Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

DEPLOYMENT PACK DOCUMENTATION

VERSION: JANUARY 2019

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP DEPLOYMENT PACK- Summary & Content List

Purpose

The present MIP Deployment Pack presents is a portfolio of documents providing an overview of the deployment process of the Medical Informatics Platform in research centres and clinical institutions.

- **Document 1**

Letter of introduction

- **Document 2**

Executive Summary - Outlines the primary issues related to the deployment of the Medical Informatics Platform (MIP) into hospitals participating to the MIP network. It provides the overall context, short description of the MIP and its functionalities, followed by an overview of deployment activities and responsibilities.

- **Document 3**

Platform Installation and License Agreement - Is the agreement to be signed by a legal representative of the Centre where the MIP will be installed and a legal representative of CHUV - the legal entity responsible for the development of the Medical Informatics Platform and the management of the infrastructure within the framework of sub-project 8 of the Human Brain Project. It includes a Service Level Agreement (SLA) as Appendix 2.

- **Document 4**

MIP Deployment Ethics and Legal Requirements - MIP Local - Present document outlines the Ethics and Legal requirements and responsibilities related to the
deployment of the Medical Informatics Platform (MIP) into hospitals participating to the MIP network.

- **Document 5**

MIP Technical Specifications and Step-by-Step guide for installation - Provides detailed specification of hardware, software, connectivity and network service configuration, including a step-by-step MIP software deployment guide.

- **Document 6**

MIP System Description - Provides a detailed description of the architecture of the MIP.

- **Document 7**

MIP User Manuel - Documents the manipulation descriptions and the functionalities for users

Note: This package was produced in January 2019 and is subject to updates and additions

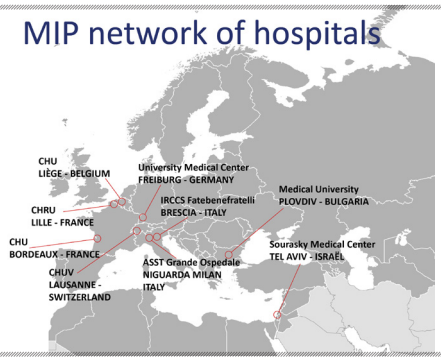
Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

PRIVACY-PRESERVING SOFTWARE FOR MEDICAL DATA SHARING

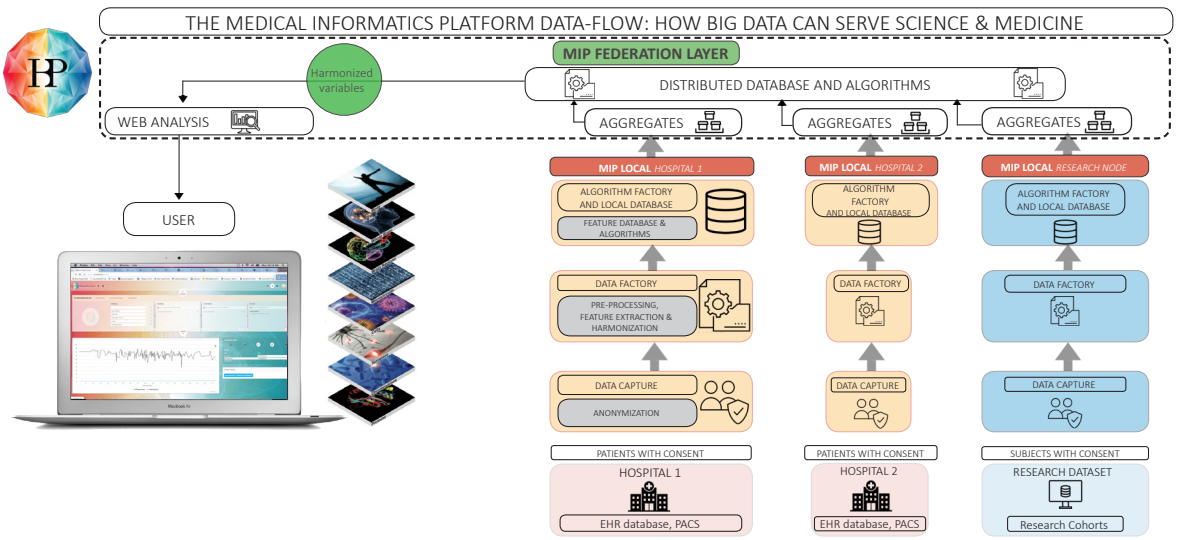
As clinicians, neuroscientists, epidemiologists, researchers, health managers, you deal with data everyday. The amount of data available is growing and represents an unprecedented opportunity to make significant progress by analyzing it with smart innovative tools and new perspectives.

Legal and ethical framework in Europe ensures the protection of personal data and delineate the purposes for which processing is legitimate. Centralizing all the data available is not possible, but solutions begin to appear.



SP8 develops and operates the Medical Informatics Platform (MIP), a privacy preserving data analytics application for clinicians and researchers, enabling you to run analysis on health-related data distributed across different hospitals and research centers, without moving the data outside their original storage.

MIP can help you investigate and compare harmonized medical data extracted from pre-processed neuroimaging, neurophysiological,-omics and medical records. It features a user-friendly interface to run statistical analysis and predictive models using machine learning on large datasets.



The present package will introduce you to the MIP, highlighting some key features and benefits, as well as give you an overview of the requirements and deployment process.
We hope it will raise your interest, in which case our team is available to provide you with a more detailed documentation and a personalized support.

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

EXECUTIVE SUMMARY

VERSION: JANUARY 2019

THE MEDICAL INFORMATICS PLATFORM

MIP DEPLOYMENT EXECUTIVE SUMMARY

Purpose

The present document outlines the primary issues related to the deployment of the Medical Informatics Platform (MIP) into hospitals participating to the MIP network. It provides the overall context, short description of the MIP and its functionalities, followed by an overview of deployment activities and responsibilities.

Introduction

Brain diseases, considered as a whole, affect 165 million European citizens, a large number of whom are being at least partly managed in hospitals. The clinical data collected from these patients represent a unique source of information for better understanding and treating brain diseases but are unfortunately not usually available for research.

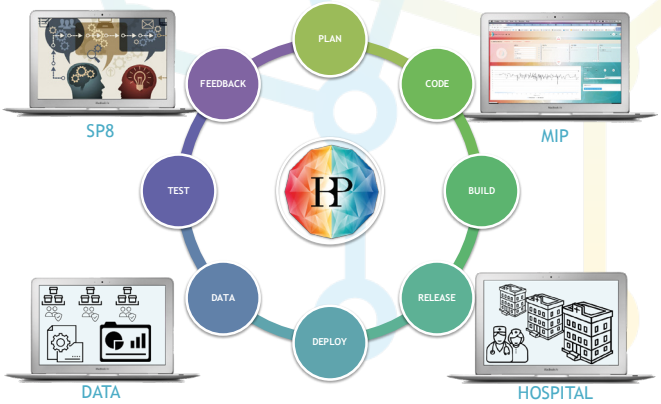
The MIP has been developed by the Human Brain Project (HBP), a EU Horizon 2020 Flagship project, in order to facilitate access to clinical data stored in hospitals, for research purpose, while preserving data privacy. The MIP aims at enabling breakthrough medical progress in the field of brain diseases through access to an unprecedented volume of patients' data.

MIP description

The MIP is an innovative data analysis and data collection system that provides an interface for various investigators (clinicians, neuroscientists, epidemiologists, researchers, health managers) enabling them to access and analyze anonymized medical data currently locked in hospitals, without moving the data from the hospital where they reside, and without infringing on patient privacy.

The MIP is designed to help clinicians and researchers aiming to adopt advance analytics for diagnosis and research in clinics and to promote collaborative neuroscience research using distributed hospital data.

The MIP is divided into two main components, called MIP-Local and MIP-Federated Node, which shall be installed on different servers within the participating hospitals. MIP-Local contains pseudonymised data that can only be accessed and analyzed by the Local Data Coordinator and its accredited staff from within the hospital. MIP-Federated Node contains anonymized data and can be connected to other MIP-Federated Nodes in other hospitals. Upon signed agreement between data providers from the MIP network, accredited investigators can query multiple MIP federated nodes and obtain aggregate results. Queries of the MIP-Federated Nodes do not allow to copy or upload any data, nor to see individual patient's data.



The 2-tier MIP architecture (MIP-Local, MIP-Federated Node) has been designed to address the specific challenges of:
1) local deployment adapted to each hospitals' environment,
2) capturing and processing heterogeneous type of data (e.g. socio-demographic, clinical, biological and neuroimaging data),
3) fulfilling privacy rules, policies and best practices to enable efficient and secure data sharing,
4) harmonizing data through Common Data Elements for cross-site comparisons, and
5) integrating readily available statistical and machine learning tools.

During the two completed phases of HBP, the MIP has been developed and installed in an increasing number of participating hospitals. The Lausanne university hospital (CHUV) is the HBP partner coordinating this activity

Key benefits to participate in the MIP

- Participate in the largest ever funded Europe-wide brain research initiative;
- Train and use novel state-of-the-art analytical tools, include machine learning algorithms;
- Investigate and discover novel findings from its own data using the MIP-Local;
- Participate or lead Federated analyses on big data available in the network of MIP-Federated Nodes
- Develop new scientific collaborations
- Increase the chance of future successful national or European competitive grant applications

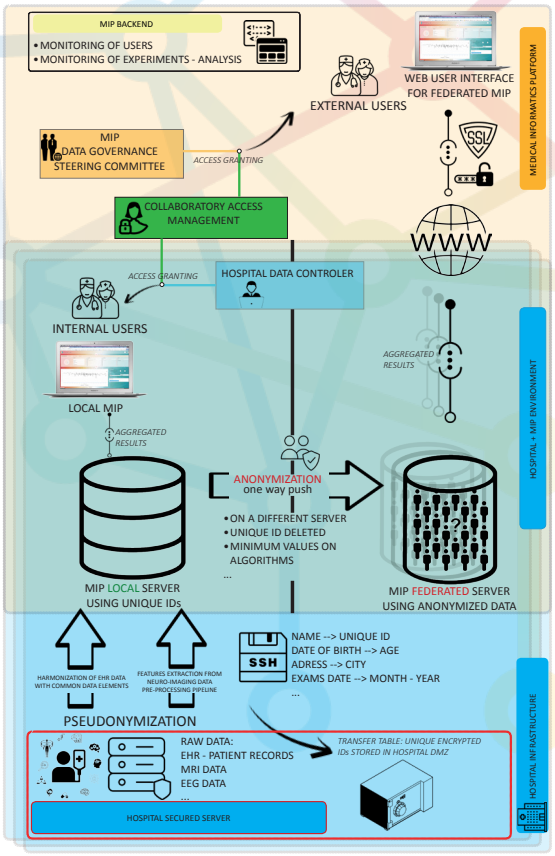
MIP Functionalities

Functionality of the Medical informatics Platform	
Components	Description
Data Capture	Set of tools allowing to capture patient data from hospital's information systems, and their de-identification
Data Factory	Configurable tool chain for pre-processing neuroimaging/biomedical/demographic patients' data to extract relevant features, harmonize and map them to the MIP standard schema, as well as anonymize the resulting database.
Algorithm Library	Standard catalogue of centralised statistical methods and predictive machine learning algorithms run by the Algorithm Factory. They range from classical ones (such as linear regression, Anova, k-nearest neighbour and other naive Bayes) to more sophisticated ones (such as predictive clustering trees).
Algorithm Factory	Framework used to execute the algorithms from the Algorithm Library, as well as to train predictive models and test them using a cross-validation schema.
Web Portal	The web-based interface to offer end-user access to MIP functionalities (see list below).
Development (SDK)	Tools for integration of additional methods and algorithms, based on Docker technology.
End-user functionality	Description
Private/Public Workspace	Every user gets an account. Once authenticated he/she can access his/her private workspace. A public workspace is also provided to users to share content among the community members. The content includes data models, experiments, articles and protocols.
Variable Exploration	Interactive exploration of available metadata and data statistics.
Descriptive Statistics	Histograms of separate variables and interaction between them
Interactive Analytics and Model Creation	Creation and configuration of data models, which are sets of explanatory and predictive variables, including filters, which define filtered subsets of selected data. Analysis of variables' interactions based on this subset of cross-sectional data. Interactive visualisations for few selected variables by mean of design matrices and other plots. Data models can be saved for reuse and sharing.
Experiment Builder	Configuration and running of experiments consisting of executing algorithms on data defined in previously configured data models. Unique execution of an algorithm is possible, as well as repetitive execution of several algorithms for a cross-validation using k-random schema or using partition based on the provenance. Experiments can be saved and shared between users.
Article writing	Creation of articles where users can include results of experiments or descriptive statistics and interactive analysis plots. Articles can be saved and shared with peers.

MIP governance and management

As delineated in the next section, installation and running of the MIP in Hospitals require good collaboration between several stakeholders defined below:

1. MIP-CHUV deployment team: This team, based at CHUV (Lausanne), has been responsible for MIP development and now ensures its deployment, maintenance and debugging when necessary. It includes the Head of the Department of Clinical Neuroscience in charge of the MIP project, the vice-director of the CHUV IT Department, as well as project managers and IT engineers involved in MIP development and maintenance, one of whom is appointed as the MIP-CHUV Deployment Manager.
2. Hospital Deployment Team: This team, based in the Hospital where the MIP is to be deployed, includes at least one readily available representative from the clinical or research department aiming to use the MIP, referred to as the Hospital Deployment Manager, and one representative from the hospital IT department. Administrative and legal staff might also participate. The Hospital Deployment Manager shall be responsible for coordinating the work of the technical (IT) and medical research staff of his Hospital in close collaboration with the MIP-CHUV Deployment manager. In addition, a Local Data Coordinator shall be identified within the Hospital medical research team to supervise all activities in relation with capturing and analyzing data into MIP-Local or MIP-Federated node. The Hospital Deployment Manager and Local Data Coordinator might be the same or a different person. In case the hospital aims at using the MIP for different brain diseases, different Local Data Coordinators might be ascribed to each of these diseases.
3. MIP Data Governance Steering Committee (DGSC): MIP DGSC has been established in order to control all activities in relation with data analyses performed over MIP-Federated nodes and ensure that all Local Data Coordinators are duly involved in analyses that include their data. To this effect, DGSC is organized in disease-specific boards for each condition currently considered in the MIP network (i.e. dementia, epilepsy, traumatic brain injury, mental health, disorders of consciousness). Each MIP DGSC disease-specific board includes all Local Data Coordinators who have agreed to participate to federated analyses that involve their data, and have their hospital signed a MIP Data Sharing Agreement.



MIP deployment process

The MIP deployment process follows 12 steps, many of which are optional or can be deferred to a later stage. The rationale behind this incremental strategy is to start rapidly and easily, and progressively allow the hospital staff to gain experience along the implementation process.

The first four steps fall within the scope of the MIP installation agreement:

1. Identify all relevant hospital staff (Hospital Deployment Team) required to proceed to MIP deployment, including the Hospital Deployment manager, the Local Data Coordinator (s), as well as the MIP-CHUV Deployment Manager counterpart.

2. Secure signature of the MIP Installation Agreement by both the Hospital and CHUV legal representatives. This agreement only covers installation of the MIP software, and not data sharing.

3. Prepare the IT infrastructure (e.g. servers) needed to install the MIP-Local only or both the MIP-Local and MIP-Federated Node together, according to their specifications. This preparation is typically performed by the Hospital IT staff under the supervision of the Hospital Deployment Manager.

4. Install the MIP-Local software as a stand-alone component, or both the MIP-Local and MIP-Federated Node software together. The installation of the MIP-Federated Node component is thus optional and can be installed at a later stage. The above installations are usually being performed jointly by the Hospital IT staff and the MIP-CHUV deployment team under the co-supervision of the Hospital and MIP-CHUV Deployment Managers. Part of the installation can be performed remotely through VPN connection. On site hospital-specific tuning is however often required.

Steps 5 to 12 are optional, fall outside the scope of the MIP installation agreement, and can be deferred in time. Steps 6 to 12 require the signature of the Data Sharing Agreement:

5. Capture into MIP-Local pseudonymized patients' data from the hospital clinical or research department aiming to use the MIP, in full compliance with all local ethics and regulatory issues. This step is undertaken under the full responsibility of the above department and its Local Data Coordinator . Once data have been captured in MIP-Local, the Local Data Coordinator and his accredited local staff can use the MIP to analyse their data. No other stakeholder has access to the data.

6. Secure the signature of the MIP Data Sharing Agreement by both the Hospital and CHUV legal representatives. This agreement covers the possibility to perform federated analyses of fully anonymized data captured in the MIP-Federated node of the hospital.

7. Secure participation of the Local Data Coordinator to the relevant MIP DGSC disease-specific board.

8. Prepare the IT infrastructure (e.g. server) to install the MIP-Federated Node software if not already done during step 3 (same procedure as previously described).

9. Install the MIP-Federated Node software if not already done during step 4 (same procedure as previously described).

10. Proceed to full anonymization of the data stored in MIP-Local and then push these data into the MIP-Federated Node database. This step is undertaken under the full responsibility of the Local Data Coordinator .

11. Enable the link to the network of MIP-Federated Nodes from other hospitals. This link is enabled/disabled by the Hospital IT staff in charge of installing and controlling the MIP, under the supervision of the Hospital Deployment Manager.

12. Participate to federated analyses according to the procedures agreed upon within the relevant MIP DGSC disease-specific board. This activity is being performed by the Local Data Coordinator or its accredited staff and can only provide aggregated results with no possibility to copy or upload the database, nor that of exploring individual patient's data.

Contacts

HBP-SP8 leadership: philipperyvlinhbp@gmail.com
HBP-SP8 leadership: Sandra.Schweighauser@chuv.ch

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP INSTALLATION AND LICENSE AGREEMENT

VERSION: JANUARY 2019

HDC – v.20.01.2019

The Medical Informatics Platform Installation and License Agreement

BETWEEN

Centre hospitalier universitaire vaudois (hereinafter : "CHUV").

AND

CENTRE: NAME – ADRESS (hereinafter: the "CENTRE").

PREAMBLE

WHEREAS the Medical Informatics Platform (MIP) is an innovative IT solution using open source software that provides an interface for various investigators (clinicians, neuroscientists, epidemiologists, researchers, health managers) enabling them to access and analyze medical data currently locked in hospitals or medical research CENTREs.

WHEREAS the MIP was developed as part of the sub-project 8 of the Human Brain Project, a EU funded H2020 FET Flagship Project, to unlock access to CENTRE data while preserving data privacy.

WHEREAS the HBP Flagship Project was launched by the European Commission's Future and Emerging Technologies (FET) scheme in October 2013 and is scheduled to run for ten years.

WHEREAS the MIP consists in two components:

- one "MIP LOCAL" which is being installed in the CENTREs and contains pseudonymised data that can only be accessed and analyzed by the accredited staff from within the CENTRE, and

- one "MIP federated node" which installed on a second server in the CENTRE, contains anonymized data and can be connected to other MIP-Federated Nodes in other hospitals/CENTRES. Upon signed agreement between data providers from the MIP network, accredited investigators can query multiple MIP federated nodes and obtain aggregate results. Queries of the MIP-Federated Nodes do not allow to copy or upload any data, nor to see individual patient's data.

WHEREAS the clinical impact of the MIP specifically addresses EU health priorities to reduce the burden of brain diseases by leveraging personalized medicine and treatment;

WHEREAS the Centre Hospitalier Universitaire Vaudois (CHUV) is attached to the Department of health and social action of the State of Vaud;

WHEREAS, subject to the approval of the State government, the General Manager of CHUV is entitled to decide about collaborations with other health institutions and to sign collaboration agreements legally binding for CHUV;

WHEREAS CHUV is the legal entity responsible for the development of the Medical Informatics Platform and the management of the infrastructure within the framework of sub-project 8 of the Human Brain Project.

NOW, THEREFORE, the Parties agree as follows:

SUBJECT-MATTER

This AGREEMENT binds the Parties in the context of the INSTALLATION and USE of the Medical Informatics Platform for research only.

It is covering the DISTRIBUTION, INSTALLATION and USE of the Medical Informatics Platform in the CENTRE.

This AGREEMENT does not cover any aspect of data sharing. These aspects will be covered in a different contract "Data Sharing Agreement".

DEFINITION OF TERMS

AGREEMENT means this Installation and License Agreement;

CENTRE means any hospital, clinic, research institute or university entering into this AGREEMENT where the Medical Informatics Platform is installed pursuant to the terms of this AGREEMENT;

EXTERNAL USER means an end user accredited by the MIP data management for the MIP FEDERATE NETWORK.

GDPR means the General Data Protection Regulation 2016/679;

INSTALLATION means the process of downloading and installing the SOFTWARE on the CENTRE IT infrastructure and servers;

LICENSE means the GNU General Public License v. 3 (https://www.gnu.org/licenses/gpl-3.0.en.html) reproduced in Annex III;

MEDICAL INFORMATICS PLATFORM (**MIP)** means an IT platform comprising a suite of open source software, including a front-end interface for EXTERNAL USERS designed to allow privacy preserving data sharing within and across hospitals/CENTREs in Europe, based on the use of software installed locally in the CENTRE;

MIP means the Medical Informatics Platform made available by CHUV under the terms of this Agreement and installed in the CENTRE.

MIP LOCAL means the primary component of the MIP software installed in the CENTRE, which contains pseudonymized data from the CENTRE and can only be accessed by users from the CENTRE.

MIP FEDERATED NODE means the optional component of the MIP software installed in the CENTRE, which contains anonymized data from the CENTRE and can be connected to the MIP FEDERATE NETWORK upon authorization by the CENTER and under its control.

MIP FEDERATE NETWORK means the network of all authorized, active and connected MIP federated nodes;

MIP IT TEAM means the team from CHUV or its partners listed in ANNEX I supporting the installation and maintenance of MIP in the CENTRE;

PACKAGE means the suite of SOFTWARE and the instructions to install the SOFTWARE on the CENTRE IT infrastructure;

SERVICE or **SERVICES** refers to the services provided to CENTRE by CHUV pursuant to the SLA;

SERVICE LEVEL AGREEMENT (SLA) means the agreement attached in ANNEX IV defining the SERVICES to be provided by CHUV to the HOSPITAL;

SOFTWARE means the suite of open source software listed in ANNEX II contained into the MIP, including all UPDATES and UPGRADES of such open source software

SYSTEM means the information system made available through the MIP.

ARTICLES

1 SCOPE

Pursuant to the terms of this AGREEMENT, CHUV and, on its behalf, the MIP IT TEAM will provide CENTRE with:

1.1. The PACKAGE;
1.2. The LICENSE;
1.3. The SERVICES.

2 LICENSE

CHUV distributes, makes available to the CENTRE and installs the SOFWARE on the CENTRE's dedicated server, pursuant to the terms of the LICENSE.

CENTRE is entitled to copy, reproduce, distribute, modify, translate, create derivative works out of, the SOFTWARE (or the resulting derivative work) in any medium, with or without modifications, in source form, provided it strictly complies with the terms of the LICENSE, meaning that the CENTRE shall ensure that:

a) The SOFTWARE carries prominent notices stating that CENTRE modified it, and giving a relevant date;

b) The SOFTWARE must carry prominent notices stating that it is released under the LICENSE and any conditions added under section 7 of the LICENSE.

c) CENTRE must license the entire SOFTWARE (including all derivative works), as a whole, under the LICENSE to anyone who comes into possession of a copy of the SOFTWARE.

d) If the SOFTWARE has interactive user interfaces, each must display appropriate legal notices.

All appropriate copyright and other proprietary notices and legends shall be retained on the SOFTWARE, and CENTRE shall maintain and reproduce such notices on all authorized copies of the SOFTWARE and related documentation including in any scientific publications.

3 SERVICES

CHUV provides to CENTRE the SERVICES listed in the SERVICE LEVEL AGREEMENT.

Such SERVICES may be, at all times, delegated by CHUV to a third party, subject to prior approval of the CENTRE.

4 CENTRE OBLIGATIONS

CENTRE is responsible for providing the required IT infrastructure dedicated for MIP to be installed and commit IT resources for the INSTALLATION, in compliance with the applicable data protection regulations.

CENTRE shall accept the INSTALLATION of the SOFTWARE locally on its IT infrastructure dedicated to MIP.

CENTRE is responsible for ensuring that MIP is used for research purpose only. MIP has not been designed as a clinical diagnostic software. CENTRE is also responsible for ensuring that any data stored into the MIP has been pseudonymised according to standards and that any data stored in the MIP FEDERATED NODE has been anonymized.

CENTRE is responsible for complying with the terms of the LICENSE.

5 TERM AND TERMINATION

This AGREEMENT is valid for an indefinite period of time (the “Term”).

Either Party may terminate this AGREEMENT by serving a thirty (30) days written notice by certified mail to the other Party.

This AGREEMENT shall be automatically terminated in the event of termination of the HBP Flagship Project.

In the event of termination of this AGREEMENT, CHUV will stop providing the SERVICES.

6 REPRESENTATIONS AND WARRANTIES

Each of the Parties represent and warrant that they have the unrestricted right and authority:

a) to enter validly into this AGREEEMENT;

b) to validly represent the party to this AGREEMENT;

c) to perform all undertakings under or in connection with this AGREEMENT;

and represent and warrant that this AGREEMENT constitutes a valid, legal and binding obligation of the Parties, enforceable against the parties in accordance with its terms.

7 NO WARRANTIES

The SOFTWARE is provided to the CENTRE "AS IS" without any warranty of any kind.

The Parties make no warranties, either explicit or implied, with respect to the MIP, the SOFTWARE, and/or the SERVICES and/or as to any matter including but not limited to, warranties of ownership, novelty, patentability, originality, accuracy, non-infringement, merchantability, quality or fitness of the MIP and/or the SOFTWARE, the SERVICES for a particular purpose.

The SERVICES are provided by CHUV to the CENTRE without any warranty and without any obligation of result.

8 LIABILITY

Each Party shall only be liable towards the other in the event of fraud or gross negligence resulting in direct damages for the other party. Any other liability incurred by a party as a result of a breach of the obligations contained in this AGREEMENT and/or as a result of the MIP, the SOFTWARE and/or the SERVICES is excluded.

9 APPLICABLE LAW AND PLACE OF JURISDICTION

This AGREEMENT shall be governed by the laws of Switzerland.

All disputes concerning intellectual property arising under this AGREEMENT shall be submitted to mediation in accordance with the WIPO Mediation Rules. The place of mediation shall be Lausanne unless otherwise agreed upon. The language to be used in the mediation shall be English unless otherwise agreed upon. If, and to the extent that, any such dispute has not been settled pursuant to the mediation within 60 calendar days of the commencement of the mediation, the courts of Lausanne shall have exclusive jurisdiction.

For all other disputes arising under this AGREEMENT, which cannot be solved amicably, the courts of Lausanne shall have exclusive jurisdiction.

10 AMENDMENT

This AGREEMENT may not be modified except by a written instrument signed by authorized representatives of the Parties.

11 ASSIGNMENT

CHUV shall be entitled to assign this AGREEMENT or delegate its obligations under this AGREEMENT either in whole or in part without the prior written consent of CENTRE.

CENTRE shall not assign this AGREEMENT or its obligations under this AGREEMENT without prior written approval given by CHUV.

12 MISCELLANEOUS

This AGREEMENT supersedes any and all prior agreements or understandings relating to the subject matter hereof. This AGREEMENT may not be modified except by a written instrument signed by authorized representatives of the Parties.

Neither Party shall be entitled to commit the other Party to any obligation in connection with this AGREEMENT, without the prior written consent of the other Party.

This AGREEMENT may be signed in counterparts, and by either party on separate counterpart, each which shall be deemed original, but all of which together constitute one and the same instrument.

Nothing whatever in this AGREEMENT shall be construed as conferring rights to use in advertising, publicity, or otherwise the name and logo of either party or any of its respective marks or name of employees.

The terms of this AGREEMENT are severable such that if any term or provision is declared by a court of competent jurisdiction to be illegal, void, or unenforceable, the remainder of the provisions shall continue to be valid and enforceable.

HDC – v.20.01.2019

IN WITNESS WHEREOF, the Parties hereto have executed this AGREEMENT as one of the date first written above.

Signed for and on behalf of:

CHUV

by _____

Title: General Manager

Date:

Signed for and on behalf of

CENTRE

by: _____

Title:

Date:

Annex I: GNU General Public License v. 3 (https://www.gnu.org/licenses/gpl-3.0.en.html)

Annex II: Service Level Agreement

ANNEX I – GNU General Public License

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

PREAMBLE

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the

work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of

the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information

must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time

you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of

the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR

OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by

the Free Software Foundation, either version 3 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

GNU General Public License for more details.

You should have received a copy of the GNU General Public License

along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>

This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it

under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <https://www.gnu.org/licenses/why-not-lgpl.html>

ANNEX II - Service Level Agreement (SLA)

1 Definitions

CLAIM means a claim submitted by CENTRE to CHUV with respect to the SOFTWARE pursuant to this SLA;

EXTERNAL CONNECTIVITY is bi-directional network traffic that can be sent and received from a public IP address ensuring a remote access;

INCIDENT is an unplanned interruption to or reduction in the quality of the IT service provided by the MIP, including the failure of a configuration item that has not yet affected service (ITIL 2011);

MIP LOCAL means the primary component of the MIP software installed in the CENTRE, which contains pseudonymized data from the CENTRE and can only be accessed by users from the CENTRE.

RELEASE means a collection of hardware, software, documentation, processes or other components required to implement one or more approved changes, update or upgrade to the SOFTWARE. The contents of each RELEASE are managed and tested as a unit, and deployed as a single entity.

SUPPORT means the services by which CHUV may provide assistance to CENTRE to resolve issues with the MIP.

Definitions of the MIP Installation and License Agreement are hereby incorporated by reference.

2 SERVICES

The SERVICES will be provided to CENTRE for the duration of the Term of the MIP Installation and License Agreement. The SLA can be updated independently from the renewal or update of the MIP Installation and License Agreement.

The SERVICES include:

- SUPPORT:
  - Level 1;
  - Level 2;
  - Level 3.
- End-user Assistance:
  - User access management;
  - User and technical documentation.
- INCIDENT Management:

- INCIDENT report;
- Root-cause analysis;
- INCIDENT resolution.

- Availability Management:
  - Design service for availability;
  - Availability testing;
  - Availability monitoring and reporting.

- SOFTWARE Maintenance:
  - SOFTWARE maintenance;
  - RELEASE planning;
  - Emergency SOFTWARE release planning;
  - RELEASE management.

3 Support

There are three levels of support.

Level 1	The first level support attempts to collect as much information and diagnostics about the INCIDENT as possible and best effort to resolve the issue on the spot. Every INCIDENT or user demand has to be reported to the first level support, which is responsible for managing the CLAIM. If the first level support is not able to resolve the INCIDENT right away, it will escalate the INCIDENT to second level support.
Level 2	The second level is devoted to INCIDENT and problem resolutions. If the second level support is not able to resolve the INCIDENT, it will escalate the INCIDENT to third level support.
Level 3	The third level is constituted by all third parties involved is the development of the SOFTWARE. If CHUV HBP MIP Team cannot fix the underlying cause of the INCIDENT, it escalates to the relevant open source software communities.

3.1 First-level Support

The first level support is provided by the HBP HLST team.

Contact address is support@humanbrainproject.eu.

Opening hours are from 9:00 to 17:00 Central European Time except bank holidays (federal and cantonal (Vaud)).

3.2 Second-level Support

The second level support is provided by the CHUV HBP MIP team.

In cases when the first level support team cannot resolve an INCIDENT, it provides a request for support to the second level support team.

The second level support provides SERVICE in its domain of competencies. The SERVICE is provided on a best-effort basis during the opening hours from 9:00 to 17:00 Central European Time except bank holidays (federal and cantonal (Vaud)).

3.3 Third Level Support

The third level support is provided by the CHUV HBP MIP team.

In cases when the second level support team cannot resolve an INCIDENT, it provides a request for support to the third level support team.

The third-level support team consists of the experienced members of the CHUV MIP team involved in the development of the MIP platform. They have all the necessary competencies to resolve the most complex INCIDENTS.

If appropriate, in cases when the source of the incident is a third-party component, third-level support team may propose integration of an alternative solution to the development team. In these cases, the development team decides on whether and when the change will be released, subject to internal HBP MIP development project coordination and prioritization.

The SERVICE is provided on a best-effort basis during the opening hours from 9:00 to 17:00 Central European Time except bank holidays (federal and cantonal (Vaud)).

4 End-user Assistance

Users of the MIP can contact the first level support for assistance.

Assistance includes support in using the MIP, simple modifications of parameters, corrections to the DATA or any other request related to the MIP usage that the user or super-user cannot respond by his own means. The CLAIM should be accompanied by the expected solution deadline.

5 INCIDENT Management

CENTRE raises an INCIDENT by contacting the MIP first level support using the e-mail address provided in section 2 of this SLA. The first level support registers the INCIDENT and provides its reference (ticket number) to the CENTRE.

The first-level support provider analyses root-cause of the INCIDENT and resolves it in the scope of its responsibilities and its domain of competence. In cases when it cannot resolve the INCIDENT, the first level support provider escalates the INCIDENT to the second-level support (CHUV HBP MIP team) and informs the end-user about the escalation.

The second level support provider analyses root-cause of the INCIDENTSs escalated by the first-level support and resolves it in the scope of its responsibilities and its domain of competence. In cases

when it cannot resolve the INCIDENT, the second-level support provider escalates the INCIDENT to the third-level support.

The third-level support provider analyses root-cause of the INCIDENT and resolves it in the scope of its responsibilities. The third-level support provider is an engineer experienced with the development of the HBP MIP who has all necessary competencies to resolve the most complex INCIDENTS. The third-level support provider is responsible for coordinating the plans for emergency software releases with the HBP MIP software development team.

6 Availability Management

THE CHUV HBP MIP team is responsible for designing the procedures and technical features to maximise the MIP availability levels.

The CHUV HBP MIP TEAM shall continuously monitor MIP availability, identify the areas where it must be improved, and implement measures for the availability maximization. The identification of areas for improvement and implementation of measures for improvement are done on a best-effort basis.

7 SOFTWARE MAINTENANCE

7.1 Maintenance Window

The maintenance windows planned for releasing MIP UPDATES or UPGRADES will be communicated at least 3 months in advance (usually year after year). There can be at most four upgrades per year.

7.2 Emergency RELEASES

Between the planned HBP MIP software maintenance windows, important and/or urgent corrections shall be released in the scope of emergency software RELEASES, subject to internal HBP MIP development project coordination and prioritization.

7.3 EXTERNAL CONNECTIVITY

To guarantee maximum level of platform availability, CENTRE shall permanently provide to at least two CHUV HBP MIP engineers a secure EXTERNAL CONNECTIVITY to super-user credentials for the MIP execution environment.

If CENTRE cannot provide permanent EXTERNAL CONNECTIVITY and credentials to CHUV HBP MIP engineers, it is recommended to at least grant a temporary EXTERNAL CONNECTIVITY and credentials to the MIP execution environment during the maintenance release windows, or intermittently on-demand, within a reasonable timeframe, for un-planned emergency operations.

In cases where CENTRE's data centre security policy strictly forbids any remote access and/or super-user credentials to third-party personnel, CENTRE and CHUV HBP MIP team will define a specific operation procedure.

7.4 RELEASE management

During the Term, RELEASES will be deployed in the CENTRE's private execution environment (MIP LOCAL) either by the CHUV HBP MIP team or by the CENTRE's IT responsible.

CENTRE shall install the RELEASES or allow MIP IT TEAM to proceed to the installation of the RELEASES.

The same terms of the LICENSE contained in the MIP Installation and License Agreement will apply to the RELEASES.

8 EXCLUSIONS

This SLA does not apply to any INCIDENTS:

- occurring during maintenance of the SOFTWARE;
- due to factors outside CHUV's reasonable control (for example, a network or device failure at the CENTRE);
- failure of CENTRE to provide EXTERNAL CONNECTIVITY to MIP IT TEAM to MIP LOCAL or to CENTRE server;
- resulting from CENTRE's or third-party hardware or software, including VPN devices that have not been tested and found to be compatible by MIP IT TEAM;
- resulting from actions or inactions of CENTRE or third parties;
- caused by CENTRE's use of the SOFTWARE after MIP IT TEAM advised CENTRE to modify its use of the SOFTWARE, if CENTRE did not modify its use as advised;
- due to any act or omission of CENTRE or CENTRE's employees, agents, contractors, or vendors, or anyone gaining access to the SOFTWARE by means of CENTRE's passwords or equipment.

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP ETHICS AND LEGAL REQUIREMENTS

VERSION: JANUARY 2019

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP DEPLOYMENT ETHICS AND LEGAL REQUIREMENTS

MIP LOCAL

Purpose

The present document outlines the **Ethics and Legal requirements and responsibilities** related to the deployment of the Medical Informatics Platform (MIP) into hospitals participating to the MIP network.

Introduction

MIP relies on citizens and patients allowing researcher to use their private personal medical data.
MIP is a platform designed to enable large scale, privacy preserving data sharing for research purpose. It is the responsibility of the hospitals to make sure that their data subjects and patients have given their consent for the collection of the data. It is also the responsibility of the hospitals to ensure that this data has been properly pseudonymized / anonymized according to the standards and the recommendations of the MIP deployment team.

Structure of the document

The following document presents information at two levels. Blue Text boxes are used to summarize and provide legal conclusions regarding application of the GDPR to the MIP.

The analysis and reasoning behind the highlighted text box conclusion is provided in more details with selected reference to the regulation. The intent of this structure is to provide an accessible or operational document while also providing expanded explanations for users requiring additional information.

Basis and Reference

MIP is complying with the GDPR with special consideration to Privacy by Design and Privacy by Default.
Legislation and Guidance
- REGULATION (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016, on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)- Applies from May 2018
- Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data, OJ L 281, 31–50 (henceforth '95/46/EC' or 'the Directive').
- Working Party 29 'Opinion 216 05/2014 on Anonymisation Techniques' (2014) 5 ('WP29 216').
- Federal Act on Research involving Human Beings (Human Research Act, HRA) of 30 September 2011 (Status as of 1 January 2014)

Data collection

Data is not collected in the specific purpose of the MIP. It is collected in the course of the patient's health care or for research projects and can be further processed and shared using the MIP. Storing the data for a longer period may require to submit an extension request with the competent body in the Member-State where the Data Provider is based. In such cases, a support from the CHUV-MIP Deployment team can be provided to streamline the process.

Consent

'consent' of the data subject means any freely given, specific, informed and unambiguous indication of the data subject's wishes by which he or she, by a statement or by a clear affirmative action, signifies agreement to the processing of personal data relating to him or her;

Article 7 of the GDPR "Conditions for consent

1.Where processing is based on consent, the controller shall be able to demonstrate that the data subject has consented to processing of his or her personal data.

2.If the data subject's consent is given in the context of a written declaration which also concerns other matters, the request for consent shall be presented in a manner which is clearly distinguishable from the other matters, in an intelligible and easily accessible form, using clear and plain language. Any part of such a declaration which constitutes an infringement of this Regulation shall not be binding.

3.The data subject shall have the right to withdraw his or her consent at any time. The withdrawal of consent shall not affect the lawfulness of processing based on consent before its withdrawal. Prior to giving consent, the data subject shall be informed thereof. It shall be as easy to withdraw as to give consent.

4.When assessing whether consent is freely given, utmost account shall be taken of whether, inter alia, the performance of a contract, including the provision of a service, is conditional on consent to the processing of personal data that is not necessary for the performance of that contract.

Data Providers are responsible for obtaining the consent from their patients and / or data subject whenever appropriate.
Support and audit to make sure the informed consent form is compliant with the regulation can be provided by the MIP Deployment Team, as well as a template.

PSEUDONYMIZATION - ANONYMIZATION

Pseudonymization

According to GDPR Art. 4(c)

'pseudonymisation' means the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately and is subject to technical and organisational measures to ensure that the personal data are not attributed to an identified or identifiable natural person;

Anonymization

According to the Data Policy Manual of the HBP,

"Anonymous data: Information which does not relate to an identified or identifiable natural person."

According to the Swiss Federal Act on research involving Human Beings, "Anonymised biological material and anonymised health-related data means biological material and health-related data which cannot (without disproportionate effort) be traced to a specific person;"

The principles of data protection in GDPR does not apply to anonymous information, namely information which does not relate to an identified or identifiable natural person or to personal data rendered anonymous in such a manner that the data subject is not or no longer identifiable. This Regulation does not therefore concern the processing of such anonymous information, including for statistical or research purposes.

Application of the GDPR

The GDPR only applies to personal data or information concerning an identified or identifiable natural person. If data are anonymised, it is no longer considered to be personal and is thus outside the scope of GDPR application. In other words, if data accessible in the MIP are anonymous, the GDPR does not apply and the data can be processed for research purposes without the restrictions of data protection law. However, given the difficulty in creating truly anonymous data, the bar for anonymisation has been set extremely high under EU data protection law.

