAG-Biomedical-Informatics/pheno-ranker/tree/main/share>. Accessed 24 Sep 2024.
77. Danis D, Barnshad MJ, Bridges Y, Cacheiro P, Carmody LC, Chong JX, et al. A corpus of GA4GH Phenopackets: case-level phenotyping for genomic diagnostics and discovery. *Human Genet Genomics Adv.* 2024;6(1):100371.
78. Gu Z, Hübschmann D. Make interactive complex heatmaps in R. *Bioinformatics.* 2022;38:1460–2.
79. Sievert C. Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC; 2020.
80. Luna A, Shah O, Sander C, Shannon P. cyjShiny: a cytoscape.js R Shiny widget for network visualization and analysis. *PLoS One.* 2023;18:e0285339.
81. Franz M, Lopes CT, Huck G, Dong Y, Sumer O, Bader GD. Cytoscape.js: a graph theory library for visualisation and analysis. *Bioinformatics.* 2016;32:309–11.
82. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol.* 1990;215:403–10.
83. Jinnin M. Mechanisms of skin fibrosis in systemic sclerosis. *J Dermatol.* 2010;37:11–25.
84. Kaul A, Gordon C, Crow MK, Tourna Z, Urowitz MB, van Vollenhoven R, et al. Systemic lupus erythematosus. *Nat Rev Dis Primers.* 2016;2:16039.
85. Kalra D, Beale T, Heard S. The open EHR foundation. *Stud Health Technol Inform.* 2005;115:153–73.
86. Rueda M, Leist IC, Gut IG. Convert-pheno: a software toolkit for the interconversion of standard data models for phenotypic data. *J Biomed Inform.* 2023;149:104558.
87. Putman TE, Schaper K, Matentzoglou N, Rubineti VP, Alquaddoomi FS, Cox C, et al. The Monarch 9 Initiative in 2024: an analytic platform integrating phenotypes, genes and diseases across species. *Nucleic Acids Res.* 2023;52:D938–49. <https://doi.org/10.1093/nar/gkad1082>.
88. Danis D, Jacobsen JOB, Wagner AH, Groza T, Beckwith MA, Rekerle L, et al. Phenopacket-tools: building and validating GA4GH phenopackets. *PLoS ONE.* 2023;18:e0285433.
89. Health Cards Framework: implementation guide and supporting material. 2023.
90. Laurie S, Pisco D, Matalonga L, Corvó A, Fernández-Callejo M, García-Linares C, et al. The RD-connect genome-phenome analysis platform: accelerating diagnosis, research, and gene discovery for rare diseases. *Hum Mutat.* 2022;43:717–33.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

4.2.2.1 Additional Pheno-Ranker-UI features only briefly mentioned in the publication

Extraction of the patients ids from the MDS plot

To make the multidimensional scaling results even more insightful I added extra interactive annotations to the rendered scatter plots. Now, when the user hovers over any point in the scatter plot the patient's id is shown. Additionally, the selection of the points with a lasso or a box select will render a textbox below the plot showcasing the corresponding patient IDs (Figure 4.17). The interactive extraction of patient IDs is especially useful when users want to leverage Pheno-Ranker's output for building analyses cohorts as described in Section 4.3.

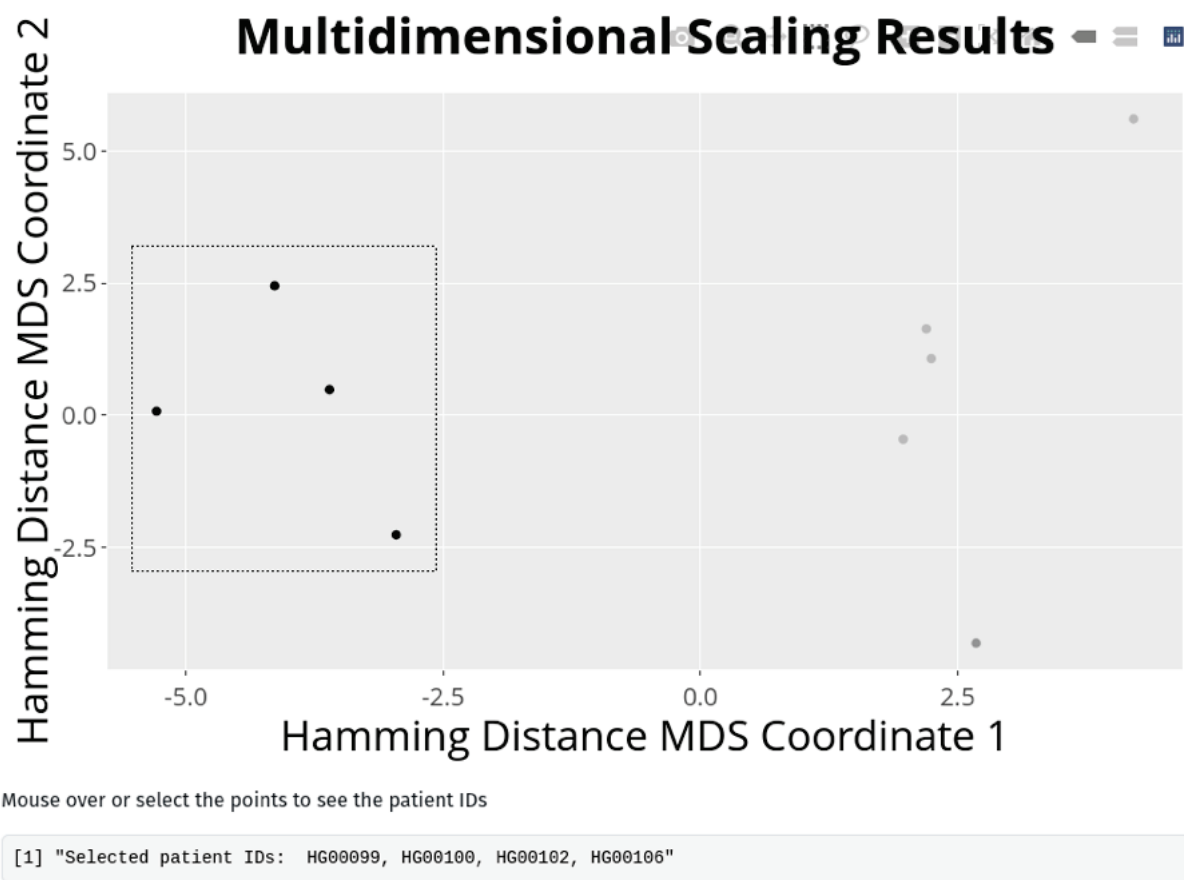


Figure 4.17: Screenshot showcasing the extraction of the patient IDs from the multidimensional scaling plot in the Pheno-Ranker-UI. Here the patients have been selected by a box select. The corresponding patient ids are shown in the textbox below the plot.

Cytoscape network visualisation

While the MDS plots and the heatmaps are very informative, sometimes a more interactive visualisation would be beneficial to examine the similarities/dissimilarities between individuals. A tool well suited for that matter is Cytoscape [125], as it is a widely used, open-source platform designed for interactive exploration of complex

networks. Thus, we decided to add the generation of Cytoscape ready JSON files as an additional Pheno-Ranker output. In the UI this is reflected by the addition of an additional tab to each of the ranking modes in which the user can explore the ranking results in an interactive network visualisation powered by Cytoscape.js [126]. Here the nodes represent the patients and the similarity scoring between them is indicated by the edge thickness and colouring. The latter could vary between three distinct colours coming from a preset colour palette. By default, red is used for low, green for medium and blue for high similarity. Additionally, the user can filter the nodes by the similarity score, change the layout of the network (grid, circle, etc.), move the nodes around and zoom in and out. Moreover, it is also possible to personalise the output of the network visualisation such as altering the node colouring (Figure 4.18). Thus, the user could easily use the network visualisations for presentations or publications.