To determine whether a person is identifiable, one must consider "all the means reasonably likely to be used,

such as singling out, either by the controller or by another person, to identify the natural person directly or indirectly." To make this determination, one must consider all "objective factors, such as the costs of and the amount of time required for identification, taking into consideration the available technology at the time of the processing and technological developments."
In making this determination, one must also consider the robustness of the anonymization techniques they apply and the potential for failure of those techniques. Like the Directive, the GDPR takes a 'risk-based approach', and obligations are scalable. Therefore, GDPR anonymization obligations require both interpretation and monitoring. Applying the GDPR to the MIP will thus remain an ongoing and continual process.
The main anonymisation techniques applied in data protection law are randomisation and generalisation. Regardless of the technique applied (e.g. addition, permutation, differential privacy, aggregation, k-anonymity, l-diversity, t-closeness, etc.), three main questions should be considered:

1) Is it still possible to single out an individual?
2) Is it still possible to link records relating to an individual?
3) Can information be inferred concerning an individual?

On a general basis, data stored on the MIP local will be pseudonymised. Data stored on the MIP local (pseudonymised data) is attributable to a natural person by the use of additional information, which is securely stored using both organizational and technical security measures.

Data on the MIP federated node will be anonymized. Taking into account all the means reasonably likely to be used for identification, data subjects cannot be identified through research data available at this level.

Data Providers are responsible for the pseudonymization and the anonymization of their data, based on the requirements provided by the MIP Deployment team.

To ensure effective pseudonymization of the data, SP8 can provide technical support for the installation and configuration of the software used during SGA1 (FedEHR Gnubilla) and also the following guidelines for the pseudonymization of EHR data.

All the identifiers are removed or coded and the patient record receives a unique encrypted identifier when it is stored on the MIP Local server. The look-up table is stored on a different server in the hospital level 3 "clinical area" which is not accessible from the outside.

A complete table of the recommended fields to modify- delete- mask is available in the Rules for Anonymization document, part of the Deployment Package.

DATA PRIVACY LEVELS

Level 3 - Data stored in hospital's clinical data storage systems (EHR, PACS)

- Contains Personal Health Identifiers (PHI)
- Raw data, including full brain images that enable reconstructing the patient's face, diagnostics and longitudinal information with exact dates
- High risk of unauthorized identification
- General regulatory requirements: Cannot be shared publicly, must be protected from any un-authorized access.

MIP policy: Such data are not accessible through the MIP

Level 2 - Pseudonymised data stored in MIP local

- No Personal Health Identifiers (PHI).
- Neuroimaging data are being processed in order to deface them in the case images are shared, or to extract features such as brain volumes.
- Medium to Low (from Raw to features) risk of unauthorized re-identification: identity can be recovered from a lookup table secured and password protected in a hospital server distinct from where the pseudonymised data are stored. In hospitals, the look-up table is stored on the level 3.
- General regulatory requirements: Can be shared by authorized investigators provided ethics approval and patient's informed consent whenever appropriate, but cannot be shared publicly and must be protected from any unauthorised access.
- MIP policy: Such data will be only accessible through the MIP local by the data provider and his local authorized staff.

Level 1 - Anonymized data stored in MIP federate nodes

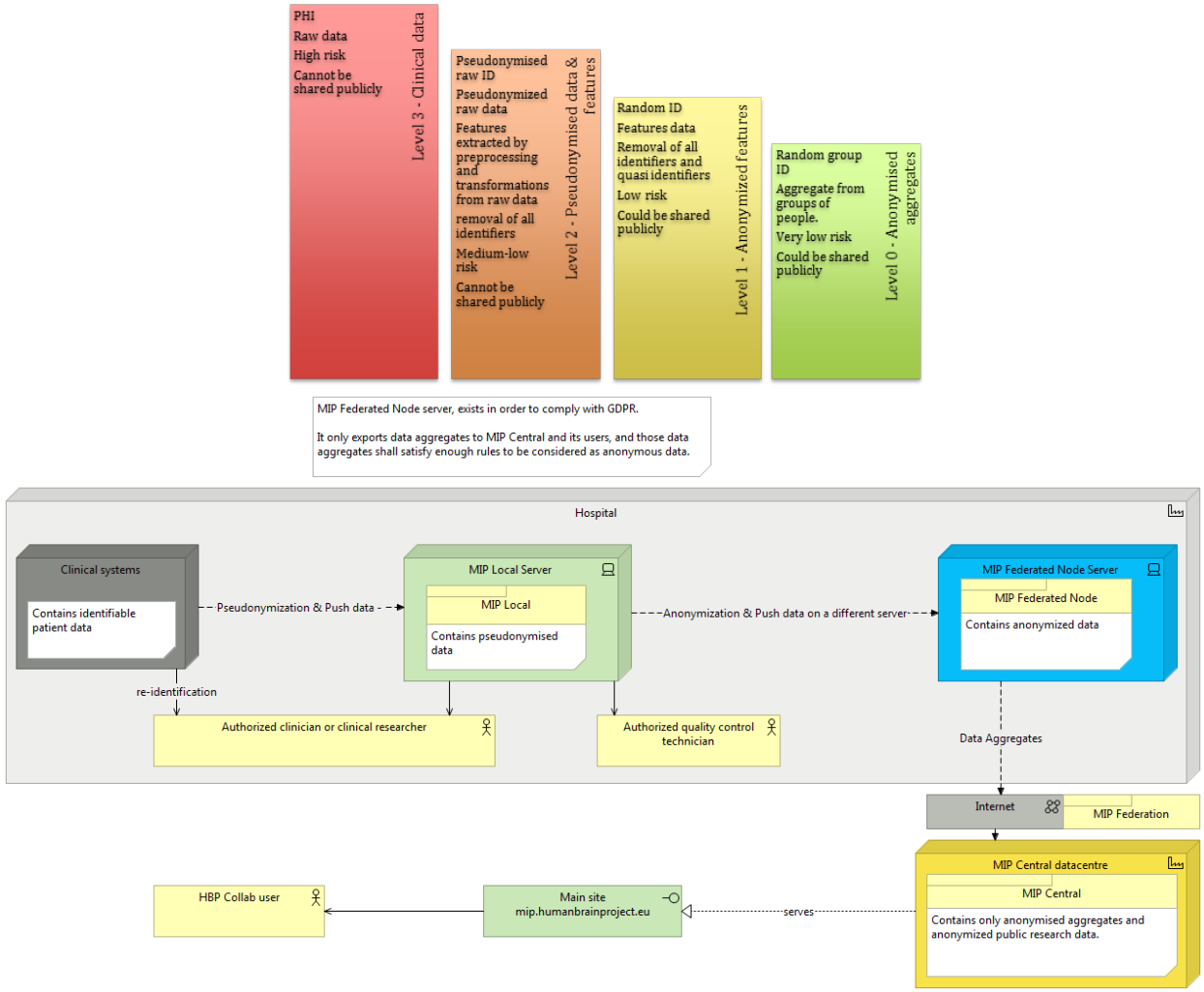
- Anonymization with no lookup table
- Only features data obtained after pre-processing of raw data (no available image that would allow reconstructing the face)
- Very low risk of unauthorized re-identification: identity cannot be recovered from a lookup table. Most features do not contain enough information to find directly or by cross-references the identity of an individual.
- General regulatory requirements: Can be shared by authorized investigators. Must be protected from any unauthorised access.

MIP policy: Such data cannot be explored at the individual level. Data are made available for aggregated queries only within the MIP federate network to investigators authorized by the MIP Data Governance Steering Committee. Will not be shared publicly, must be protected from any unauthorised access. An expert determination report assessing the strength of protection from re-identification should be provided for each dataset (similar to https://www.hipaajournal.com/de-identification-protected-health-information/)

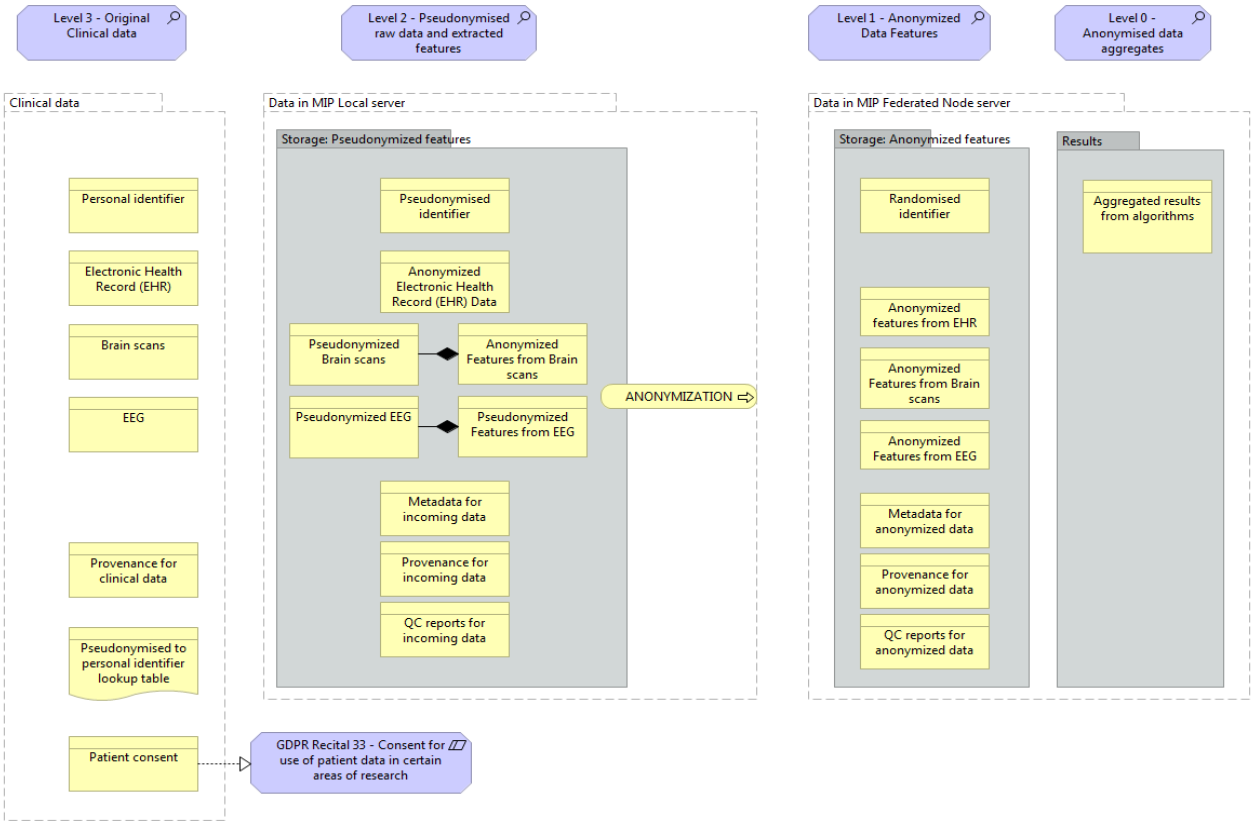
Level 0 - Anonymized data aggregates transmitted to MIP central

- Same as above with the following additional features:
- only aggregated data (minimum values are set to the algorithms to ensure there is no singling out)

MIP policy: Data will be made available for aggregated queries only to any MIP registered users.



To ensure data security, Data Providers shall allocate two different servers, one for the MIP Local and one for the MIP Federated Node. The MIP Federated Node Server will not contain any information allowing the reidentification of patients.

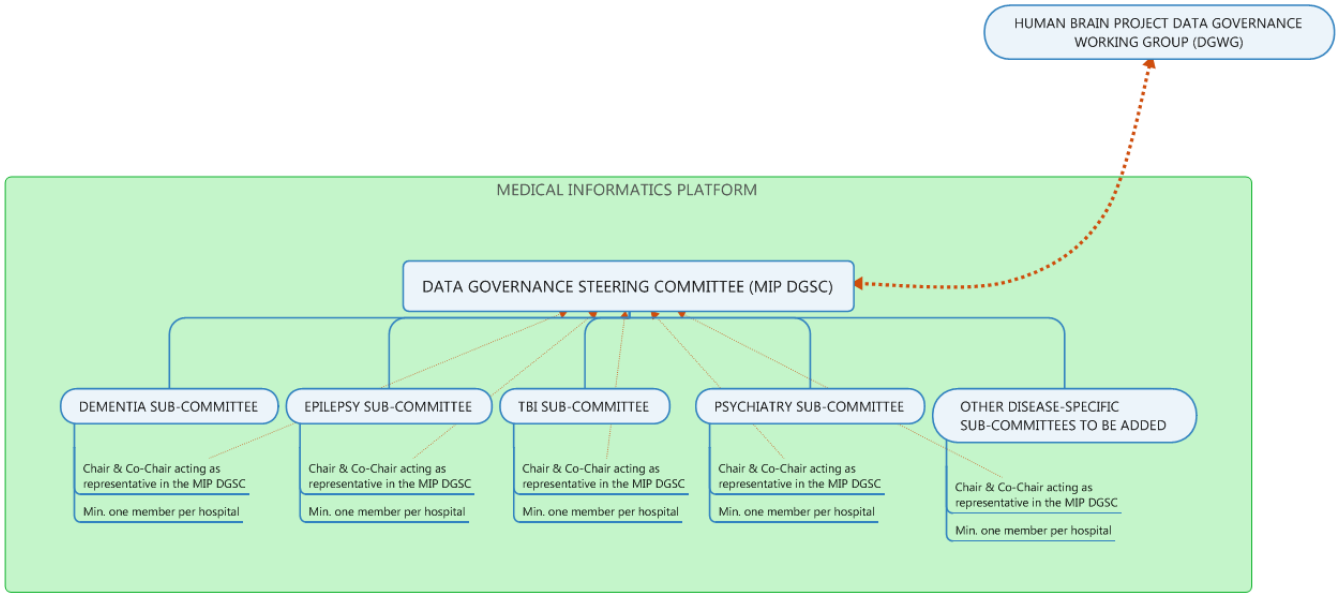


MIP Installation Agreement, MIP Data Sharing Agreement and Governance

As delineated in the Executive Summary of this Package, the Legal Entity representing the MIP is the CHUV, in Lausanne. Two agreements are signed between the CHUV and the Hospitals providing data.
- An Installation Agreement to cover the installation of the platform in the hospital, the responsibilities, the level of service provided by CHUV MIP Deployment Team and other relevant elements pertaining to the software installation
- A Data Sharing Agreement, covering the aspects of implementing and sharing the data at the MIP Federation Level.

Moreover, as stated in the Executive Summary, a MIP Data Governance Steering Commitee (MIP DGSC) is created to discuss all matters of Governance and produce the Charter for Data Sharing in the MIP and the Publication and Authorship Policy.

It is organized with 2 levels, a global MIP DGSC covering all the high level aspects, rules and guidelines, and Pathology-specific sub-committees to go deeper into each specific disease, as illustrated below.



Available Documentation

For in-depth documentation on compliance, ethics and governance issues, plsease contact Florent Gaillard.

CONTACT:

CHUV MIP Ethics and Governance Officer: Florent Gaillard, florent.gaillard@chuv.ch

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP TECHNICAL SPECIFICATIONS
&
STEP-BY-STEP SOFTWARE DEPLOYMENT GUIDE

VERSION: JANUARY 2019

Table of Contents

List of Figures

List of Tables

1. Introduction

Convergence of biology and technology and increasing capability to:

- assess pan-omic biological data of an individual, including detailed brain features (morphology, connectivity, functionality), DNA sequence analysis, proteome, metabolome, microbiome, physiome and phenome, and
- correlate biological data with sign and symptoms, observational data and diagnostic

provide the opportunity to discover new biological signatures of diseases, develop preventive strategies and improve medical treatment. Opportunities to use these data to improve health outcomes – to develop preventive strategies and improve medical care – is the motivation for the development of the Medical Informatics Platform.

The Medical Informatics Platform (MIP) provides expert tools for:

- Data processing
- Data exploration and selection, and
- Analysis of patients biological and other health-relevant data distributed across remote locations in hospitals, research institutes and universities

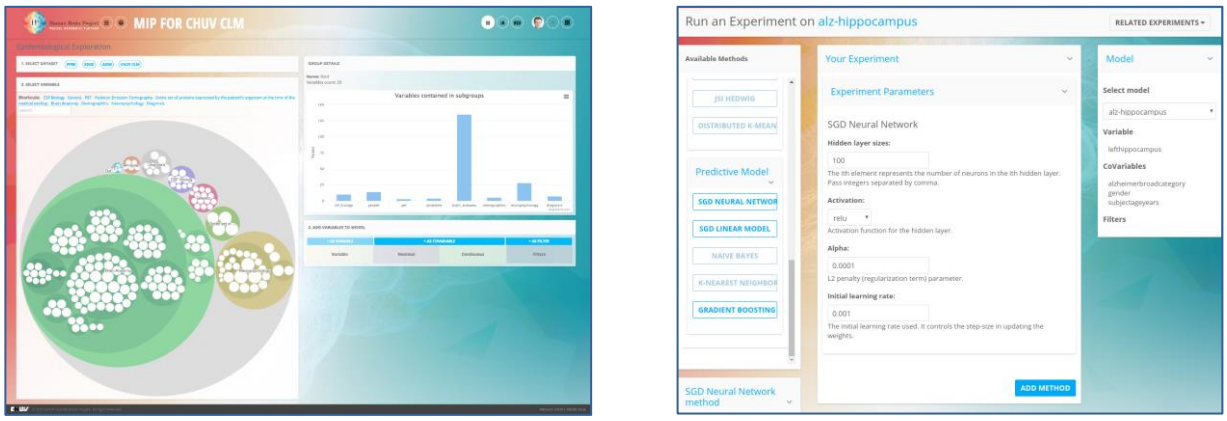


Figure 1 – Medical Informatics Platform Web Portal

The use of a large volume and variability of normalised patient data provides for more rapid advancement in understanding neurological and psychiatric diseases. This will, in turn, allow identification of the associated biological mechanisms and open possibilities for prevention, early diagnosis and personalised medicine.

The Medical Informatics Platform has three key objectives:

1) Provide analytical tools for in situ clinical data analysis and federation of results
2) Recruit hospitals, research institutes and universities to contribute to and benefit from using the platform
3) Develop tools for extracting biological signatures of diseases from multi-level pan-omic data

The MIP provides methods for federated analysis of patient data from hospitals, research centres and biobanks. Clinical scientists can develop, share and release results of their research. The MIP aims to bring together people across professional and scientific fields encourages them to actively contribute to the design and development of the services which the MIP provides.

The users of the MIP are:

- **Clinicians**, for objective diagnoses and treatment of brain disease
- **Neuroscientists**, for the application and testing of new models and methods
- **Pharmaceutical or biotech researchers**, for disease target discovery

2. Hybrid Deployment Model

The MIP is distributed, cloud-ready patient data analysis ecosystem, which connects patients' data from hospitals and research cohort datasets and provides set of pre-integrated statistical methods and predictive machine learning algorithms for patient data exploration, data modelling, integration and execution of experiments (data analysis methods), and visualisation of the results.

The Platform makes data on populations of patients broadly available for research use, by providing software-as-a-service to clinicians, neuroscientists and epidemiologists both for diagnosis and research in clinics and for collaborative neuroscience research using hospital data and open patient research cohort datasets.

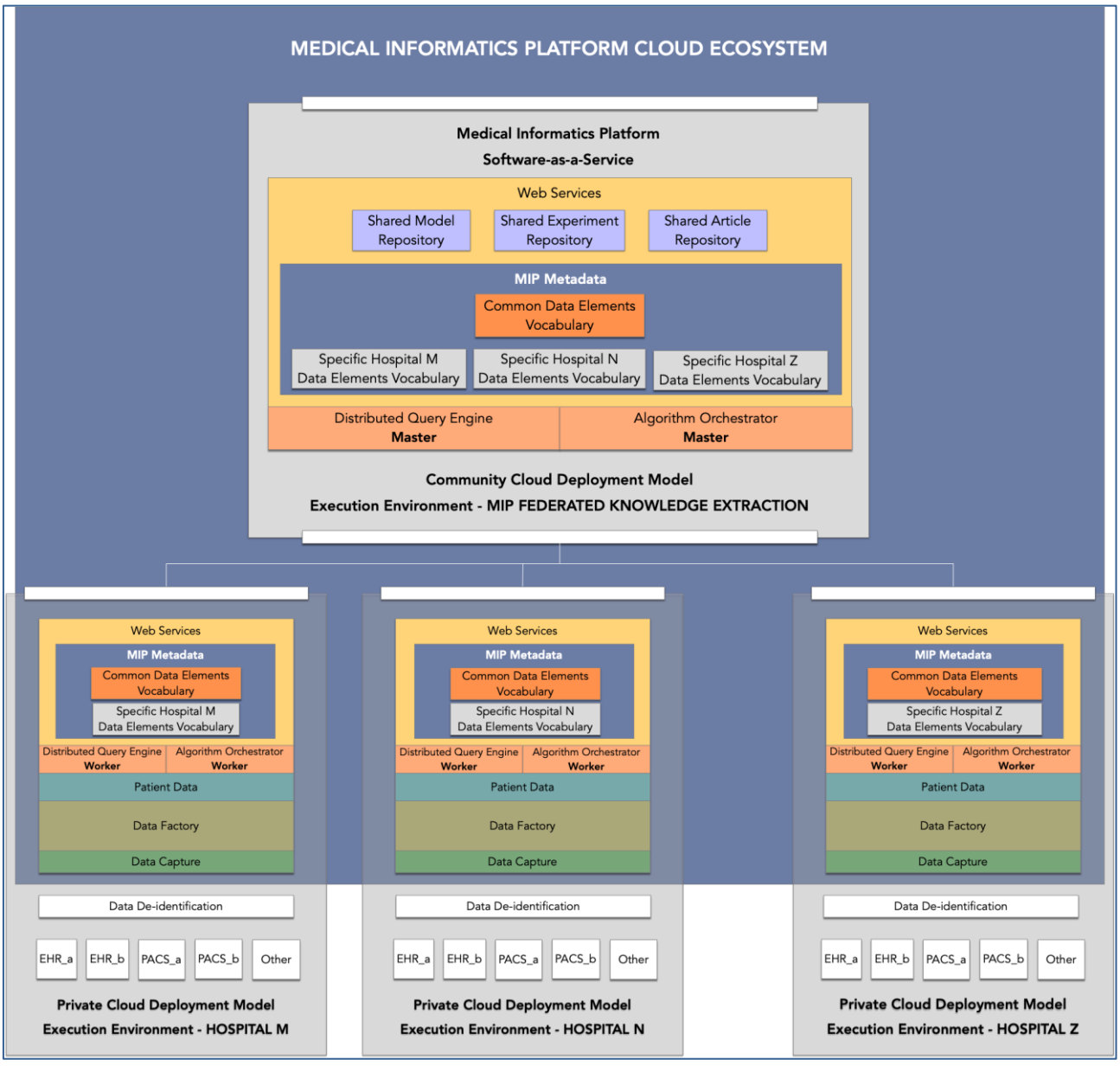


Figure 2 - Medical Informatics Platform Deployment Model

Figure 2 illustrates the cloud-ready MIP federated knowledge extraction software-as-a-service deployed in community execution environment. The MIP community execution environment provides advanced multi-dataset, cross-centre descriptive and predictive analytics. It runs software that orchestrates the execution of statistical and machine-learning algorithms in private hospital MIP execution environments and aggregates the results. The algorithms are executed locally, in private hospital environments where the de-identified patient data is stored. Master orchestrator components that are running in community execution environment, connected to the

distributed private MIP execution environments via web services, fetch the aggregate results of the algorithms executed in the private execution environments and aggregate them in a cross-centre data analysis result.

The MIP is engineered with a privacy by design approach. De-identified patient data stored in private hospital's execution environments are accessible only locally, either by the algorithms running there or by other means of data exploration within the private execution environment, using the locally deployed web services.

Users of the MIP can access the community execution environment or the local private hospital execution environment through the MIP web portal (Figure 1). The MIP web applications allow for the statistical/aggregated (not individual) data exploration, selection of data types for analytics, execution of algorithms/experiments and visualisation of the results.

The hybrid community and private hospital deployment model, microservice architecture coupled with continuous integration and continuous deployment technology, distributed hospital patient data storage and federated algorithm execution are software architecture-related prerequisites to having a cross-centre data analytics.

This distributed, patient privacy preserving software architecture is a necessary but not a sufficient condition to having multi-dataset clinical studies. Hospital datasets have overlapping data types but different ontological representations. Data is described, stored and formatted in different data structures. For executing a multi-dataset analytics, data models need to be harmonised in a common MIP data model, which is shared and synchronised between the distributed private hospital instances and community execution environment.

The data model harmonisation is, therefore, a key technology enabler for cross-centre multiple dataset clinical studies. It is a well-defined process supported by the workflow orchestration, application ontology software architecture, and the organisation, which establishes and maintains the rules and controls the quality and the integrity of the data harmonisation process.

Data governance and data selection (DGDS) committee is a centrally coordinated MIP organisational entity responsible for establishing and maintaining data governance methodology and data harmonisation rules. The members of the DGDS committee are MIP software architects and data managers from the MIP R&D team, supported by the expert medical/scientific committee consisting of the medical doctors, clinical researchers and data managers of the participating centres.

Data harmonisation and re-harmonisation is an on-going process. With the introduction of a new dataset, the whole process has to be repeated, starting with the analysis of the incoming dataset ending with the synchronisation of (re-)harmonised data models across the distributed MIP ecosystem.

The distributed, privacy-preserving MIP software deployed across hospitals and institutes using a hybrid community-private deployment model with centralised orchestration of statistical inference and machine learning algorithms, and managed harmonisation and synchronisation of the data model provide IT prerequisites for execution of cross-centre, multi-dataset clinical studies across the participating centres.

For example, using the unsupervised machine learning on patients' data in one or several centres to train a classifier which differentiates between the frontotemporal dementia and Alzheimer's disease, then applying learned classifier to patients' data in other participating centre for a differential diagnosis between the two neurodegenerative disorders. Or, using the clinical and pathological data of deceased patients from available datasets to train a machine-learning model that can be used to predict the disease progression with patients in other hospitals.

A centre hosting the MIP has to provide two servers, install required middleware and set up the network ensuring the adequate level of security. The MIP is designed using privacy-by-design approach and it contributes to data protection by allowing a remote access only to irreversibly anonymised patients' data. For maximising data security, it is data centre responsibility to design, implement and operate an appropriate data centre networking.

3. Hardware Requirements

MIP requires configuration and installation of the following two nodes in a participating centre's private execution environment, each on a single server or on a cluster of servers:

- **Local MIP Node** – a server designed for working with pseudonymised clinical data
- **Local MIP Federation Node** – a server designed for working with irreversibly anonymised data

The first server, a local MIP node, is an instance of the MIP containing:

- Pseudonymised patient data capture sub-system
- Data factory sub-system with pipelines for patient data and MRI processing and normalisation
- Feature data store for storing normalised pseudonymised patient biological and other health-relevant features
- Knowledge extraction sub-system with distributed query engine, algorithm factory and algorithm orchestrator
- Web-based front-end used by authorised clinicians to perform local data analysis

The second server, a local MIP federation node, is an instance of the MIP designed to secure full patients' privacy by allowing a remote access only to irreversibly anonymised data. It receives normalised and irreversibly anonymised patient data from the local MIP node, does not perform heavy data pre-processing, but may execute computationally intensive machine learning algorithms.

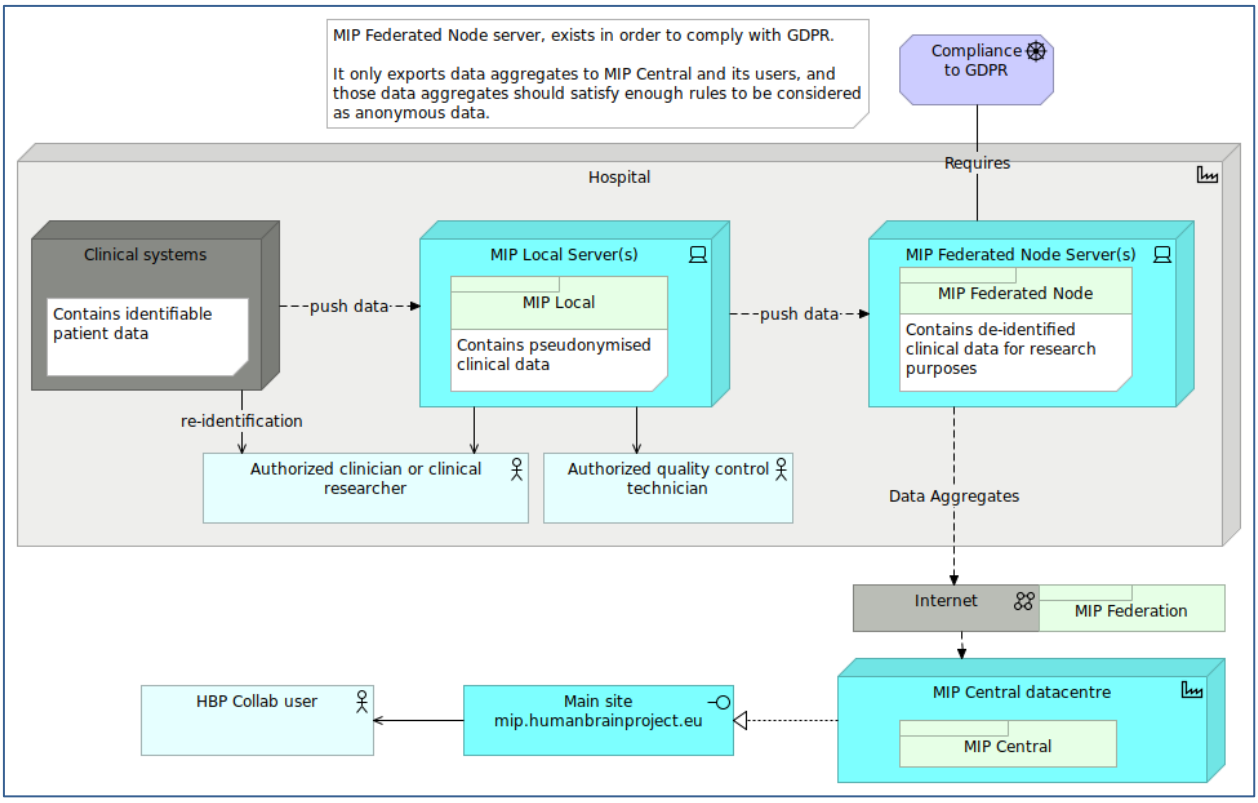


Figure 3 - MIP Deployment in a Private (Hospital) Execution Environment

The MIP does not provide any feature for identifying individual patients. It is the responsibility of the participating centre to secure a look up mechanism alongside the local MIP node for mapping patient pseudonymes with their real identifiers.

Table 1 - Hardware Requirements Specification

MIP Hardware Requirements

MIP Node can be installed on a single server of on a cluster of servers

Local MIP Node

The following scenarios are supported:

- **On-site pre-processing of MRI scans or EEGs**
  (backlog of up to 500 patients, daily processing of about a dozen patients)
  - **One 'Large Processing Server' or at least two 'Medium Processing Servers'**
- **No pre-processing of MRI scans or EEGs or in the low hundreds of patients**
  - **One 'Medium Processing Server'**

The exact specifications may change depending on the volume and nature of pre-processing operations. To support additional computational requirements, it is always possible to add more processing nodes in the cluster

Name	CPU	Memory	Disk	Network Zone
Large Processing Server	x64 12 – 16 cores	32GB+	**Local Disk**: 200 GB **Shared Disk**: 11 times the expected size of the provided imaging data	Research Network
Medium Processing Server	x64 8 cores	16GB+	**Local Disk**: 200 GB **Shared Disk**: 11 times the expected size of the provided imaging data	Research Network

Local MIP Federation Node

This node received de-identified and pre-processed data from MIP Local, it does not perform any heavy data pre-processing but may execute computationally intensive machine learning algorithms.

The exact specifications may change depending on the volume of data and nature of machine learning algorithms. To support additional computational requirements, it is always possible to add more processing nodes in the cluster. GPU is not supported at the moment.

Name	CPU	Memory	Disk	Network Zone
Medium Analytics Server	x64 8 cores	16GB+	**Local Disk**: 500 GB	Research Network or DMZ

4. Software Requirements

The local MIP node runs on Linux operating system supporting Docker technology. Installation of the software components specified in Table 2 are required for automatic deployment of MIP software packaged in Docker images using Ansible installation script.

Table 2 - Software Requirements Specification

MIP Software Requirements		
Local MIP Node		
Operating System	**Connectivity**	**Licenses**
Ubuntu 16.04 LTS RedHat Enterprise Linux 7.2+ CentOS 7.2+	OpenSSH	MATLAB 2016b
Local MIP Federation Node		
Operating System	**Connectivity**	**Licenses**
Ubuntu 16.04 LTS RedHat Enterprise Linux 7.2+ CentOS 7.2+	OpenSSH	N/A

MATLAB 2016b license is needed by the Statistical Parametric Mapping SPM12 tool, a component orchestrated by the neuromorphometric processing pipeline situated in the data factory sub-system of the local MIP node. Local MIP federation node is not used for neurommorphometric data processing; hence it does not need MATLAB license.

5. Connectivity Settings

Centre that hosts MIP nodes have to provide two servers, configure the adequate operating system and connectivity software as well as to setup network connectivity and provide an adequate level of security.

5.1 Security Recommendation

Local MIP node local MIP federation nodes should not be placed in the clinical network security zone. Any connectivity from the local MIP and local MIP federation nodes through the firewalls into the clinical network security zone should be prevented.

It is also recommended not to place local MIP federation node in the network security zone of the local MIP node. Any connectivity from the local MIP federation node, through the firewalls, into the security zone of the local MIP node should be prevented.

5.2 Network Service and Connectivity

Configuration of the network services and opening of L3 ports specified in Table 3 and Figure 4 is the last step of private execution environment preparation for installation of local MIP and local MIP federation nodes.

Table 3 - Network Service and Connectivity Requirements

MIP Network Service and Connectivity Requirements		
Local MIP Node		
Subnet	Configure a dedicated sub-network for the node	
DNS	Addressable from the Internet for the server, incl. web portal	
Firewall	Prevent access from the node to the clinical network security zone Allow outgoing connections to other services on Internet (see chapter 5.3) Allow outgoing connections to local MIP federation node Allow incoming connections to the ports below (see ingress connectivity)	
Ingress Connectivity		
Port #	**Protocol**	**Traffic Type**
22	**SSH**	Pushing data from the clinical systems to local MIP node Remote administrative access for the CHUV HBP MIP team
80	**HTTP**	Used by Let's Encrypt CA to setup its SSL certificate Used by MIP for health tests
443	**HTTPS**	MIP Web Portal Connections only from locally authorised users
Local MIP Federation Node		
Subnet	Configure a dedicated sub-network for local MIP federation node	
DNS	Addressable from the Internet for the server, incl. web portal	
Firewall	Prevent access from the node to the clinical network security zone Prevent access from the node to the local MIP node's network zone Allow outgoing connections to other services on Internet (see chapter 5.3) Allow incoming connections from the central MIP server to the ports below	
Ingress Connectivity		
Port #	**Protocol**	**Traffic Type**
22	**SSH**	Pushing data from local MIP node to local MIP federation node Remote administrative access for the CHUV HBP MIP team
80	**HTTP**	Used by Let's Encrypt CA to setup its SSL certificate Used by MIP for health tests
443	**HTTPS**	MIP federated API Connections only from local users and the central MIP server

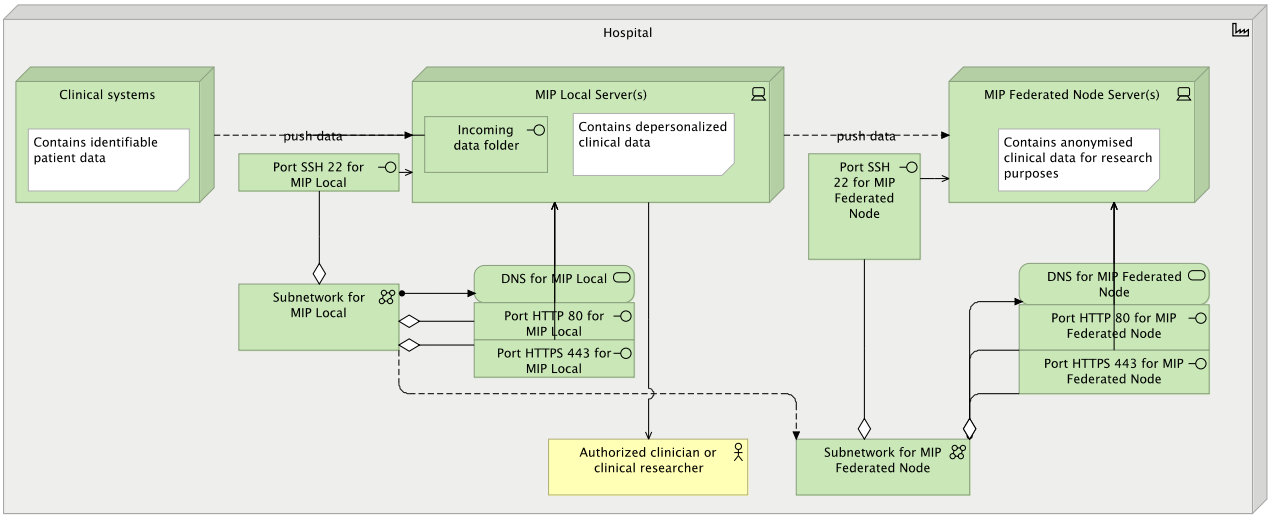


Figure 4 - Private Execution Environment Connectivity Diagram

5.3 External Services

The external services that must be accessible from any MIP dedicated machines for installation, configuration, update, and maintenance purposes are listed in Table 4.

Table 4 - External Service Specification

Name	Description	IP/Hostname	L3P>PORT>L7P>IN/OUT	Required by
Remote Access SSH	Only inbound service required. It is mainly used by our automated deployment scripts and by our deployment and support team to connect the machine using the provided VPN access.	Inside Clinical Network (VPN)	**TCP>22>SSH>IN**	**Local MIP**
Web Portal	Main entry-point for MIP local users	Inside Clinical Network	**TCP>80>HTTP>IN**	**Local MIP**
Ubuntu French APT Servers	Ubuntu France's official APT server	fr.archive.ubuntu.co security.ubuntu.com	**TCP>443>HTTPS>OUT**	**Local MIP on Ubuntu**
Docker APT	Docker's official APT server	download.docker.com	**TCP>443>HTTPS>OUT**	**Local MIP on Ubuntu**

Name	Description	IP/Hostname	L3P>PORT>L7P>IN/OUT	Required by
Docker YUM	Docker's official YUM server	yum.dockerproject.org	**TCP>443>HTTPS>OUT**	**Local MIP on RHEL**
Launchpad	Binary repository containing Ansible up-to-date versions	launchpad.net	**TCP>443>HTTPS>OUT**	**Local MIP**
Mesosphere APT	Mesosphere's official APT server	repos.mesosphere.com	**TCP>443>HTTPS>OUT**	**Local MIP on Ubuntu**
PyPI	Python package repository	pypi.python.org	**TCP>443>HTTPS>OUT**	**Local MIP**
Docker Hub	Docker Hub (To be replaced by our own private Docker registry)	hub.docker.com	**TCP>443>HTTPS>OUT**	**Local MIP**
MATLAB License Server	Required only if the institution uses its own MATLAB licence server		**TBD**	**Local MIP**
GitHub	Main source repository for SP8's software	github.com	**TCP>443>HTTPS>OUT**	**Local MIP**
Bitbucket	Private source repository for deployments configurations	bitbucket.org	**TCP>443>HTTPS>OUT**	**Local MIP**
CHUV Server	Private git repository/Docker registry	hbps1.chuv.ch	**TCP>443>HTTPS>OUT**	**Local MIP**

6. Software Deployment Package

The MIP is deployed on Mesos stack with added support for automated deployment/upgrade of services managed by Mesos Marathon and hardened security of the Ubuntu operating system. The services are built using Ansible scripts, unifying operation system configuration, middleware and application software deployment. Automated installation and configuration of MIP software on bare metal or preconfigured virtual machines supports clustering, security and monitoring.

HBP MIP deployment software package consists of third-party (Table 5) and HBP MIP software (Table 6). Participating centres are required to install them on local MIP servers.