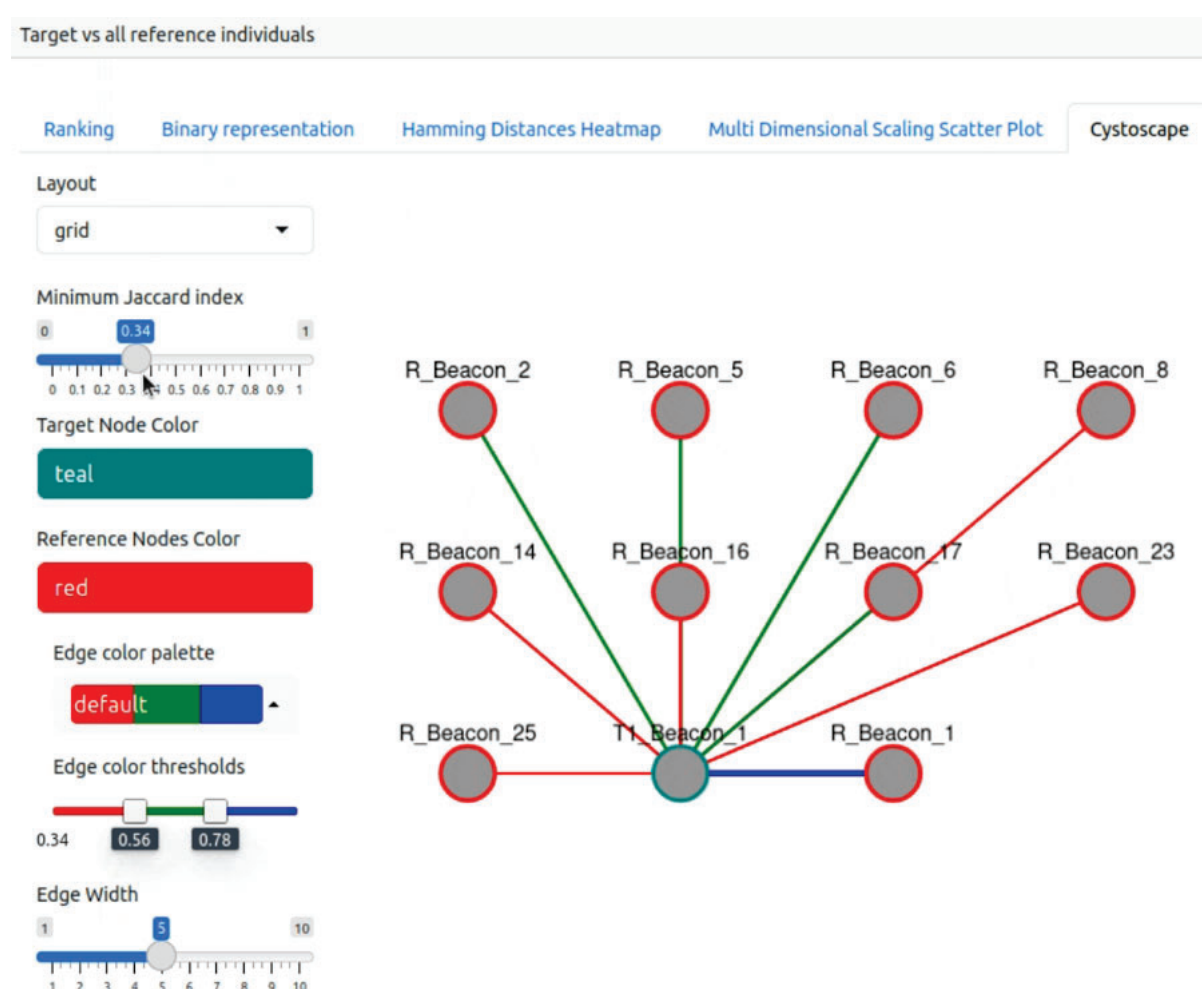


Figure 4.18: Screenshot of Pheno-Ranker-UI's Cytoscape driven network visualisation of a target patient against the patients in the reference cohort. The user can change the node colour of the target patient, as well as the node colour of the patients in the reference cohort. The edges could be customised by selecting one of the preset colour palettes and adjusting the ranges for the colours indicating the similarity. Finally, the default edge width can be set as well.

4.3 OmicsDM in concert with Convert-Pheno and Pheno-Ranker

Since the pheno-clinical data conversion with Convert-Pheno can be a complex and manual process, I decided against directly integrating it into OmicsDM. In fact, producing a good conversion result is an iterative process, where the user must repeatedly convert the data, check the conversion result and if necessary, adjust the conversion input files (in the case of REDCap data the mapping file) or change the conversion parameters before converting the data again. Thus, I created an extra user role in OmicsDM, the “Pheno-clinical data harmoniser”, who is responsible for the conversion of the pheno-clinical data. After the conversion runs, the user with the aforementioned role uploads the conversion results directly to the data warehouse. The generated PXF and BFF files are associated with the respective project and dataset where the original data was uploaded. From there these files can be accessed by the data owner to give feedback to the “Pheno-clinical data harmoniser” or by the data analyst for further analyses. Additionally, the data analyst can use the Pheno-Ranker tool to assess the similarity between patients in the dataset which can be beneficial information for the building of cohorts across studies and diseases. Figure 4.19 illustrates how Convert-Pheno and Pheno-Ranker are used in concert with OmicsDM.

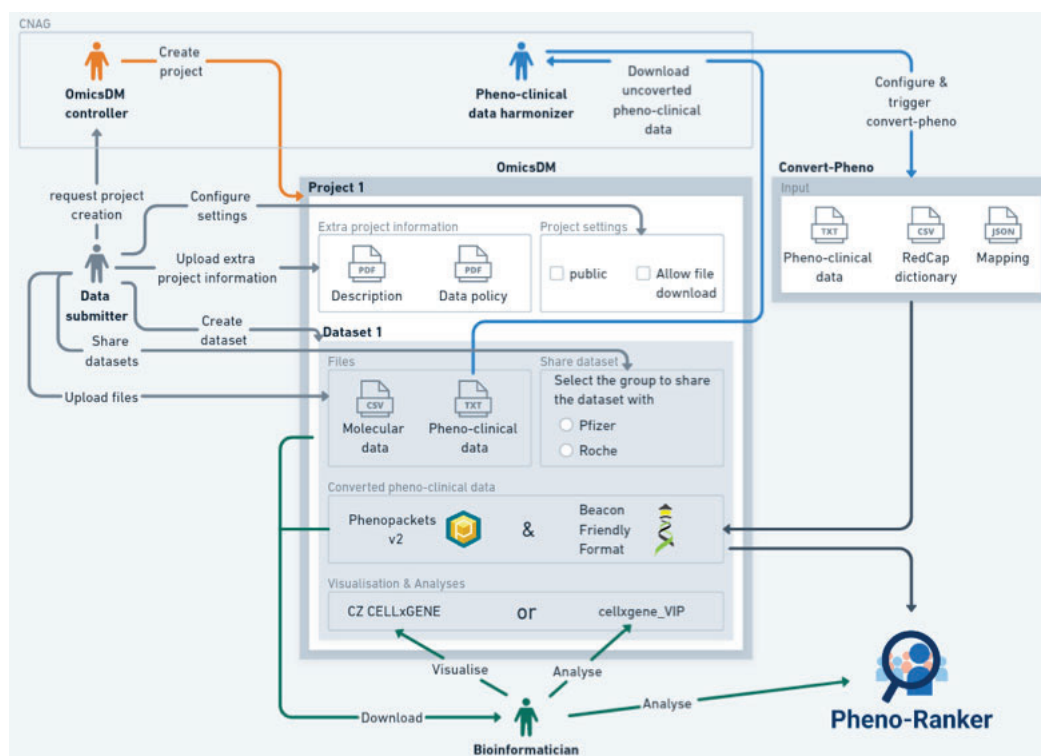


Figure 4.19: Diagram representing how Convert-Pheno & Pheno-Ranker is used in concert with OmicsDM.

Chapter 5: Discussion

5.1 OmicsDM

5.1.1 OmicsDM as a future proof data warehouse solution for the 3TR project

The 3TR project is a large-scale project in terms of the number of participating partners (69) and in regard to the expected data volume (data of 50,000 patients). The data could be of a multitude of different data types such as different formats of various omics data types (Requirement 1 - R1). It is also important to keep in mind that users have different levels of technical skills. The data warehouse solution should not only cater to users familiar with accessing APIs but also to users who are not familiar with the command line at all (R2). On top of that, at all times the data should be secure at all times and only accessible to authorised users. This requires a fine-grained layer for data access control, whereby the sharing of data should be as easy as possible while not compromising security (R3). Furthermore, the data warehouse should not be limited to solely to data management but also allows the users to visualise the data, especially from single cell experiments (R4), and run analyses on the data directly, preferably with the aid of a user-friendly web interface (R5). Table 5.1 shows how all these requirements are met by OmicsDM, including potential limitations.

Table 5.1: How OmicsDM meets the requirements of the 3TR project and potential limitations.

Requirement		Addressed by	Potential limitation
R1	Adaptable to different data types	S3 storage allows all kind of data types	no in-file query support (PL1)
R2	Cater to users with different levels of technical skills	Web interface allows upload and download of data	2FA on the API level is cumbersome (PL2)
R3	Easy/secure sharing of datasets	Dataset (collection of files) sharing on dataset level leveraging Keycloak groups	No fine-grained sharing on a per file basis (PL3)
R4	Single cell visualisations	Employing ShinyProxy for orchestrating CZ CELL×GENE Annotate instances	CZ CELL×GENE Annotate can be computational intensive for large datasets (PL4)
R5	Run analyses	Run Snakemake/Nextflow workflows directly from the UI	Due to security reasons only by the OmicsDM administrators vetted pipelines are allowed to run (PL5)

In the following each of the potential limitations (PL1-5) listed in Table 5.1 will be discussed in detail, including possible mitigation strategies.

PL1: S3 not allowing to search within the uploaded files

S3 only allows for object-level operations such as upload, download, and delete objects. Consequently, it is not possible to run in-file queries such as SQL queries on the data stored in S3. If we would want to allow the users to return files based on the content in the files we would have to index them. This could be done by using a tool such as Apache Lucene⁷ or Elasticsearch⁸. By indexing, the content of the files could be extracted and stored in a way that allows fast search queries. However, indexing all the uploaded data would require substantial computing resources with no apparent direct benefit for the users, as for instance omics data such as RNA-Seq data is usually stored in the form of matrices without any meta data. Instead, the metadata is stored separately in pheno-clinical data files. Unfortunately, these often come in the form of tabular data files, without an accompanying schema file defining the columns, like the expected value format (numerical/string) per column. This makes it hard to index these files to enable efficient in-file queries. However, with Convert-Pheno we have developed a tool to convert all the pheno-clinical data to the GA4GH standard exchange formats PXF and BFF (introduced in Section 3.1.2). Both PXF and BFF files can come in JSON format. Thus, they could be either directly indexed and then queried with e.g, Elasticsearch or uploaded to a database such as MongoDB (mongodb.com) which allows for querying hierarchical data structures. For the latter approach there is already a reference implementation (github.com/EGA-archive/beacon2-ri-api) available which could be taken as inspiration.

PL2: Cumbersome 2FA on the API level

OmicsDM is built with a stronger focus on the web interface than on the API. If someone would like to use the API directly, they would first have to login into OmicsDM's 2FA protected web UI and then extract the authentication token from the browser's local storage. The token could then be used to query the API. While this can be seen as not the most user-friendly way to use the API, it is a trade-off that is justified

⁷ lucene.apache.org

⁸ elastic.co/elasticsearch

as sensitive patient information is stored in OmicsDM. Consequently, preventing unauthorised access is more important than the ease of use of the API.