Table 5 - Third-party Software

Name	Level	License	Deployment	Required by
OpenSSH	Infrastructure		Native	Local MIP
Java Open JDK SE	Infrastructure		Native/Container	Local MIP
Python2.7	Infrastructure		Native	Local MIP
Python3	Infrastructure		Native/Container	Local MIP
R	Infrastructure		Container	Local MIP
Docker	Infrastructure		Native	Local MIP
Mesos	Infrastructure		Native	Local MIP
Zookeeper	Infrastructure		Native	Local MIP
Chronos	Infrastructure		Container	Local MIP
Marathon	Infrastructure		Native	Local MIP
Git	Infrastructure		Native	Local MIP
docker_py	Infrastructure		Native	Local MIP
python-simplejson	Infrastructure		Native	Local MIP
Træfik	Infrastructure		Native/Container	Federated MIP
Consul	Monitoring/Security		Native/Container	Federated MIP
Logwatch	Monitoring/Security		Native	Local MIP
fail2ban	Monitoring/Security		Native	Local MIP
ufw	Monitoring/Security		Native	Local MIP
PostgreSQL	Everywhere		Container	Local MIP
slackclient_py	Data Factory		Native	Optional
Airflow	Data Factory		Native	Local MIP

Name	Level	License	Deployment	Required by
MATLAB	Data Factory	Proprietary	Native	Local MIP
SPM12	Data Factory		Native	Local MIP
Spring Framework	Web Portal		Container	Local MIP
Flyway	Web Portal		Container	Local MIP
Nginx	Everywhere		Container	Local MIP

Software packages listed in Table 6 have been developed by the HBP MIP development teams in the scope of the Human Brain Project's sub-project 8.

Table 6 - HBP MIP Software

Name	Level	License	Deployment	Required by
data-tracking	Data Factory		Container	Local MIP
i2b2-import	Data Factory		Container	Local MIP
i2b2-setup	Data Factory		Container	Local MIP
data-catalog-setup	Data Factory		Container	Local MIP
hierarchizer	Data Factory		Container	Local MIP
mri-preprocessing-pipeline	Data Factory		Container	Local MIP
airflow-imaging-plugins	Data Factory		Container	Local MIP
data-factory-airflow-dags	Data Factory		Container	Local MIP
MIPMap	Hospital Database Bundle		Container	Local MIP
PostgresRAW	Hospital Database Bundle		Container	Local MIP

Name	Level	License	Deployment	Required by
PostgresRAW-UI	Hospital Database Bundle		Container	Local MIP
Exareme	Hospital Database Bundle		Container	MIP Federated
woken	Algorithm Factory		Container	Local MIP
base-docker-images	Algorithm Factory		Container	Local MIP
python-base-docker-images	Algorithm Factory		Container	Local MIP
functions-repository	Algorithm Factory		Container	Local MIP
Label Propagation Framework	Algorithm Library		Container	MIP Federated
Exareme mip-algorithms	Algorithm Library		Container	MIP Federated
hbpjdbcconnect	Algorithm Library		Container	Local MIP
hbplregress	Algorithm Library		Container	Local MIP
hbpsummarystats	Algorithm Library		Container	Local MIP
CCC	Algorithm Library		Container	Federated MIP
jsi-functions	Algorithm Library		Container	Federated MIP
bhtsne	Algorithm Library		Container	Federated MIP
Rtsne	Algorithm Library		Container	Federated MIP
portal-backend	Web Portal		Container	MIP-Local
portal-frontend	Web Portal		Container	MIP-Local

7. MIP Software Deployment

This chapter provides a step-by-step guide for the MIP software deployment. The steps describMed in the GitHub project documentation are described in somewhat more details The software deployment scripts referred here are available on the GitHub at:

https://github.com/HBPMedical/mip-microservices-infrastructure.

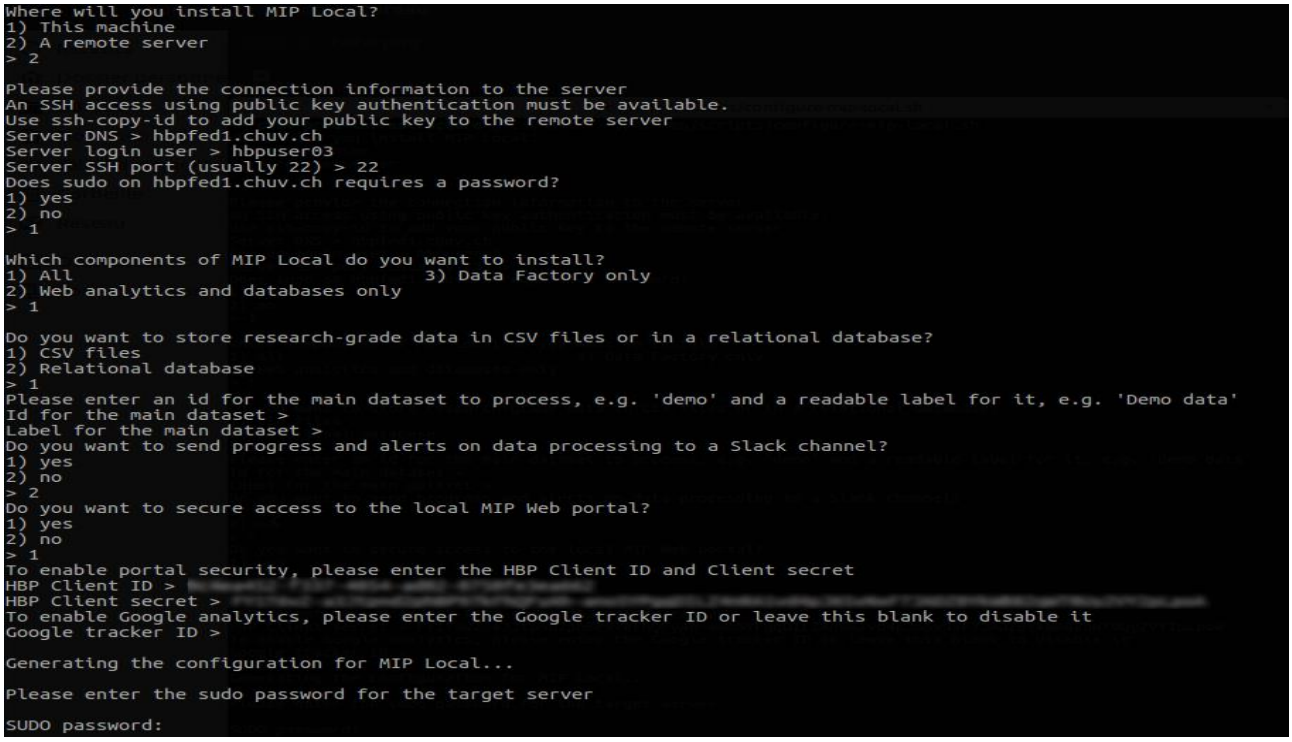
7.1 Step 1 – Clone the Project

Create git project based on the mip-microservices-infrastructure project v2.5.3 (abb98e9).

7.2 Step 2 – Generate a Software Configuration

Run the following configuration script to generated some configuration files:

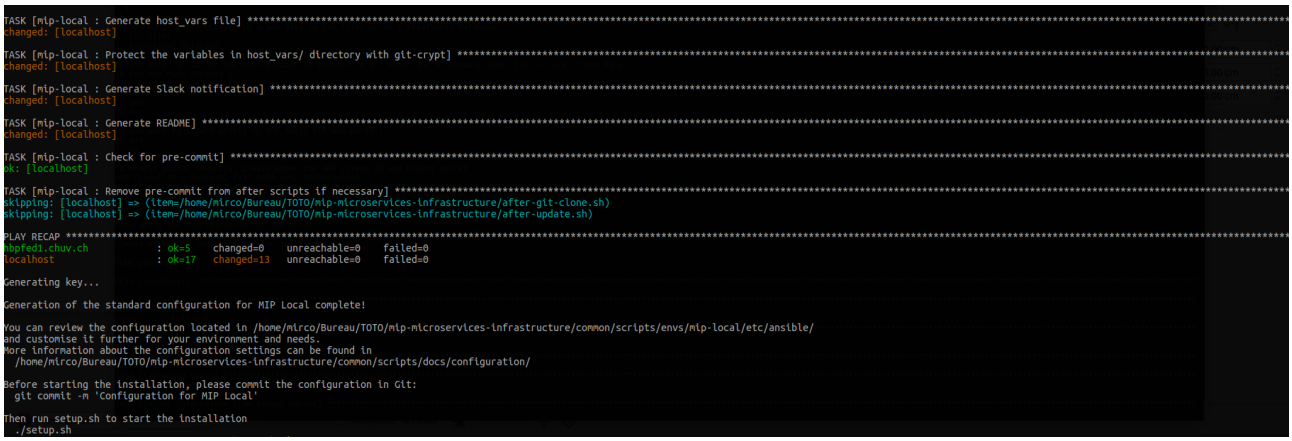
```
./common/scripts/configure-mip-local.sh
```



Note that if you enable the portal security (in order to use the HBP credentials to log in to the MIP), you'll be asked to provide HBP Client ID and a HBP Client secret. Those values must be provided by the LREN-CHUV team.

In order to install the research datasets, you'll have to provide us with your GitLab account so that we'll be able to invite you to join those datasets projects as a guest. We can provide you with a generic account if needed.

If you don't have a valid GPG key, the script will automatically ask you to create one.



At the end of this step, newly created configuration files (and a few files that might have been updated) should be staged and git-crypt should have been used to encrypt the files containing secret information (please check it). You are invited to manually check and update the configuration files and commit the changes.

A specific DNS server from within the portal-backend Docker container had to be used in this step-by-step guide. Therefore, the following was added to the configuration:

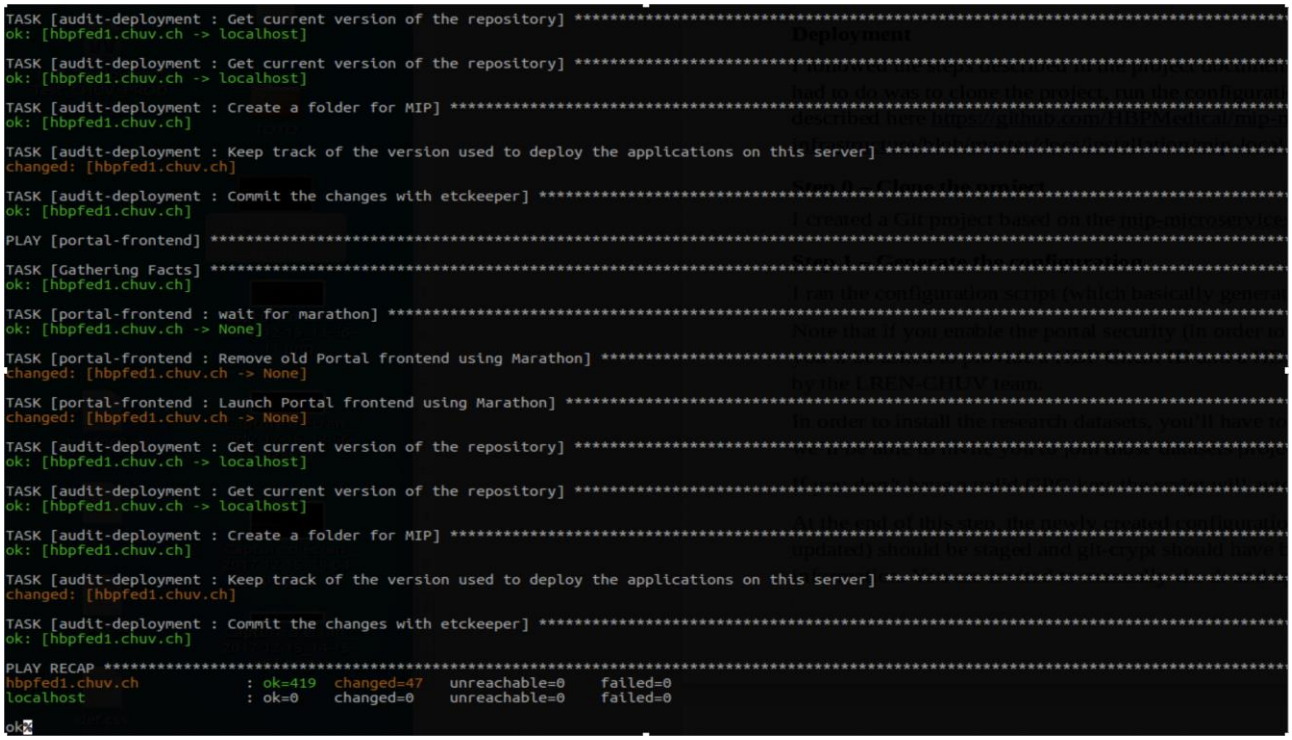
```
diff --git a/envs/mip-local/etc/ansible/group_vars/infrastructure b/envs/mip-local/etc/ansible/group_vars/infrastructure
index e9423b8..55a728b 100644
--- a/envs/mip-local/etc/ansible/group_vars/infrastructure
+++ b/envs/mip-local/etc/ansible/group_vars/infrastructure
@@ -1,5 +1,13 @@
 ---
-# Nothing defined here
+# Overlay storage should help avoiding freezing Docker containers with a process that exits with error
+# See https://github.com/docker/docker/issues/12738
+# Also a kernel issue seems to be the cause
+# See https://github.com/docker/docker/issues/18180#issuecomment-167042078
+docker_options:
+  - "--dns 155.105.251.102"
+  - "--dns 155.105.251.86"
+  - "--log-driver=journald"
+  - "--disable-legacy-registry"
+  - "--storage-driver=overlay"
 ...
diff --git a/envs/mip-local/etc/ansible/host_vars/hbpfed1.chuv.ch b/envs/mip-local/etc/ansible/host_vars/hbpfed1.chuv.ch
index 16ec884..5cd3a5b 100644
--- a/envs/mip-local/etc/ansible/host_vars/hbpfed1.chuv.ch
+++ b/envs/mip-local/etc/ansible/host_vars/hbpfed1.chuv.ch
@@ -5,6 +5,9 @@
 # is safe to store secrets such as passwords in it and then commit the file in a
 # Git repository

+docker_root_dir: '/var/lib/docker'
+docker_opts: "--storage-driver=overlay2 --graph={{ docker_root_dir }} --dns 155.105.251.102 --dns 155.105.251.86"
+
 ldsm_db_password: "Hph1lVROV7htG55"
 ldsm_db_admin_password: "bD6b9lbhT8bAcBh"

(END)
```

7.3 Step 3 – Install the Software

Now that the configuration files are created, the installation script can be executed. If a MATLAB license is not available yet, add the '--skip-tags=spm' parameter to the command:

```
./setup.sh –skip-tags=spm
```

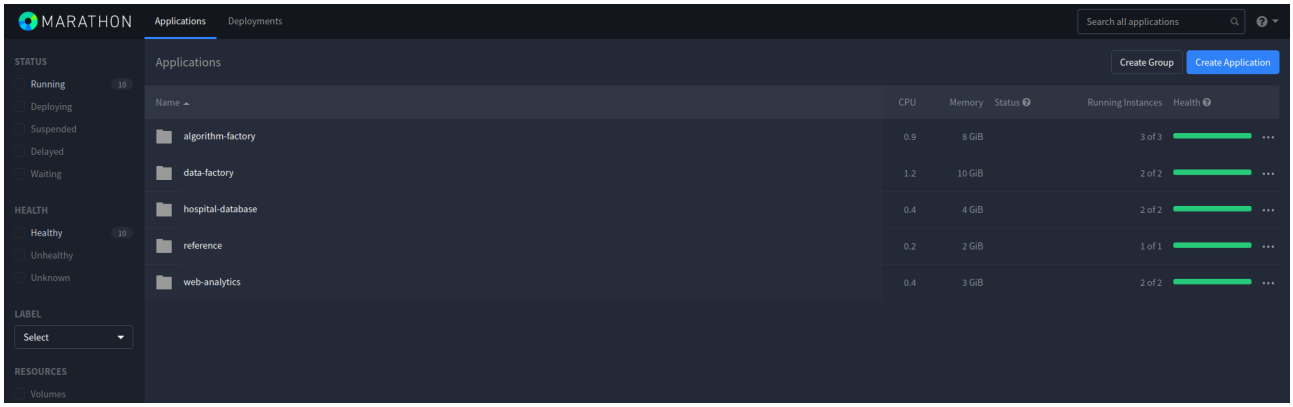
IMPORTANT: At some point, the installation process might stop, requiring to re-run the installation script. This is NOT a problem! As indicated, setup.sh script needs just to be re-launched.



The script was executed with a success. After that, the MIP is deployed.

7.4 Step 4 – Technical Verification

In order to check that all the components are correctly running, use the Marathon UI at http://hbpfed1.chuv.ch:5080:



As you can see on the right side (green indicators), all the components are working OK.

8. References

[1] SP8 Medical Informatics Platform – Architecture and Deployment Plan
D. Milovanovic & E. Miquel Fernandez, http://bit.ly/HBP_MIP_SystemDescription

[2] Deployment Requirements
L. Claude, https://hbpmedical.github.io/deployment/

[3] External Services
L. Claude, https://hbpmedical.github.io/deployment/services/

[4] MIP Software
L.Claude, https://hbpmedical.github.io/deployment/software/

[5] SP8 Medical Informatics Platform – System Validation Plan at end of SGA1
F. Kherif & D. Milovanovic, http://bit.ly/HBP_MIP_SystemValidation

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP SYSTEM DESCRIPTION

VERSION: JANUARY 2019

Table of Contents

List of Figures

List of Tables

1. Introduction

Convergence of biology and technology and the increasing capabilities to perform comprehensive "omic" assessments of an individual, including detailed brain features (morphology, connectivity, functionality), DNA sequence analysis, proteome, metabolome, microbiome, autoantibodies, physiome, phenome, etc., provide opportunities to discover new biological signatures of diseases, develop preventive strategies and improve medical treatments. Opportunities to use these data to improve health outcomes – to develop preventive strategies and improve medical care – are the motivation for the development of the Medical Informatics Platform (MIP).

The MIP is a cloud-ready patient data analysis ecosystem, which connects patient data from hospitals and research cohort datasets and provides a set of pre-integrated statistical methods and predictive machine learning algorithms for patient data exploration, data modelling, integration and execution of experiments (data analysis methods), and visualisation of the results.

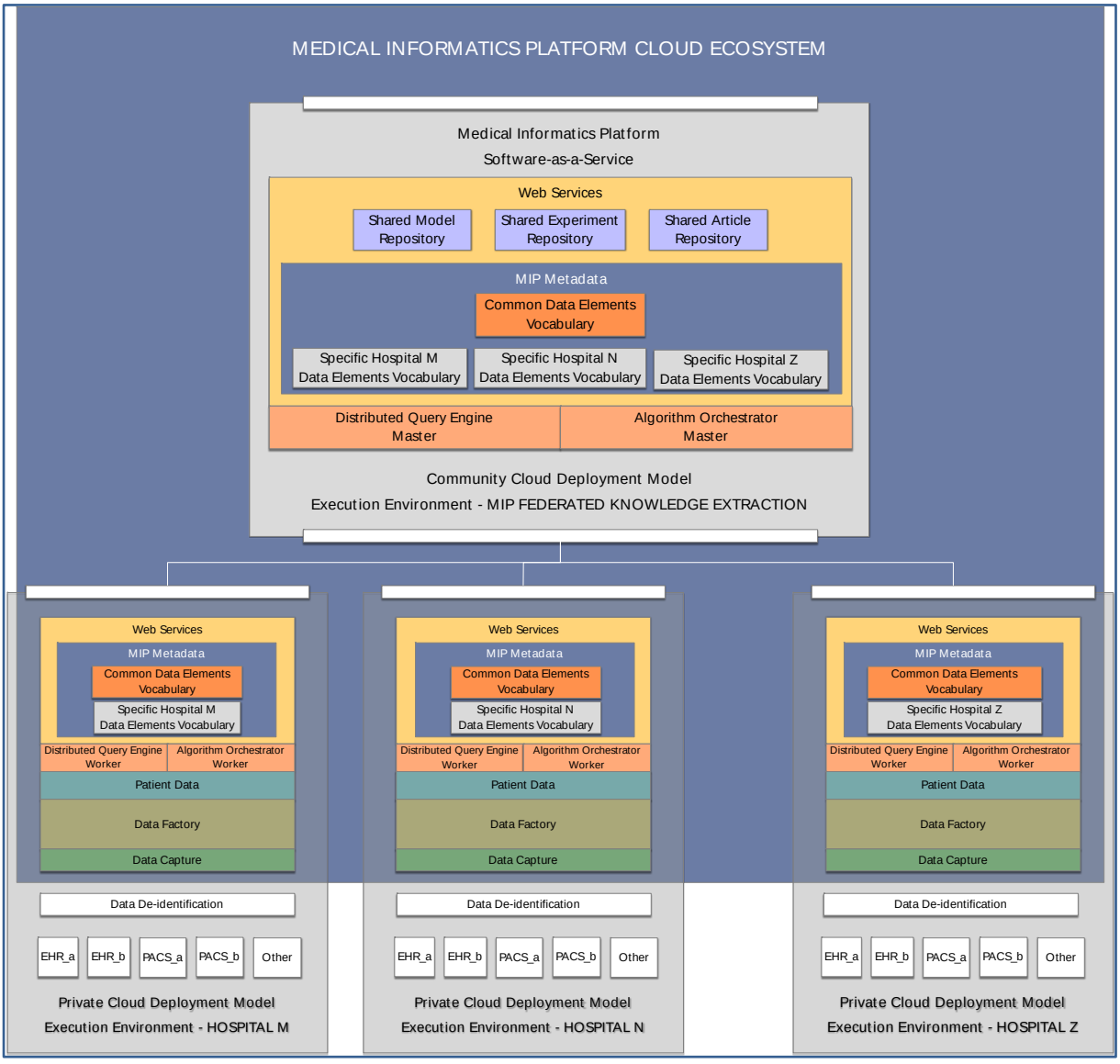


Figure 1: Medical Informatics Platform Architecture

The platform, developed during the SGA1 project phase, makes data on populations of patients broadly available for research use, by providing software-as-a-service to clinicians, neuroscientists and epidemiologists, for diagnosis and research in clinics, and for collaborative neuroscience research using hospital data and open patient research cohort datasets.

Figure 1 illustrates the cloud-ready MIP federated knowledge extraction software-as-a-service deployed in a community execution environment. It provides centralised access to the Medical Informatics Platform software and data deployed in private hospital execution environments. The MIP community execution environment orchestrates the execution of statistical and machine-learning algorithms for advanced multi-datasets, cross-centre descriptive, and predictive analytics and federates the results. The algorithms are executed locally, in private hospital execution environments where patient de-identified data is stored. Master orchestrator components that are running in community execution environment, connected to the distributed private MIP execution environments via web services, fetch the aggregated results of the algorithms executed in the private execution environments and aggregate them into a cross-centre data analysis result.

The MIP is engineered according to the privacy by design principle. De-identified patient data stored in private hospital execution environments are accessible only locally, either by the algorithms running there or by other means of data exploration within the private cloud using the locally deployed web services.

MIP users can access a community execution environment or a local private hospital execution environment through the MIP web portal. The MIP web applications allow statistical/aggregated (not individual) data exploration, selection of data types for analytics, execution of algorithms/experiments and visualisation of results. Figure 2 illustrates one instance of the web portal for the local execution environment in the University Hospital in Lausanne (CHUV), Switzerland.

Figure 2: Medical Informatics Platform Web Portal

2. Use Case Model

This section provides an overview of the Medical Informatics Platform use case model. Platform operational capabilities and user needs are formally defined using a use case modelling approach.

MIP use cases are identified using the traditional use case modelling approach: each use case specifies a complete functional unit, i.e. it handles the entire process, from its initiation by an external actor until it has performed the requested functionality. A use case always delivers some value to an actor.

There is a conceptual difference between MIP use cases and MIP use scenarios. MIP use cases represent a set of complete functions of the system, such as Data Exploration, Testing Correlation, Clinical Validity Assessment, etc. MIP use scenarios represent the workflows of the MIP functionalities to achieve a final result. Therefore, workflows of the MIP user scenarios, such as Measuring Clinical Utility of the Volumes of Medial Temporal Lobe Subregions for Diagnosing Alzheimer's Disease, consist of number of different MIP use cases used in a certain order and with a defined purpose.

This chapter gives an overview of the MIP use cases. Examples of MIP use scenarios are provided in SGA1 Deliverables D8.6.3 and D8.6.4.

2.1 Software Installation

The objective of this use case is to configure and install the Medical Informatics Platform software in a hospital's data centre.

The MIP microservices deployment architecture enables agile continuous integration and continuous component deployment developed or modified by different European-wide teams. This architecture enables efficient future upgrades of the platform with new technologies and new features needed to support evolved clinical needs. Automation of configuration and installation of the MIP software minimises IT efforts to keep the maximum focus on the scientific and clinical aspects of the projects.

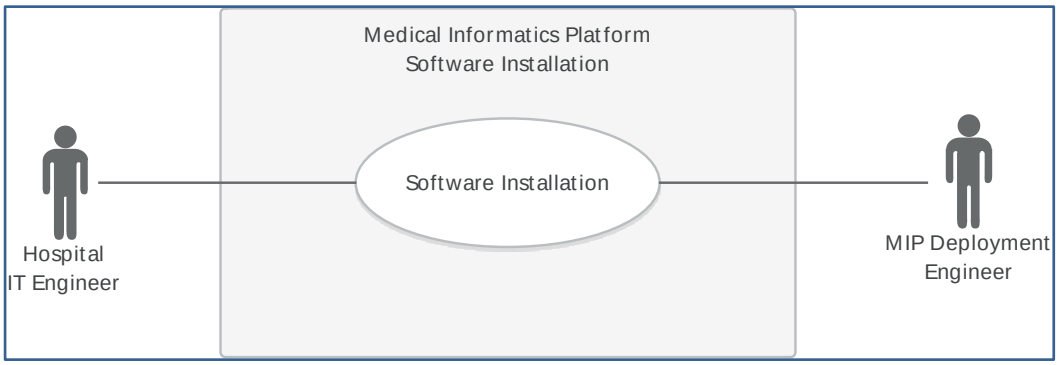


Figure 3: MIP Software Installation Use Case

Scientific Added Value

Hospital's data centre has a centralised platform for processing, storing and analysing de-identified and harmonised neuroimaging, neuropsychological, biological and demographic data of its patient population.

Efficient, configurable and automated end-to-end software installation, unifying operation system configuration, middleware installation and microservice building minimises the IT efforts to keep the focus on using the MIP platform for the scientific and clinical activities

2.2 Data Factory

The objective of the Data Factory use case group is to process patient data from different sources – hospitals and open research cohort datasets, EHR and PACS systems for:

1) Extraction of individual patient biomedical and health-related features
2) Transformation of source patient biomedical and health-related features to harmonised data structure and data vocabulary
3) Loading of transformed source datasets to permanent harmonised feature data store for federated multi-centre multi-dataset analytics

Patient source data from both hospitals and open research cohorts is typically structured and organised to capture the type and time of clinical observations, the type, modality, time and results of workups as well as the diagnoses. The Medical Informatics Platform is processing de-identified patient source data to extract biomedical and other health-related patient features, i.e. neuromorphometric, cognitive, biological, genetic, molecular and demographic, harmonises the extracted features across the different data sources, and permanently stores harmonised features for multi-centre, multi dataset clinical research studies.

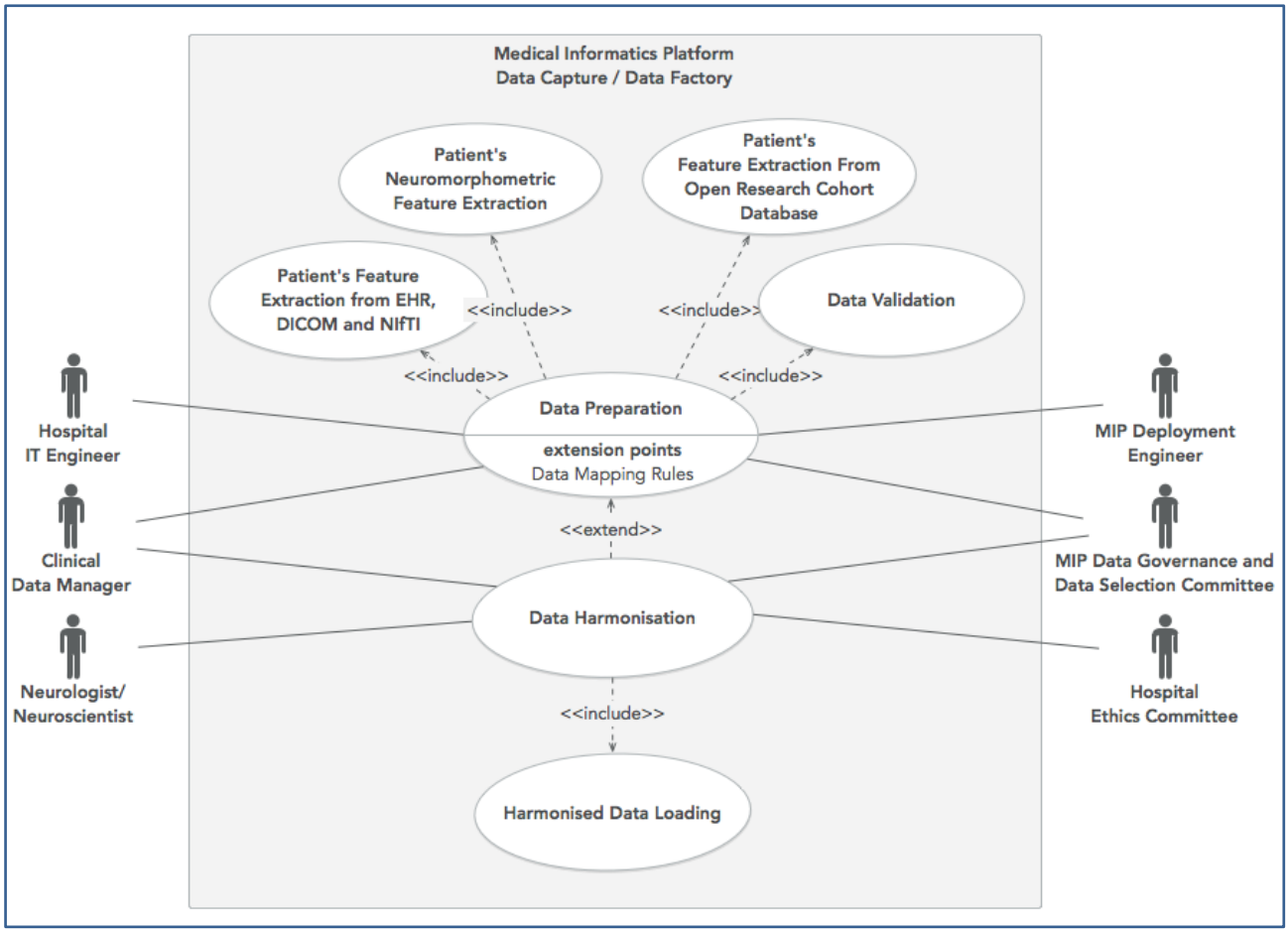


Figure 4: MIP Data Factory Use Cases

Clinical studies involving multiple open research cohort datasets and patient datasets from multiple hospitals are challenging because data sources have different structures and use different coding systems.Erreur ! Source du renvoi introuvable. The Medical Informatics Platform supports harmonisation of data from different sources and provides harmonised data to clinicians and researchers for further analysis. This process is becoming more and more significant since the need for multi-centre studies is rapidly growing and the volume of the available open research cohort data have a tendency to explode.

Scientific Added Value

Extraction and harmonisation of patient biomedical and other health-related features from the source patient data is a first step in the process of creation of a data model for comprehensive

molecular-level data analysis of both individual patients and populations, including their brain features, DNA sequence, proteome, metabolome, microbiome, autoantibodies, etc. Unification of biomedical and other health-related data provides the best opportunity to discover new biological signatures of diseases, improve taxonomy of diseases, develop preventive strategies, and improve medical treatment. This approach shall support the development of individualised medicine and enable cross-comparison between the individual patients to make diagnosing of complex cases more efficient and precise.

Harmonisation of the full set of Medical Informatics Platform's patient biomedical and other health-related features enables large multi-centre, multi-data source studies, increasing the accuracy of analysis methods and the probability for new scientific discoveries.

2.3 Web Applications

A web sub-system provides a web portal and the following applications:

- **Collaboration Space** – landing page of the Medical Informatics Platform displaying a summary of statistics (users, available variables, written articles), and the latest three shared models and articles. It provides a link to the Article Builder web application
- **Data Exploration** – a statistical exploration of patient feature data (i.e. variables). It is possible to explore only statistically aggregated data, not an individual patient's information. This web application provides on-the-fly generation of the descriptive statistics and contains a caching mechanism to handle any future data import in an automated way. It uses information stored in a Metadata database to display additional information about the displayed statistical data, such as data acquisition methodology, units, variable type (nominal or continuous), etc. This web application provides the functionality to search, select and classify data elements as variables, co-variables and filters for configuration of the statistical or machine learning models
- **Model Builder** – configuration/design of statistical or predictive machine learning models. It also provides visualisation for searching the data element types, select and classify data elements as variables, co-variables (nominal and continuous) and filters. Once the model is designed, a design matrix is populated with the selected data. Model Builder provides a visual representation of the design matrix and the selected data for inspection before running a statistical, feature extraction or a machine learning algorithms. It also provides an option to save the designed models
- **Model Validation -** measuring machine-learning models' accuracy by calculating predictive error rate of the model trained on training data against a test dataset. The results guide the user to select the best-performing algorithm and fine-tune its parameters as well as to understand how well the model performs before it is used. The Model benchmark and Validation component from Algorithm Factory is used to measure machine-learning model accuracy. In MIP SGA1 it supports cross-validation method – data split using K-Fold cross-validation. This method of data sampling divides the complete dataset into K disjoint parts of roughly the same size. K different models are trained on K-1 parts each while being tested on the remaining one part of the data. That is done on all K parts exactly once to ensure that every data row is used equally often for training and exactly once for testing. Resulting K test errors are then averaged to get the final error estimate of the model, which was built on the complete dataset
- **Experiment Builder & Disease Models** – a selection of a statistical, feature extraction or machine learning method, the configuration of the method's parameters and the parameters for the trained model validation for supervised machine learning, as well as launching of the machine learning experiment. This application displays experiment validation results as bar charts and confusion matrices
- **Article Builder** – writing the articles using the results of the executed experiments

- **Third-party Applications and Viewers** – portal for accessing third-party web applications for data exploration and visualisation

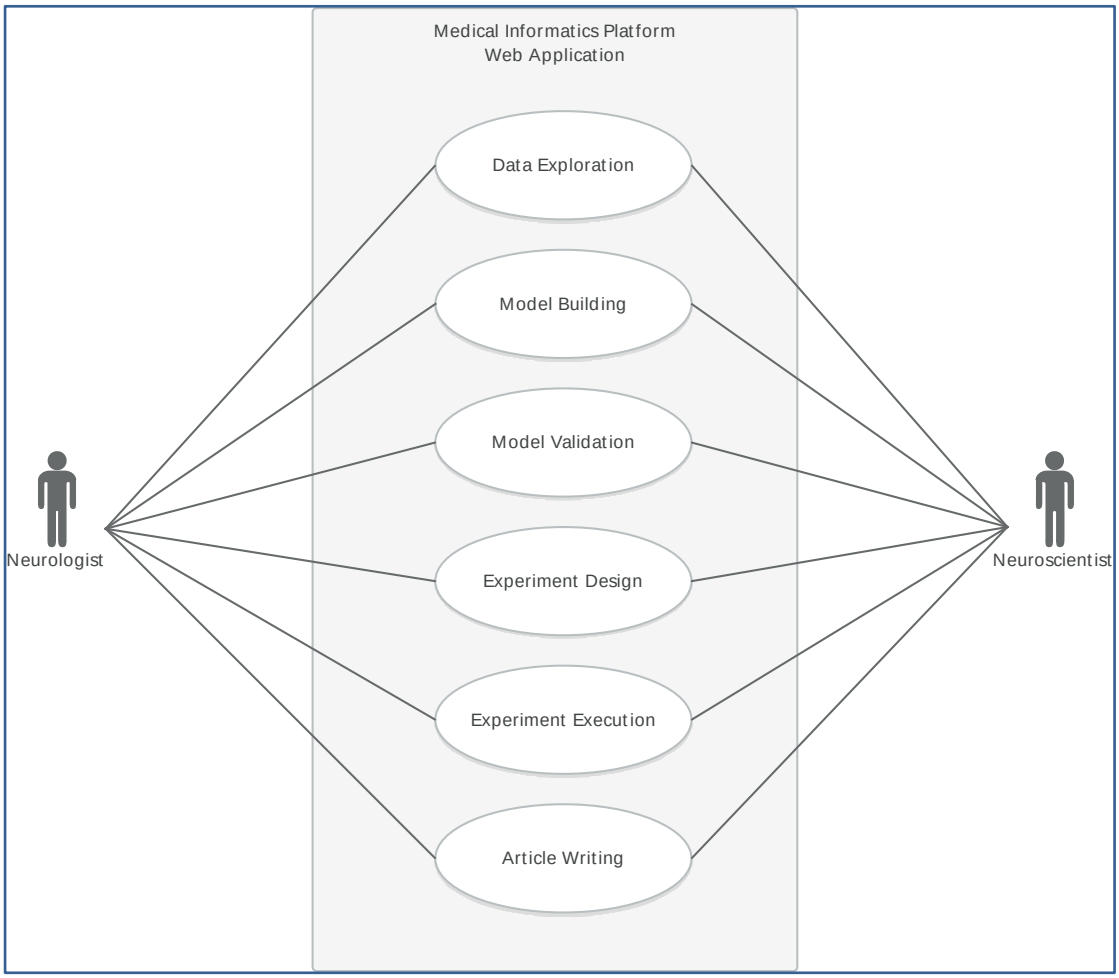


Figure 5: MIP Web Application Use Cases

2.4 Data Mining

The objective of data mining of a group of use cases is the discovery of properties of data in datasets. Out-of-the-box statistical and machine learning algorithms are used to realise MIP data mining use cases.

In case of using machine-learning algorithms for data mining, measurement of the learned model's accuracy and consequently the assessment of the accuracy of the discovered data properties is supported through using the algorithms from the Algorithm Factory's repository. Note that it is not possible to validate algorithms from the Distributed Query Processing Engine's repository in MIP SGA1.

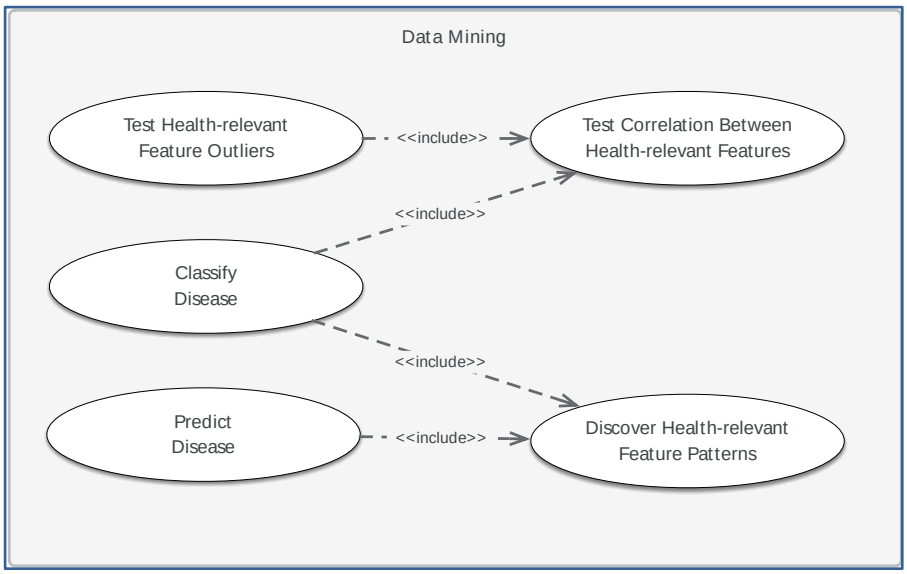


Figure 6: MIP Data Mining Use Cases

Scientific Added Value

This set of use cases specifies the core functionality of the MIP platform – data analytics. Any clinical / research operational scenario executes one or more of the data mining use cases. The four examples of scientific operational scenarios that execute all of the MIP data mining use cases are described in Chapter 8.

Example:

A correlation between brain volume and cognitive decline has been discovered. It was tested whether there are outliers: persons with brain volume decline but no cognitive decline. This gives the idea to include additional health-relevant features to discover whether they may correlate with the observed exceptions. For example, outliers have been discovered and with further data mining it was found that the age of the persons that have brain volume decline but no cognitive decline is in the same range – younger people who have brain volume decline do not have cognitive decline.

2.5 Data Analysis Accuracy Assessment

2.5.1 Analytical Validity

The MIP can be used to measure the analytical validity of tests, i.e. to measure the ability of the tests to accurately detect and measure patient health-related features of interest. MIP SGA1 can measure analytical validity of the following: brain MRI scans, scanning protocols, neuromorphometric feature extraction software applications, neuromorphometric feature extraction methods, neuropsychological instruments and methods, laboratory instruments and methods, etc.

The measured analytical validity using the MIP is the probability that the test results in a dataset chosen for the study will be in the same expected range with the results of the same test under the same conditions in different control datasets, i.e. other research cohorts with available data in the MIP. Analytical validity is a measurement of the MIP data quality.

When there are more data available in the MIP, meaning both the number of patients and the diversity of the test conditions and datasets, the measurement of analytical validity will be more accurate and reliable

The MIP can be used to measure analytical validity on its own, or to include measurement of analytical validity as a research dataset validation step prior to executing a scientifically relevant clinical or biomedical research study using that dataset.

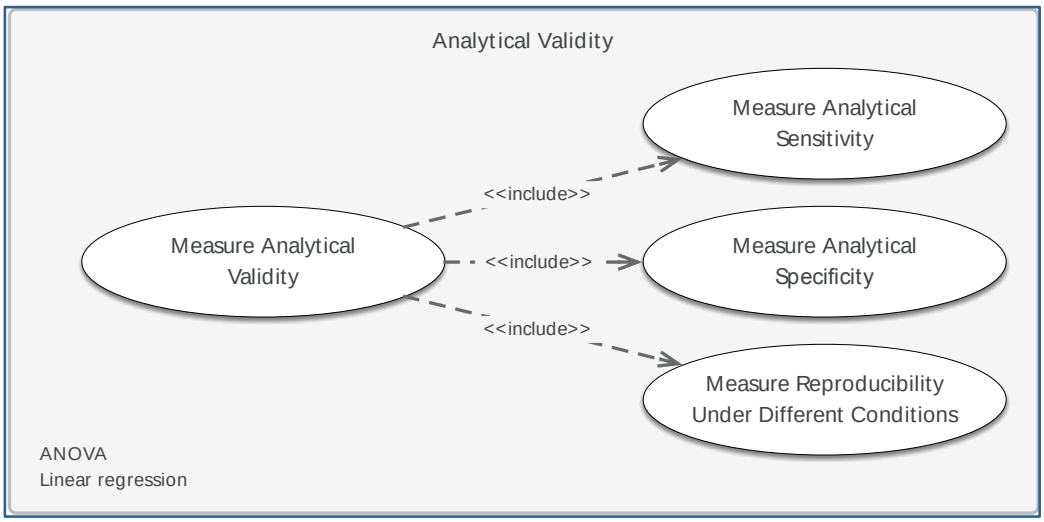


Figure 7: Analytical Validity Use Case

Analytical validity is the test's ability to accurately detect and measure the biomarker of interest (i.e. protein, DNA, RNA). Are the test results repeatable when performed under identical conditions? Are the test results reproducible when the test is performed under different conditions? Is the test sensitive enough to detect biomarker levels as they occur in a real-life setting?

For DNA-based tests, analytical validity requires establishing the probability that a test will be positive when a particular sequence (analyte) is present (analytical sensitivity) and the probability that the test will be negative when the sequence is absent (analytical specificity). In contrast to DNA-based tests, enzyme and metabolite assays measure continuous variables (enzyme activity or metabolite concentration). One key measure of their analytical validity is accuracy, or the probability that the measured value will be within a predefined range of the true activity or concentration. Another measure of analytical validity is reliability, or the probability of repeatedly getting the same result.

2.5.2 Clinical Validity

The MIP can be used to measure clinical validity of a biomarker or other health-relevant feature, i.e. to assess whether the biomarker or other health-relevant patient feature tested is associated with a disease or outcome or the response to a treatment.

Testing of whether a test is accurately detecting and measuring a biomarker or other health-relevant patient feature, i.e. the assessment of test's analytical validity, is a prerequisite for accurate and reliable measurement of the biomarker's or other health-relevant feature's clinical validity. To measure biomarkers' or other health-relevant features' clinical validity, the values for the tested biomarker or the other health-relevant feature, i.e. the data stored in MIP Feature Data Store, must be accurate and reliable. The MIP SGA1 can measure clinical validity of the following types of health-related features: neuromorphometric, cognitive, demographic, genetic, molecular and other biomedical metrics.

Assessment of clinical validity involves measurement of biomarker's or other health-relevant feature's clinical performance, including: (1) clinical sensitivity (ability to identify those who have or will get the disease), (2) clinical specificity (ability to identify those who do not have or will not get the disease), (3) positive predictive value (PPV) - the probability that a person with a positive test result for a predictor, i.e. a biomarker or other health-relevant feature, has or will get the disease, and negative predictive value (NPV) - the probability that a person with a negative test result for a predictor, i.e. a biomarker or other health-relevant feature, does not have or will not get the disease.

When there are more data available in MIP, meaning the number of patients and the diversity of their conditions and profiles, the measurement of clinical validity will be more accurate and reliable.

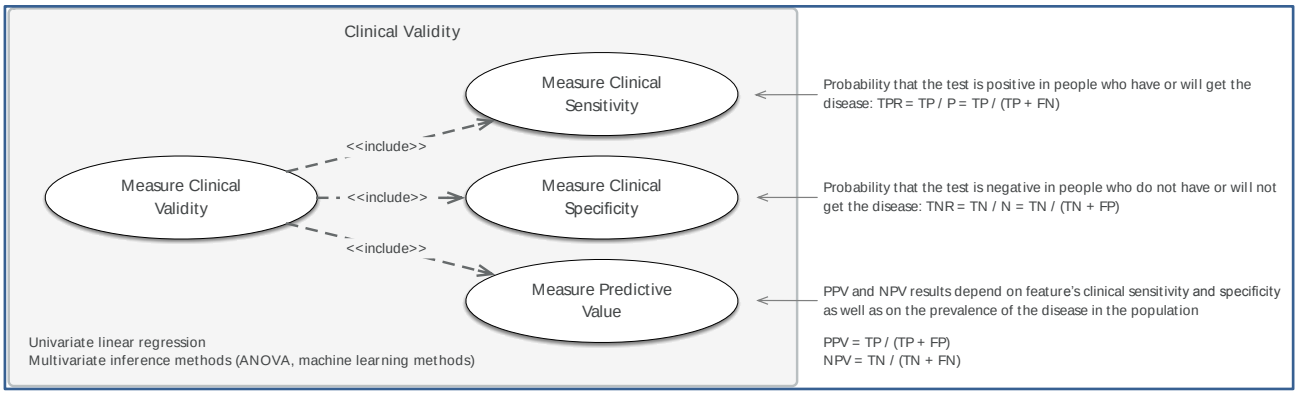


Figure 8: Clinical Validity Use Case

MIP can be used to measure clinical validity on its own, or to include measurement of clinical validity as a research dataset validation step prior to executing a scientifically relevant clinical or biomedical research study using that dataset.

2.5.3 Clinical Utility

Clinical utility is perhaps one of most important considerations when determining whether or not to order or cover a biomedical or other health-relevant feature test. While the meaning of the term has some variability depending on the context or source, there is a largely agreed-upon definition. Four factors are generally considered when evaluating the clinical utility of a test:

- **Patient outcomes** – do the results of the test ultimately lead to improvement of health outcomes (e.g. reduce mortality or morbidity) or other outcomes that are important to patients such as quality of life?
- **Diagnostic thinking** – does the test confirm or change a diagnosis? Does it determine the aetiology for a condition or does it clarify the prognosis?
- **Decision-making guidance** – will the test results determine the appropriate dietary, physiological, medical (including pharmaceutical), and/or surgical intervention?
- **Familial and societal impacts** – does the test identify family members at risk, high-risk race/ethnicities, and the impact on health systems and/or populations?

The development of tests to predict future disease often precedes the development of interventions to prevent, ameliorate, or cure that disease. Even during this therapeutic gap, benefits might accrue from testing. However, in the absence of definitive interventions for improving outcomes in those with positive test results, the clinical utility of the testing will be limited. To improve the benefits of testing, efforts must be made to investigate the safety and effectiveness of new interventions while the tests are developed.

Clinical utility is not always evident in testing for inherited disorders for which treatments have not yet been developed. The clinical utility of a genetic diagnosis for an incurable or untreatable disease, without knowing the outcome, just looking for a predisposition to disease, is not useful.

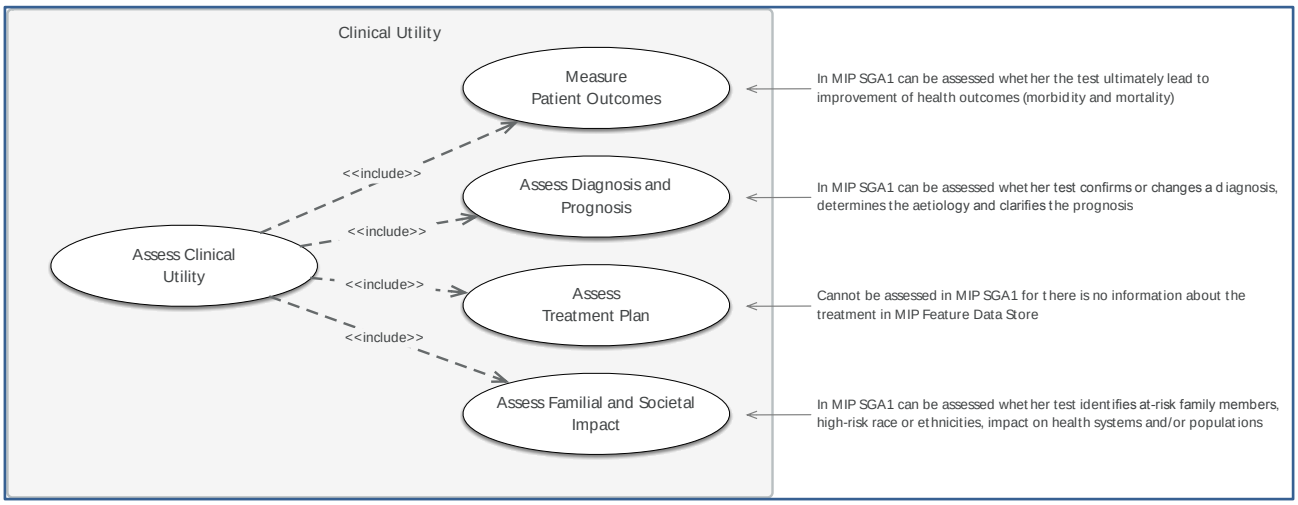


Figure 9: Clinical Utility Use Case

2.6 Overview of MIP Use Cases

This section of the document gives a summary of all MIP use cases discussed in the previous subchapters. MIP use cases are grouped in two tables:

1) MIP use cases involved in MIP deployment use scenarios

2) MIP use cases involved in MIP data analysis, i.e. clinical study scenarios

MIP deployment use scenarios consist of the actions for software installation and patient data extraction and processing.

MIP clinical study scenarios consist of data analysis actions, including data examination, creation of data models, selection and configuration of statistical or machine learning methods for descriptive or predictive data analytics.

Table 1: Overview of MIP Deployment Use Cases

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Deployment Use Cases			
Software Installation			
UC_ITL_01	Software Installation	MIP execution environment configuration and software installation	
Data Capture / Data Factory			
UC_DFY_01	Data Preparation	Orchestration of source EHR and brain imaging data extraction, data transformation and data loading pipelines, including data quality assurance and data provenance storage	

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Deployment Use Cases			
UC_DFY_02	Patient's Feature Extraction from EHR, DICOM and NIfTI	Extraction of patient demographic, biological, genetic and cognitive data from HER and extraction of the metadata from patient's brain scan DICOM or NIfTI files	Included in UC_DFY_01
UC_DFY_03	Patient's Neuromorphometric Feature Extraction	Extraction of neuromorphometric data from patient brain scans	Included in UC_DFY_01
UC_DFY_04	Patient's Feature Extraction From Open Research Cohort Dataset	Extraction of patient feature data from open research cohort datasets	Included in UC_DFY_01
UC_DFY_05	Data Validation	Checking of pre-processed brain images for artefacts and quality metrics, check data for confound and biases, check metadata	Included in UC_DFY_01
UC_DFY_06	Data Harmonisation	Transformation of source patient biomedical and health-related features to harmonised data structure and data vocabulary	Extends UC_DFY_01
UC_DFY_07	Harmonised Data Loading	Loading of transformed source datasets to permanent harmonised feature data store for federated multi-centre multi-dataset analytics	Included in UC_DFY_05

Table 2: Overview of MIP Clinical Study Use Cases

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
Web Application			
UC_WEB_01	Data Exploration	Statistical exploration of patient feature data (i.e. variables)	

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
UC_WEB_02	Model Building	Configuration/design of statistical or predictive machine learning models	
UC_WEB_03	Model Validation	Validation of learned model against the test dataset. Calculation of the predictive error rate	
UC_WEB_04	Experiment Design	Selection of a statistical, feature extraction or machine learning method, the configuration of the method's parameters and the parameters for the trained model validation for supervised machine learning	
UC_WEB_05	Experiment Execution	Launching of the machine learning experiment. Displays experiment validation results as bar charts and confusion matrices	
UC_WEB_06	Article Writing	Writing scientific articles using the results of the executed experiments	
Data Mining			
UC_DTM_01	Test Correlation Between Health-relevant Features	Testing the correlation between two or more variables using a statistical or machine learning method	Included in UC_DTM_02 Included in UC_DTM_03
UC_DTM_02	Test Health-relevant Feature Outliers	Discovering outliers after testing the correlation between variables	
UC_DTM_03	Classify Disease	Using classification machine learning algorithms to create (learn), validate and/or apply the classifier	
UC_DTM_04	Predict Disease	Apply a learned classifier to predict pathology	
UC_DTM_05	Discover Health-relevant Feature Patterns	Discover patterns of correlated variables in a population	Included in UC_DTM_03 Included in UC_DTM_04
Data Analysis Accuracy Assessment			

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
UC_ACC_01	Measure Biomarker's Analytical Validity	Measure analytical validity of tests – assess the ability of the test to accurately detect and measure patient's health-related features of interest. Analytical validity measured using MIP is the probability that the test results in a dataset chosen for the study will be in the same expected range with the results of the same test under the same conditions in different control datasets, i.e. other research cohorts whose data are part of the MIP. Analytical validity is a measurement of the MIP data quality.	
UC_ACC_02	Measure Biomarker's Analytical Sensitivity	Measure the probability that a test will detect an analyte when it is present in a specimen	Included in UC_ACC_01
UC_ACC_03	Measure Biomarker's Analytical Specificity	Measure the probability that a test will be negative when an analyte is absent from a specimen	Included in UC_ACC_01
UC_ACC_04	Measure Biomarker's Reproducibility Under Different Conditions	Evaluating the results of the a test when it is performed under different conditions	Included in UC_ACC_01
UC_ACC_05	Measure Health-relevant Feature's Clinical Validity	Measure clinical validity of a biomarker or other health-relevant feature, i.e. to assess whether the biomarker or other tested health-relevant patient's feature is associated with a disease or outcome or the response to a treatment	
UC_ACC_06	Measure Health-relevant Feature's Clinical Sensitivity	Probability that the test is positive in people who have or will get the disease: TPR = TP / P = TP / (TP + FN)	Included in UC_ACC_05
UC_ACC_07	Measure Health-relevant Feature's Clinical Specificity	Probability that the test is negative in people who do not have or will not get the disease: TNR = TN / N = TN / (TN + FP)	Included in UC_ACC_05

Medical Informatics Platform Use Case List			
ID	**Name**	**Short Description**	**Relationship**
Clinical Study Use Cases			
UC_ACC_08	Measure Health-relevant Feature's Clinical Predictive Value	Positive Predictive Value (PPV) and Negative Predictive Value (NPV) results depend on feature's clinical sensitivity and specificity as well as on the prevalence of the disease in the population. PPV = TP / (TP + FP) NPV = TN / (TN + FN)	Included in UC_ACC_05
Clinical Utility Assessment			
UC_CLU_01	Assess Health-relevant Feature's Clinical Utility	Three factors are generally considered when evaluating the clinical utility of a test: 3) Patient outcomes, 4) Diagnostic thinking, 5) Societal impacts	
UC_CLU_02	Measure Patient Outcomes	Do the results of the test ultimately lead to improvement of health outcomes (e.g. reduced mortality or morbidity) or other outcomes that are important to patients such as quality of life?	Included in UC_CLU_01
UC_CLU_03	Assess Diagnosis and Prognosis	Does the test confirm or change a diagnosis? Does it determine the aetiology for a condition or does it clarify the prognosis?	Included in UC_CLU_01
UC_CLU_04	Assess Societal Impact	Does the test identify high-risk race/ethnicities, and the impact on health systems and/or populations?	Included in UC_CLU_01

2.7 Data Analysis Accuracy Assessment Use Case Overview

The assessment of the accuracy of analysed patient data – the MIP diagnostic measure – is a crucial feature of any clinical study. As discussed in previous subchapters, an assessment of data analysis results is performed by the three use case groups: Analytical Validity, Clinical Validity and Clinical Utility. Here, we provide an overview of the three groups, the object of measurement, a short description and a list of statistical and machine learning methods available in the MIP.

Table 3: Measuring Analytical and Clinical Validity and Clinical Utility With The MIP

MIP Diagnostic Measure	Measured Object	Description of the MIP Diagnostic Measure	Method of Measure
Analytical Validity	**Data Quality**	Measurement of the data quality: accuracy and reliability. Accuracy is the probability that the values of the patient features in a dataset chosen for the study will be in the same expected range as the values of those features in "gold standard" – control research cohort datasets. Reliability is the probability of repeatedly getting the same data analysis result when using MIP's integrated statistical methods and machine learning algorithms. Analytical validity assessment is a prerequisite for accurate and reliable measurement of the feature's clinical validity. To measure clinical validity of the feature, data stored in MIP must be accurate and reliable Reliability of the predictive (machine-learning) models is measured using model validation methods integrated into the Medical Informatics Platform	Analytical validity of data: ANOVA, linear regression, logistic regression. Visual methods: histogram, density plot, scatter plot, box plot Analytical validity of predictive models: cross-validation
Clinical Validity	**Clinical Feature**	Measurement of the feature's clinical performance: (1) clinical sensitivity (ability to identify those who have or will get the disease), (2) clinical specificity (ability to identify those who do not have or will not get the disease), (3) positive predictive value (PPV, the probability that a person with a positive test result for a predictor, has or will get the disease), and negative predictive value (NPV the probability that a person with a negative test result for a predictor does not have or will not get the disease). The MIP can be used to measure clinical validity of the features (biomarkers and other relevant data), or to measure clinical validity of the descriptive and predictive mathematical models by executing integrated model validation methods. Clinical validity of the models with different set of features can be compared using ROC curves, C-statistics, etc.	Clinical validity of features: ANOVA, linear regression, logistic regression. Visualisation: heatmap

MIP Diagnostic Measure	Measured Object	Description of the MIP Diagnostic Measure	Method of Measure
		The more data available in the MIP (patient number and diversity of patient conditions and profiles), the more accurate and reliable the measurement of clinical validity	
Clinical Utility	Result of Analytics	Evaluation of the clinical utility of the results of the data analytics using the Medical Informatics Platform: 1) **diagnostic relevance**: do the results of the predictive analytics confirm or change a diagnosis in a new group of patients, do they determine the aetiology for a condition or clarify the prognosis 2) **disease outcomes**: do the results of the predictive analytics lead to the improvement of health outcomes (e.g. reduce mortality or morbidity – prescriptive implication of machine learning models) or other outcomes that are important to patients, such as quality of life 3) **familial and societal impacts**: do the results of the predictive analytics identify family members at risk, high-risk race/ethnicities, and the impact on health systems and/or population The important part of the assessment of the clinical utility of the results of predictive analytics is the evaluation of the accuracy of the hypothesis function. The method used in this release of MIP is cross-validation. The measured accuracy of the learned model shall determine the level of clinical utility of the model with the real patient population.	Machine learning models (supervised and unsupervised): univariate and multivariate linear and polynomial regression using gradient decent, KNN, Naïve Bayes; K-means; SVM. Validation of machine learning models using cross-validation integrated into the MIP

3. MIP Architecture

The Medical Informatics Platform is a complex information system comprising numerous software components designed and integrated by different SP8 partners.

This Chapter provides an end-to-end functional overview of the Platform, describing the logical component architecture and the components' roles, showing how the functionality is designed inside the Platform, regarding the static structure of the Platform and the interaction between its components.

This Chapter also contains a brief overview of the key deployment architecture concepts, without providing a detailed specification of the deployment of components into the Platform's physical architecture. Some deployment terminology, such as "local hospital MIP" and "central MIP federation node" is used here only in the context of describing the function of relevant components.

3.1 Functional Architecture Overview

3.1.1 Data Capture Subsystem

The Data Capture sub-system provides a local interface to other hospital information systems. It is a single point of entry for all the data that contain personally identifiable information.

The purpose of the Data Capture sub-system is de-identification of patient data exported from hospital information systems (EHRs, PACS). De-identified data is uploaded to De-identified Data Version Control Storage, belonging to the Data Factory sub-system, for processing and feature extraction.

The flow of data between the Data Capture component (Data De-identifier) and, on one side, other local hospital information systems and, on the other side, the MIP Data Factory sub-system is as follows:

4) MIP captures personal health sensitive data from the following hospital information systems:
    - Electronic Health Record (EHR) Systems
    - Picture Archiving and Communication Systems (PACS)
5) Data De-identifier replaces the following personally identifiable information with pseudonyms:
    - Information exported from EHR systems in CSV format
    - Information from neuroimages stored in the headers of DICOM files
6) Data De-identifier saves the files with de-identified data to storage in the Data Factory sub-system

Anonymised patient cohort datasets (for example, ADNI, EDSD, PPMI) are stored directly in the De-identified Data Version Controlled Storage belonging to the Data Factory sub-system.

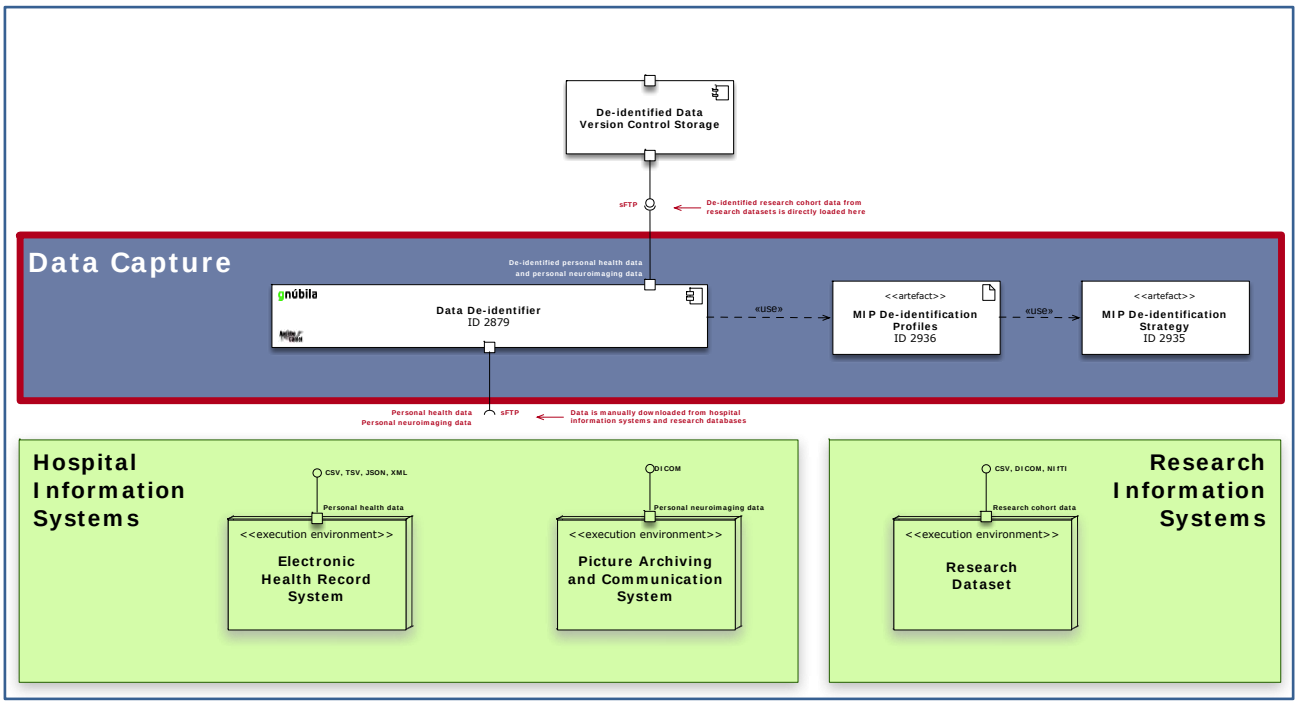


Figure 10: Data Capture Sub-system

```
/data
├── anonymiser              -- This folder contains all the data being processed.
│   ├── db                  -- This contains the database of the internal IDs to public IDs mappings.
│   ├── in                  -- Files to be processed.
│   ├── out                 -- Files successfuly anonymised.
│   ├── quarantine          -- Files for which the anonymisation failed:
│   │   ├── csv             --   * EHR data files
│   │   ├── dicom           --   * Imaging data files
│   │   ├── pacs_csv        --   * EHR data retrieved through a PACS connexion
│   │   └── unknown         --   * Any other kind of failures
│   └── scripts             -- Anonymisation configurations for:
│       ├── csv             --   * EHR data files
│       └── dicom           --   * Imaging data files
└── ldsm
```

Figure 11: Data Folder Organisation for the De-identification Processing

The Electronic Health Record (EHR) is a collection of a patient health information stored by EHR systems in a digital format. EHR systems are designed for capturing and storing of patient data over time. Well-designed EHR systems are online transaction processing systems that collect and store patient data in a normalised database, therefore minimising data redundancy and improving data integrity.

Picture Archiving and Communication System (PACS) provides storage and access to digital images originating from multiple modalities (imaging machine types). The universal format for PACS image storage and transfer is DICOM (Digital Imaging and Communications in Medicine). Non-image data, such as image-related metadata and scanned PDF documents, can be encapsulated in DICOM files.

MIP captures patient personally identifiable demographic, diagnostic and biomedical data from EHR systems in CSV file format and neuroimaging MRI data from PACS systems in DICOM file format. Patient data are captured periodically for batch processing in the MIP.

Authorised hospital staff that exported the data, manually imports them into the MIP Data De-identifier component for de-identification.

In coordination with local hospital's data management team and ethics committee, the MIP data governance and data selection team (DGDS) is responsible for the specification of data de-

personalisation rules in compliance with data protection regulations, such as EU/GDPR, CH/FADP and US/HIPAA. The Data de-identifier component's rule engine is configured using configuration scripts derived from these rules.

The third-party Gnubila FedEHR Anonymizer data de-identification solution has been chosen for the Data De-identifier component. This component is a profile-based, rule-based asynchronous message-oriented mediation engine, developed using an Apache Camel framework. It can be extended to support new data formats and de-identification algorithms. It replaces all personally identifiable information from the captured data with pseudonyms using out-of-the-box data de-identification techniques, such as generalisation, micro-aggregation, encryption, swapping and sub-sampling.

Discussion About Data Re-identification

Data re-identification is not a feature of the Medical Informatics Platform. It is not possible to re-identify a patient using any of the designed functions of the MIP (data privacy by design). Administratively and organisationally, re-identification of patient data is the responsibility of their hospitals. Technically, for re-identifying patient data stored in the de-identified form in their hospitals' local MIP data storage, hospital IT staff needs to develop standalone lookup applications to map personally identifiable information with the pseudonyms at the point of de-identification. Those applications shall never be integrated with the MIP.

3.1.2 Data Factory Subsystem

The components of the logical Data Factory sub-system perform batch neuroimaging and EHR data pre-processing, extraction, transformation and loading into the normalised permanent data storage.

The ETL processes of the Data Factory sub-system are orchestrated as directed acyclic graphs (DAG's) of tasks in programmatically configurable pipelines using an open-source Apache Airflow workflow management platform. Additional components are built for data transformation and data provenance tracking, including the complex neuroimaging processing and brain feature extraction, brain scan metadata and EHR data extraction as well as data transformation and loading tasks.

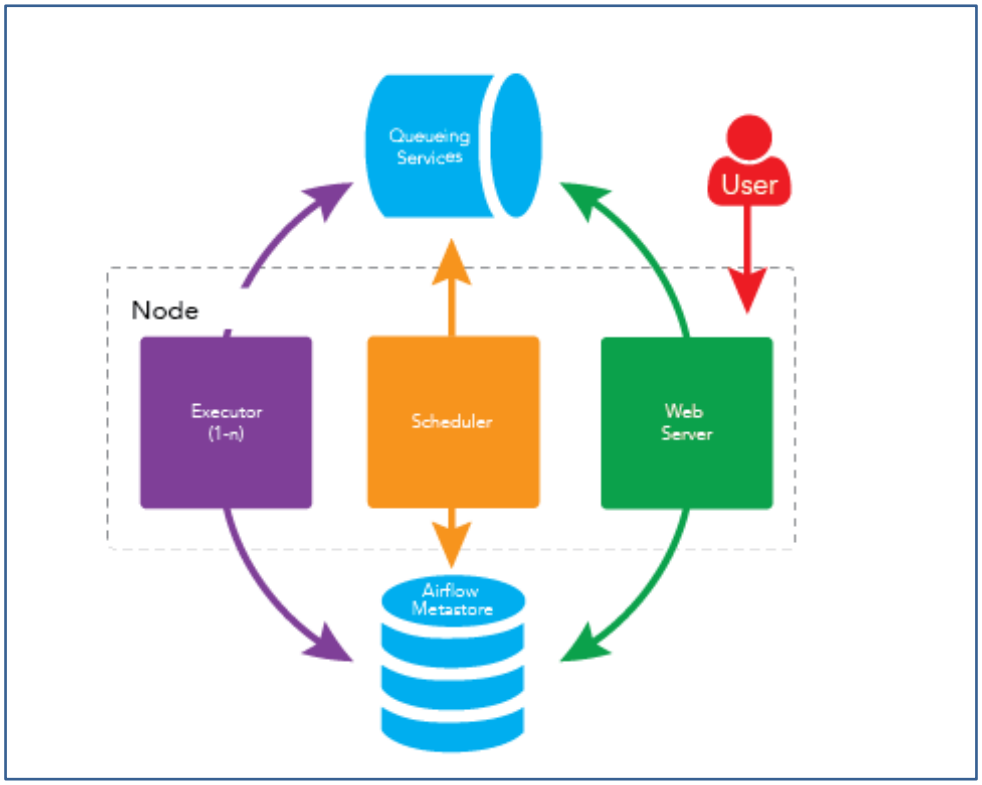


Figure 12: Apache Airflow Concept

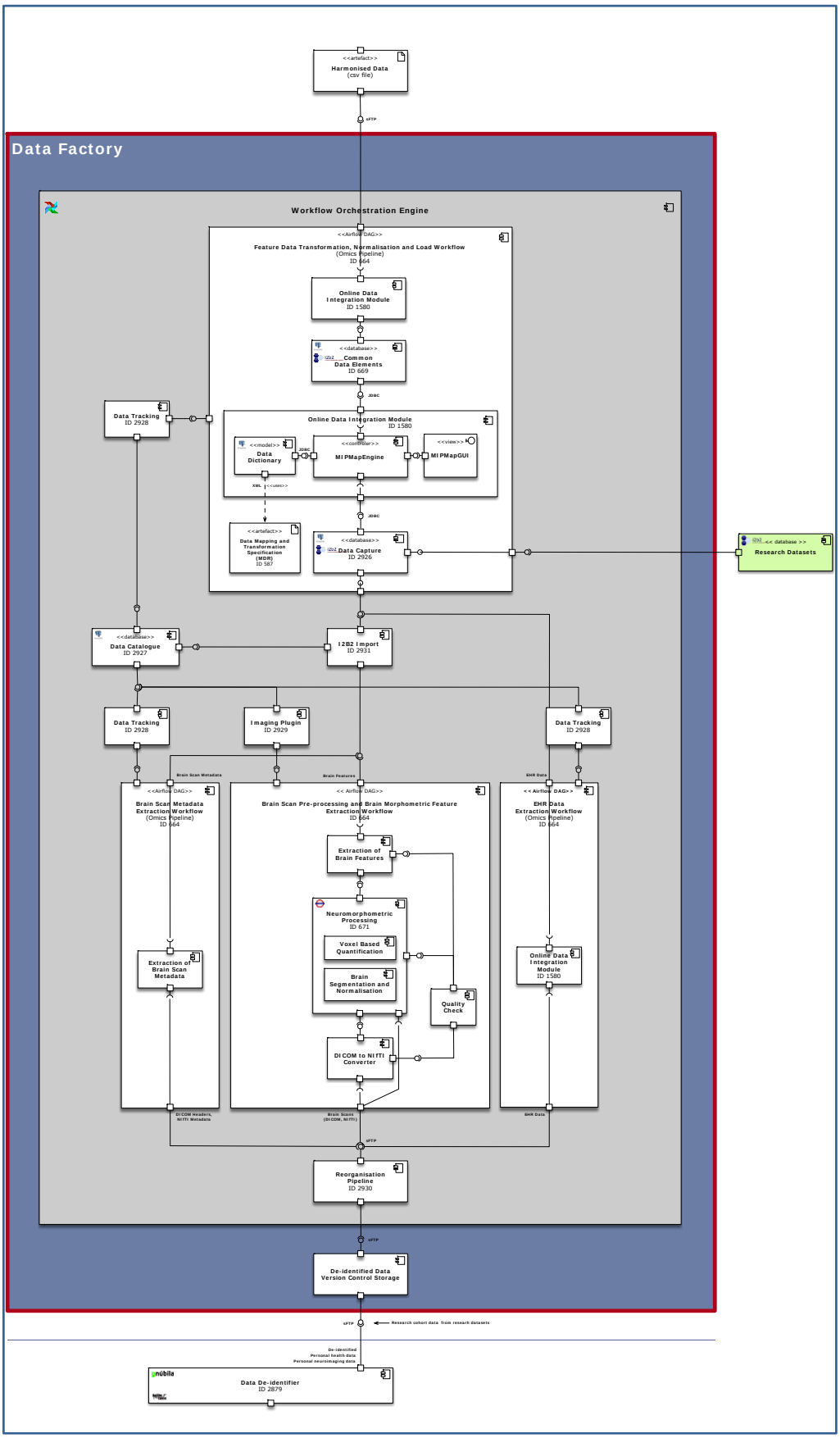


Figure 13: Data Factory Sub-system

Airflow is an open source solution for defining, scheduling, and monitoring of jobs. Pipelines are defined as a code using Python and the jobs are scheduled using cron expressions. The scheduler executes tasks on an array of workers according to the specified dependencies. The user interface makes it easy to visualise pipelines running in production, monitor progress, and troubleshoot issues when needed.

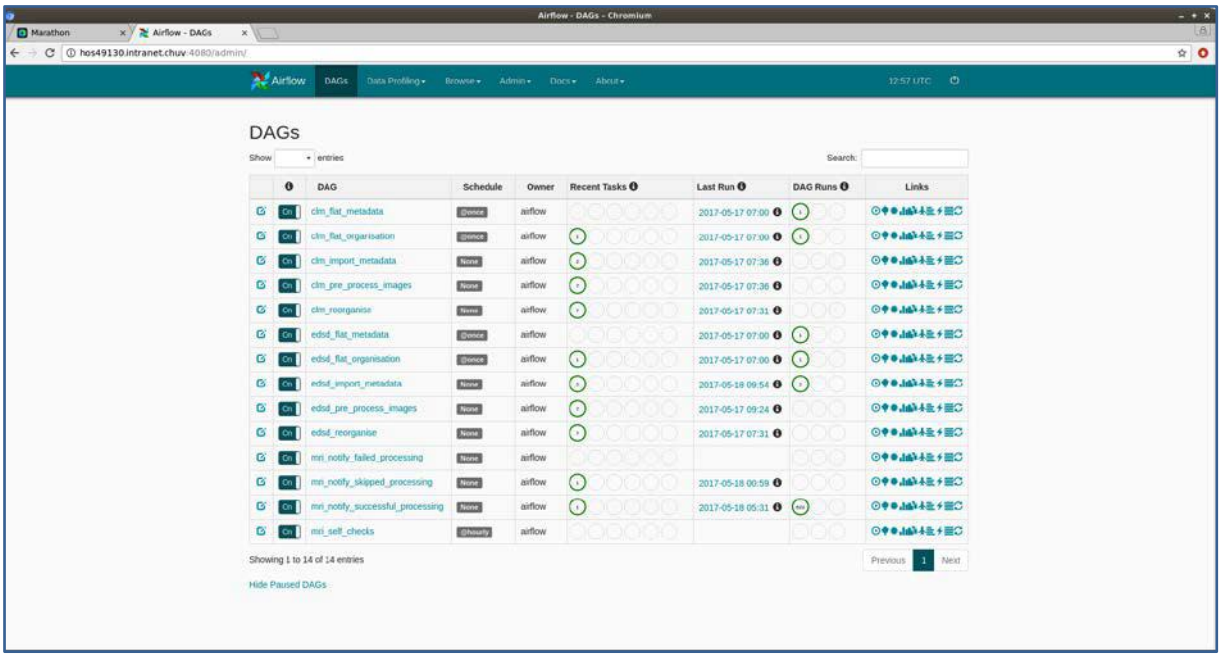


Figure 14: Apache Airflow Dashboard

The Data Factory sub-system provides the following extraction, transformation and load functionality:

7) Pulling de-identified data out of the files stored in De-identified Data Version Control Storage

```
├─ DICOM
│  └─ 2016                                  -- yearly folder, date represents the date of export
│     └─ 20161029                           -- daily folder, date represents the date of export
│        └─ scan_research_id                -- see description below
│           └─ dicom_name_generated_01.dcm  -- set of DICOM files
│           └─ dicom_name_generated_02.dcm  -- set of DICOM files
│           └─ dicom_name_generated_03.dcm  -- set of DICOM files
└─ EHR
   └─ 2016                                  -- yearly folder, date represents the date of export
      └─ 20161029                           -- daily folder, date represents the date of export
         ├─ table1.csv                      -- pre-defined name for 1st table containing EHR data, depends on hospital data
         ├─ table2.csv                      -- pre-defined name for 2nd table containing EHR data, depends on hospital data
         └─ ...                             -- more (or less) tables as needed, depends on hospital data
```

Figure 15: De-identified DICOM and EHR Data

```
├─ NIFTI
│  └─ 2016                                  -- yearly folder, date represents the date of export
│     └─ 20161029                           -- daily folder, date represents the date of export
│        └─ scan_research_id                -- see description below
│           └─ dicom_name_generated_01.nifti -- Nifti file
│           └─ dicom_name_generated_01.json  -- metadata for the Nifti file
│           └─ dicom_name_generated_02.nifti -- Nifti file
│           └─ dicom_name_generated_02.json  -- metadata for the Nifti file
└─ EHR
   └─ 2016                                  -- yearly folder, date represents the date of export
      └─ 20161029                           -- daily folder, date represents the date of export
         ├─ table1.csv                      -- pre-defined name for 1st table containing EHR data, depends on hospital data
         ├─ table2.csv                      -- pre-defined name for 2nd table containing EHR data, depends on hospital data
         └─ ...                             -- more (or less) tables as needed, depends on hospital data
```

Figure 16 – De-identified NIfTI and EHR Data

8) Processing de-identified data to extract a patient's raw health-related features:
   a) Brain morphometric features (grey matter volume, shape and dimensions)
   b) Brain scan metadata
   c) Data from EHR files (demographic, biomarkers, neuropsychological assessments, diagnoses)
9) Harmonising data types from different source datasets into a common data element (CDE) model
10) Transformation of the extracted feature data and its permanent storage into the CDE Database
11) Placing feature data into files accessible by Features Data Store sub-system components

In addition to the components for extracting personal health features, the Data Factory sub-system contains a set of quality assurance components:

- **Quality Check** for a computational check of the quality of processed and extracted data
- **Imaging Plugin** to track all data changes during brain scan data processing and extraction
- **Data Tracking** to track all data changes except during brain scan data processing and extraction
- **Data Catalogue** to store data provenance/data version information

Reorganisation Pipeline

The Reorganisation pipeline is a component conditional to reorganise datasets pulled from the De-identified data version control storage to prepare them to enter the workflows for processing and extracting brain scan metadata, brain scan pre-processing and brain morphometric feature extraction and EHR data extraction.

The configuration of this pipeline needs to be tailored to every new hospital and research data set. The structure of the brain scan files (DICOM or NIfTI), including the metadata in their headers, depends on the non-standardised procedures specific for each hospital. The structure and the content of EHR files also need to be inspected, and configuration of the pipeline tailored accordingly.