PL3: Lack of fine-grained sharing on a per file basis

Not being able to share any file individually can be seen as a limitation. However, this was a deliberate decision since allowing to share files on a per file basis could easily lead to a situation where for instance user A shares an omics data file with user B but not its corresponding pheno-clinical data file. Since the omics data is usually meaningless without the corresponding pheno-clinical data user B would have to ask user A to share the pheno-clinical data file as well. To avoid this type of situations, we only allow sharing on the *dataset* level. Note that in the context of OmicsDM the term *dataset* refers to a collection of files. One of these files should contain all the pheno-clinical data corresponding to the omics files within each *dataset*. This link of the pheno-clinical data to the molecular information also helps to prevent potential unnecessary molecular data downloads since the users could first download the pheno-clinical data in the *dataset* overview, check if the patients fit to their inclusion criteria, before downloading the actual omics data. However, it must be noted that we at no point enforce the user to upload pheno-clinical data since, but only encourage the users to do so by showing the link to the respective file very prominently in the datasets overview table.

PL4: Running CZ CELL×GENE Annotate for large datasets is computationally intensive

The observation that running CZ CELL×GENE Annotate for large datasets turns out to be computationally intensive should neither be understood as a limitation of OmicsDM's infrastructure nor as a criticism of the CZ CELL×GENE Annotate software per se. Instead it is to be seen as an acknowledgment of the current reality that nowadays it is not uncommon to have single cell experiments with millions of cells [127]. Easy to imagine, visualising the resulting AnnData objects (introduced in Section 3.1.1) can be a RAM heavy task. To put this into perspective in Table 5.2 the mean memory usage of CZ CELL×GENE Annotate extracted from a benchmark conducted by Conor Klamann *et al.* 2024 [128] is shown.

As apparent from Table 5.2 even only medium sized single cell datasets (around 50,000 cells) would require over 10 GB of RAM. Since for the OmicsDM playground only 64 GB RAM were dedicated (Section 3.4.7) this means a maximum of five users are able to use the playground in parallel, under the assumption each is exploring a medium sized dataset. To mitigate this limitation ShinyProxy constantly checks if there are CZ CELL×GENE Annotate instances which are no longer in use and stops them. This approach helps to free up unused resources as quickly as possible. Thus, minimising the potential waiting time for other users to explore their datasets.

Table 5.2: CZ CELL×GENE Annotate mean RAM usage across different cell counts. Adapted from [127]

Number of cells	Mean RAM usage (in GB)	Dataset
18,859	3.5	Tabula Muris [128], [129] (subset)
41,668	8.0	Tabula Muris [129]
483,152	222.44	Tabula Sapiens [130]

PL5: Administrators need to vet new pipelines before they can be run on OmicsDM

Requiring the OmicsDM administrator to vet any pipeline to be run on OmicsDM can be seen by the users as an annoying “bureaucratic” hurdle. However, allowing the users to run arbitrary code on the OmicsDM servers would not only be a security risk but could also cause unforeseen issues in terms of the overall stability of the system. Since the pipelines run inside docker containers it could be argued that this extra administrative step is not necessary. However, since the docker containers running the analysis pipelines are run in the same network as the S3 storage docker container we decided to rather be too cautious than risking a data breach. In addition this thorough testing also helps to catch any unwanted side effect of a pipeline, such as the unexpected rapid filling up of the provided disk space. Thus, vetting the pipelines is not only a security measure, but also guarantees the overall stability of the system.

In conclusion none of the above discussed potential limitations hinders OmicsDM’s applicability to the 3TR project. In fact, the developed data warehouse solution has been well received by its users.

5.1.1.1 Decision for a centralised data warehouse architecture despite the trend towards federated solutions

There seems to be a trend towards federated data warehouse solutions in the biomedical field. Consequently, at the onset of the 3TR project the idea of setting up a federated data warehouse solution was discussed. While the idea of a federated data warehouse solution is appealing since it allows the users to keep their data on their own servers this would have meant a substantial increase in complexity. This holds especially true in the light of the 69 partners involved in the 3TR project. In a federated environment this would have meant that for each of these partners a middleware would have to be set up. This could have been a challenging undertaking for (a) partners with limited technical personnel available and (b) pharmaceutical companies which are usually very restrictive in terms of what can be installed on their systems. Regarding the latter, one could argue that the potential additional legal difficulties entailing it could have been worth the effort, since this middleware could not only be used for data access but also for running non-disclosive analytics. However, the latter was not required in the 3TR context because its consortium agreement included the necessary legal framework in terms of GDPR compliance. Thus, an extra level of privacy-preserving measures was not required.

The most important aspect which led to the decision to have a centralised data warehouse solution was the envisioned cross-disease analyses. To this end, not only the omics data has to be compatible with each other but also the pheno-clinical data would need to be harmonised. To facilitate the latter in principle Convert-Pheno [72] could also be leveraged in a federated environment to convert the pheno-clinical data to a common format such as a GA4GH standard (Section 1.3.4.1). However, in this setting each partner would be responsible for converting their data to a common format on their own. Running Convert-Pheno should not be a problem for any of the partners, since it is distributed in the form of a docker image (hub.docker.com/r/manuelrueda/convert-pheno). Nevertheless, depending on the input data format Convert-Pheno requires manually created auxiliary files. In the 3TR context this is far from trivial since the data is collected from multiple REDCap instances (Section 1.3.3.1) with differing variable labels. Consequently, before the actual conversion a manual harmonisation of the variable names across multiple

studies would be necessary. As easily imaginable it simplifies the process when all the pheno-clinical data is stored in one place.

Thus, a centralised data warehouse architecture was the best choice to address 3TR's harmonisation needs and the scale of the whole project. Note that regarding the latter we tried to mitigate future scalability difficulties by incorporating S3 as the storage backend. This setting makes it possible to scale by simply adding new storage nodes to the S3 cluster. Finally, having all the data stored at one location also facilitates data reuse beyond the project's lifetime, as this approach requires only one web server, one database and one S3 storage cluster to be maintained.

5.1.2 OmicsDM vs existing data warehouse solutions

OmicsDM vs existing data warehouse solutions for omics data

Before risking reinventing the wheel, multiple already available open-source self-hostable data warehouse solutions available such as tranSMART [77], Opal [78], MOLGENIS EMX2 [89], or SODAR [80] and their applicability to the 3TR use case were assessed. I was unable to test a fifth option, called Terra (app.terra.bio), because there does not seem to be a self-deployment manual available⁹.

The main shortcoming in common between the evaluated data warehouse solutions was that it is hard to prevent the emergence of a complex nested data (folder) structure. This is problematic because it not only makes it hard to find the data but makes it especially difficult to share data with other users. Therefore, in our data warehouse we decided to have the lowest level of nesting possible. Thus, in OmicsDM there are only three levels of nesting: project, dataset, and files. Another critical issue I encountered when evaluating tranSMART and Opal was that these tools were struggling with the upload of large datasets (greater than 10 GB). For clarification it is not that the upload is failing, but rather renders the web UI unresponsive. Thus the user no longer receives any feedback if the upload is progressing or not. In contrast, MOLGENIS-EMX2 updates the user about the progress of the upload. However, the upload of large datasets (greater than 10 GB) fails silently. SODAR has no method to upload files directly from the web interface but only via the command line, by mounting a WebDAV network drive or by using external software such as Cyberduck (cyberduck.io). While this might work well for the upload of large datasets it might be

⁹ support.terra.bio/hc/en-us/community/posts/360042857092-Where-is-the-Terra-source-code

difficult for not so tech-savvy users. OmicsDM in contrast comes with a file upload mechanism which enables the user to upload large files directly from the web interface, while providing feedback about the progress of the upload. Moreover, when a file with the same name is uploaded to OmicsDM, the existing file is not overwritten but a new version of the file is created. This is a feature which is not available in any of the other tested tools.

On top, OmicsDM is similar to other solutions such as SODAR not only a data management platform but also also has analytical and visualisation capabilities inbuilt. However, none of the existing tools allows to programmatically add custom annotations, such as results from a functional enrichment analyses (as shown in Figure 4.13). Thus, OmicsDM is to our knowledge the only self-hostable data warehouse solution which covers the whole chain of an analytical workflow in one platform directly from an intuitive UI: (i) upload a large dataset, (ii) run analyses, (iii) integrate the analysis results into the input dataset, (iv) visualise it and ultimately (v) share it with other collaborators.

OmicsDM vs existing data warehouse solutions tailored to single-cell data

Before integrating the single cell data visualisation module into OmicsDM I evaluated multiple existing open-source, self-hostable data warehouse solutions tailored to single-cell data such as CZ CELL×GENE Discover (cellxgene.cziscience.com), Single Cell Portal (singlecell.broadinstitute.org) and CellDepot (celldepot.bxgenomics.com) regarding their applicability to the 3TR use case. CellDepot, has been assessed solely based on their playground. Despite the availability of a manual, I was unable to set up a local CellDepot instance. Same holds true for the Single Cell Portal (github.com/broadinstitute/single_cell_portal_core). Here the blocking issue was the necessity to have a Google Cloud platform account to set up a local instance. Thus, CZ CELL×GENE Discover was the only solution we managed to run locally. However, its software stack is quite complex. Consequently, the developers of CZ CELL×GENE Discover do not advise to self-host it but instead encourage interested parties to contribute their data to CZ CELL×GENE Discover's corpus [131].

We believe that in contrast to the other data warehouse solutions OmicsDM is easier to set up. In fact, the OmicsDM server and client, accompanied by its auxiliary services,

such as Keycloak and ShinyProxy can be launched with Docker compose. Additionally, OmicsDM's code repository¹⁰ includes configuration templates.