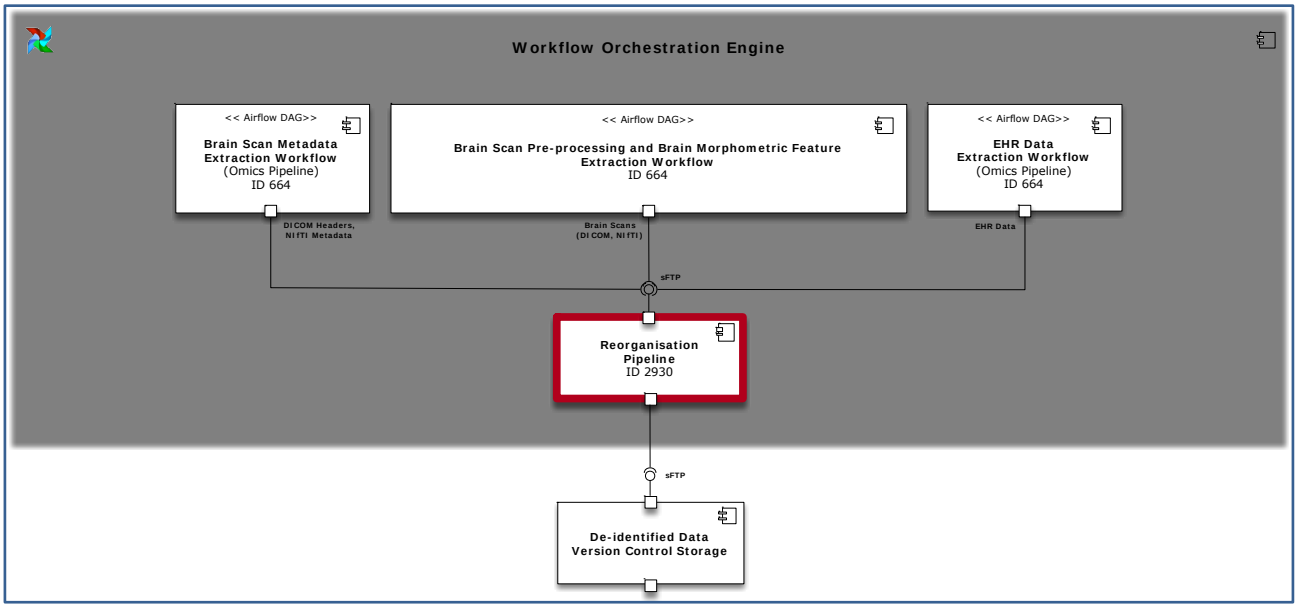


Figure 17: Reorganisation Pipeline

Brain Scan Pre-processing and Brain Morphometric Feature Extraction Pipeline

Software systems are essential in all stages of neuroimaging, allowing scientists to control highly sophisticated imaging instruments and to make sense of the vast amounts of generated complex data. For magnetic resonance imaging (MRI), software systems are used to design and implement signal-capturing protocols in imaging instruments, reconstruct the resulting signals into a three-dimensional representation of the brain, correct for and suppress noise, statistically analyse the data, and visualise the results. Collected neuroimaging data can then be stored, queried, retrieved and shared using PACS, XNAT, CBRAIN, LORIS or any other system. Neuro-anatomical data can be extracted from neuroimages, compared and analysed using other specialised software systems, such as SPM and FreeSurfer.

After capturing and de-identifying neuroimaging DICOM data from PACS systems, the MIP's Data Factory sub-system extracts neuroanatomical data from captured brain magnetic resonance images, permanently stores that data into the Feature Data Store sub-system where it is made available for data mining and analysis together with the rest of biomedical and other health-related information.

The flow of data between Brain Scan Pre-processing and Brain Feature Extraction pipeline components is as follows:

12) **A visual quality check of the neuroimages performed by a neuroradiologist.**

Pre-processing of magnetic resonance (MR) images strongly depends on the quality of input data. Multi-centre studies and data-sharing projects need to take into account varying image properties due to different scanners, sequences and protocols

Image format requirements:

- Full brain scans
- Provided either in DICOM or NIFTI format
- High-resolution (max. 1.5 mm) T1-weighted sagittal images.
- If the dataset contains other types of images (that is not meeting the above description, e.g. fMRI data, T2 images, etc.), a list of protocol names used and their compatibility status regarding the above criterion has to be provided
- Images must contain at least 40 slices

13) **The DICOM to NIfTI Converter converts brain scan data captured in DICOM format to NIfTI data format**

14) **The Neuromorphometric Processing component (SPM12) uses NIfTI data for computational neuro-anatomical data extraction using voxel-based statistical parametric mapping of brain image data sequences:**

a) Each T1-weighted image is normalised to MNI (Montreal Neurological Institute) space using non-linear image registration SPM12 Shoot toolbox

b) The individual images are segmented into three different brain tissue classes (grey matter, white matter and CSF)

c) Each grey matter voxel is labelled based on Neuromorphometrics atlas (constructed by manual segmentation for a group of subjects) and the transformation matrix obtained in the previous step. Maximum probability tissue labels were derived from the "MICCAI 2012 Grand Challenge and Workshop on Multi-Atlas Labelling". These data were released under the Creative Commons Attribution-Non-Commercial (CC BY-NC. The MRI scans originate from the OASIS project, and the labelled data were provided by Neuromorphometrics, Inc. under an academic subscription

15) **The Voxel-Based Quantification (VBQ) component, through its sensitivity to tissue microstructure, provides absolute measures for neuroimaging biomarkers for myelination, water and iron levels comparable across imaging sites and in time**

16) **The I2B2 Import component stores extracted brain morphometric features in I2B2 Capture Database, alongside the brain scan metadata and patient EHR data**

The Quality Check component evaluates essential image parameters, such as signal-to-noise ratio, inhomogeneity and image resolution. It evaluates images for problems during the processing steps. It allows comparing quality measures across different scans and sequences.

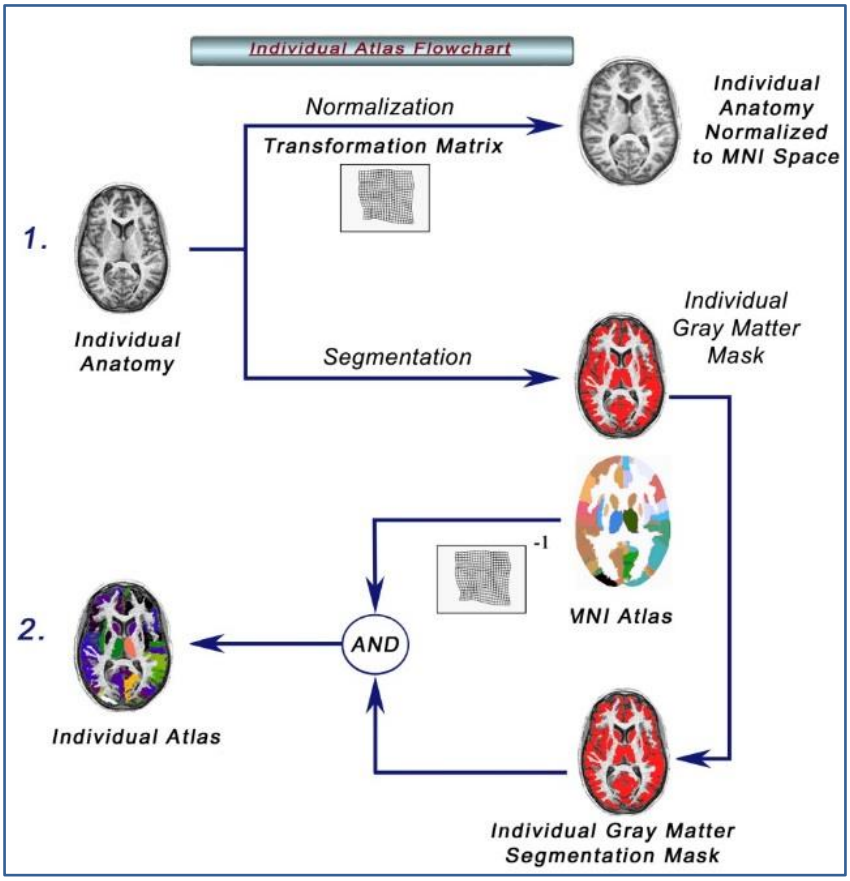


Figure 18: Neuromorphometric Processing

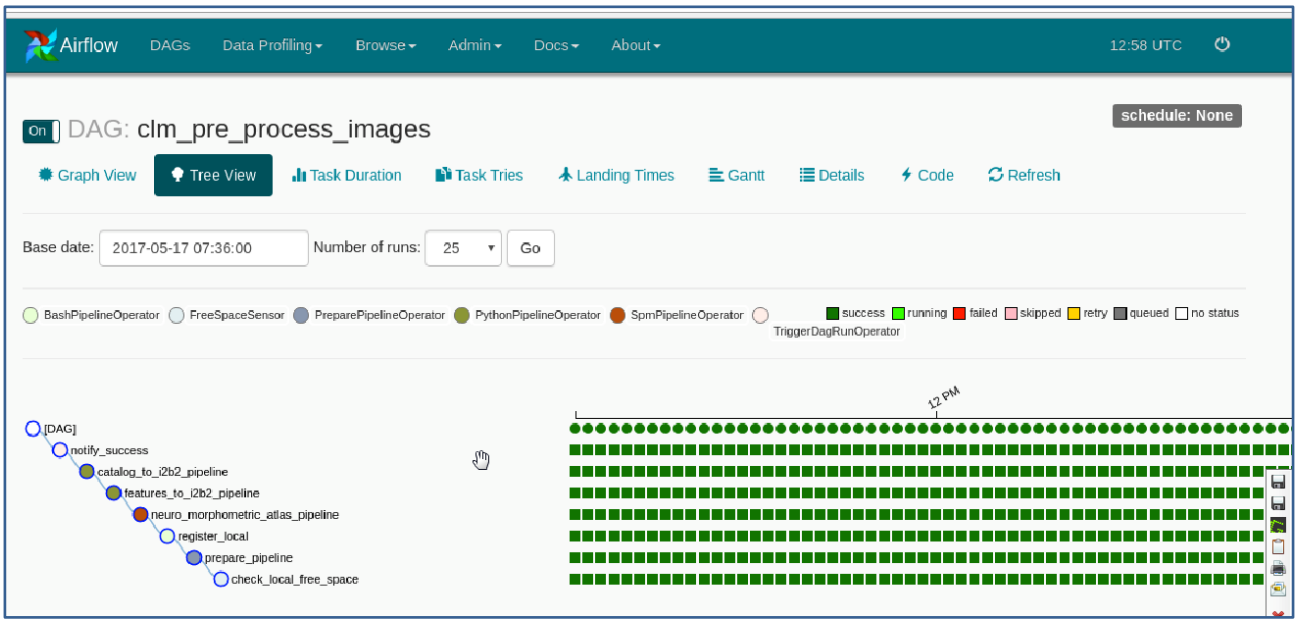


Figure 19: Apache Airflow Image Processing Pipeline Status

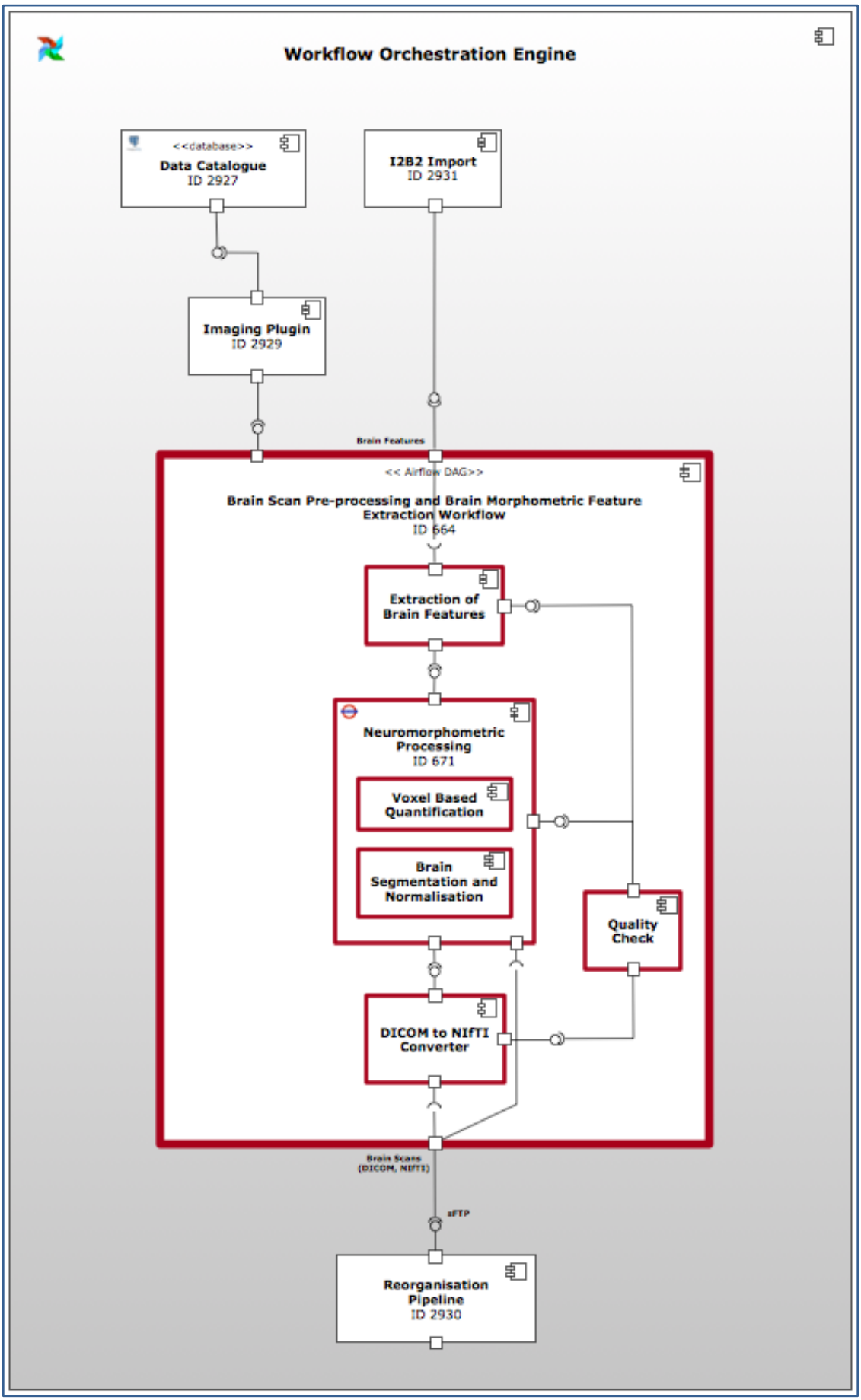


Figure 20: Brain Scan Pre-processing and Brain Feature Extraction Workflow

The Brain Scan Pre-processing and Brain Morphometric Feature Extraction pipeline contains components for processing T1-weighted brain image data sequences and extracting morphometric brain features – grey matter volume and shape – using voxel-based morphometry (VBM). VBM provides insight into macroscopic volume changes that may highlight differences between groups, be associated with pathology or be indicative of plasticity.

For neuromorphometric processing, the MIP uses SPM12 software running within the MATLAB software environment. For image pre-processing and morphometric feature extraction, SPM requires input data in a standard format used by neuromorphometric tools for computation and feature extraction: the NIfTI format.

The T1-weighted images are automatically segmented into 114 anatomical structures using the Neuromorphometrics atlas.

In addition to voxel-based neuromorphometric processing of T1-weighted images for classification of tissue types and measuring of macroscopic anatomical shape, the MIP uses a voxel-based quantification (VBQ) toolbox as a plugin for SPM12 that can analyse high-resolution quantitative imaging and can provide neuroimaging biomarkers for myelination, water and iron levels that are absolute measures comparable across imaging sites and in time.

Single NIfTI volumes of the brain are first partitioned into three classes: grey matter, white matter and background. This procedure also incorporates an approximate image alignment step and a correction for image intensity non-uniformities. This procedure uses the SPM12 Segment5 tool.

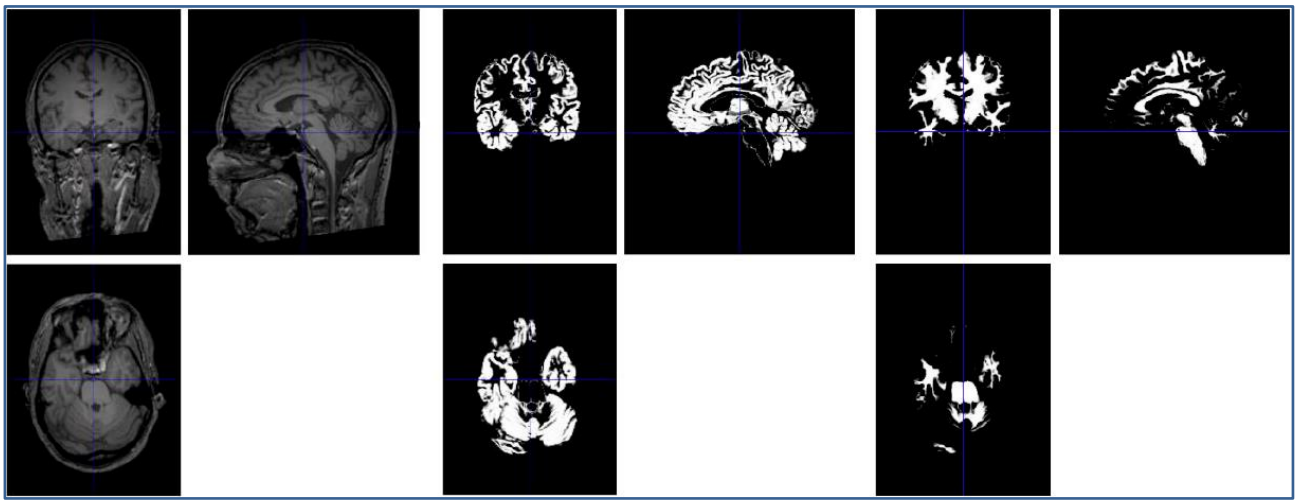


Figure 21: Original T1-weighted MRI scan (left), along with automatically extracted grey (middle) and white matter (right) tissue maps. The tissue maps encode the probability of each tissue type calculated using the given model and data

Tissue atlases, pre-computed from training data are then spatially registered with the extracted grey and white matter maps, using the Shoot5 tool from SPM12. The warps estimated from this registration step are then used to project other pre-computed image data into alignment with the original scans (and their grey and white matter maps).

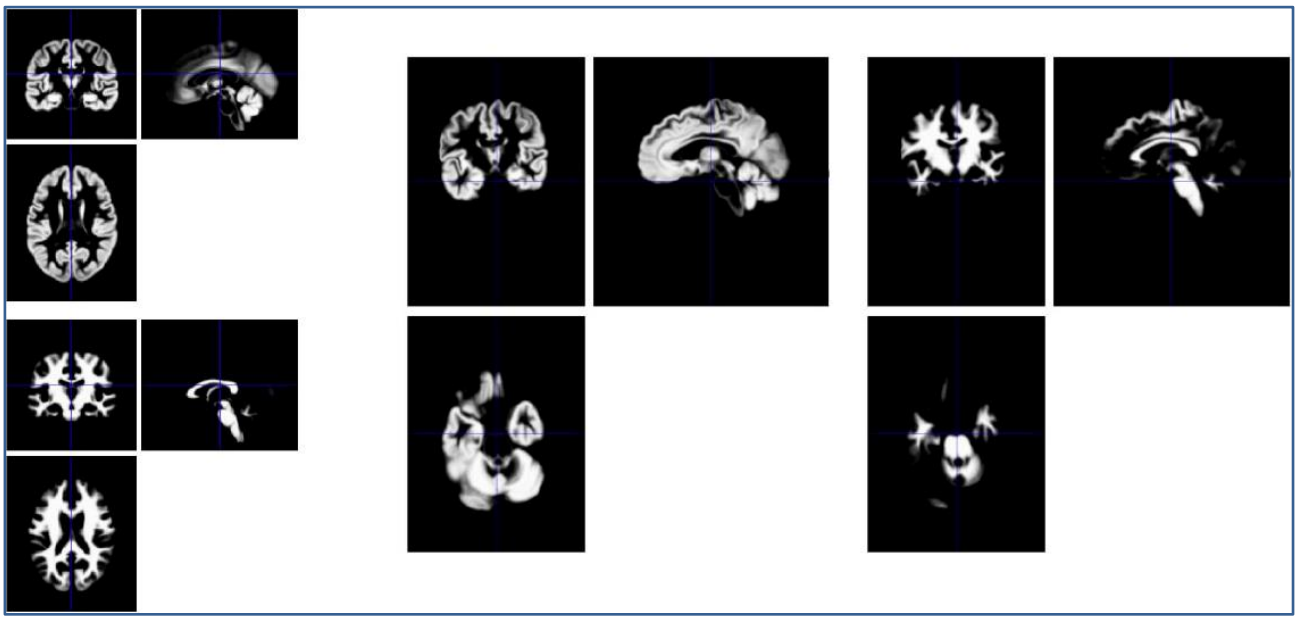


Figure 22: Grey and white matter from the original tissue atlases (left) along with registered versions (middle and right)

The rules of probability are then used to combine the various images to give a probabilistic label map for each brain structure. These probabilities are summed for each structure, to provide probabilistic volume estimates. These estimates are saved in the MIP platform as brain morphometric features.

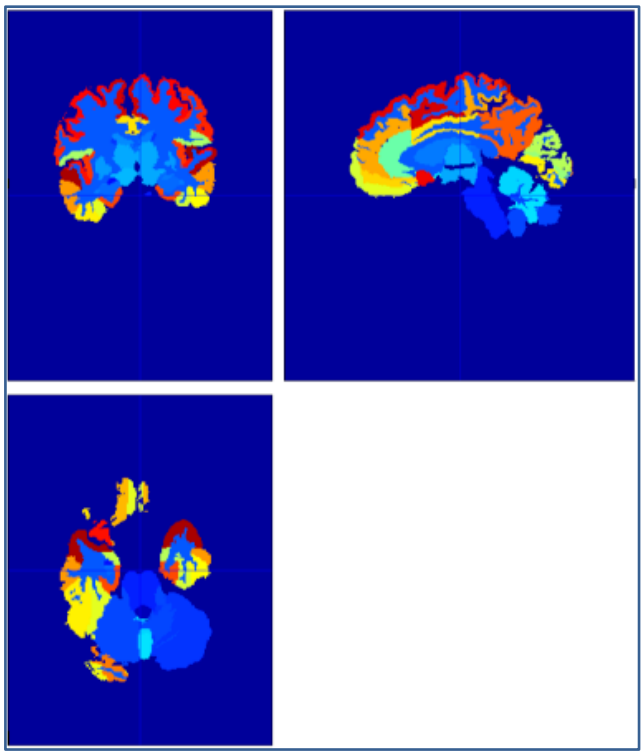


Figure 23: Automatically labelled image, showing most probable macro anatomy structure labels

While Voxel-based morphometry classifies tissue types and measures anatomical shape (Brain Segmentation and Normalisation component), the Voxel-Based Quantification component provides complementary information through its sensitivity to tissue microstructure. The Multi-parameter Mapping (MPM) imaging protocol is used to provide whole-brain maps of relaxometry measures ($R_1 = 1/T_1$ and $R_2^* = 1/T_2^*$), magnetisation transfer saturation (MT) and effective proton density (PD*) with the isotropic resolution of 1mm or higher.

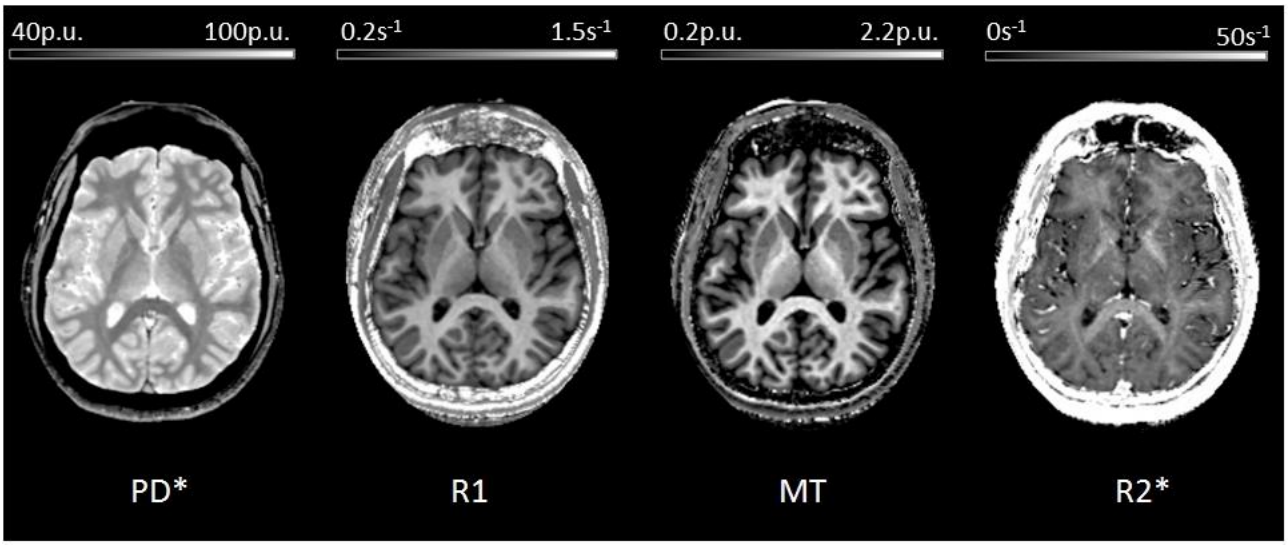


Figure 24: Multi Parameter Mapping high-resolution quantitative MRI acquisition protocol

MPM is a high-resolution quantitative imaging MRI protocol which, combined with VBQ data analysis, opens new windows for studying the microanatomy of the human brain *in vivo*. With T1-weighted images, the signal intensity is in arbitrary units and cannot be compared across sites or even scanning sessions. Quantitative imaging can provide absolute measures for neuroimaging biomarkers for myelination, water and iron levels comparable across imaging sites and in time.

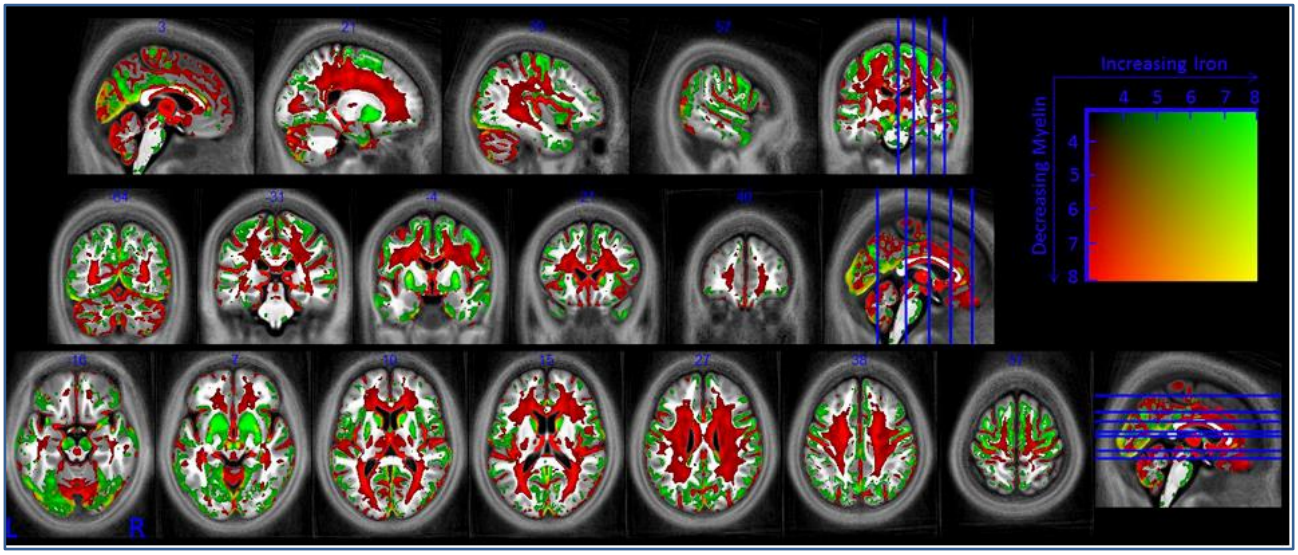


Figure 25: Voxel Based Quantification data analysis for studying microanatomy of the human brain *in vivo*

Brain Scan Metadata Extraction and EHR Data Extraction Pipelines

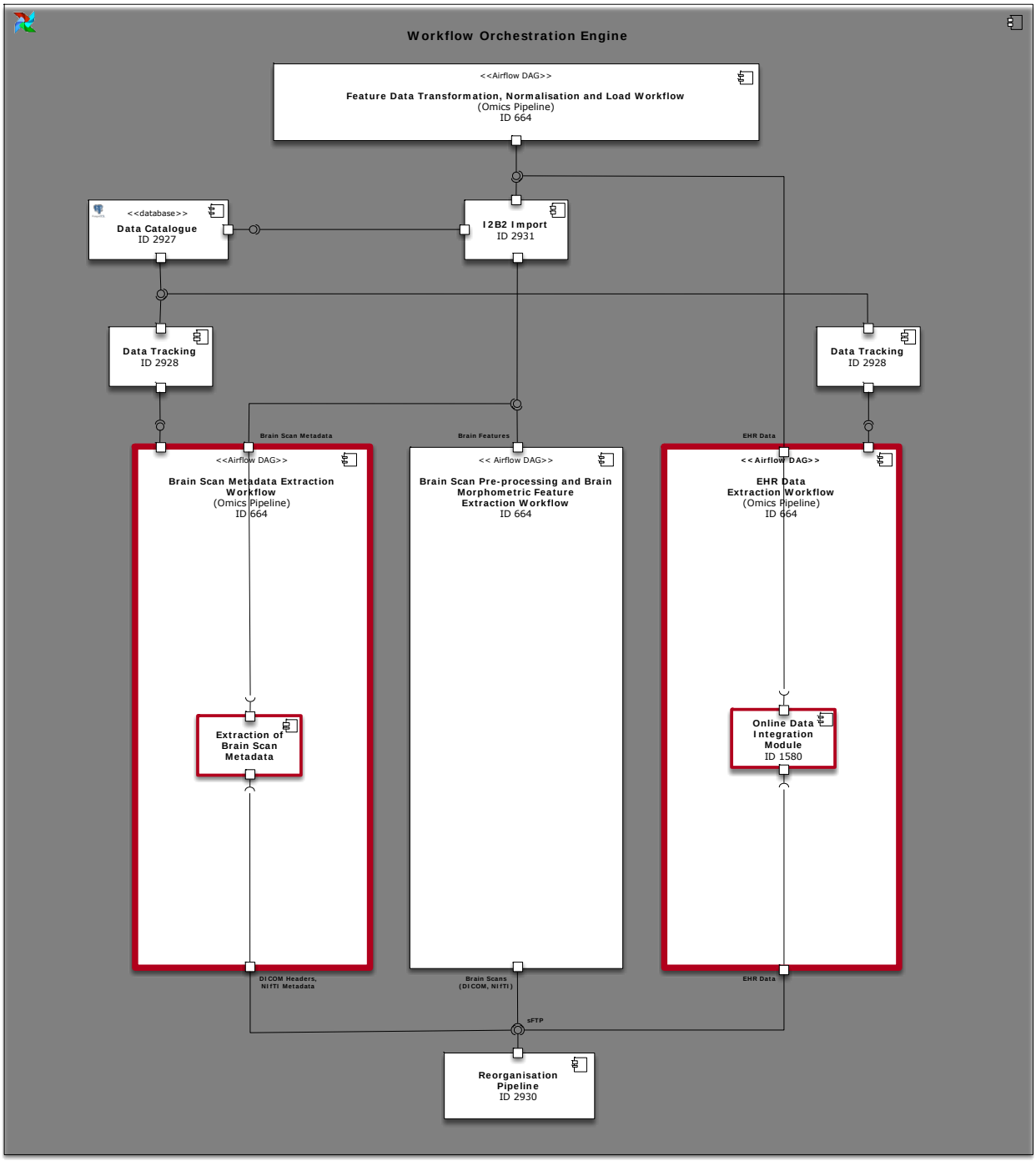


Figure 26: Brain Scan Metadata and EHR Data extraction pipelines

A patient's brain scan metadata and EHR data are extracted from the corresponding de-identified files and stored in I2B2 Capture Database alongside extracted brain morphometric features. Data provenance is stored in Data Catalogue.

Feature Data Transformation, Normalisation and Load Pipeline

This pipeline contains the following components:

- **Data Capture Database** – for storing patient health features extracted from brain scans and EHR files
- **Data Mapping and Transformation Specification** – data mapping rules – the results of harmonising data types from different source datasets into a common data element (CDE) model

- **Online Data Integration Module** – for transformation of the extracted patient feature data into the common data elements format, according to the Data Mapping and Transformation Specification rules. Also for exporting CDE Database to CSV file for storing the harmonised data into the local data store mirror (Features Table) in Features Data Store sub-system
- **Common Data Elements Database** – for permanently storing the transformed patient feature data into a normalised I2B2 schema

Data Capture Database

De-identified data, extracted from patient electronic health records and brain scans, is stored in the original data format in the Data Capture Database, implemented using I2B2 schema managed by PostgreSQL database management system.

The I2B2 schema allows for an optional direct update of Data Capture Database with data from a large number of I2B2-compliant anonymised patient cohort datasets. I2B2 is widely used for implementing clinical data warehouses as well as research data warehouses. Over the years, it became a de facto standard for bridging the gap between clinical informatics and bioinformatics, providing large datasets for clinical, biomedical and pharmaceutical research.

In cases when research datasets are stored in different formats, such as ADNI or BIDS files, they are initially saved in the Data Factory sub-system's version controlled storage before the data is extracted using the extraction pipelines and then finally stored in the Capture Database.

Data Mapping and Transformation Specification

The MIP Data Governance and Data Specification (DGDS) team receive information from hospitals about new data elements that shall be captured from patient EHR and brain scan datasets. In collaboration with hospital clinicians and data managers, the MIP DGDS team analyses new data types and harmonises them into a common data elements model. Data Mapping and Transformation Specification is updated with new harmonisation rules. This artefact is used for transformation of original data extracted from hospitals into the common data element format using the Online Data Integration Module.

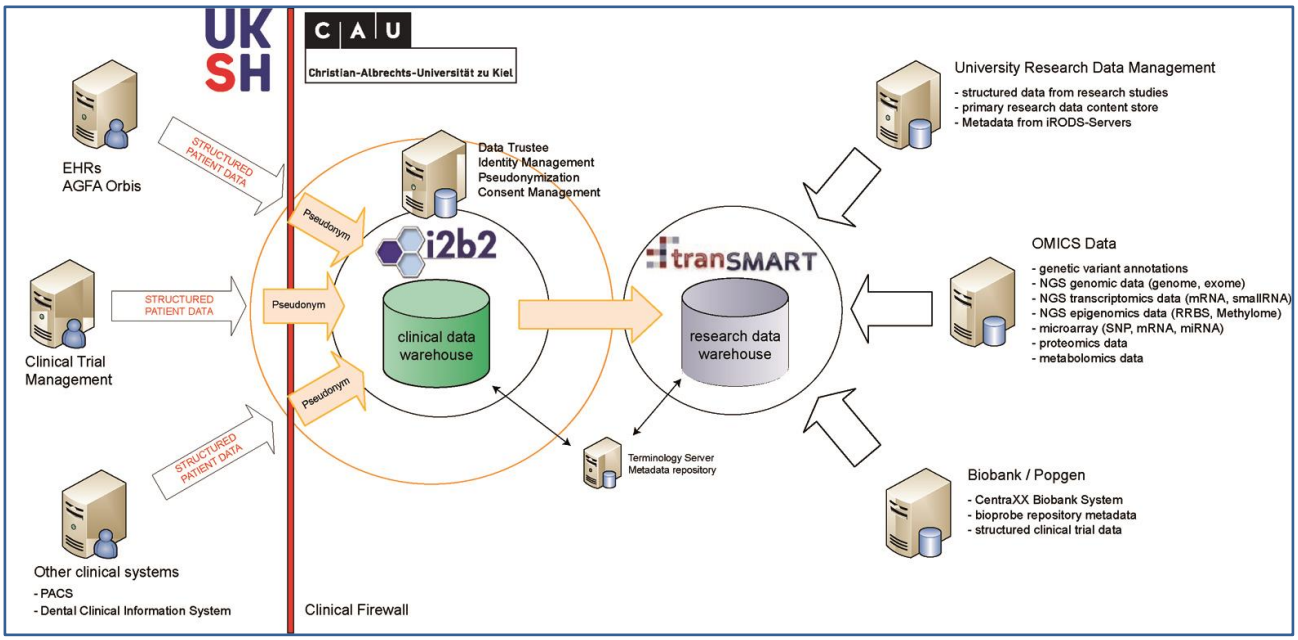


Figure 27: I2B2 tranSMART Foundation's research data warehouse for clinical, biomedical and pharmaceutical research

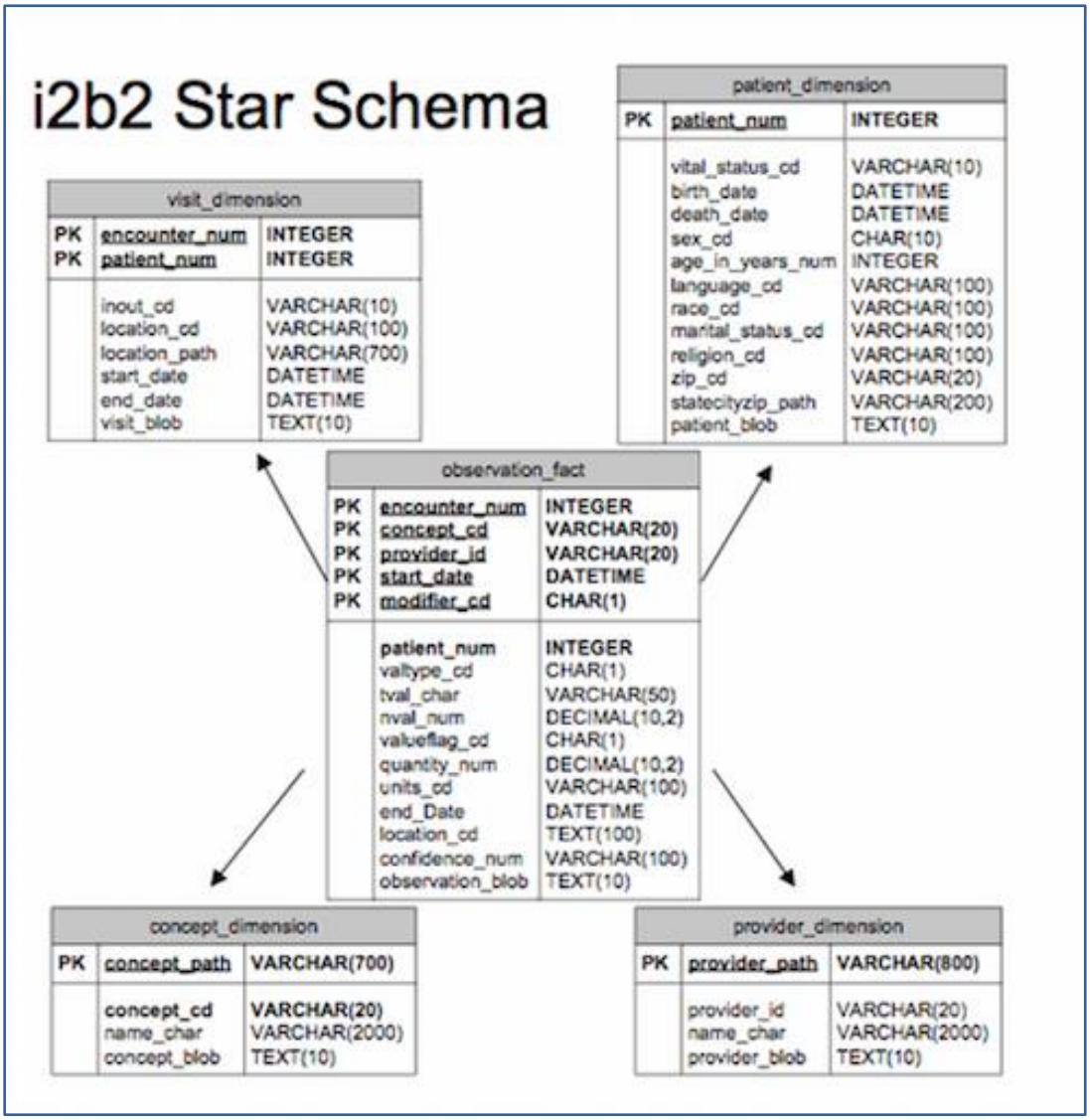


Figure 28: I2B2 Schema

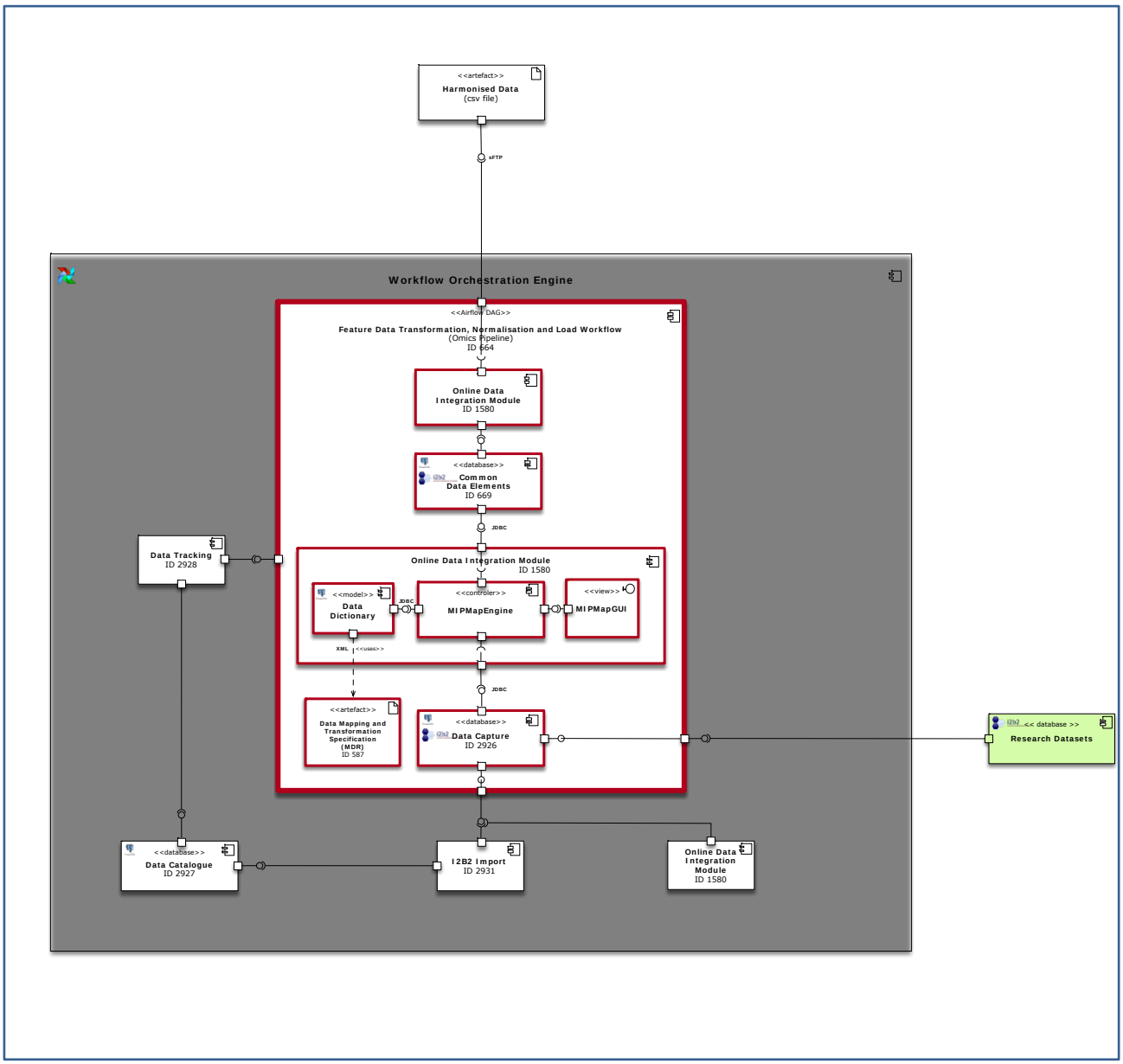


Figure 29: Feature Data Transformation, Normalisation and Load Pipeline

Online Data Integration Module for Data Transformation and Load to CDE Database

The Online Data Integration Module component is used for extracted data transformation, and loading into the normalised I2B2-compliant Common Data Element Database, managed by PostgreSQL database management system. This component is also used to export harmonised data from CDE Database to CSV files, out of which the Feature Table in the Feature Data Store sub-system is populated. The Online Data Integration Module is implemented using an open source ++Spicy data exchange tool. The adaptation of this application for the MIP is called MIPMap. This tool, which has been developed in Java using the NetBeans platform, applies Data Mapping and Transformation Specification rules for transformation of data stored in I2B2 Capture Database to the normalised I2B2 Common Data Elements Database.