5.1.3 Applicability of OmicsDM to other consortia

The version of OmicsDM presented in this thesis is a proof of concept strictly tailored to the needs of the 3TR projects. However, the platform has the potential to be extended to other projects and consortia. To achieve this, the underlying code base would need to be refactored and OmicsDM's modules decoupled from each other more strictly. This decoupling could give the OmicsDM's admin the option to (a) disable or (b) not to deploy a for that particular project unneeded module, as for instance the single-cell visualisation module.

Another aspect to be considered for the potential reuse of OmicsDM in another consortium is the scalability of the system. Not only in terms of the data needed to be stored, but for instance also the number of expected simultaneous single-cell visualisation sessions. While the data storage can be easily scaled by adding new storage nodes to the S3 cluster (horizontal scaling), the server hosting ShinyProxy, the tool orchestrating the CZ CELL×GENE Annotate instances, can only be scaled vertically. Thus, either forcing the OmicsDM's admins to replace the virtual machine that is running ShinyProxy with one with more RAM or even the whole physical server. While this could be overcome by linking ShinyProxy to Kubernetes, an open-source container orchestration system, the combination of two container orchestration systems can be considered unnecessary overhead. In fact, in this context there is already some exploratory work underway in our laboratory by Etienne Jodry, to whose endeavours I am contributing conceptually. The proof of concept of orchestrating CZ CELL×GENE Annotate instances directly with Kubernetes can be found here: github.com/bag-cnag/cxg_on_k8. However, running a Kubernetes cluster requires quite some technical skills. Thus, the burden of maintaining a Kubernetes solution is only justified if the consortium has a large number of simultaneous users visualising large single-cell datasets. However, at least for now this can be considered an edge case. I am confident that for other cases the current ShinyProxy-based architecture of OmicsDM would be sufficient.

¹⁰ github.com/CNAG-Biomedical-Informatics/omicsdm

5.2 Standalone tools for pheno-clinical data harmonisation & exploration

5.2.1 Convert-Pheno

The tool Convert-Pheno is our contribution to a future of pheno-clinical data management in which no longer standard after standard is developed ([69], [132], [133], [134], [135]) in the seemingly never ending hunt for a gold standard for every potential use case. Instead with Convert-Pheno we aimed to come up with an extensible translator of different pheno-clinical data standards. The at the time of the compilation of this thesis (June 2025) supported inter-conversion are shown in Figure 5.1.

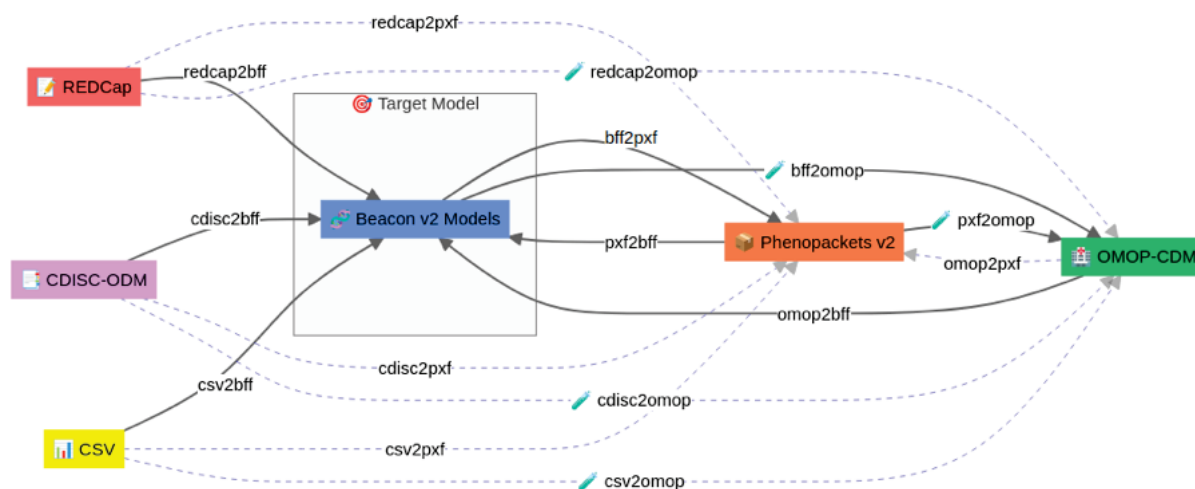


Figure 5.1: Convert-Pheno supported pheno-clinical data inter-conversions as of June 2025. Coloured boxes show the possible input, respectively output formats. It has to be noted that all conversions go via the intermediate target model, namely Beacon v2 Models. Not shown for the sake of readability but instead indicated by dotted lines. A test tube indicates that this conversion was at the time of this thesis compilation (June 2025) still experimental. Adapted from cnag-biomedical-informatics.github.io/convert-pheno/supported-formats.

While Convert-Pheno dramatically facilitates the conversion of pheno-clinical data from one format to another, it is not yet a fully automated process. Depending on the input format, the user has to provide additional files. For example, the conversion from the REDCap exports and CDISC-ODM requires the user to provide two additional files, namely a dictionary and a mapping file. Taking the advances in the field of machine learning (artificial intelligence) into account, in the future these files could likely be generated automatically. Be it with the help of tools powered by rule-based algorithms or supervised learning such as spaCy [136], or by generative artificial intelligence. Thus, minimising the effort required by the user to convert their pheno-clinical data. In conclusion, Convert-Pheno serves as potential stepping stone for fully automated data harmonisation.

5.2.2 Pheno-Ranker

Pheno-Ranker's power lies in its ability to compare the patient data across all variables in a pheno-clinical data file. Thus, potentially revealing systematic differences in the data. However, from the output of Pheno-Ranker's CLI on its own the differences between the individuals might not be immediately visible. This holds especially true in case of the cohort mode, *i.e.*, all patients compared to each other, since its output is in form of a matrix. Thus, users either need to have programming skills, such as being able to write a script to visualise the output or use the web interface to benefit from Pheno-Ranker's full potential.

Nevertheless, Pheno-Ranker's output, such as the generated plots in its UI, should be interpreted with caution. Since the tool highlights (dis)similiarties between individuals but does not automatically explain which variables are (mis)matching. Therefore, a domain expert would still be needed to interpret the results. Otherwise, users might falsely attribute the observed clustering to a clinical/biological root. To elucidate the reason behind the potential unexpected clusters of similar individuals users could take advantage of Pheno-Ranker's ability to include only selected variables in the comparison. The UI can be especially helpful in this regard since it allows the user to easily select variables of interest via drag and drop. In concert with the ability to quickly switch between past comparisons runs, using the history sidebar, it has the potential to give the user an idea which phenotypic information is driving the clustering. Note that the drag and drop functionality was developed with the GA4GH standards in mind, which hierarchical nature allows to easily restrict the comparison to a certain category of the patient such as diseases. However, in some cases it might be necessary to create more complex plots to unravel the potential reason for unexpected differences between different cohorts. An example for such a deep dive into Pheno-Ranker's output using custom R scripts is shown in the tool accompanying publication [120], namely the reanalysis of the PRECISESADS dataset. In the section "Use case—PRECISESADS dataset" one of the generated plots (Fig. 5B) shows patients from recruitment centre "A" being unexpectedly dissimilar to the patients of other centres. Further investigation uncovered that the phenol-clinical data of centre "A" had the highest proportion of missing values (not shown). This example showcased Pheno-Ranker's potential to assess the quality of pheno-clinical data.

Chapter 6: Conclusions

6.1 Overall

Throughout my thesis I have built a web-based data platform, called OmicsDM, that facilitates secure upload, storage, sharing, visualisation, and analyses of multi-omics data of the 3TR project.

To support the cross-disease analyses envisioned by the 3TR partners, I also contributed to the development of two tools that facilitate the harmonisation and cleaning of pheno-clinical data. The first tool, Convert-Pheno, is a command line tool that converts pheno-clinical data from one format to another. The second tool, Pheno-Ranker, is for the comparison of pheno-clinical data on an individual level. For each, my main contribution was the development of a web user interface and the set up of a public playground (<https://convert-pheno.cnag.cat>; <https://pheno-ranker.cnag.eu>), secured by Life Science Login. The latter enables the user to login for instance via ORCID. Thus, the capabilities of both toolkits cannot only be explored using a shared account with test data but also with private data in a secured environment.

6.2 Development of OmicsDM, an integrative data management, analysis and visualisation platform for multi-omics data

OmicsDM's hallmarks are:

- ◆ A 3-step data submission process:
 - ▶ Project selection
 - ▶ Dataset creation (set metadata and upload pheno-clinical data)
 - ▶ File submission (including large multi file upload from its web UI)
- ◆ Bulk data sharing with all users or specific user groups
- ◆ Automatic versioning of uploaded files
- ◆ Configuration and execution of prebuilt Snakemake or Nextflow pipelines
(Example use case: RNA-Seq data analysis)
- ◆ Single cell data visualisation employing CZ CELL×GENE

6.3 Standalone tools for pheno-clinical data exploration

6.3.1 Development of a web UI for Convert-Pheno, a software toolkit for the interconversion of standard data models for phenotypic data

The UI enables users to:

- ◆ Convert the following input data:
REDCap CSV exports, PXF (Phenopackets v2), BFF (Beacon Models v2), OMOP-CDM, CDISC-ODM to the GA4GH standards PXF and BFF
- ◆ Browse the conversion outputs in a HTML table
- ◆ Filter the conversion output table with a drag and drop query builder allowing to set multiple inclusion and/or exclusion criteria based on the top-level elements

6.3.1.1 Development of an R shiny UI for Pheno-Ranker, a toolkit for comparison of phenotypic data stored in GA4GH standards and beyond

Pheno-Ranker-UI's hallmarks are:

- ◆ Low entry barrier thanks to utilities for simulating input data, getting example data from Phenopacket Store (Monarch Initiative), querying data from Beacon APIs or to convert CSV data into a Pheno-Ranker ready format
- ◆ Two different modes:
Cohort, to perform inter- or intra cohort comparisons using Hamming distances. Patient, to determine which individuals in the cohort(s) are the closest to a selected patient by ranking them using the (dis)similarity metrics: Hamming distance and Jaccard Index
- ◆ Visualisation of the Hamming distances in the form of interactive heatmaps and scatter plots
- ◆ Sortable HTML tables in the patient mode to quickly grasp which individual is the most (dis)similar to a selected patient
- ◆ A history sidebar to access the results of past Pheno-Ranker runs, respectively the output of its utilities

Chapter 7: References

- [1] G. Barturen *et al.*, “Integrative Analysis Reveals a Molecular Stratification of Systemic Autoimmune Diseases,” *Arthritis & Rheumatology*, vol. 73, no. 6, pp. 1073–1085, 2021, doi: 10.1002/art.41610.
- [2] C. Walton *et al.*, “Rising prevalence of multiple sclerosis worldwide: Insights from the Atlas of MS, third edition,” *Mult Scler*, vol. 26, no. 14, pp. 1816–1821, Dec. 2020, doi: 10.1177/1352458520970841.
- [3] J. Tian, D. Zhang, X. Yao, Y. Huang, and Q. Lu, “Global epidemiology of systemic lupus erythematosus: a comprehensive systematic analysis and modelling study,” *Annals of the Rheumatic Diseases*, vol. 82, no. 3, pp. 351–356, Mar. 2023, doi: 10.1136/ard-2022-223035.
- [4] R. J. Black *et al.*, “Global, regional, and national burden of rheumatoid arthritis, 1990–2020, and projections to 2050: a systematic analysis of the Global Burden of Disease Study 2021,” *The Lancet Rheumatology*, vol. 5, no. 10, pp. e594–e610, Oct. 2023, doi: 10.1016/S2665-9913(23)00211-4.
- [5] C. Le Berre, S. Honap, and L. Peyrin-Biroulet, “Ulcerative colitis,” *The Lancet*, vol. 402, no. 10401, pp. 571–584, Aug. 2023, doi: 10.1016/S0140-6736(23)00966-2.
- [6] R. Wang, Z. Li, S. Liu, and D. Zhang, “Global, regional and national burden of inflammatory bowel disease in 204 countries and territories from 1990 to 2019: a systematic analysis based on the Global Burden of Disease Study 2019,” *BMJ Open*, vol. 13, no. 3, p. e065186, Mar. 2023, doi: 10.1136/bmjopen-2022-065186.
- [7] S. Momtazmanesh *et al.*, “Global burden of chronic respiratory diseases and risk factors, 1990–2019: an update from the Global Burden of Disease Study 2019,” *eClinicalMedicine*, vol. 59, May 2023, doi: 10.1016/j.eclinm.2023.101936.
- [8] E. Boers *et al.*, “Global Burden of Chronic Obstructive Pulmonary Disease Through 2050,” *JAMA Network Open*, vol. 6, no. 12, p. e2346598, Dec. 2023, doi: 10.1001/jamanetworkopen.2023.46598.
- [9] S. M. Hayter and M. C. Cook, “Updated assessment of the prevalence, spectrum and case definition of autoimmune disease,” *Autoimmunity Reviews*, vol. 11, no. 10, pp. 754–765, Aug. 2012, doi: 10.1016/j.autrev.2012.02.001.
- [10] L. Fugger, L. T. Jensen, and J. Rossjohn, “Challenges, Progress, and Prospects of Developing Therapies to Treat Autoimmune Diseases,” *Cell*, vol. 181, no. 1, pp. 63–80, Apr. 2020, doi: 10.1016/j.cell.2020.03.007.
- [11] M. Colquhoun, M. Gulati, Z. Farah, and M. Mouyis, “Clinical features of rheumatoid arthritis,” *Medicine*, vol. 50, no. 3, pp. 138–142, Mar. 2022, doi: 10.1016/j.mpmed.2021.12.002.
- [12] A. Kaul *et al.*, “Systemic lupus erythematosus,” *Nat Rev Dis Primers*, vol. 2, no. 1, pp. 1–21, Jun. 2016, doi: 10.1038/nrdp.2016.39.
- [13] A. N. Bismillah and S. D. P. Fielden, “Holding your nerve in academia whilst managing multiple sclerosis,” *Nat Rev Chem*, vol. 8, no. 11, pp. 793–794, Nov. 2024, doi: 10.1038/s41570-024-00661-8.
- [14] S. Jarmakiewicz-Czaja, M. Zielińska, A. Sokal, and R. Filip, “Genetic and Epigenetic Etiology of Inflammatory Bowel Disease: An Update,” *Genes (Basel)*, vol. 13, no. 12, p. 2388, Dec. 2022, doi: 10.3390/genes13122388.
- [15] D. Piovani, S. Danese, L. Peyrin-Biroulet, G. K. Nikolopoulos, T. Lytras, and S. Bonovas, “Environmental Risk Factors for Inflammatory Bowel Diseases: An Umbrella Review of Meta-analyses,” *Gastroenterology*, vol. 157, no. 3, pp. 647–659.e4, Sep. 2019, doi: 10.1053/j.gastro.2019.04.016.
- [16] F. A. R. Silva, B. L. Rodrigues, M. de L. S. Ayrizono, and R. F. Leal, “The Immunological Basis of Inflammatory Bowel Disease,” *Gastroenterology Research and Practice*, vol. 2016, no. 1, p. 2097274, 2016, doi: 10.1155/2016/2097274.
- [17] Z. Cai, S. Wang, and J. Li, “Treatment of Inflammatory Bowel Disease: A

- Comprehensive Review,” *Front Med (Lausanne)*, vol. 8, p. 765474, Dec. 2021, doi: 10.3389/fmed.2021.765474.
- [18] S. P. Jadhav *et al.*, “Introduction to Lung Diseases,” in *Targeting Cellular Signalling Pathways in Lung Diseases*, K. Dua, R. Löbenberg, Å. C. Malheiros Luzo, S. Shukla, and S. Satija, Eds., Singapore: Springer, 2021, pp. 1–25. doi: 10.1007/978-981-33-6827-9_1.
 - [19] S. M. Levine and D. D. Marciniuk, “Global Impact of Respiratory Disease,” *Chest*, vol. 161, no. 5, pp. 1153–1154, May 2022, doi: 10.1016/j.chest.2022.01.014.
 - [20] G. A. Jacobson *et al.*, “Long-Acting β 2-Agonists in Asthma: Enantioselective Safety Studies are Needed,” *Drug Saf*, vol. 41, no. 5, pp. 441–449, May 2018, doi: 10.1007/s40264-017-0631-1.
 - [21] M. Cazzola, C. P. Page, P. Rogliani, and M. G. Matera, “ β 2-Agonist Therapy in Lung Disease,” *Am J Respir Crit Care Med*, vol. 187, no. 7, pp. 690–696, Apr. 2013, doi: 10.1164/rccm.201209-1739PP.
 - [22] R. Buhl *et al.*, “Effective Management of Severe Asthma with Biologic Medications in Adult Patients: A Literature Review and International Expert Opinion,” *The Journal of Allergy and Clinical Immunology: In Practice*, vol. 10, no. 2, pp. 422–432, Feb. 2022, doi: 10.1016/j.jaip.2021.10.059.
 - [23] A. L. Kersul and B. G. Cosio, “Biologics in COPD,” *Open Respir Arch*, vol. 6, no. 2, p. 100306, Feb. 2024, doi: 10.1016/j.opresp.2024.100306.
 - [24] S. P. Bhatt *et al.*, “Dupilumab for COPD with Type 2 Inflammation Indicated by Eosinophil Counts,” *New England Journal of Medicine*, vol. 389, no. 3, pp. 205–214, Jul. 2023, doi: 10.1056/NEJMoa2303951.
 - [25] V. L. Kronzer, C. S. Crowson, J. A. Sparks, E. Myasoedova, and I. I. I. John Davis, “Family history of rheumatologic, autoimmune, and non-autoimmune diseases and risk of rheumatoid arthritis,” *Arthritis care & research*, vol. 73, no. 2, p. 180, Feb. 2021, doi: 10.1002/acr.24115.
 - [26] J. Cárdenas-Roldán, A. Rojas-Villarraga, and J.-M. Anaya, “How do autoimmune diseases cluster in families? A systematic review and meta-analysis,” *BMC Medicine*, vol. 11, no. 1, p. 73, Mar. 2013, doi: 10.1186/1741-7015-11-73.
 - [27] M. P. C. Santos, C. Gomes, and J. Torres, “Familial and ethnic risk in inflammatory bowel disease,” *Annals of Gastroenterology*, vol. 31, no. 1, p. 14, Oct. 2017, doi: 10.20524/aog.2017.0208.
 - [28] M. G. Sikjær, A. Klitgaard, O. Hilberg, and A. Løkke, “Parental COPD as a Risk Factor for the Development of COPD and Disease Severity in Offspring: A Systematic Scoping Review,” *International Journal of Chronic Obstructive Pulmonary Disease*, vol. 17, p. 1323, Jun. 2022, doi: 10.2147/COPD.S364899.
 - [29] S. F. Thomsen, “Genetics of asthma: an introduction for the clinician,” *European Clinical Respiratory Journal*, vol. 2, p. 10.3402/ecrj.v2.24643, Jan. 2015, doi: 10.3402/ecrj.v2.24643.
 - [30] A. Balcerac and C. Louapre, “Genetics and familial distribution of multiple sclerosis: A review,” *Revue Neurologique*, vol. 178, no. 6, pp. 512–520, Jun. 2022, doi: 10.1016/j.neurol.2021.11.009.
 - [31] R. Fernandez-Ruiz and T. B. Niewold, “Type I Interferons in Autoimmunity,” *Journal of Investigative Dermatology*, vol. 142, no. 3, Part B, pp. 793–803, Mar. 2022, doi: 10.1016/j.jid.2021.11.031.
 - [32] J.-Y. Li *et al.*, “IRF/Type I IFN signaling serves as a valuable therapeutic target in the pathogenesis of inflammatory bowel disease,” *International Immunopharmacology*, vol. 92, p. 107350, Mar. 2021, doi: 10.1016/j.intimp.2020.107350.
 - [33] H. E. Rich, D. Antos, N. R. Melton, J. F. Alcorn, and M. L. Manni, “Insights Into Type I and III Interferons in Asthma and Exacerbations,” *Front Immunol*, vol. 11, p. 574027, Sep. 2020, doi: 10.3389/fimmu.2020.574027.
 - [34] F. Atzeni, M. C. Gerardi, G. Barilaro, I. F. Masala, M. Benucci, and P. Sarzi-Puttini, “Interstitial lung disease in systemic autoimmune rheumatic diseases: a