MIPMap provides a graphical user interface where a hospital data manager or a MIP DGDS data manager can create mapping correspondences between source data elements and targets by drawing lines between them. This forms a mapping scenario that is stored in XML format. The mapping process is performed once for every hospital.

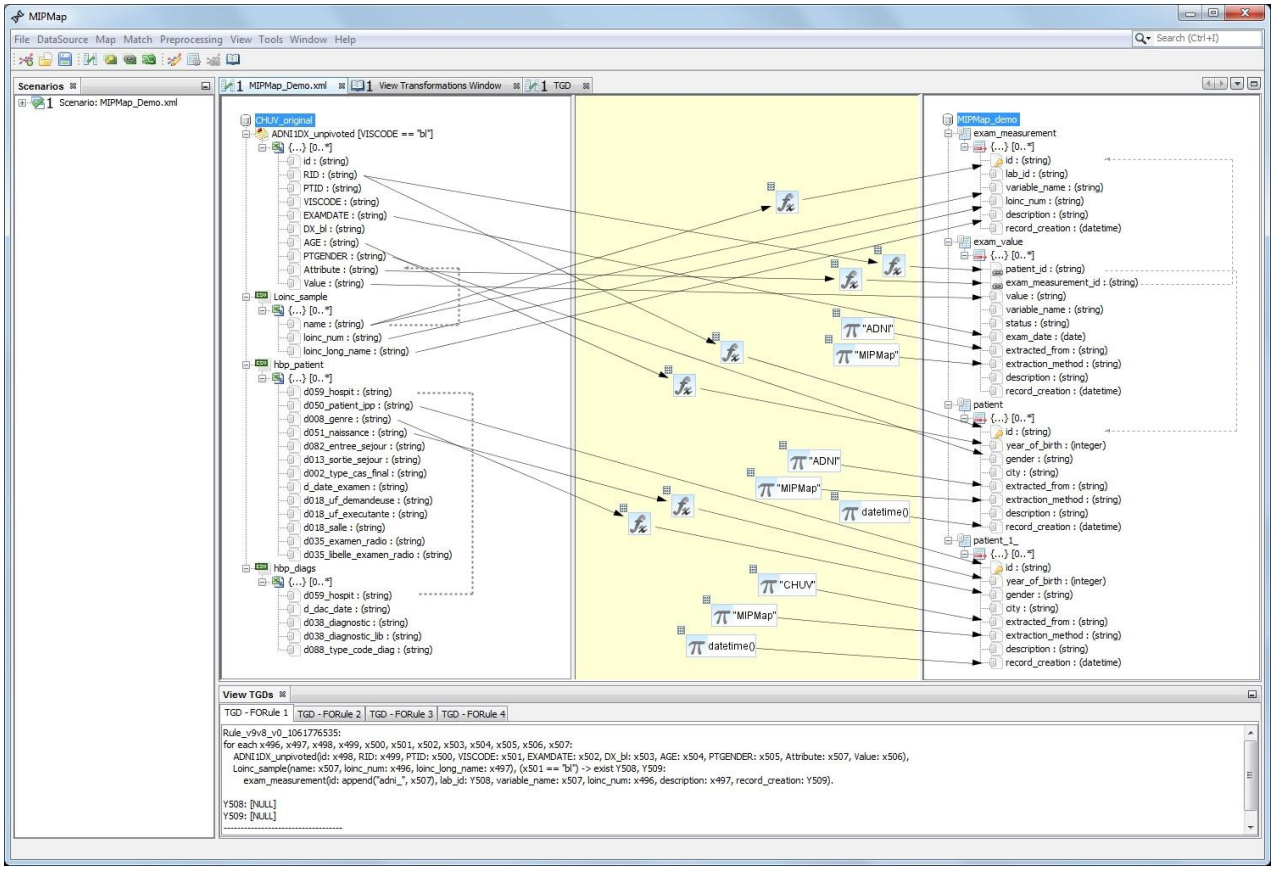


Figure 30: MIPMap user interface

Having created a mapping scenario, the MIPMap Engine generates an optimised SQL script that translates the data from the source (CSV file or a database schema) to the target database schema and then updates the target database.

Common Data Elements Database (delivered by CHUV team)

After the source data types have been mapped to the destination common data elements schema using the Data Mapping and Transformation Specification, the Online Data Integration Module loads the data from the Capture Database to the Common Data Elements Database.

An I2B2-compliant Common Data Elements (CDE) database schema is incorporated on top of the PostgreSQL database management system for permanently storing harmonised patient data from different hospitals and research datasets.

One of the key added-value characteristics of the MIP is the harmonisation of data elements from diverse source systems – EHR systems from different hospitals, imaging and PACS systems and research datasets. The harmonised data model is implemented as an I2B2-compliant database schema, which allows for a prospective easy integration with a large research datasets compliant with I2B2.

Online Data Integration Module for Transformation of CDE Database to Harmonised Data CSV File

Harmonised data from the CDE Database is transformed using the Online Data Integration Module component into a Harmonised Data CSV File in the Feature Data Store sub-system. The MIPMap Engine executes a pivoting script, for pivoting the variables and their values stored in the dimensional I2B2 (data mart) schema of CDE Database into a flat comma-separated value representation. The Harmonised Data CSV File is processed by the Query Engine and stored in the Feature Table to be available to the components of the Knowledge Extraction sub-system for data mining, statistical analysis and predictive machine learning.

3.1.3 Feature Data Store Subsystem

The Feature Data Store Sub-system contains components for mirroring harmonised patient data in the form appropriate for querying and using by machine learning algorithms. The components of this subsystem operate on and store the data belonging to one and only one hospital. The data is made available both for the local knowledge extraction MIP subsystem and to the remote, federated knowledge extraction MIP sub-system.

The components of the Feature Data Store sub-system are as follows:

- **Harmonised Data CSV file** – for mirroring harmonised CDE data exported from CDE database
- **Query Engine** – hospital DB back end, executing queries on extracted patient health sensitive data
- **Features Database** – hospital local data store mirror, data ready for querying and machine learning
- **PostgresRAW-UI** – user interface for Query Engine administration, including CSV files monitor

Harmonised Data CSV File

Using the Online Data Integration Module component, harmonised de-identified health-related patient data is exported from the CDE Database in the Data Factory sub-system into the CSV files accessible from the Feature Data Store sub-system components. The Query Engine component queries data stored in these files. The Query Engine also makes the data available for fetching by data mining and machine learning algorithms by storing it in the Hospital Dataset table of the Features Database.

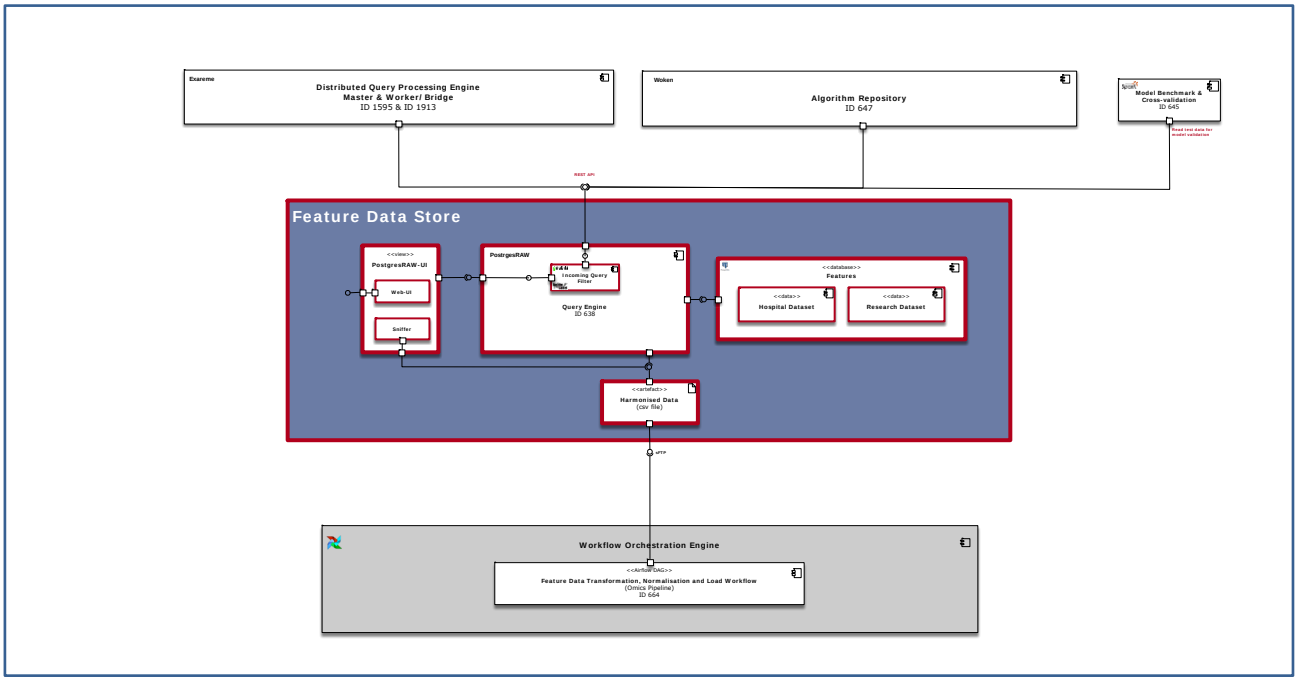


Figure 31: Feature Data Store Sub-system

Query Engine

The main purpose of the Query Engine component is to provide querying of the harmonised patient data stored in CSV files. The MIP Query Engine component is a database management system named PostgresRAW, based on PostgreSQL.

The input to the Query engine is data stored in CSV files. The output of the Query Engine is provided in JSON file format using REST services API or regular PostgreSQL connections.

Features Database

The Flat Hospital Dataset Table of the Features Database is updated with the data queried directly from the files. There it is made available for further querying by the Distributed Query Processing Engine or fetching by machine learning algorithms from the Algorithm Factory, both in the Knowledge Extraction sub-system.

The querying and fetching of data from the Feature Database is performed locally. For the privacy reasons, de-identified patient data is not allowed to be copied outside the hospital's MIP execution environment. The necessary computation is distributed throughout the hospital environments and only the results are fetched by the federation execution environment, either for visualisation or for further processing.

In addition to the Hospital Dataset flat table, the Features Database contains the Research Dataset flat table populated with the data captured from open research cohort datasets.

PostgresRAW-UI

PostgresRAW-UI automates detection and registration of raw files by providing a file monitor (Sniffer component). The folder containing the files with data that should update the Hospital Dataset table is provided as an argument when starting the database server.

3.1.4 Knowledge Extraction Subsystem

The components of the Knowledge Extraction sub-system are deployed both within the local hospital MIP execution environments and within the central MIP federation execution environment.

This MIP sub-system provides the functions for processing of the harmonised patient data, for local or distributed data mining and local or distributed execution of statistical inference and machine learning algorithms.

The two major complementary components of Knowledge Extraction sub-system are:

- **Algorithm Factory (Woken)** – orchestration of machine learning algorithm execution, including model benchmarking and cross-validation and storing of the trained models and their estimated predictive errors. Does not have out-of-the-box support for database query processing
- **Distributed Query Processing Engine (Exareme)** – query processing orchestration engine optimised for execution of distributed database queries extended with user-defined functions. Does not have out-of-the-box support for estimating trained machine learning model predictive errors

3.1.4.1 Algorithm Factory

Algorithm Orchestrator (Woken)

This component is a workflow orchestration platform, which runs statistical, data mining and machine learning algorithms encapsulated in Docker containers. Algorithms and their runtime are fetched from the Algorithm Repository, a Docker registry containing approved and compatible algorithms and their runtimes and libraries.

This component runs on top of the runtime environment containing Mesos and Chronos to control and execute the Docker containers over a cluster.

This component provides a web interface for on-demand execution of algorithms. It fetches the algorithms from the Algorithm Repository, monitors the execution of the algorithms also from the other execution environments in the cluster, collects the results formatted as a PFA document and returns a response to the web front end.

The Algorithm Orchestrator tracks data provenance information, runs model benchmarking and cross-validation of the models learned by the machine learning algorithms, using random K-Fold Sampling methods (Model Benchmark & Cross-validation), and stores PFA models in the Predictive Disease Model Repository.

Algorithm Repository

This component is a repository of Docker images that can be used by the Algorithm Orchestrator. It provides a workflow that allows contributors to provide new algorithms in a secured manner.

Algorithms, written in their native language (Python, MATLAB, R, Java, etc.), are encapsulated in a Docker container that provides them with the libraries and runtime environment necessary to execute this function. Currently, the MIP SGA1 platform supports Python-, Java- and R-based algorithms that are packaged in three Docker containers, respectively. The environment variables provided to the Docker container are used as algorithm parameters.

Algorithm Docker containers are autonomous:

- Connecting to the Features Database in the Features Data Store sub-system to retrieve feature data
- Processing data, taking into account Docker container environment variables
- Storing results into the Predictive Disease Model Repository

The Algorithm Registry database, implemented using PosgtreSQL database management system, is used to keep track of results created by the execution of an algorithm.

New algorithms can be easily integrated with the others by packaging them in the relevant Docker container. The supported algorithm results format is PFA, described in YAML or JSON configuration file. PFA enables vendor-neutral exchange and execution of complex predictive machine learning models. For visualisations, MIP SGA1 supports different formats, including Highcharts, Vis.js, PNG and SVG.

Machine learning algorithms planned for integrated by the end of SGA1 phase are:

Table 4: List of supported machine learning algorithms

Name	Methods	Federation/Local	PFA cross-validation
java-jsi-clus-fire	Clustering methods	Local	no
java-jsi-clus-fr	Clustering methods	Local	no
java-jsi-clus-pct-ts	Clustering methods	Local	no
java-jsi-clus-pct	Clustering methods	Local	yes
java-jsi-streams-modeltree	Tree-based methods	local	yes
java-jsi-streams-regressiontree	Tree-based methods	Local	yes
java-rapidminer-knn	Classification	Local	yes
java-rapidminer-naivebayes	Classification	Local	yes
python-anova	Classical inference	Local and Federation	yes
python-correlation-heatmap	Classical inference	Local and Federation	no
python-distributed-kmeans	Clustering	Local and Federation	yes
python-histograms	Descriptive	Local and Federation	no
python-jsi-hedwig	Tree-based	Local	no
python-jsi-hinmine	Tree-based	Local	no
python-knn	Classification	Local	yes
python-linear-regression	Predictive linear regression	Local and Federation	yes
python-longitudinal	Longitudinal analyses	Local	yes
python-sgd-regression	Gradient descent	Local and Federation	yes

Name	Methods	Federation/Local	PFA cross-validation
python-summary-statistics	Descriptive	Local and Federation	no
python-tsne	descriptive	Local	yes
r-3c	classification	Local	no
r-ggparci	Exploration	Local	no
r-heatmaply	Correlation	Local	no
r-linear-regression	Baysesian regression	Local and Federation	yes
Exareme k-means	Clustering	Federation	no
Exareme regression	Regression	Federation	no

Model Benchmark & Cross-validation

The Model benchmark and Cross-validation component is used to measure machine-learning models' accuracy. The results can guide the user to select the best-performing algorithm and fine-tune its parameters as well as to understand how well the model performs before it's used in production.

A model trained on training data needs to be validated. Its quality is measured by estimating its predictive error. Several techniques for assessing predictive errors exist, cross-validation being the most frequently used one. The predictive error is calculated by using the two disjoint datasets – training data set, to train the model, and test dataset to calculate the predictive error rate. The calculation of model predictive error rates is called validation.

Data used for both training and test datasets are stored in the Features Database, in the Features Data Store sub-system. The Model Benchmark & Cross-validation component performs data split using K-Fold cross-validation. This method of data sampling divides the complete dataset into K disjoint parts of roughly the same size. K different models are trained on K-1 parts each, while being tested on the remaining one part of the data. That is done on all K parts exactly once to ensure that every data row is used equally often for training and exactly once for testing. Resulting K test errors are then averaged to get the final error estimate of the model, which was built on the complete dataset.

The Algorithm Orchestrator stores the trained machine learning models and the results of cross-validations in the Predictive Disease Model Repository.

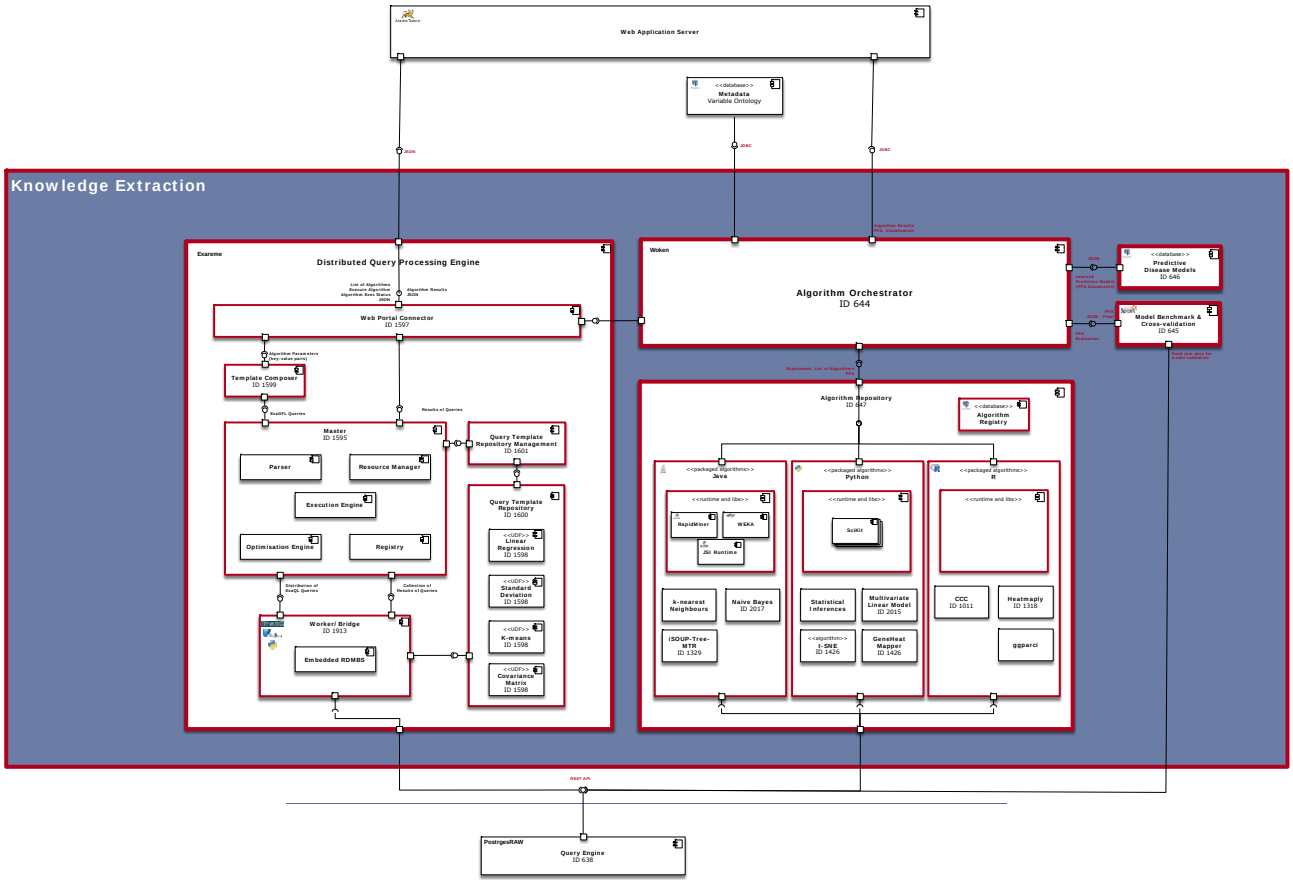


Figure 32: Knowledge Extraction Subsystem

Figure 33depicts the interaction between the Algorithm Factory components for a typical use case of running an experiment, ordered from the MIP Web sub-system.

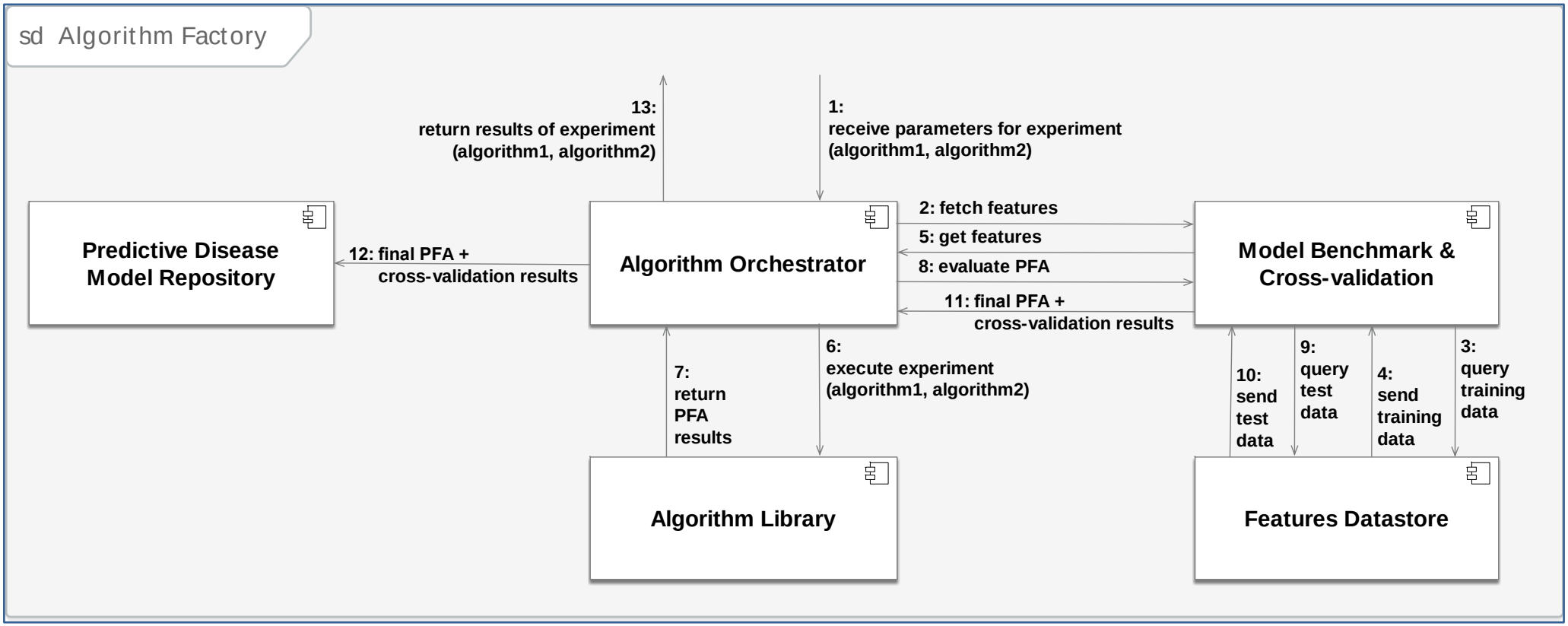


Figure 33: Algorithm Factory Communication Diagram

Predictive Disease Model Repository

This component serves as a permanent storage and search service for trained PFA models and their predictive error estimates.

3.1.4.2 Distributed Query Engine - Exareme

The Distributed Query Processing Engine plays a role in the Knowledge Extraction sub-system of the MIP platform. Master components deployed in the central federation node communicate with workers deployed in each of the hospitals, on one side, and with the Web sub-system components, on the other side. The Distributed Query Processing Engine does not allow direct communication between workers in different hospitals. Worker components, deployed in the hospitals, fetch the data from the local Feature Tables in the Features Data Store sub-system using the REST API and transfer the data to the master component for aggregation.

Systems Overview

The Distributed Query Engine, based on the open source project Exareme, is used as a traditional database system for: (1) data definition (creating, modifying, and removing database objects such as tables, indexes, and users), (2) data manipulation (data querying), and (3) external data import (from files or other databases). It is a distributed relational database management system extended with the support for complex field types – JSON, CSV and TSV.

The Distributed Query Engine uses a proprietary data manipulation language ExaDFL for specifying and orchestration of data processing. The Distributed Query Engine organises data processing in workflows designed as direct acyclic graphs (DAGs) – relational query operators are graph vertices, and the data flows between the operators are graph edges. ExaDFL is based on SQL extended with user-defined functions (UDFs) and data parallelism primitives. User-defined functions are used for specifying local data processing workflows and performing complex calculations on distributed data set partitions. ExaDFL primitives that support parallelism are declarative statements supporting parallel execution of partial queries on partitioned data sets.

The Distributed Query Engine translates ExaDFL queries to its internal declarative data manipulation language ExaQL, based on SQL-92 with extensions, for execution of query operators and user-defined functions on the distributed data set partitions.

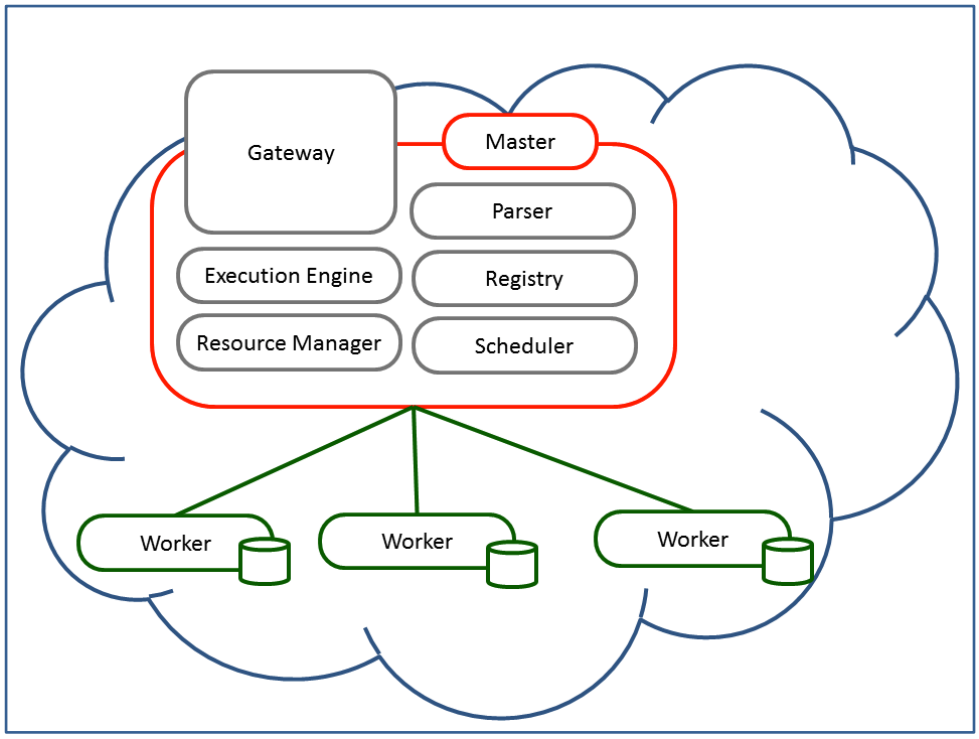


Figure 34: Distributed Query Engine Architecture Overview

The three main components of the Distributed Query Engine are:

1) **Worker** – an embedded SQLite relational database management system with Another Python SQLite Wrapper (APSW) – a Python API for SQLite running on the local hospital execution environments. It fetches local Feature Data Set sub-system's Features Table data set partitions needed for the execution of query operators and user defined functions and cashes those data set partitions to its local data storage for subsequent querying using ExaQL primitives. Worker is also a data processing system with functions for file import/export, keyword analysis, data mining tasks, fast indexing, pivoting, statistics and data processing workflow execution
2) **Master** – main entry point to the distributed query engine, running on the central federation execution environment, responsible for the coordination of the execution of other components. It aggregates query results transmitted by the Worker components distributed throughout the local hospital execution environments. Master consists of the following components:

- **Registry** – stores all information about the data and allocated resources, i.e. allocated data set partitions and their execution environments
- **Resource Manager** – allocates and de-allocates data processing resources on demand of the Execution Engine
- **Execution Engine** – requests allocation of resources from the Resource Manager, resolves the dependencies between the query operators to create a schedule of their execution in direct acyclic graph-oriented workflows, monitors the execution of the workflows and handles failures
- **Optimizer/Scheduler** – transforms ExaDFL queries into the distributed ExaQL statements and creates query execution plan by assigning operators to their respective workers
- **Gateway (Web Portal Connector)** – provides web for the communication between the Master component and the Web sub-system components

3) **Query Template Repository** – version-controlled source code store for the query templates in the form of User Defined Functions (UDFs). It is used both by the Worker and the Master components

Supported Data Processing Workflow Types

The source code of each algorithm is split into a set of local queries executed in parallel by the Worker components on the local data sets and one or more global processing executed by the Master component on the central federation node. The source code of each local and global data processing is written in a form of a workflow of SQL queries extended with user-defined functions. The source code is stored in .sql files in the Query Template Repository component. Supported data processing workflow types are:

1) **Local-global workflow** – local data processing executed in local execution environments, the aggregated results merged on the master node followed by additional data processing steps, if needed
2) **Multistep local-global workflows** – data processing workflow of predefined number of local-global data processing
3) **Iterative local-global workflows** – execution of the local-global data processing until a convergence criterion is reached (under development)

Algorithm Execution Steps

1) The Gateway component receives a user request for running an algorithm with submitted parameter values
2) The Template Composer fetches the stored local and global query templates needed for executing the selected algorithm from the Query Template Repository and creates an Algorithm template using ExaDFL primitives. Each algorithm template has an associated JSON

properties file that contains meta-information such as the algorithm's name, description, type, and parameters. Based on the type of the algorithm the type of the data processing workflow is determined

3) The Algorithm template that describes parameterized distributed workflows are forwarded to the Optimizer for generating the execution plan
4) The execution plan is forwarded to the Scheduler for determining partial algorithm execution plans, which are dispatched to Worker components running in local hospital execution environments
5) Each of the Workers executes the local data processing, then sends a confirmation of the successful execution to the Execution Engine in the central federation execution environment
6) Upon receiving success confirmations from all the Workers, the Scheduler determines global data processing plan and sends it to the Execution Engine
7) The Execution Engine then merges the aggregated results of all the workers, executes the global data processing plan and confirms its successful execution back to the Scheduler
8) The Scheduler checks if the complete local-global data processing plan has been completed
9) In case of the successful completion of the plan, it forwards the aggregated results to the user. In case the plan has not been completed, the Scheduler determines the next set of local data processing plans and the whole process of local-global plan execution is repeated until the successful completion of the algorithm

Overview of The Supported Features

The Distributed Query Processing Engine provides the following features:

1) List of the available algorithms
2) Requesting the execution of any of the available algorithms, and submission of relevant parameters
3) Execution status of the executing algorithms
4) Execution results of completed algorithms

The Distributed Query Processing Engine does not support automatic machine learning model validation. It does not provide out-of-the box predictive error estimation nor is there a component for recording the estimated accuracy of the trained machine learning models. The Algorithm Factory component can be used alongside the Distributed Query Processing Engine for trained model benchmarking and validation.

The MIP Distributed Query Processing Engine supports the following algorithms implemented as UDFs:

- **K-Means**
- **Linear Regression**

3.1.5 Web Subsystem

This section provides a brief overview of the functionality of the MIP Web sub-system. A detailed description of the front end functionality is provided in the MIP Web UI – User Guidelines, V2.0 Public Release:

(https://hbpmedical.github.io/documentation/HBP_SP8_UserGuide_latest.pdf).

The Web Sub-system provides a web portal and web applications for the end-users of the Platform. Users can explore only aggregated statistical data and perform data analysis using machine-learning methods provided by the Knowledge Extraction sub-system components. Web sub-system components have no direct access to the Feature Data Store sub-system where the individual

patient de-identified health-related feature data are stored. For privacy reasons, the MIP allows exploration only of statistical data.

The Web Sub-system provides the following applications:

- **Collaboration Space** – the landing page of the Medical Informatics Platform, displaying a summary of statistics (users, available variables, written articles), and the latest three shared models and articles. It also provides a link to the Article Builder web application
- **Data Exploration** - a statistical exploration of patient feature data (i.e. variables). It is possible to explore only statistically aggregated data, not information from an individual patient. This web application provides on-the-fly generation of the descriptive statistics and contains a caching mechanism to handle any future data import in an automated way. It uses information stored in a Metadata database to display additional information about the displayed statistical data, such as data acquisition methodology, units, variable type (nominal or continuous), etc. This web application provides the functionality to search, select and classify data elements as variables, co-variables and filters for configuration of the statistical or machine learning models.
- **Model Builder** - configuration/design of statistical or predictive machine learning models. It also provides visualisation for searching data element types, select and classify data elements as variables, co-variables (nominal and continuous) and filters. Once the model is designed, a design matrix is populated with the selected data. The Model Builder provides a visual representation of the design matrix and the selected data for inspection before running a statistical, feature extraction or machine learning algorithms. It also provides an option to save the designed models
- **Experiment Builder & Disease Models** – a selection of a statistical, feature extraction or machine learning method, the configuration of the method parameters and the parameters for the trained model validation for supervised machine learning, as well as launching of the machine learning experiment. This application displays experiment validation results as bar charts and confusion matrices
- **Article Builder** – writing articles using the results of the executed experiments
- **Third-party Applications and Viewers** – a portal for accessing third-party web applications for data exploration and visualisation

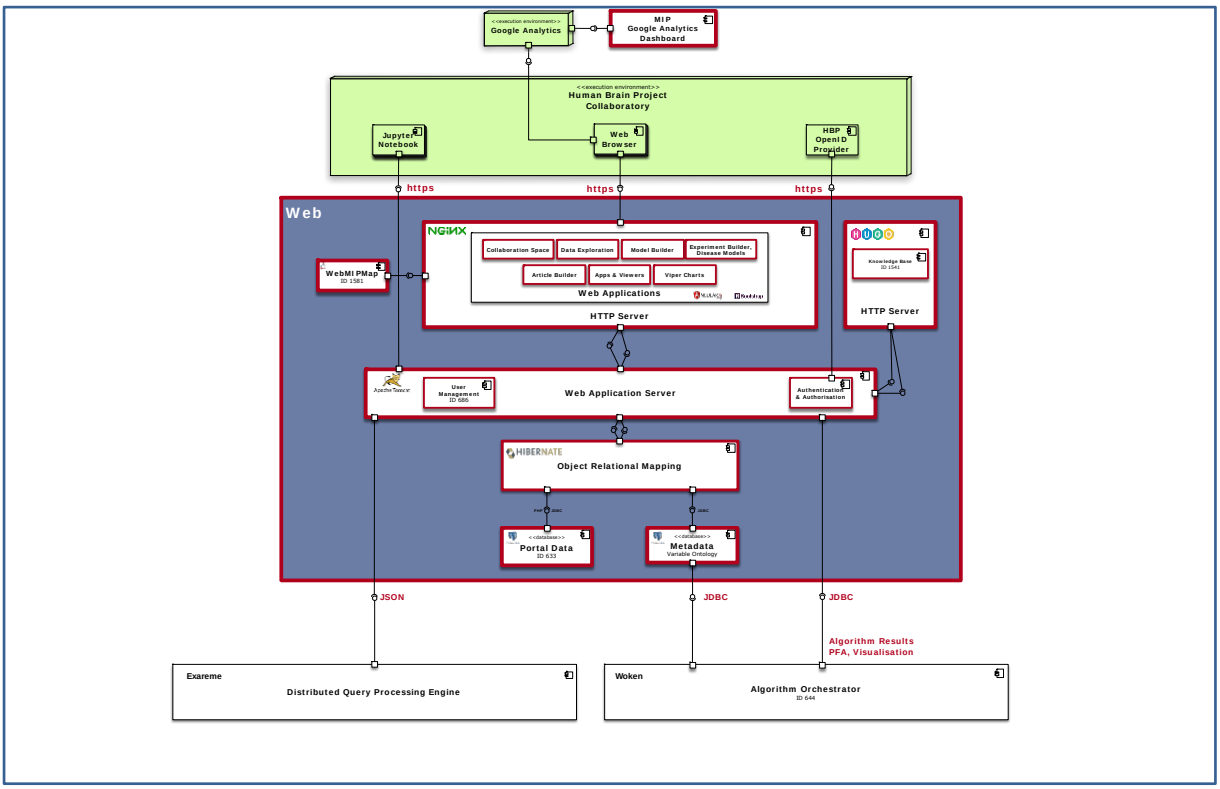


Figure 35: Web Subsystem

The Web-Sub-system allows access to its back end services and the Knowledge Extraction sub-system's Algorithm Factory through Jupyter notebooks running in the Human Brain Project's Collaboratory environment.