- comprehensive review,” *Expert Review of Clinical Immunology*, vol. 14, no. 1, pp. 69–82, Jan. 2018, doi: 10.1080/1744666X.2018.1411190.
- [35] Y. Liu *et al.*, “Identification of common susceptibility genes and drug target genes in multiple sclerosis, systemic lupus erythematosus, and rheumatoid arthritis and its value to guide clinical treatment,” *Multiple Sclerosis and Related Disorders*, vol. 58, p. 103504, Feb. 2022, doi: 10.1016/j.msard.2022.103504.
- [36] Y.-Y. Zhao and R.-C. Lin, “UPLC–MSE application in disease biomarker discovery: The discoveries in proteomics to metabolomics,” *Chemico-Biological Interactions*, vol. 215, pp. 7–16, May 2014, doi: 10.1016/j.cbi.2014.02.014.
- [37] I. Parodis, A. E. Voskuyl, M. E. Alarcón-Riquelme, and L. Beretta, “PO.4.92 Taxonomy, treatment, targets and remission in systemic lupus erythematosus: the 3TR-SLE study protocol,” *Lupus Science & Medicine*, vol. 9, no. Suppl 2, Oct. 2022, doi: 10.1136/lupus-2022-elm2022.118.
- [38] N. Mishra *et al.*, “Longitudinal multi-omics analysis identifies early blood-based predictors of anti-TNF therapy response in inflammatory bowel disease,” *Genome Medicine*, vol. 14, p. 110, Sep. 2022, doi: 10.1186/s13073-022-01112-z.
- [39] L. M. Steinheuer, S. Canzler, and J. Hackermüller, “Benchmarking scRNA-seq imputation tools with respect to network inference highlights deficits in performance at high levels of sparsity,” Apr. 02, 2021, *bioRxiv*. doi: 10.1101/2021.04.02.438193.
- [40] M. N. Bernstein, “Computational Methods for Transcriptome-based Cellular Phenotyping,” University of Wisconsin-Madison, 2019. Accessed: Oct. 23, 2024. [Online]. Available: <https://www.proquest.com/openview/af4f51ec373a0b13438c59e7731adeed/1?pq-origsite=gscholar&cbl=18750&diss=y>
- [41] L. Zeng *et al.*, “Research progress of single-cell transcriptome sequencing in autoimmune diseases and autoinflammatory disease: A review,” *Journal of Autoimmunity*, vol. 133, p. 102919, Dec. 2022, doi: 10.1016/j.jaut.2022.102919.
- [42] Z. Xie *et al.*, “A combined analysis of bulk RNA-seq and scRNA-seq was performed to investigate the molecular mechanisms associated with the occurrence of myocardial infarction,” *BMC Genomics*, vol. 25, no. 1, p. 921, Oct. 2024, doi: 10.1186/s12864-024-10813-1.
- [43] Z. Ruohan *et al.*, “Advanced analysis and applications of single-cell transcriptome sequencing,” *All Life*, vol. 16, no. 1, p. 2199140, Dec. 2023, doi: 10.1080/26895293.2023.2199140.
- [44] F. Tang *et al.*, “mRNA-Seq whole-transcriptome analysis of a single cell,” *Nat Methods*, vol. 6, no. 5, pp. 377–382, May 2009, doi: 10.1038/nmeth.1315.
- [45] A. G. Rivenbark, S. M. O’Connor, and W. B. Coleman, “Molecular and Cellular Heterogeneity in Breast Cancer: Challenges for Personalized Medicine,” *The American Journal of Pathology*, vol. 183, no. 4, pp. 1113–1124, Oct. 2013, doi: 10.1016/j.ajpath.2013.08.002.
- [46] W. Jin *et al.*, “Single-cell RNA-Seq reveals transcriptional heterogeneity and immune subtypes associated with disease activity in human myasthenia gravis,” *Cell Discov*, vol. 7, no. 1, pp. 1–18, Sep. 2021, doi: 10.1038/s41421-021-00314-w.
- [47] Z. He *et al.*, “Single-cell transcriptomics analysis of cellular heterogeneity and immune mechanisms in neurodegenerative diseases,” *European Journal of Neuroscience*, vol. 59, no. 3, pp. 333–357, 2024, doi: 10.1111/ejn.16242.
- [48] P. A. Harris, R. Taylor, R. Thielke, J. Payne, N. Gonzalez, and J. G. Conde, “Research electronic data capture (REDCap) - A metadata-driven methodology and workflow process for providing translational research informatics support,” *J Biomed Inform*, vol. Apr;42(2):377-81, 2009.
- [49] P. Angerer, L. Simon, S. Tritschler, F. A. Wolf, D. Fischer, and F. J. Theis, “Single cells make big data: New challenges and opportunities in transcriptomics,” *Current Opinion in Systems Biology*, vol. 4, pp. 85–91, Aug. 2017, doi: 10.1016/j.coisb.2017.07.004.

- [50] C. Vesteghem *et al.*, “Implementing the FAIR Data Principles in precision oncology: review of supporting initiatives,” *Briefings in Bioinformatics*, vol. 21, no. 3, p. 936, Jun. 2019, doi: 10.1093/bib/bbz044.
- [51] Y. W. Yu and G. M. Weber, “Balancing Accuracy and Privacy in Federated Queries of Clinical Data Repositories: Algorithm Development and Validation,” *J Med Internet Res*, vol. 22, no. 11, p. e18735, Nov. 2020, doi: 10.2196/18735.
- [52] R. C. Wilson *et al.*, “DataSHIELD – New Directions and Dimensions,” *Data Science Journal*, vol. 16, no. 0, Apr. 2017, doi: 10.5334/dsj-2017-021.
- [53] J. Scheibner *et al.*, “Revolutionizing Medical Data Sharing Using Advanced Privacy-Enhancing Technologies: Technical, Legal, and Ethical Synthesis,” *J Med Internet Res*, vol. 23, no. 2, p. e25120, Feb. 2021, doi: 10.2196/25120.
- [54] P. N. Robinson, S. Köhler, S. Bauer, D. Seelow, D. Horn, and S. Mundlos, “The Human Phenotype Ontology: a tool for annotating and analyzing human hereditary disease,” *Am J Hum Genet*, vol. 83, no. 5, pp. 610–615, Nov. 2008, doi: 10.1016/j.ajhg.2008.09.017.
- [55] A. W. Forrey *et al.*, “Logical observation identifier names and codes (LOINC) database: a public use set of codes and names for electronic reporting of clinical laboratory test results,” *Clin Chem*, vol. 42, no. 1, pp. 81–90, Jan. 1996.
- [56] Wellcome Trust Case Control Consortium, “Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls,” *Nature*, vol. 447, no. 7145, pp. 661–678, Jun. 2007, doi: 10.1038/nature05911.
- [57] L. Bastarache, J. C. Denny, and D. M. Roden, “Phenome-Wide Association Studies,” *JAMA*, vol. 327, no. 1, pp. 75–76, Jan. 2022, doi: 10.1001/jama.2021.20356.
- [58] R. C. Wang and Z. Wang, “Precision Medicine: Disease Subtyping and Tailored Treatment,” *Cancers (Basel)*, vol. 15, no. 15, p. 3837, Jul. 2023, doi: 10.3390/cancers15153837.
- [59] G. Nyamundanda, K. Eason, J. Guinney, C. J. Lord, and A. Sadanandam, “A Machine-Learning Tool Concurrently Models Single Omics and Phenome Data for Functional Subtyping and Personalized Cancer Medicine,” *Cancers*, vol. 12, no. 10, Art. no. 10, Oct. 2020, doi: 10.3390/cancers12102811.
- [60] F. Cascini, A. Pantovic, Y. A. Al-Ajlouni, V. Puleo, L. D. Maio, and W. Ricciardi, “Health data sharing attitudes towards primary and secondary use of data: a systematic review,” *eClinicalMedicine*, vol. 71, May 2024, doi: 10.1016/j.eclinm.2024.102551.
- [61] A. Mahmud, O. Zalay, A. Springer, K. Arts, and E. Eisenhauer, “Barriers to participation in clinical trials: a physician survey,” *Curr Oncol*, vol. 25, no. 2, pp. 119–125, Apr. 2018, doi: 10.3747/co.25.3857.
- [62] J. T. Mitchel *et al.*, “Evaluation of Data Entry Errors and Data Changes to an Electronic Data Capture Clinical Trial Database,” *Drug Inf J*, vol. 45, no. 4, pp. 421–430, Jul. 2011, doi: 10.1177/009286151104500404.
- [63] A. Le Jeannic, C. Quelen, C. Alberti, and I. Durand-Zaleski, “Comparison of two data collection processes in clinical studies: electronic and paper case report forms,” *BMC Med Res Methodol*, vol. 14, no. 1, p. 7, Jan. 2014, doi: 10.1186/1471-2288-14-7.
- [64] P. A. Harris *et al.*, “The REDCap consortium: Building an international community of software platform partners,” *Journal of Biomedical Informatics*, vol. 95, p. 103208, Jul. 2019, doi: 10.1016/j.jbi.2019.103208.
- [65] K. Ohashi, P. Dykes, K. McIntosh, E. Buckley, M. Wien, and D. W. Bates, “Evaluation of intravenous medication errors with smart infusion pumps in an academic medical center,” *AMIA Annu Symp Proc*, vol. 2013, pp. 1089–1098, 2013.
- [66] H. L. Rehm *et al.*, “GA4GH: International policies and standards for data sharing across genomic research and healthcare,” *Cell Genom*, vol. 1, no. 2, p. 100029, Nov. 2021, doi: 10.1016/j.xgen.2021.100029.
- [67] J. M. Banda, “Fully connecting the Observational Health Data Science and Informatics (OHDSI) initiative with the world of linked open data,” *Genomics Inform*, vol. 17, no. 2,