The Web Sub-system's Authentication and Authorisation component is integrated with the HBP Collaboratory's OpenID authentication service. The User Management component maintains an access control list and logging of user activities on the Data Exploration, Model Builder and Experiment Builder web applications.

Google Analytics Dashboard is set up for monitoring the usage of the Platform web services: tracking users and their behaviour and keeping an audit log with all user activities to detect potential Platform abuse and take preventive measures.

3.2 Deployment Architecture Overview

This section contains a brief overview of the key MIP deployment architecture concepts, relevant to understand the context of the MIP Software Installation use case specification.

A detailed description of the deployment architecture and its components is out of the scope of this document. It will be provided in the Deployment Specification document, including the following:

- Deployment model (execution environments, deployment artefacts and runtime components)
- Use case specifications (Software Installation, Data Preparation and Data Harmonisation)
- Deployment project configuration guide

3.2.1 Microservice Architecture

Each of the SP8 teams was focusing on delivering software components in their specific area of expertise using different technology stacks – Java, Python, R, MATLAB, Scala. As opposed to a monolithic application architecture, microservice architecture allowed the teams to work

independently in their specific functional areas: Web, distributed query processing, algorithm orchestration, data-mining, statistics and machine learning algorithms, integration and verification, local data store mirror, brain scan processing and ETL, data transformation and data harmonisation.

Another significant advantage of the microservice architecture is a possibility to adopt new technology and add new features incrementally. For example, encapsulated and loosely coupled permanent data storage can be replaced with a distributed big data-ready technology packaged and deployed in Docker containers, having no impact on the surrounding data processing, ETL and analytic software components.

Operating-system level virtualisation using Docker containers on top of Linux operating system has been chosen to build and deploy microservices and run corresponding processes. Software modules are packaged as Docker images and then integrated into a production version of the distributed MIP application using continuous integration software.

3.2.2 Docker Images as Microservices

MIP software developed by the HBP partners and 3rd party software components is packaged as microservices implemented as Docker images: independently deployable, small, loosely coupled services, each one running a unique process and communicating through a well-defined lightweight mechanism. Updating a component does not require redeployments of the entire application. MIP microservice deployment architecture supports a continuous integration and continuous deployment approach.

Docker image	Organisation	License	Build status	Image version	Image layers
docker hbpmip/flyway					
docker lren/xnat	CHUV LREN	license MIT	FAILED	version 1.6.5	993.6MB 34 layers
docker lren/labkey	CHUV LREN	license Apache-2.0		version latest	447.3MB 35 layers
docker hbpmip/woken	CHUV LREN	license Apache-2.0		version 2.1.4	144.9MB 25 layers
docker hbpmip/portal-backend	CHUV LREN	license AGPL-3.0	PASSED	version 2.5.4	121MB 21 layers
docker hbpmip/portal-frontend	CHUV LREN	license AGPL-3.0		version 2.5.2	32.6MB 25 layers
docker hbpmip/mipmap	EPFL DIAS	license MIT		version latest	65.9MB 21 layers
docker hbpmip/webmipmap	EPFL DIAS	license MIT		version latest	87.2MB 30 layers
docker hbpmip/postgresraw	EPFL DIAS	license MIT		version v1.0	13MB 20 layers
docker hbpmip/postgresraw-ui	EPFL DIAS	license MIT		version v1.2	36.9MB 12 layers
docker hbpmip/exaremelocal	UOA madgik	license MIT		version latest	852.9MB 42 layers

Figure 36: List of MIP Docker Images

3.2.3 Automated Installation and Configuration of MIP Software

Platform for fast deployment of services on bare metal or preconfigured virtual machines supporting clustering, security and monitoring is based on Cisco's Mantl rapid deployment project. The MIP is deployed on Mesos stack with added support for automated deployment/upgrade of services managed by Mesos Marathon and hardened security of the Ubuntu operating system. The services are built using Ansible scripts, unifying operation system configuration, middleware and application software deployment.

The MIP Hospital Deployment use cases planned for demonstration are specified in the next Chapter (MIP Software Installation Use Case Specification). Installation of the MIP in each new hospital is considered as a new git project, created as a clone of the generic Microservice Infrastructure project with configuration parameters updates tailored to a specific new hospital execution environment. Generic automatic MIP installation and configuration is stored and documented here:

https://github.com/HBPMedical/mip-microservices-infrastructure

3.2.4 MIP Software Installation Use Case Specification

The MIP microservice deployment architecture enables agile continuous integration and continuous deployment of components developed or modified by different European-wide teams. This architecture enables efficient future upgrades of the Platform with new technologies and new features needed to support evolved clinical needs. Automation of configuration and installation of the MIP software minimises IT efforts to keep the maximum focus on the scientific and clinical aspects of the projects.

Table 5 – Use Case Specification: Medical Informatics Platform Software Installation

Actors		HIT: Hospital IT Engineer MIT: MIP Deployment Engineer
Use Case Objective		Installation of the Medical Informatics Platform hardware and software in a hospital data centre
Pre-conditions		1) Formally approved investment in infrastructure, software, time and material 2) Signed Medical Informatics Platform Deployment and Evaluation Agreement 3) Infrastructure, software, time and material procured by hospital
Main Flow of Events		
Event ID	**Actor ID**	**Event Description**
E01	**HIT**	Prepare data centre for installing and configuring new MIP servers, storage and network

E02	**HIT**	Install MIP all-in-one server or separate servers (typical hospital configuration is provided below): a. Data capture and de-identification server CPU: 2-core x64; RAM: 2 GB; Storage: 50 GB; Security level: Highly secure clinical network b. Pre-processing server CPU: 12-core x64; RAM: 32 GB; Storage: 16 TB; Security Level: Secure research network c. Knowledge extraction and web server CPU: 8-core x64; RAM: 32 GB; Storage: 2 TB; Security Level: Secure research network or DMZ
E03	**HIT**	Install operating system on MIP servers • recommendation: Ubuntu 16.04 LTS / RHEL 7.2+ / CentOS 7.2+
E04	**HIT**	Provide sudo access rights for each MIP server to MIP deployment engineer
E05	**HIT**	Configure IPv4/IPv6 settings for each MIP server
E06	**HIT**	Configure SSH VPN tunnelling for remote connection with the MIP deployment team environment a. Install and run OpenSSH server on each MIP server b. Configure TCP port 22 for ingress SSH traffic on each MIP server c. Open port 22 for ingress traffic through firewall(s) between each MIP server and the Internet
E07	**HIT**	Configure TCP port 443 for egress HTTPS traffic on MIP servers and open port in firewall(s) for: a. Software package repositories (Ubuntu, Mesosphere, PyPI) b. Source code repositories (GitHub, Bitbucket, Launchpad, CHUV git) c. Docker registries (Docker Hub, CHUV private Docker registry)
E08	**MIT**	Install and configure MIP software automatically, using Ansible: a. Clone the generic Microservice Infrastructure project to create a git project for storing the new MIP environment configuration b. Prepare a configuration for automatic installation: • Install Python2 on the MIP servers – Ansible requires Python2 to run • Install MATLAB 2016b – required by SPM software for neuromorphometric processing • Server names and TCP/IP configuration c. Store the configuration in git, encrypt the passwords and confidential information d. Run a single Ansible script for the new MIP installation and configuration to: • Install middleware – libraries, runtimes, DBMSs and open source software • Deploy Docker images with software developed by MIP teams
E09	**MIT**	Confirm that all the processes are up and running from Marathon administrator's dashboard

E10	**MIT**	Backup the installation and configuration scripts on external server: • MIP team uses a private storage space on Bitbucket.org • Using the private repository, it is possible to safely and securely backup work, share it with other members of MIP for code review and receive upgrades of the platform
E11	**MIT**	Configure MIP backup for each MIP server in standard data centre backup environment
Special Requirements		Open relevant ports on firewalls, subject to the specific hospital IT security configuration
Post-conditions		5) MIP software is installed on all servers with all processes up and running 6) MIP platform is ready for data processing, storing and analysis
Scientific Added-value		7) The hospital data centre has a centralised platform for processing, storing and analysing de-identified and harmonised neuroimaging, neuropsychological, biological and demographic data of its patient population 8) Efficient, configurable and automated end-to-end software installation, unifying operation system configuration, middleware installation and microservice building minimises IT efforts to keep the focus on using the MIP platform for the scientific and clinical activities

4. MIP Product Structure

This section describes the version-controlled MIP product structure at the end of the SGA1 project phase and provides a detailed list of all the components. The MIP components are grouped in 4 groups, as illustrated in Figure 37.

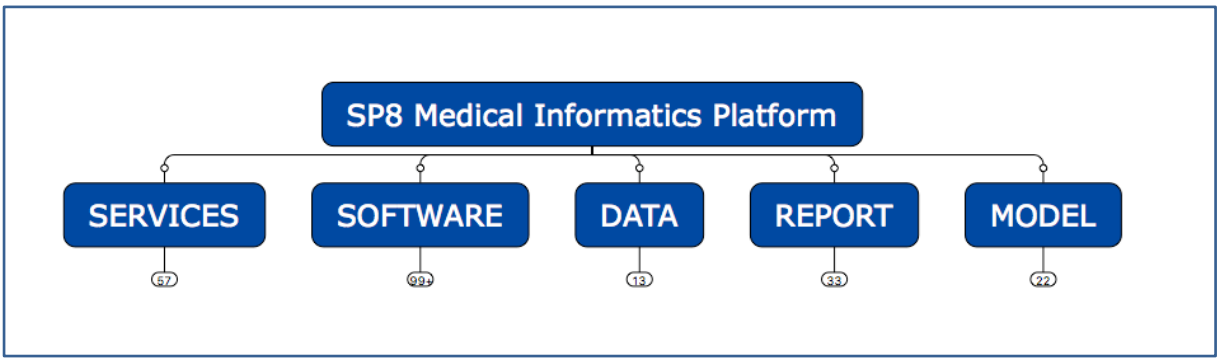


Figure 37: MIP Component Groups

The services component group contains artefacts of the research and development project activities other than software, data and models. The software group contains software components. The data group contains version controlled hospital data, metadata and test data. The report group contains communication and project management artefacts. The model group contains data models, such as disease models, reference brain models and patient de-identification profiles.

Within each of the MIP product structure groups, individual components are classified in the packages, i.e. building blocks, as illustrated in Figure 38.

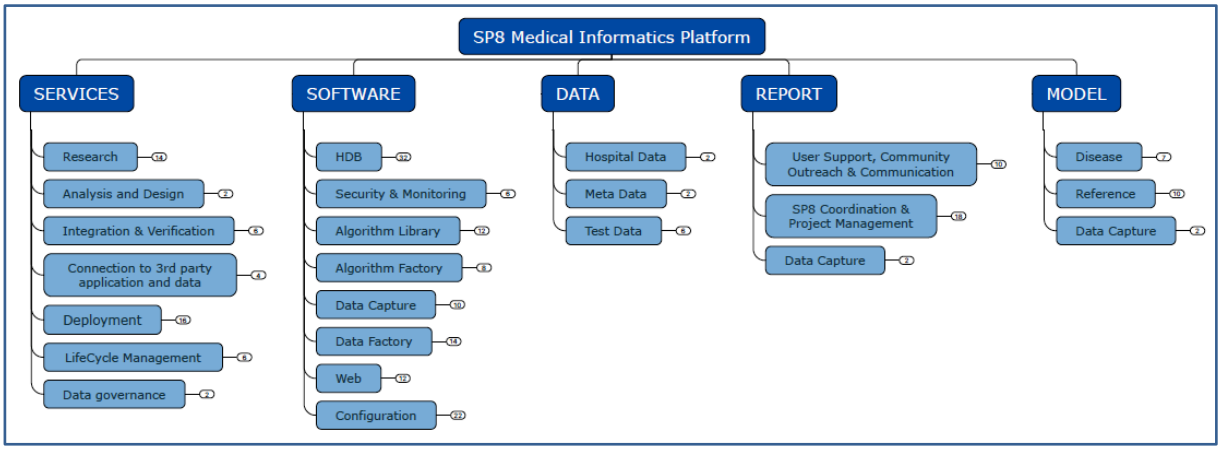


Figure 38: MIP Component Packages

The following five Figures (Figure 39, Figure 40, Figure 41, Figure 42, Figure 43) give a detailed breakdown of the component packages into individual components.

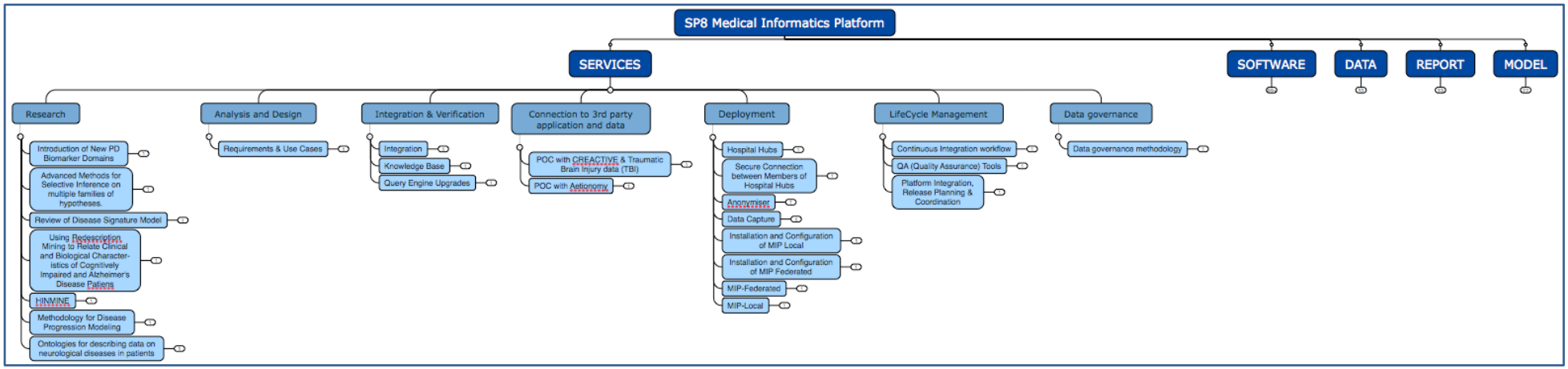


Figure 39: MIP Services Component Structure

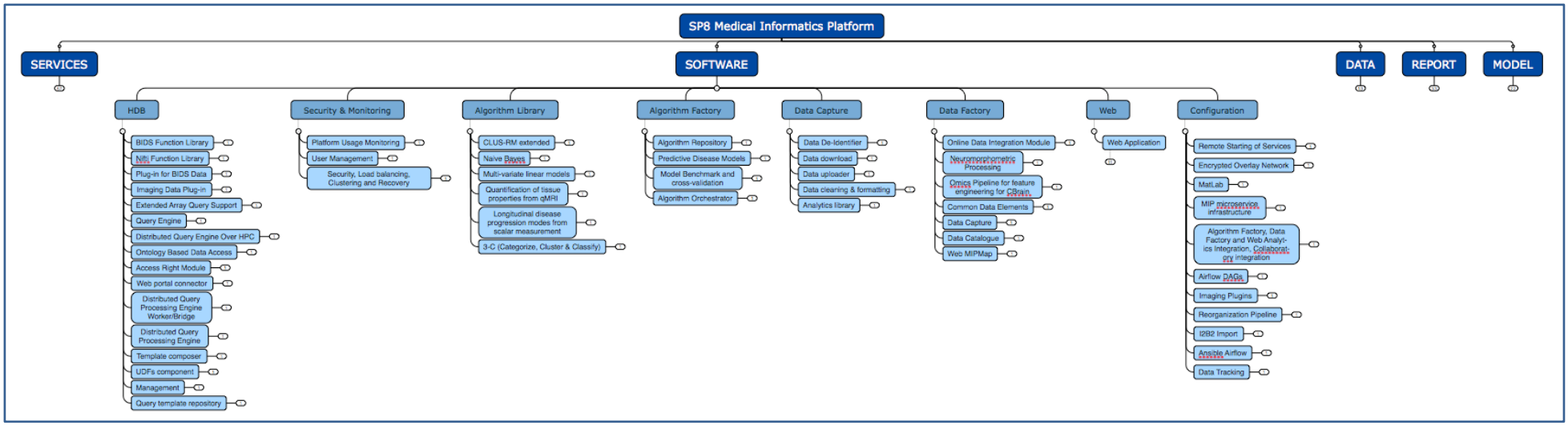


Figure 40: MIP Software Component Structure

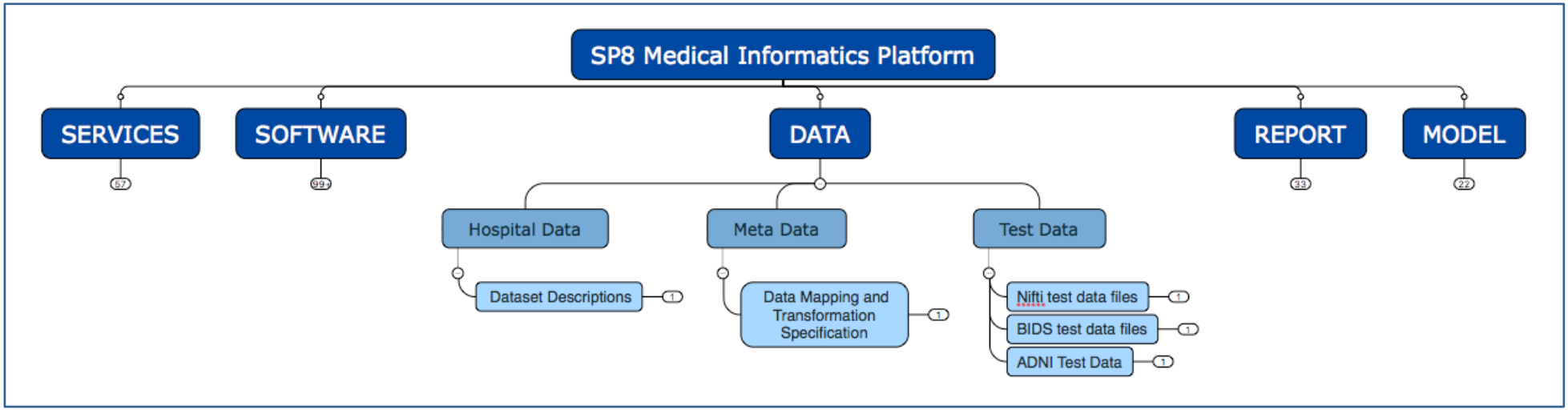


Figure 41: MIP Data Component Structure

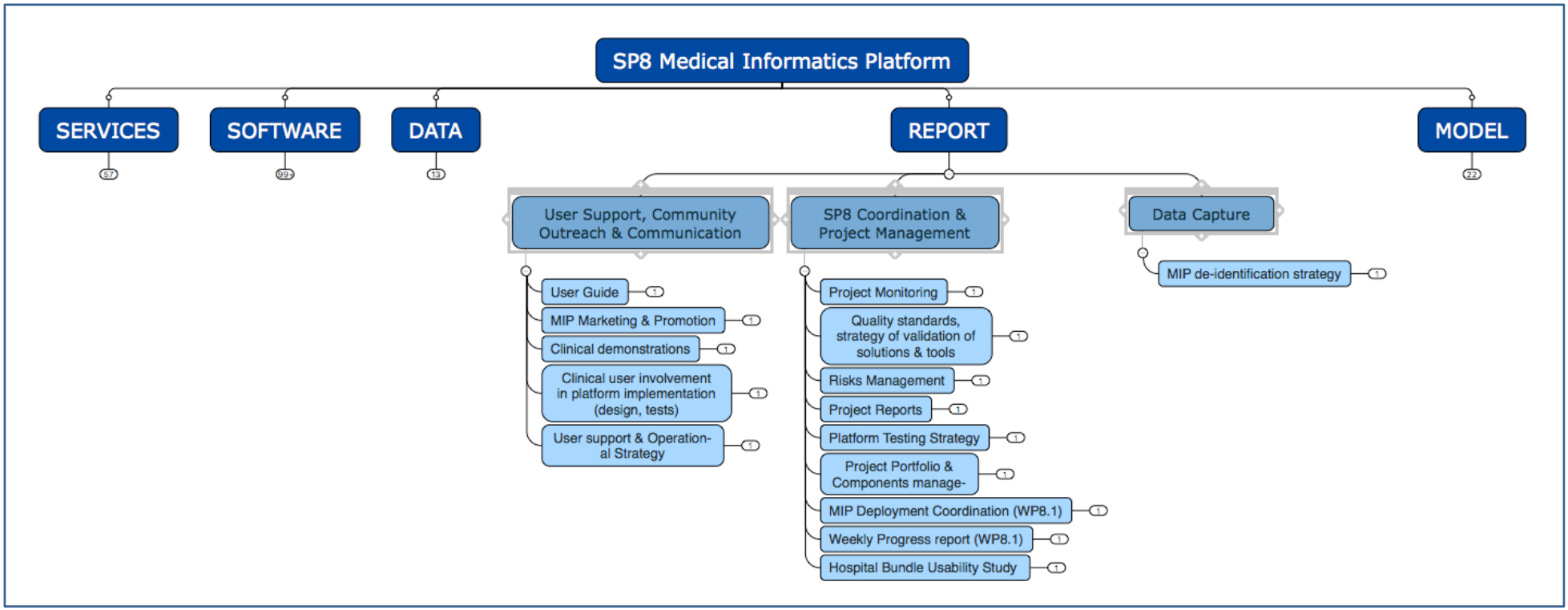


Figure 42: MIP Report Component Structure

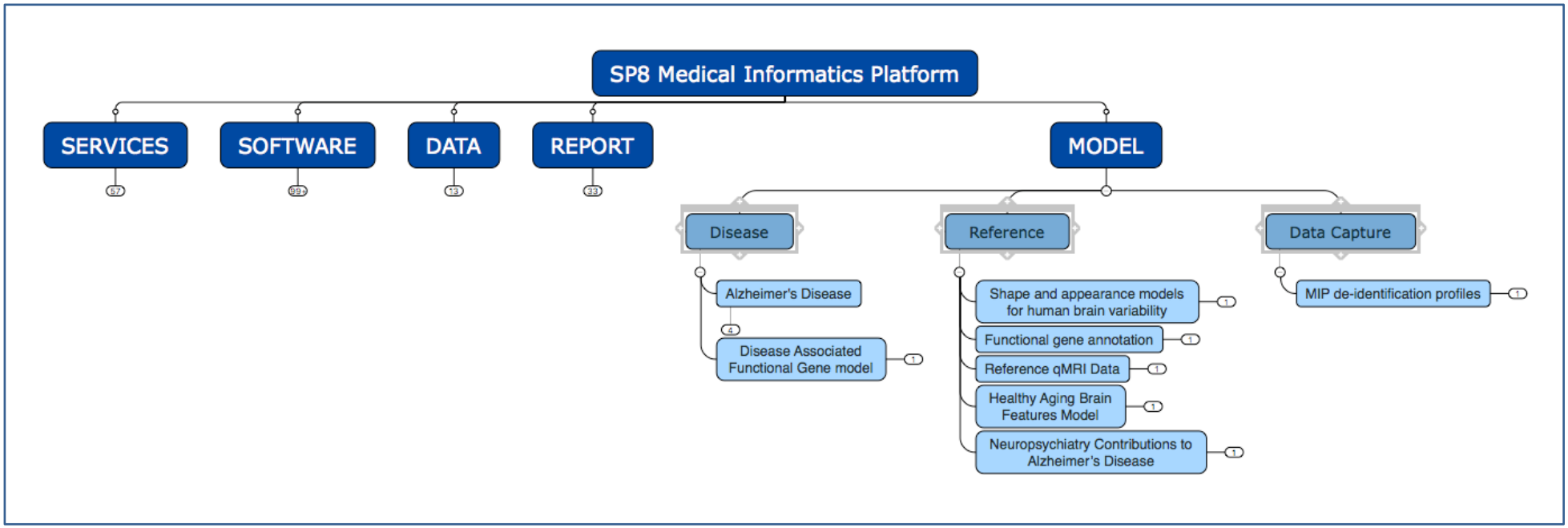


Figure 43: MIP Model Component Structure

4.1 Software Components

The software group contains software packages and corresponding components.

Table 6: MIP Software Components

building block	component	ID	task	WP
Algorithm Factory	Algorithm Repository	647	T8.5.2	WP8.5
Algorithm Factory	Predictive Disease Models	646	T8.5.2	WP8.5
Algorithm Factory	Model Benchmark and cross-validation	645	T8.5.2	WP8.5
Algorithm Factory	Algorithm Orchestrator	2938	T8.5.2	WP8.5
Algorithm Library	CLUS-RM extended	1329	T8.3.5	WP8.3
Algorithm Library	Naive Bayes	2017	T8.4.2	WP8.4
Algorithm Library	Multi-variate linear models	2015	T8.4.2	WP8.4
Algorithm Library	Quantification of tissue properties from qMRI	1287	T8.4.3	WP8.4
Algorithm Library	Longitudinal disease progression modes from scalar measurement	2416	T8.3.12	WP8.3
Algorithm Library	3-C (Categorize, Cluster & Classify)	1011	T8.3.1	WP8.3
Algorithm Library	Web Application > Heatmaply	1318	T8.3.2	WP8.3
Configuration	Remote Starting of Services	1759	T8.1.3	WP8.1
Configuration	Encrypted Overlay Network	1760	T8.1.3	WP8.1
Configuration	MatLab	665	T8.5.2	WP8.5
Configuration	MIP microservice infrastructure	102	T8.5.2	WP8.5
Configuration	Algorithm Factory, Data Factory and Web Analytics Integration, Collaboratory integration	2939	T8.5.2	WP8.5
Configuration	Airflow DAGs	664	T8.5.2	WP8.5
Configuration	Imaging Plugins	2929	T8.5.2	WP8.5
Configuration	Reorganization Pipeline	2930	T8.5.2	WP8.5
Configuration	I2B2 Import	2931	T8.5.2	WP8.5
Configuration	Ansible Airflow	2932	T8.5.2	WP8.5
Configuration	Data Tracking	2928	T8.5.2	WP8.5
Data Capture	Data De-Identifier	2940	T8.1.1	WP8.1

building block	component	ID	task	WP
Data Capture	Data download	2865	T8.4.5	WP8.4
Data Capture	Data uploader	2862	T8.4.5	WP8.4
Data Capture	Data cleaning & formatting	2863	T8.4.5	WP8.4
Data Capture	Analytics library	2864	T8.4.5	WP8.4
Data Factory	Omics Pipeline for feature engineering for Cbrain	670	T8.5.2	WP8.5
Data Factory	Online Data Integration Module	1580	T8.1.4	WP8.1
Data Factory	Neuromorphometric Processing	671	T8.5.2	WP8.5
Data Factory	Common Data Elements	669	T8.5.2	WP8.5
Data Factory	Data Capture	2926	T8.5.2	WP8.5
Data Factory	Data Catalogue	2927	T8.5.2	WP8.5
Data Factory	WebMIPMap	1581	T8.1.4	WP8.1
HDB	BIDS Function Library	1754	T8.1.1	WP8.1
HDB	Nifti Function Library	1753	T8.1.1	WP8.1
HDB	Plug-in for BIDS Data	1752	T8.1.1	WP8.1
HDB	Imaging Data Plug-in	1751	T8.1.1	WP8.1
HDB	Extended Array Query Support	1750	T8.1.1	WP8.1
HDB	Query Engine	638	T8.1.1	WP8.1
HDB	Distributed Query Engine Over HPC	1755	T8.1.2	WP8.1
HDB	Ontology Based Data Access	1579	T8.1.4	WP8.1
HDB	Access Right Module	1578	T8.1.4	WP8.1
HDB	Web portal connector	1597	T8.1.5	WP8.1
HDB	Distributed Query Processing Engine Worker/Bridge	1596	T8.1.5	WP8.1
HDB	Distributed Query Processing Engine Master	1595	T8.1.5	WP8.1
HDB	Template composer	1599	T8.1.6	WP8.1
HDB	UDFs component	1598	T8.1.6	WP8.1
HDB	Management	1601	T8.1.7	WP8.1

building block	component	ID	task	WP
HDB	Query template repository	1600	T8.1.7	WP8.1
Security & Monitoring	Platform Usage Monitoring	685	T8.5.1	WP8.5
Security & Monitoring	User Management	686	T8.5.1	WP8.5
Security & Monitoring	Security, Load balancing, Clustering and Recovery Services	684	T8.5.2	WP8.5
Web	Web Application > Knowledge Base > Research Dataset List	2286	T8.2.2	WP8.2
Web	Web Application > Brain insight > GeneHeatMapper	1426	T8.3.10	WP8.3
Web	Web Application > Portal DB (articles, experiments, models)	633	T8.2.3	WP8.2
Web	Web Application > Knowledge Base	1541	T8.2.3	WP8.2

4.2 Service Components

The services component group contains artefacts of research and development project activities other than software, data and models.

Table 7: MIP Service Components

building block	component	ID	task	WP
Analysis and Design	Requirements & Use Cases	690	T8.6.1	WP8.6
Connection to 3rd party application and data	POC with CREACTIVE & Traumatic Brain Injury data (TBI)	1560	T8.5.2	WP8.5
Connection to 3rd party application and data	POC with Aetionomy	1562	T8.5.2	WP8.5
Data governance	Data governance methodology	687	T8.6.2	WP8.6
Deployment	Hospital Hubs	1757	T8.1.2	WP8.1
Deployment	Secure Connection between Members of Hospital Hubs	1756	T8.1.2	WP8.1
Deployment	Anonymiser	1816	T8.1.3	WP8.1
Deployment	Data Capture	1815	T8.1.3	WP8.1
Deployment	Installation and Configuration of MIP Local	1817	T8.1.3	WP8.1

building block	component	ID	task	WP
Deployment	Installation and Configuration of MIP Federated	1818	T8.1.3	WP8.1
Deployment	MIP-Federated	1557	T8.6.2	WP8.6
Deployment	MIP-Local	1556	T8.6.2	WP8.6
Integration & Verification	Knowledge Base	617	T8.5.1	WP8.5
Integration & Verification	Integration	1819	T8.1.3	WP8.1
Integration & Verification	Query Engine Upgrades	1820	T8.1.3	WP8.1
LifeCycle Management	Continuous Integration workflow	1551	T8.5.1	WP8.5
LifeCycle Management	QA (Quality Assurance) Tools	1552	T8.5.2	WP8.5
LifeCycle Management	Platform Integration, Release Planning & Coordination	1555	T8.5.3	WP8.5
Research	Introduction of New PD Biomarker Domains	1021	T8.3.1	WP8.3
Research	Advanced Methods for Selective Inference on multiple families of hypotheses.	1319	T8.3.3	WP8.3
Research	Review of Disease Signature Model	1321	T8.3.4	WP8.3
Research	Using Redescription Mining to Relate Clinical and Biological Characteristics of Cognitively Impaired and Alzheimer's Disease Patients	1323	T8.3.6	WP8.3
Research	HINMINE	1325	T8.3.7	WP8.3
Research	Methodology for Disease Progression Modelling	1424	T8.3.8	WP8.3
Research	Ontologies to describe neurological disease patient data	1331	T8.3.9	WP8.3

4.3 Data Components

The data group contains version controlled hospital data, metadata and test data.

Table 8: MIP Data Components

building block	component	ID	task	WP
Hospital Data	Dataset Descriptions	455	T8.2.1	WP8.2

building block	component	ID	task	WP
Meta Data	Data Mapping and Transformation Specification	587	T8.2.1	WP8.2
Test Data	Nifti test data files	1734	T8.1.1	WP8.1
Test Data	BIDS test data files	1733	T8.1.1	WP8.1
Test Data	ADNI Test Data	2716	T8.5.2	WP8.5

4.4 Model Components

The model group contains data models, such as disease models, reference brain models and patient de-identification profiles.

Table 9: MIP Model Components

building block	component	ID	task	WP
Data Capture	MIP de-identification profiles	2936	T8.1.3	WP8.1
Disease	Alzheimer's Disease > Longitudinal ADNI Dataset	1013	T8.3.1	WP8.3
Disease	Alzheimer's Disease > Brain Features Model	608	T8.4.1	WP8.4
Disease	Disease Associated Functional Gene model	1605	T8.4.1	WP8.4
Reference	Shape and appearance models for human brain variability	2171	T8.3.11	WP8.3
Reference	Functional gene annotation	1604	T8.4.1	WP8.4
Reference	Reference qMRI Data	1305	T8.4.3	WP8.4
Reference	Healthy Aging Brain Features Model	611	T8.4.1	WP8.4
Reference	Neuropsychiatry Contributions to Alzheimer's Disease	2018	T8.4.2	WP8.4

5. MIP Data Management

Software architecture-related prerequisites for cross-centre data analytics are:

- a hybrid community and private execution environment deployment model
- a microservice architecture coupled with continuous integration and continuous deployment technology
- a distributed patient data storage and federated algorithm execution

This distributed, patient privacy preserving software architecture is a necessary but not sufficient condition for multi-dataset clinical studies comprising patient data from hospitals and open research cohort datasets. The datasets have overlapping data types but different ontological representations. Data is described, stored and formatted in different data structures. For execution of multi-dataset analytics, data models need to be harmonised in a common MIP data model, which is, together with dataset-specific data models, shared and synchronised between the distributed private hospital instances and community execution environment (Figure 1).

Data model harmonisation is, therefore, a key technology enabler for cross-centre multiple dataset clinical studies. It is a well-defined process supported by the application ontology software architecture, and the organisation, which establishes and maintains the rules and controls quality and integrity of the data harmonisation process.

The Data Governance and Data Selection (DGDS) committee is a centrally coordinated MIP organisational entity responsible for establishing and maintaining data governance methodology and data harmonisation rules. The members of the DGDS committee are MIP software architects, members of the expert medical committee consisting of medical doctors and clinical researchers of participating hospitals and institutes, and data managers, from the participating hospitals and the MIP R&D team.

Data harmonisation and re-harmonisation is an on-going process. With the introduction of a new dataset the whole process has to be repeated, starting with the analysis of the incoming dataset and ending with the synchronisation of (re-)harmonised data models across the distributed MIP ecosystem.

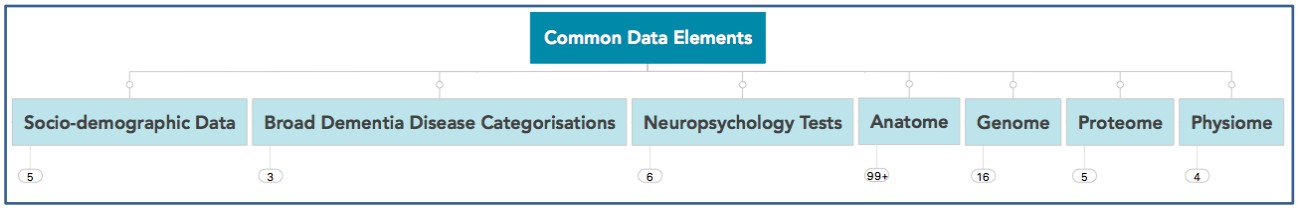


Figure 44: MIP SGA1 Common Data Elements

A diagnostic framework, i.e. the type and categorisation of diagnoses is typically hospital specific. Each hospital has its own naming classification of diseases. It is usually based on a standard classification, like ICD-10, but often a more detailed classification is needed for some disease domains. In case of dementia disorders, for example, CHRU Lille has adopted the recommendation of the Banque National Alzheimer (BNA). The CHUV in Lausanne has recently provided their adaptation of the BNA disease classification, which is planned for integration in the next release of the MIP. System validation has been based on the old ICD-10 classification. To have multi-dataset analytics involved in diagnoses, the MIP has introduced three broad disease categories – Alzheimer's Disease, Parkinson's Disease and Neurodegenerative Diseases. These broad disease categories are mapped to the disease definitions of each of the disease frameworks of the participating hospitals.