- p. e13, Jun. 2019, doi: 10.5808/GI.2019.17.2.e13.
- [68] A. M. Kathryn North, "The global alliance for genomics and health: towards international sharing of genomic and clinical data," *Pathology*, vol. 47, pp. S28–S29, Jan. 2015, doi: 10.1097/01.PAT.0000461407.88852.73.
 - [69] M. S. Ladewig *et al.*, "GA4GH phenopackets: A practical introduction," *Advanced Genetics*, p. 2200016, 2022.
 - [70] M. Rueda and A. R., "Beacon v2 Reference Implementation: a toolkit to enable federated sharing of genomic and phenotypic data," *Bioinformatics*, vol. btac568.
 - [71] J. O. B. Jacobsen *et al.*, "The GA4GH Phenopacket schema defines a computable representation of clinical data," *Nat Biotechnol*, vol. 40, no. 6, pp. 817–820, Jun. 2022, doi: 10.1038/s41587-022-01357-4.
 - [72] M. Rueda, I. C. Leist, and I. G. Gut, "Convert-Pheno: A software toolkit for the interconversion of standard data models for phenotypic data," *Journal of Biomedical Informatics*, vol. 149, p. 104558, Jan. 2024, doi: 10.1016/j.jbi.2023.104558.
 - [73] S. Laurie *et al.*, "The RD-Connect Genome-Phenome Analysis Platform: Accelerating diagnosis, research, and gene discovery for rare diseases," *Human Mutation*, vol. 43, no. 6, pp. 717–733, 2022, doi: 10.1002/humu.24353.
 - [74] C. Megill *et al.*, "cellxgene: a performant, scalable exploration platform for high dimensional sparse matrices," Apr. 06, 2021, *bioRxiv*. doi: 10.1101/2021.04.05.438318.
 - [75] F. A. Wolf, P. Angerer, and F. J. Theis, "SCANPY: large-scale single-cell gene expression data analysis," *Genome Biology*, vol. 19, no. 1, p. 15, Feb. 2018, doi: 10.1186/s13059-017-1382-0.
 - [76] I. Virshup, S. Rybakov, F. J. Theis, P. Angerer, and F. A. Wolf, "anndata: Access and store annotated data matrices," *Journal of Open Source Software*, vol. 9, no. 101, p. 4371, Sep. 2024, doi: 10.21105/joss.04371.
 - [77] E. Scheufele *et al.*, "transSMART: An Open Source Knowledge Management and High Content Data Analytics Platform," *AMIA Jt Summits Transl Sci Proc*, vol. 2014, pp. 96–101, Apr. 2014.
 - [78] 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," *International Journal of Epidemiology*, vol. 46, no. 5, pp. 1372–1378, Oct. 2017, doi: 10.1093/ije/dyx180.
 - [79] K. J. van der Velde *et al.*, "MOLGENIS research: advanced bioinformatics data software for non-bioinformaticians," *Bioinformatics*, vol. 35, no. 6, pp. 1076–1078, Mar. 2019, doi: 10.1093/bioinformatics/bty742.
 - [80] M. Nieminen *et al.*, "SODAR: managing multiomics study data and metadata," *GigaScience*, vol. 12, p. giad052, Jan. 2023, doi: 10.1093/gigascience/giad052.
 - [81] B. D. Athey, M. Braxenthaler, M. Haas, and Y. Guo, "transSMART: An Open Source and Community-Driven Informatics and Data Sharing Platform for Clinical and Translational Research," *AMIA Jt Summits Transl Sci Proc*, vol. 2013, pp. 6–8, Mar. 2013.
 - [82] C. I. Barash, K. Elliston, and R. Potenzzone, "transSMART Foundation Datathon 1.0: The cross neurodegenerative diseases challenge," *Appl Transl Genom*, vol. 6, pp. 42–44, Sep. 2015, doi: 10.1016/j.atg.2015.08.004.
 - [83] J. Christoph *et al.*, "Usability and Suitability of the Omics-Integrating Analysis Platform transSMART for Translational Research and Education," *Appl Clin Inform*, vol. 08, no. 04, pp. 1173–1183, Oct. 2017, doi: 10.4338/ACI-2017-05-RA-0085.
 - [84] D. Doiron *et al.*, "Data harmonization and federated analysis of population-based studies: the BioSHaRE project," *Emerg Themes Epidemiol*, vol. 10, no. 1, p. 12, Nov. 2013, doi: 10.1186/1742-7622-10-12.
 - [85] A. Pinot de Moira *et al.*, "The EU Child Cohort Network's core data: establishing a set of findable, accessible, interoperable and re-usable (FAIR) variables," *Eur J Epidemiol*, vol. 36, no. 5, pp. 565–580, 2021, doi: 10.1007/s10654-021-00733-9.

- [86] C. Pang *et al.*, “MOLGENIS/connect: a system for semi-automatic integration of heterogeneous phenotype data with applications in biobanks,” *Bioinformatics*, vol. 32, pp. 2176–2183, 2016.
- [87] W. T. K. Maassen *et al.*, “MOLGENIS VIP: an open-source and modular pipeline for high-throughput and integrated DNA variant analysis,” Apr. 26, 2024, *medRxiv*. doi: 10.1101/2024.04.11.24305656.
- [88] K. J. van der Velde *et al.*, “FAIR Genomes metadata schema promoting Next Generation Sequencing data reuse in Dutch healthcare and research,” *Sci Data*, vol. 9, p. 169, Apr. 2022, doi: 10.1038/s41597-022-01265-x.
- [89] K. J. van der Velde *et al.*, “FAIR Genomes metadata schema promoting Next Generation Sequencing data reuse in Dutch healthcare and research,” *Sci Data*, vol. 9, no. 1, p. 169, Apr. 2022, doi: 10.1038/s41597-022-01265-x.
- [90] M. A. Swertz *et al.*, “The MOLGENIS toolkit: rapid prototyping of biosoftware at the push of a button,” *BMC Bioinformatics*, vol. 11, no. 12, Art. no. 12, Dec. 2010, doi: 10.1186/1471-2105-11-S12-S12.
- [91] B. Cakir, M. Prete, N. Huang, S. van Dongen, P. Pir, and V. Y. Kiselev, “Comparison of visualization tools for single-cell RNAseq data,” *NAR Genomics and Bioinformatics*, vol. 2, no. 3, p. lqaa052, Sep. 2020, doi: 10.1093/nargab/lqaa052.
- [92] K. Li *et al.*, “Cellxgene VIP unleashes full power of interactive visualization and integrative analysis of scRNA-seq, spatial transcriptomics, and multiome data,” Apr. 14, 2022, *bioRxiv*. doi: 10.1101/2020.08.28.270652.
- [93] A. E. Ahmed *et al.*, “Design considerations for workflow management systems use in production genomics research and the clinic,” *Sci Rep*, vol. 11, no. 1, p. 21680, Nov. 2021, doi: 10.1038/s41598-021-99288-8.
- [94] O. Spjuth *et al.*, “Experiences with workflows for automating data-intensive bioinformatics,” *Biol Direct*, vol. 10, p. 43, Aug. 2015, doi: 10.1186/s13062-015-0071-8.
- [95] M. Djaffardjy *et al.*, “Developing and reusing bioinformatics data analysis pipelines using scientific workflow systems,” *Computational and Structural Biotechnology Journal*, vol. 21, pp. 2075–2085, Jan. 2023, doi: 10.1016/j.csbj.2023.03.003.
- [96] F. Strozzi *et al.*, “Scalable Workflows and Reproducible Data Analysis for Genomics,” *Methods Mol Biol*, vol. 1910, pp. 723–745, Jan. 2019, doi: 10.1007/978-1-4939-9074-0_24.
- [97] M. Sztuka, K. Kotlarz, M. Mielczarek, P. Hajduk, J. Liu, and J. Szyda, “Nextflow vs. plain bash: different approaches to the parallelization of SNP calling from the whole genome sequence data,” *NAR Genomics and Bioinformatics*, vol. 6, no. 2, p. lqae040, Jun. 2024, doi: 10.1093/nargab/lqae040.
- [98] T. Reiter *et al.*, “Streamlining data-intensive biology with workflow systems,” *GigaScience*, vol. 10, no. 1, p. g1aa140, Jan. 2021, doi: 10.1093/gigascience/g1aa140.
- [99] F. Mölder, K. P. Jablonski, and L. B., “Sustainable data analysis with Snakemake [version 2; peer review: 2 approved],” *F1000Research*, vol. 10, no. 33, 2021.
- [100] P. Tommaso, M. Chatzou, E. W. Floden, P. P. Barja, E. Palumbo, and C. Notredame, “Nextflow enables reproducible computational workflows,” *Nature Biotechnology*, vol. 35, no. 4, pp. 316–319, 2017.
- [101] U. Bhadra, N. Thakkar, P. Das, and M. Pal Bhadra, “Evolution of circadian rhythms: from bacteria to human,” *Sleep Medicine*, vol. 35, pp. 49–61, Jul. 2017, doi: 10.1016/j.sleep.2017.04.008.
- [102] A. Pascual-Ahuir, J. Fita-Torró, and M. Proft, “Capturing and Understanding the Dynamics and Heterogeneity of Gene Expression in the Living Cell,” *International Journal of Molecular Sciences*, vol. 21, no. 21, p. 8278, Nov. 2020, doi: 10.3390/ijms21218278.
- [103] W. V. Li and J. J. Li, “Modeling and analysis of RNA-seq data: a review from a statistical perspective,” *Quant Biol*, vol. 6, no. 3, pp. 195–209, Sep. 2018, doi: 10.1007/s40484-018-0144-7.

- [104] M. Mistry, M. Piper, J. Liu, and R. Khetani, "hbctraining/DGE_workshop_salmon_online: Differential gene expression workshop lessons from HCBC (first release)." Zenodo, May 2021. doi: 10.5281/zenodo.4783481.
- [105] N. Hermann *et al.*, "Clinical Presentation and Prognostic Features in Patients with Immunotherapy-Induced Vitiligo-like Depigmentation: A Monocentric Prospective Observational Study," *Cancers*, vol. 14, no. 19, p. 4576, Sep. 2022, doi: 10.3390/cancers14194576.
- [106] M. Ashburner *et al.*, "Gene Ontology: tool for the unification of biology," *Nat Genet*, vol. 25, no. 1, pp. 25–29, May 2000, doi: 10.1038/75556.
- [107] M. Kanehisa and S. Goto, "KEGG: kyoto encyclopedia of genes and genomes," *Nucleic Acids Res*, vol. 28, no. 1, pp. 27–30, Jan. 2000, doi: 10.1093/nar/28.1.27.
- [108] G. Joshi-Tope *et al.*, "Reactome: a knowledgebase of biological pathways," *Nucleic Acids Research*, vol. 33, no. suppl_1, pp. D428–D432, Jan. 2005, doi: 10.1093/nar/gki072.
- [109] M. Milacic *et al.*, "The Reactome Pathway Knowledgebase 2024," *Nucleic Acids Research*, vol. 52, no. D1, pp. D672–D678, Jan. 2024, doi: 10.1093/nar/gkad1025.
- [110] A. Subramanian *et al.*, "Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles," *Proc Natl Acad Sci U S A*, vol. 102, no. 43, pp. 15545–15550, Oct. 2005, doi: 10.1073/pnas.0506580102.
- [111] K. Zhao and S. Y. Rhee, "Interpreting omics data with pathway enrichment analysis," *Trends in Genetics*, vol. 39, no. 4, pp. 308–319, Apr. 2023, doi: 10.1016/j.tig.2023.01.003.
- [112] M. I. Love, W. Huber, and S. Anders, "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2," *Genome Biology*, vol. 15, no. 12, p. 550, Dec. 2014, doi: 10.1186/s13059-014-0550-8.
- [113] V. K. Mootha *et al.*, "PGC-1 α -responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes," *Nat Genet*, vol. 34, no. 3, pp. 267–273, Jul. 2003, doi: 10.1038/ng1180.
- [114] L. Kolberg, U. Raudvere, I. Kuzmin, J. Vilo, and H. Peterson, "gprofiler2 – an R package for gene list functional enrichment analysis and namespace conversion toolset g:Profiler," *F1000Res*, Jul. 2020.
- [115] S. Hänzelmann, R. Castelo, and J. Guinney, "GSVA: gene set variation analysis for microarray and RNA-Seq data," *BMC Bioinformatics*, vol. 14, no. 1, p. 7, Jan. 2013, doi: 10.1186/1471-2105-14-7.
- [116] G. Korotkevich, V. Sukhov, N. Budin, B. Shpak, M. N. Artyomov, and A. Sergushichev, "Fast gene set enrichment analysis," Feb. 01, 2021, *bioRxiv*. doi: 10.1101/060012.
- [117] N. Borchering *et al.*, "Mapping the immune environment in clear cell renal carcinoma by single-cell genomics," *Commun Biol*, vol. 4, no. 1, pp. 1–11, Jan. 2021, doi: 10.1038/s42003-020-01625-6.
- [118] Z. Gu and D. Hübschmann, "Make Interactive Complex Heatmaps in R," *Bioinformatics*, vol. 38, no. 5, pp. 1460–1462, Mar. 2022, doi: 10.1093/bioinformatics/btab806.
- [119] A. Luna, O. Shah, C. Sander, and P. Shannon, "cyjShiny: A cytoscape.js R Shiny Widget for network visualization and analysis," *PLOS ONE*, vol. 18, no. 8, p. e0285339, Aug. 2023, doi: 10.1371/journal.pone.0285339.
- [120] I. C. Leist *et al.*, "Pheno-Ranker: a toolkit for comparison of phenotypic data stored in GA4GH standards and beyond," *BMC Bioinformatics*, vol. 25, no. 1, p. 373, Dec. 2024, doi: 10.1186/s12859-024-05993-2.
- [121] M. Tabaka, J. Gould, and A. Regev, "scSVA: an interactive tool for big data visualization and exploration in single-cell omics," Jan. 06, 2019, *bioRxiv*. doi: 10.1101/512582.
- [122] D. Feng, C. E. Whitehurst, D. Shan, J. D. Hill, and Y. G. Yue, "Single Cell Explorer, collaboration-driven tools to leverage large-scale single cell RNA-seq data," *BMC Genomics*, vol. 20, no. 1, p. 676, Aug. 2019, doi: 10.1186/s12864-019-6053-y.

- [123] M. L. Speir *et al.*, “UCSC Cell Browser: visualize your single-cell data,” *Bioinformatics*, vol. 37, no. 23, pp. 4578–4580, Dec. 2021, doi: 10.1093/bioinformatics/btab503.
- [124] D. Danis *et al.*, “A corpus of GA4GH phenopackets: Case-level phenotyping for genomic diagnostics and discovery,” *HGG Adv*, vol. 6, no. 1, p. 100371, Oct. 2024, doi: 10.1016/j.xhgg.2024.100371.
- [125] P. Shannon *et al.*, “Cytoscape: a software environment for integrated models of biomolecular interaction networks,” *Genome Res*, vol. 13, no. 11, pp. 2498–2504, Nov. 2003, doi: 10.1101/gr.1239303.
- [126] M. Franz *et al.*, “Cytoscape.js 2023 update: a graph theory library for visualization and analysis,” *Bioinformatics*, vol. 39, no. 1, p. btad031, Jan. 2023, doi: 10.1093/bioinformatics/btad031.
- [127] J. Cao *et al.*, “The single-cell transcriptional landscape of mammalian organogenesis,” *Nature*, vol. 566, no. 7745, pp. 496–502, Feb. 2019, doi: 10.1038/s41586-019-0969-x.
- [128] C. Klamann, C. J. Lau, J. Ruiz-Ramírez, and G. W. Schwartz, “TooManyCellsInteractive: A visualization tool for dynamic exploration of single-cell data,” *GigaScience*, vol. 13, p. giae056, Jan. 2024, doi: 10.1093/gigascience/giae056.
- [129] N. Schaum *et al.*, “Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris,” *Nature*, vol. 562, no. 7727, pp. 367–372, Oct. 2018, doi: 10.1038/s41586-018-0590-4.
- [130] THE TABULA SAPIENS CONSORTIUM, “The Tabula Sapiens: A multiple-organ, single-cell transcriptomic atlas of humans,” *Science*, vol. 376, no. 6594, p. eabl4896, May 2022, doi: 10.1126/science.abl4896.
- [131] CZI Cell Science Program *et al.*, “CZ CELLxGENE Discover: a single-cell data platform for scalable exploration, analysis and modeling of aggregated data,” *Nucleic Acids Research*, vol. 53, no. D1, pp. D886–D900, Jan. 2025, doi: 10.1093/nar/gkae1142.
- [132] E. A. Voss *et al.*, “Feasibility and utility of applications of the common data model to multiple, disparate observational health databases,” *J Am Med Inform Assoc*, vol. 22, no. 3, pp. 553–564, May 2015, doi: 10.1093/jamia/ocu023.
- [133] S. N. Duda *et al.*, “HL7 FHIR-based tools and initiatives to support clinical research: a scoping review,” *Journal of the American Medical Informatics Association*, vol. 29, no. 9, pp. 1642–1653, Sep. 2022, doi: 10.1093/jamia/ocac105.
- [134] T. Souza, R. Kush, and J. P. Evans, “Global clinical data interchange standards are here!,” *Drug Discovery Today*, vol. 12, no. 3, pp. 174–181, Feb. 2007, doi: 10.1016/j.drudis.2006.12.012.
- [135] S. Garde, P. Knaup, E. J. S. Hovenga, and S. Heard, “Towards Semantic Interoperability for Electronic Health Records,” *Methods Inf Med*, vol. 46, no. 3, pp. 332–343, 2007, doi: 10.1160/ME5001.
- [136] H. Matthew, I. Montani, S. Van Landeghem, and A. Boyd, “spaCy: Industrial-strength Natural Language Processing in Python,” Oct. 2023, doi: 10.5281/zenodo.1212303.