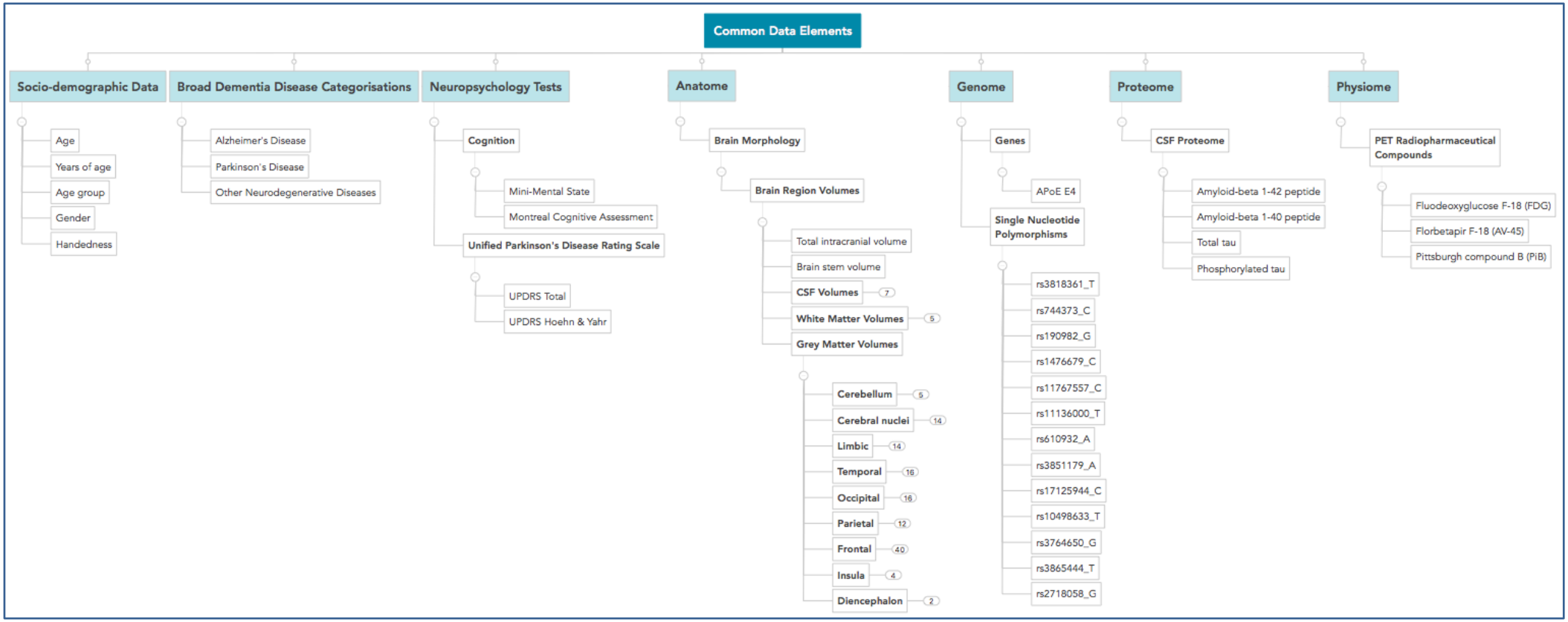


Figure 45: MIP SGA1 Common Data Element Taxonomy[1]

[1] **Note:** Brain Volume group contains 135 data elements representing brain regions classified using the standard brain anatomy classification

Table 10: High-level data model harmonisation process description

Activity Number	Activity	Description
1	Analysis of the new dataset	Initial profiling of the original de-identified patient data exported from EHR's and research datasets in CSV format, stored in clinical research data warehouses or other OLAP systems (for example I2B2). Analysis of the brain scan dataset, including the number of scan sessions, and preliminary examination of the DICOM file header information
2	Understanding the meaning of the data	Analysis of the formats and structures of received datasets. Informal description of the original data types confirmed and approved by the originating hospital/institute experts
3	Creation of data vocabularies / application ontologies	Creation of data vocabularies / MIP application ontologies for: socio-demographic data, brain regions, genome, proteome, metabolome, physiome, phenome. Creation of a hospital-specific diagnostic framework, including mapping to MIP broad disease categories. Creation of a hospital-specific neuropsychological assessment framework
4	Re-harmonisation of the common data model	Updating of the MIP common data model in coordination with expert representatives of participating hospitals and institutes
5	Update and formal approval of the Data Mapping and Transformation Specification	Formal, version controlled specification of the harmonisation and naming rules updated and formally approved by originating hospital/institute experts, MIP medical consultants and MIP software architects
6	Integration of common and dataset-specific data models	Integration and verification of common and dataset-specific data models in the MIP testing environment. Regression testing using the open research cohort datasets
7	Synchronisation of harmonised data models across the distributed MIP ecosystem	Synchronisation of (re-)harmonised data models – common hospital/insitute-specific across the private hospital/institute MIP execution environments, common and all hospital/insitute specific in the community execution environment (architecture details in chapters 1 and 3)

6. MIP Hospital Deployment Results

During the Ramp-Up Phase of the project, nineteen European university hospitals expressed interest in providing patient datasets, deploying and evaluating Medical Informatics Platform. Deployment and Evaluation Agreements were signed with seven of them (Figure 46):

- University Hospital of Lausanne, Switzerland (CHUV)
- Regional University Hospital Center of Lille, France (CHRU Lille)
- The IRCCS (Istituto di Ricovero e Cura a Carattere Scientifico) St. John of God Clinical Research Center, Brescia, Italy
- Metropolitan Hospital Niguarda in Milano, Italy (ASST Grande Ospedale Metropolitano Niguarda)
- Medical Center – University of Freiburg, Germany (Universitätsklinikum Freiburg)
- Medical University of Plovdiv, Bulgaria
- Tel Aviv Sourasky Medical Center in Tel Aviv, Israel

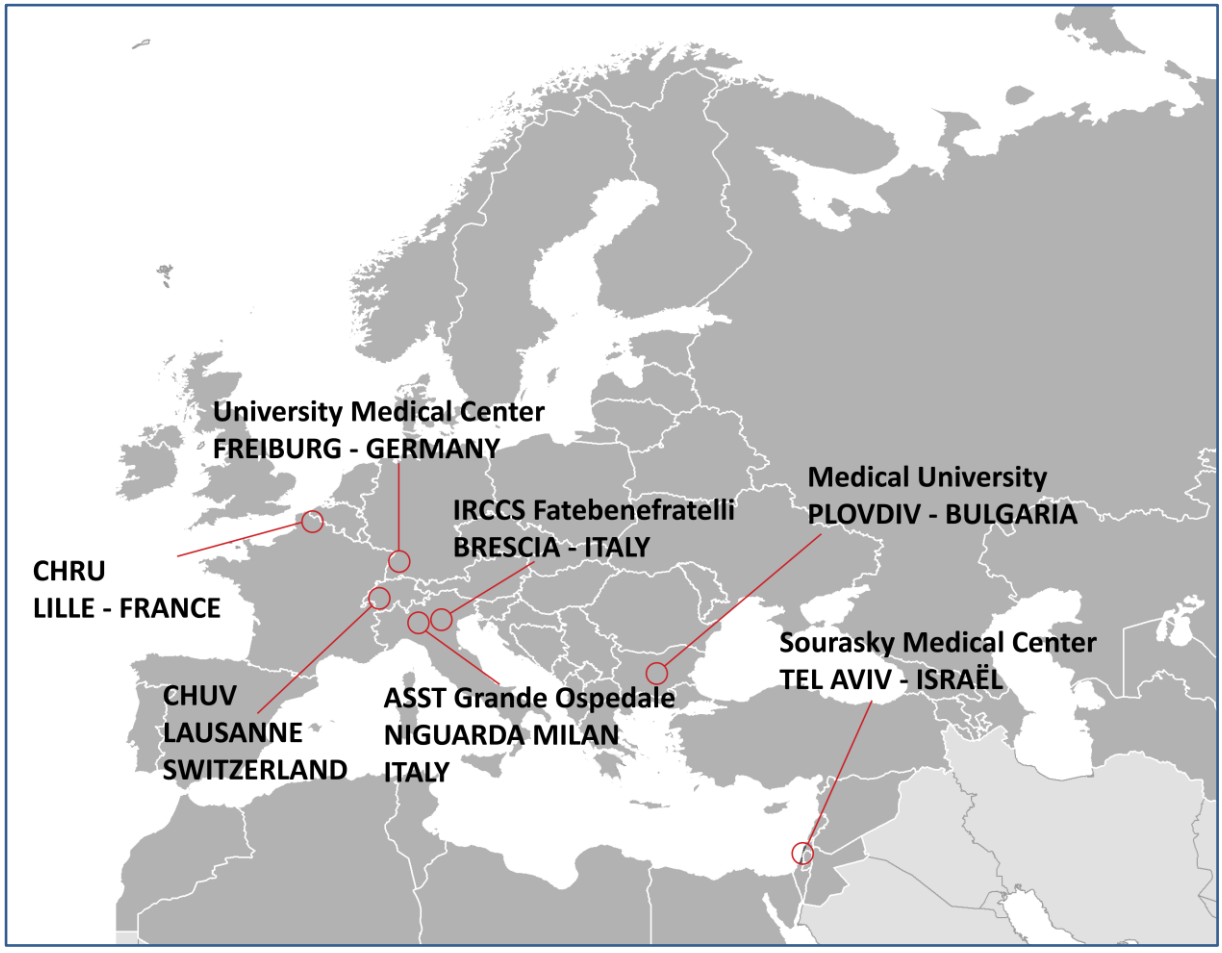


Figure 46: Deployment and Evaluation Agreements with European University Hospitals

The criteria for selecting the seven European university hospitals for providing patient datasets, deploying and evaluating the MIP are provided in Table 11:

Table 11: Criteria for selecting participating hospitals

Criteria	Description

Diversity	Hospitals in different countries. Objective: to test the MIP in different healthcare systems, using data of patients with different exposure to risk factors, disease prevalence, etc.
Size	Hospitals that have a significant number of patients and large patient datasets
Clinical Excellence	The best national hospitals with expertise in clinical neuroscience and clinical care, willingness to share data, with well-established ethics consent procedures
Available resources	Hospitals that have the personnel and IT equipment resources, and a long-term commitment to maintain the Medical Informatics Platform infrastructure
Influence	Hospitals that will promote the Medical Informatics Platform through collaboration with other hospitals in the same region or country

The Medical Informatics Platform provides support for analysing diverse biomedical and other health-relevant patient data. That includes support for multi-centre, multi-dataset studies to bridge the gap between fundamental research and clinical practice.

The scientific research significance of this project is a realistic possibility to discover hidden data patterns by combining multi-centre patient clinical datasets with available open research cohort data, such as ADNI, EDSD and PPMI, and compliance of the MIP platform concept with WHO's action areas. The Information Systems for Dementia and Dementia Research and Innovation were the reasons to select dementia and in particular Alzheimer's disease clinical study scenarios for the demonstration of MIP functionality and its scientific utility. The scientific utility of the Platform, defined as a key SP8 result, is discussed in the M24 deliverable D8.6.3.

Clinicians and clinical researchers from the three selected university hospitals have been chosen because of their expertise in the domain of dementia syndromes and of the profiles of the available patient datasets. They presented data of a significant number of patients with neurodegenerative and neurocognitive disorders, different types of dementia, high Alzheimer's disease incidence, and a variety of biological, cognitive, neuroimaging and other relevant patient data available for analytics.

Data profiles for three university hospitals from France, Italy and Switzerland, including the number of patients in each cohort dataset and the counts of patients with diagnosed Alzheimer's disease (AD), mild cognitive disorder (MCI), other neurodegenerative disorders, cognitive normal (CN) control group and not defined (N/A – not available) diagnostic are provided in Table 12.

The patient cohort dataset of the Regional University Hospital Center of Lille , France (CHRU Lille), consisted of multiple visits per patient. Data profiles in Table 12 give information about the first and the last visit that are recorded in the dataset. The patient cohort datasets of the IRCCS FBF in Brescia, Italy and the University Hospital of Lausanne (CHUV) in Switzerland consisted of a single visit per patient.

Table 12: Hospitals selected to participate in MIP system validation - data profiles[2]

Hospital	Patient Count	Recorded Visit	Diagnosis - Alzheimer's Broad Category CDE				
			AD	MCI	Other	CN	N/A
CHRU Lille France	1436	First	591	227	551	67	0
		Last	813	7	604	12	0
IRCCS FBF Brescia Italy	1960	First	151	201	192	1240	176
		Last	N/A	N/A	N/A	N/A	N/A
CHUV/CLM Lausanne Switzerland	699	First	164	78	414	41	2
		Last	N/A	N/A	N/A	N/A	N/A
ADNI	1066		222	576	0	268	0
EDSD	474		141	76	0	151	106
TOTAL	**5635**		**1269**	**1158**	**1157**	**1767**	**284**

Common data models (Figure 45) have been integrated and synchronised across the participating hospitals' private MIP execution environments. Both common and hospital-specific data models have been integrated in the central MIP community execution environment (Figure 1).

MIP SGA1 common data taxonomy is illustrated in Figure 45. Specific data taxonomies for each of the three hospitals participating in the SP8 system validation project phase are illustrated in Figure 47, Figure 48, and Figure 49.

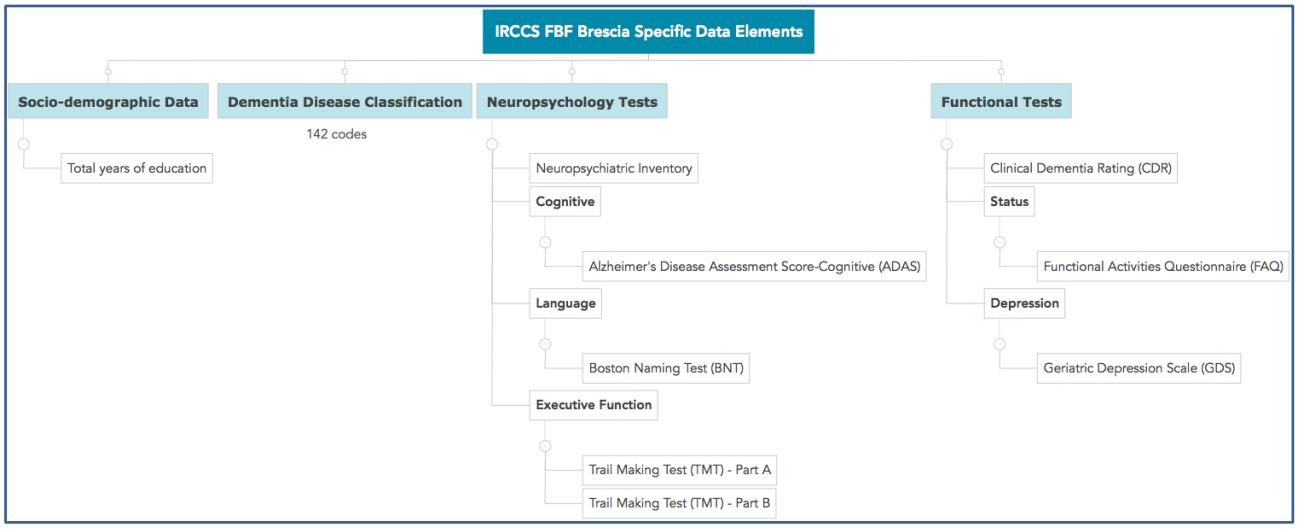


Figure 47: IRCCS Brescia Specific Data Taxonomy

[2] **Diagnosis:** AD - Alzheimer's disease, MCI - mild cognitive impairment, CN - cognitive normal, Other - other neurodegenerative disorder, N/A - disease information not available

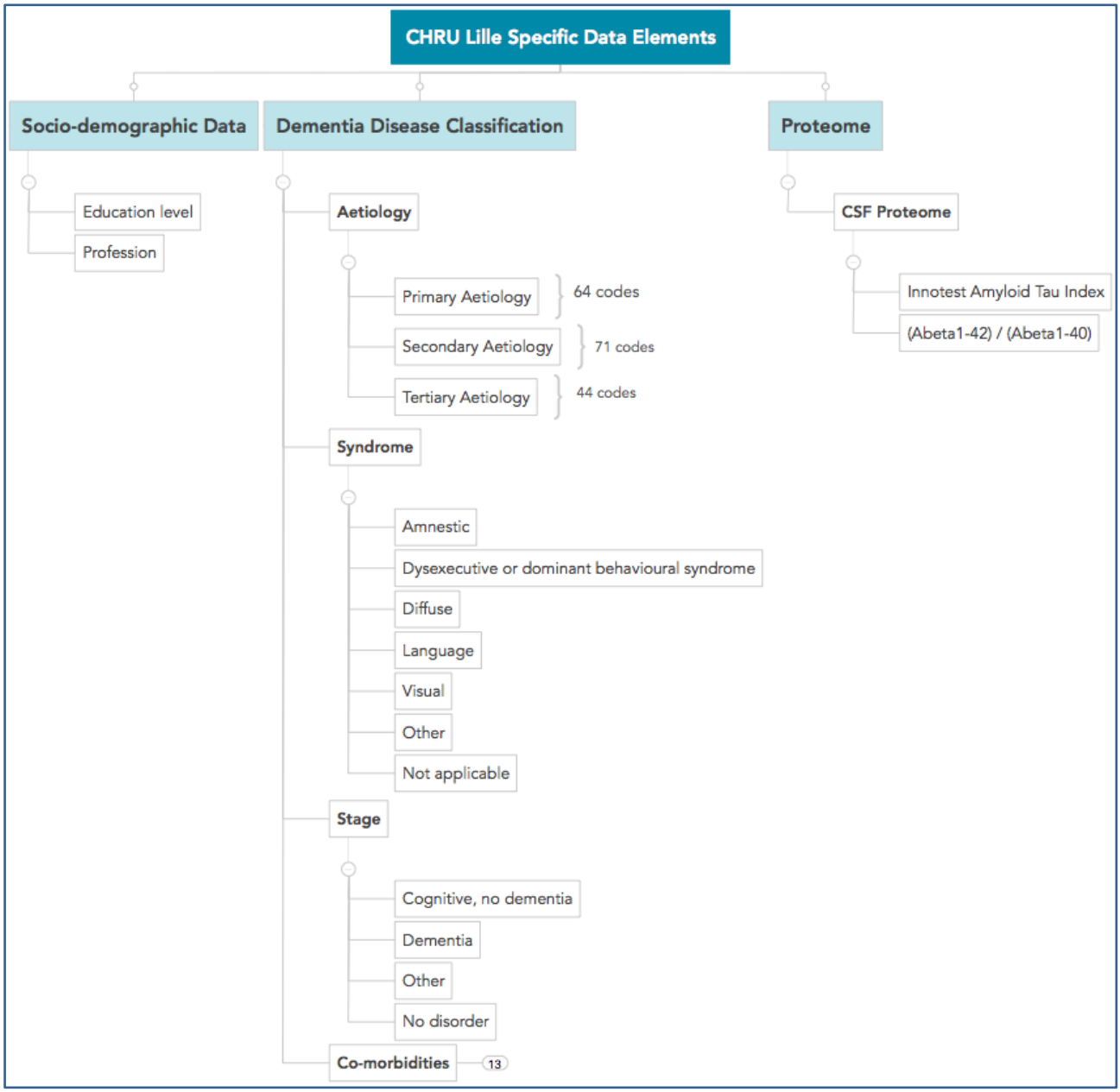


Figure 48: CHRU Lille Specific Data Taxonomy

The MIP software provides the IT prerequisites for the execution of cross-centre, multi-dataset clinical studies across the three university hospitals participating in the SP8 system validation project phase. The software harmonises and synchronises the data model. This software is privacy-preserving, it uses a hybrid community-private deployment model with centralised orchestration of statistical and machine learning algorithms.

The unsupervised machine-learning can be used, for example, to train a classifier, using CHRU-Lille data, to be able to differentiate between frontotemporal dementia and Alzheimer's disease. The learned classifier can then be applied to obtain a differential diagnosis (between frontotemporal dementia and Alzheimer's disease) in the IRCCS in Brescia and the CHUV in Lausanne. Or, we can use clinical and pathological data of deceased patients from the CHRU Lille dataset to train a machine-learning model that can be used to predict disease progression from patients in the other two hospitals. Detailed results of these studies are presented in the final M24 deliverable D8.6.3.

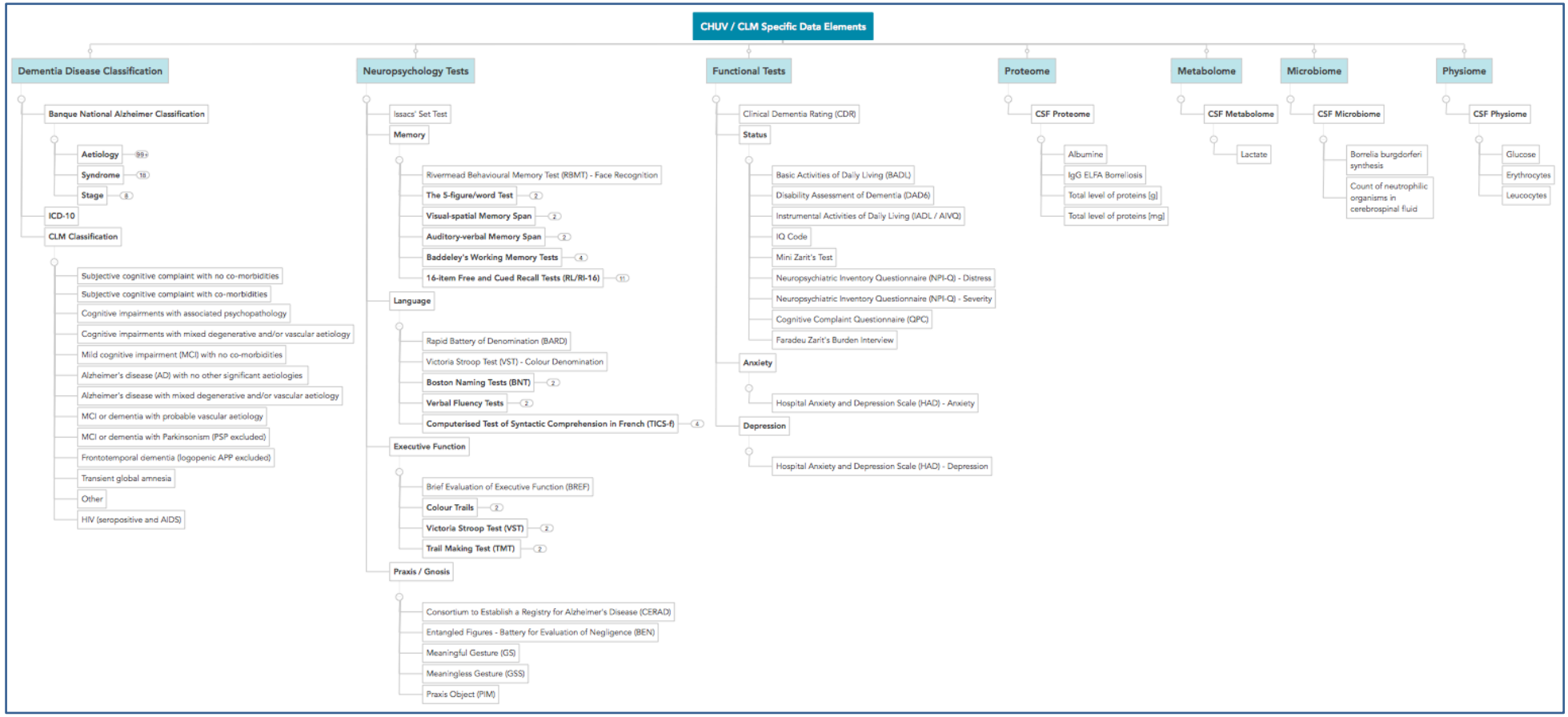


Figure 49: CLM/CHUV Lausanne Specific Data Taxonomy

7. Technology Readiness Level Assessment

The Medical Informatics Platform is a sophisticated software system developed out of many individual technologies (i.e. components) integrated into a complex functional solution for descriptive and predictive analysis of patient datasets, including the combination of data originating from their hospital health records and processed brain scans.

7.1 Adaptation of the standard EC TRL scale

HBP adaptations of the source EC TRL scale addresses the aspects of research solutions that will need integration of various technologies, interaction with users and validation of the systems in user environments. These adaptations are essential for the comprehensive evaluation of the technological maturity of the Medical Informatics Platform, because its technological value and value for users depend on the maturity of the fully integrated and operational system.The platform is a complex solution developed out of a number of individual technologies/components.

The Medical Informatics Platform is a data-intensive analytics solution. It uses available data (patient biomedical and other health-relevant information from their hospital medical records and neuroimaging data from brain scans) to produce more data (the results of the descriptive and predictive data analysis). The technological maturity of such a solution, and its value for the users is a function of a quality of new data (or knowledge) production, i.e. it is a function of the quality of the data analysis results. The quality of the data analytics ultimately depends on the type, quality, variability and volume of the analysed data.

Therefore, the technological maturity of the Medical Informatics Platform and its value for the users directly depends on the number and variety of participating hospitals and the number and type of datasets that are available to the Platform for analytics. The number and variety of participating hospitals and research institutes will not only depend on the technological maturity of the Platform. Financial and organisational aspects determine equally much the success of the widespread deployment of the solution.

For an accurate evaluation of the MIP technological maturity and a precise communication of the technology readiness level in any of the project stages, it is crucial to take the following aspects into account:

- The TRL setback mechanisms need to be incorporated, as their exclusion would mean that when (not if!) they occur, funding of specific activities would be (temporarily) stopped, leading to an unnecessary destruction of capital. In contrast to the implicit linear character of both EC TRL scale and its HBP adaptation, feedback models show that research is needed even at the higher TRL levels, i.e. an increase in maturity also requires additional R&D. The implication is that in every stage certain kinds of R&D should be incorporated
- Innovation is usually built up from different technologies. Therefore the TRL scaling should make a distinction between R&D on individual technologies, integration of those technologies and pilot production. Most of the relevant aspects are provided for the HBP adaptation of the EC TRL scale, but the focus of the higher TRLs seem to be on the “small number of users”. In the case of the Medical Informatics Platform, the TRL scaling should account for the wide deployment and the maturity of the corresponding technologies. The software “manufacturing” technologies needed (CI/CD, system monitoring, O&M tools, version control, operation processes, etc.), can be seen as just another set of technologies
- Innovation is not about technology (product and process) alone. Financial and organisational activities can be crucial to commercial success. Both the EC TRL scale and its HBP adaptation are clearly about product oriented technologies. Their focus is apparently on product development, but very little on the ability of the production on a broader scale and there is no explicit mentioning of organisational requirements. Non-technological aspects, the readiness of an organisation to implement the innovation, for example, should be incorporated

into the TRL definitions. For example, the development of accompanying services, including tools, processes and organisation, is just one example that is crucial in case of the Medical Informatics Platform, as it determines the success and sustainability of the wide-spread platform deployment

An integrative TRL assessment approach, combining different technologies and addressing market and organisational issues, is recommended to assess and communicate the MIP technology readiness level. We have decided to adopt the recommendation of European Association of Research and Technology Organisations (EARTO) for its close compliance to the needs of the SP8 strategic objectives and for the nature of the Platform. The different maturity stages are summarised in Figure 50 and details are provided in Table 13: MIP TRL definition overview table.

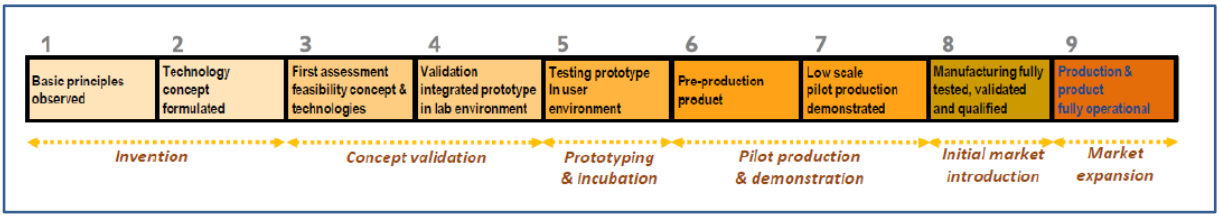


Figure 50: The adaptation of EC TRL scale to the SP8-MIP needs

Finally, an interaction between disciplines, a trans-disciplinary and user-centric approach is needed to solve societal challenges by connecting various technologies, connecting one technology to multiple applications, connecting technologies to non-technological disciplines allowing to take users perspective into account as well as to look at solutions bridging commercial interests and society needs. These aspects are all relevant and essential criteria for the assessment of a complex data intensive solution, such as the Medical Informatics Platform.

Furthermore, the successful wide-scale use of the new technologies inevitably changes the perceptions and needs of the users and society as a whole. As a consequence of the ever-lasting evolution of the user needs, an organisation needs to provide for the sustainable phased development of the solutions. In a typical mature R&D organisation, while phase N of the complex solution is deployed and in operation (TRL9 level), product phase N+1 is in the TR5-7 developmental stage, phase N+2 is in the conceptualisation and prototyping stage at TR level 2-4, and the product phase N+3 can be in the early research stage at TRL1 level.

Table 13: MIP TRL definition overview table

Cluster	TRL	HBP Terminology	MIP/EARTO Reading	MIP Definition and Description
Invention	TRL1	Project Initiation	Basic principles observed	Basic scientific research is translated into potential new basic principles that can be used in new technologies
	TRL2	Conceptualisation	Technology concept formulated	Potential application of the basic (technological) principles is identified, including their technological concept. Also, the first wide-scale software deployment principles are exploited, as well as possible markets identified. A small research team is established to facilitate assessment of technological feasibility
Concept validation	TRL3	Proof of Concept Implementation	First assessment of feasibility of the concept and technologies	Based on the preliminary study, an analysis is conducted to assess technical and market feasibility of the concept. This includes active R&D on a laboratory scale and first discussions with potential clients from major European university hospitals. The research team is further expanded and an early market feasibility assessed
	TRL4	Prototype Component	Validation of integrated prototype in a laboratory	Basic technological components are integrated to assess early feasibility by testing in a laboratory environment. Wide-scale software deployment is actively researched and analysed, identifying main production principles. Lead hospitals and institutes are engaged to ensure connection with demand. Organisation is prepared to enter into scale up, possible services prepared and full market analysis conducted
Prototyping and incubation	TRL5	Prototype Integration	Testing of the prototype in a user environment	The system is tested in a user environment, connected to the broader technological infrastructure. Actual use is tested and validated. Wide-scale deployment is prepared and tested in a laboratory environment and lead hospitals and institutes can test pre-production products. First activities within the organisation are established to further scale up to pilot production and marketing
Pilot production and demonstration	TRL6	Prototype-to-Real-world Integration	Pre-production of the product, including testing in a user environment	Product and manufacturing technologies are now fully integrated in a pilot line or pilot plant (low-rate software deployment). The interaction between the product and wide-scale software deployment technologies are assessed and fine-tuned, including additional R&D. Lead hospitals and institutes test the early products and wide-scale software deployment process and the organisation of production is made operational (including marketing, logistics, production and others)

Cluster	TRL	HBP Terminology	MIP/EARTO Reading	MIP Definition and Description
	TRL7	Operational Integration	Low-scale pilot production demonstrated	Wide-scale software deployment process is now fully operational at a low rate, producing actual final developed products. Lead hospitals and institutes test these final products and organisational implementation is finalised (full marketing established, as well as all other production activities fully organised). The product is formally launched into first early adopter hospitals and institutes
Initial market introduction	TRL8	Deployment	Wide-scale software deployment process fully tested, validated and qualified	Wide-scale software deployment of the product and the final version of the product are now full established, as well as the organisation of production and marketing. Full-launch of the product is now established at the European markets
Market expansion	TRL9	Production	Production and product fully operational and competitive	Full production is sustained, the product is expanded to worldwide markets and incremental changes of the product create new versions. Wide-scale software deployment and overall production is optimised by continuous incremental innovations of the process. Worldwide markets are fully addressed

7.2 Integrated system technology readiness level assessment

As discussed in the previous sub-chapter, for the precisse assessment of the MIP's TRL at the end of SGA1 project phase, and for full compliance with the plans for the technology maturation as defined in the SP8 SGA2 proposal, we decided to adopt the EARTO adaptation of the EC TRL definitions (see Table 13: MIP TRL definition overview table).

The MIP is a data-intensive solution. MIP of a higher level of technological maturity requires access to big data for technologically more advanced ways to discover biological signatures of diseases by applying predictive machine learning and deep learning algorithms. The emphasis is therefore also on the development of a mature wide-scale production technology of the Platform, with the corresponding processes and organisational aspects as prerequisites for its wide-scale deployment to get access to more patient datasets.

Table 14: Technology readiness level assessment of the key technologies / components

ID	Component Name	TRL	Component Type	Description
2938	Algorithm Orchestrator	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment.
647	Algorithm Repository	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
645	Model Benchmark and Validation	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
646	Predictive Disease Models	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
633	Portal DB (Articles, Experiments, Models)	TRL5	SOFTWARE	The component is integrated into the MIP ecosystem and tested in a user environment
1595	Distributed Query Processing Engine - Master	TRL4	SOFTWARE	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment
1596	Distributed Query Processing Engine - Worker	TRL4	SOFTWARE	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment
638	Query Engine	TRL4	SOFTWARE	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment

ID	Component Name	TRL	Component Type	Description
687	Data Governance Methodology	TRL3	service	Based on the preliminary study, technical and market feasibility of the concept is analysed and discussed with potential clients from major European hospitals
587	Data Mapping and Transformation Specification	TRL3	data	Based on the preliminary study, technical and market feasibility of the concept is analysed and discussed with potential clients from major European hospitals and tested in one hospital (CHRU Lille)
1580	Online Data Integration Module	TRL4	software	The component is integrated into the MIP ecosystem to assess early feasibility by testing in a laboratory environment
671	Neuromorphometric Processing	TRL7	software	Lead hospitals and institutes are using the solution. The component is formally launched and training is established. The solution is developed and managed in an academic organisation. Product manufaturing and marketing organisation needed for TRL8 categorisation is not established
664	Airflow DAGs	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL8/9 categorised Apache Airflow solution
2927	Data Catalogue	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL9 PostgreSQL DBMS
2926	Data Capture Database	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL9 PostgreSQL DBMS and the star database schema is compatible with TRL7 I2B2 solution
669	Common Data Elements Database	TRL4	software	The component is integrated into the MIP ecosystem and tested in a laboratory environment. It is based on the TRL9 PostgreSQL DBMS and the star database schema is compatible with TRL7 I2B2 solution

ID	Component Name	TRL	Component Type	Description
102	MIP Microservice Infrastructure	TRL5	software	The component is integrated into the MIP ecosystem and tested in a user environment
2940	Data De-identifier	TRL5	software	The component is integrated into the MIP ecosystem and tested in a user environment.
2936	MIP De-identification Profiles	TRL5	model	The component is integrated into the MIP ecosystem and tested in a user environment.
2935	MIP De-identification Strategy	TRL5	report	The component is integrated into the MIP ecosystem and tested in a user environment.

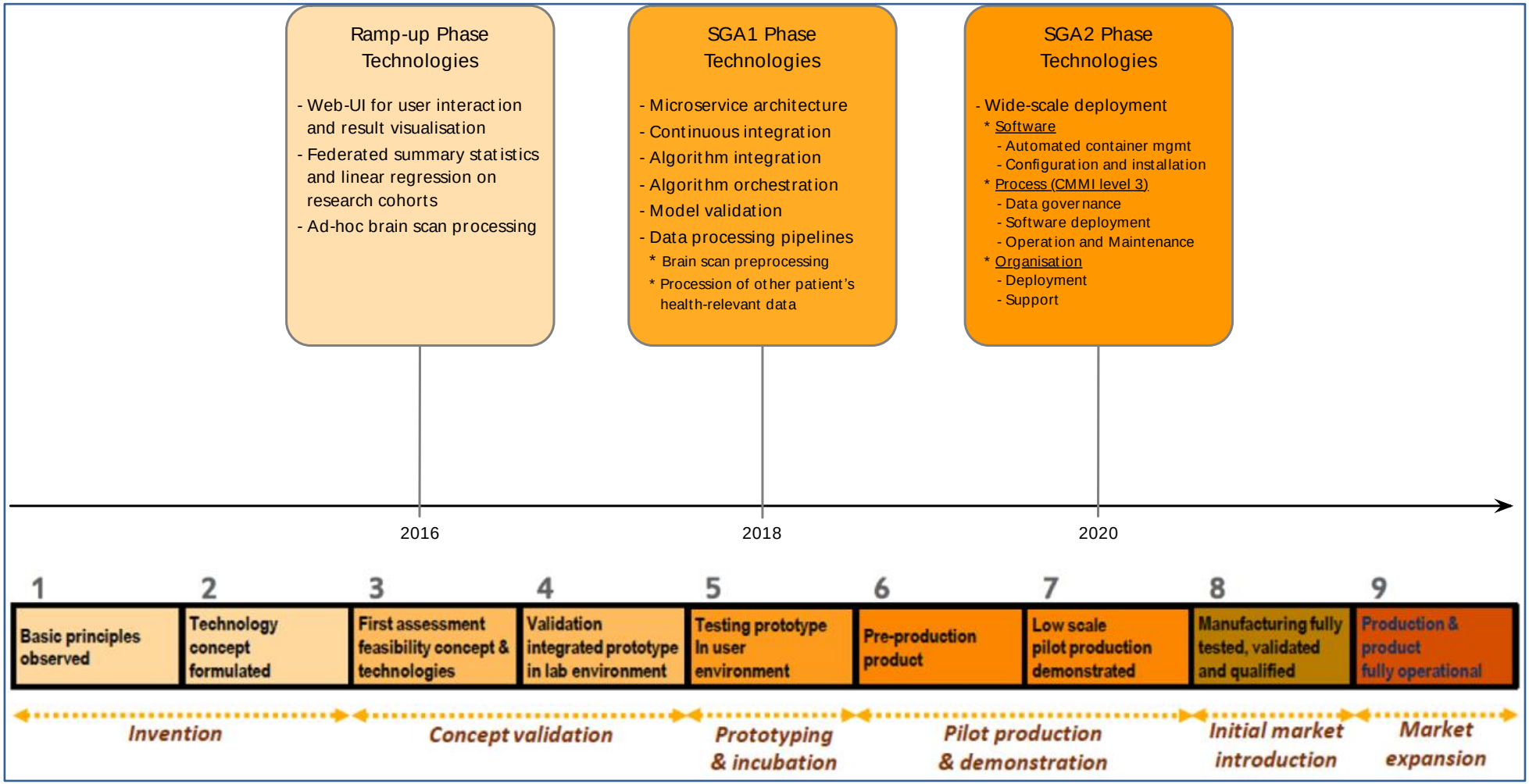


Figure 51: Transition of MIP technology readiness level and future roadmap

Human Brain Project

THE MEDICAL INFORMATICS PLATFORM

MIP USER MANUAL

VERSION: JANUARY 2019

Table of Contents

1. Introduction

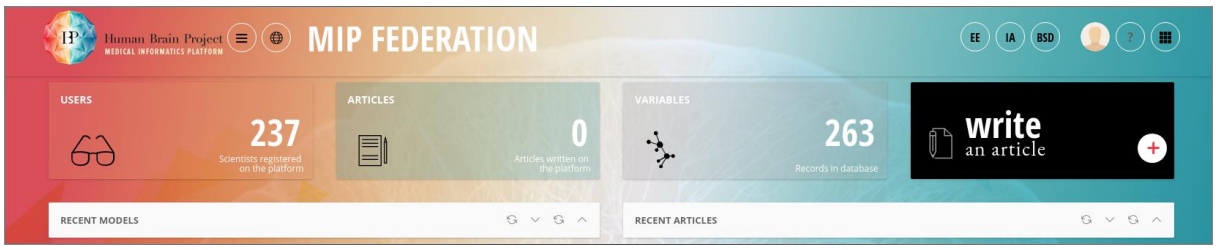
The Medical Informatics Platform (MIP) aims to provide tools to analyse medical data and advance more rapidly in understanding the neurological and psychiatric diseases. The users can access the platform via a Web user interface (UI) (password restricted) where they can run exploratory data analyses, create and share analysis models, execute descriptive statistics, inferential statistics and machine-learning algorithms on user-defined analysis models, as well as collaboratively write articles.

2. MIP User Guidelines

2.1 General Navigation with the MIP

After log-in, the platform opens with the main dashboard.

The dashboard shows a summary of statistics, users, available variables, as well as the latest three saved analysis models and articles (those of the current user or shared among all users). From here, the user may also start writing articles (description of data analyses performed).



At any time the user can return to this page by clicking on the HBP logo on the top left corner.

From the top banner, user can at any time access:

- **My Data**: personal dashboard displaying own work (saved analysis models and articles)
- **My Community**: all work labelled for sharing by any user within its MIP community
- **Functionalities**:
  - Epidemiological Exploration (**EE**)
  - Interactive Analyses (**IA**)
  - Biological Signatures of Diseases (**BSD**)
  - Personal Profile
  - Third-party web applications

2.2 Variables Exploration, Analysis Model and Experiment Design

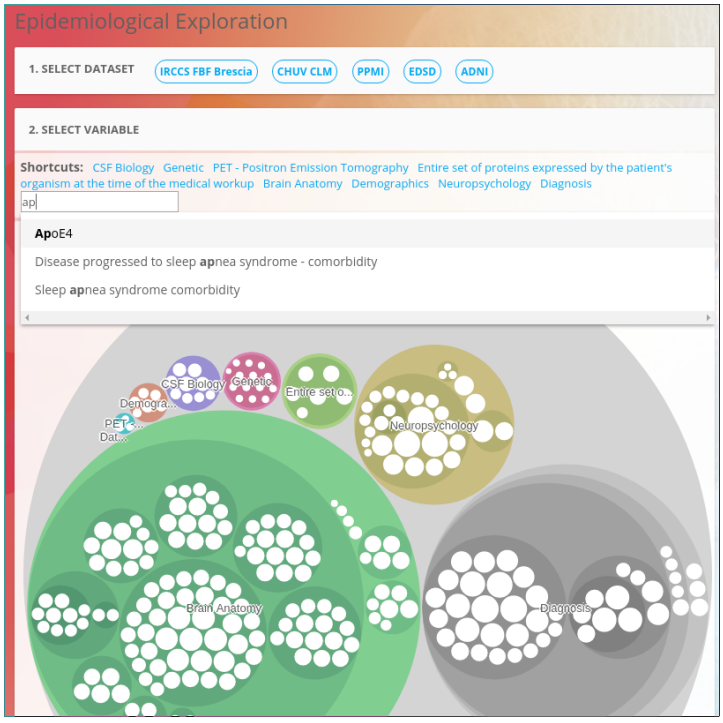
The MIP provides the following functionalities:

- **EE**: Epidemiological Exploration – allows exploration of the available variables, including visualisation of variables' types, selection of variables, visualisation of variables' descriptive information, and definition of analysis models (selection of response and explanatory variables for data analysis);
- **IA**: Interactive Analyses – provides descriptive summary statistics for the defined analysis models, in tabular and graphical formats;
- **BSD**: Biological Signatures of Disease – provides selection and configuration of data analysis algorithms – descriptive and inferential statistics, machine-learning and validation, using analysis models defined with IA functionality.

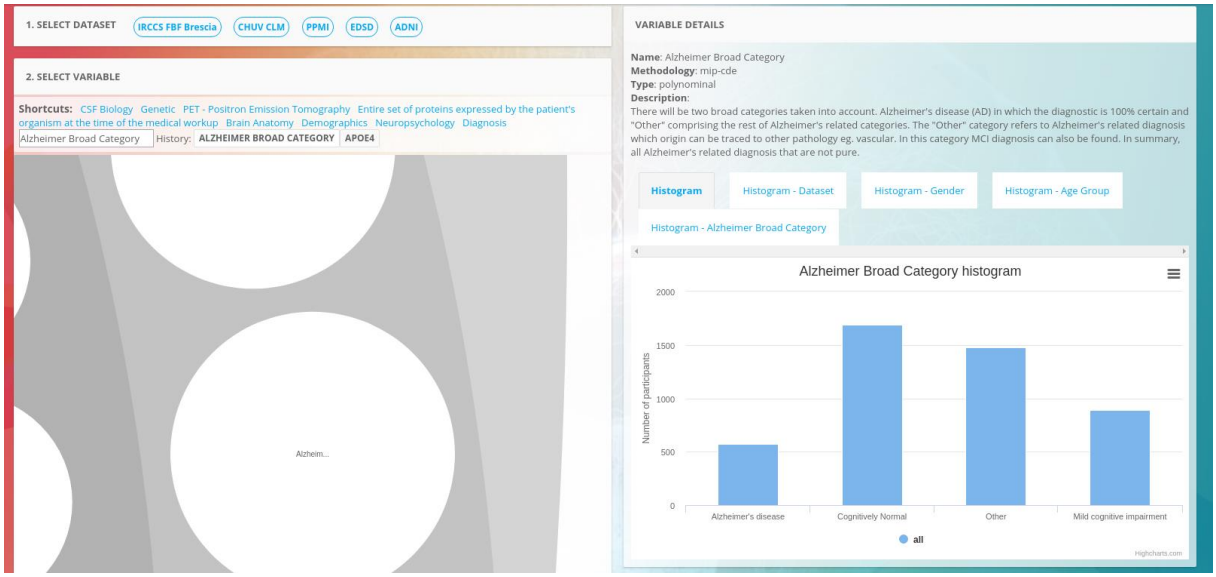
2.2.1 EE: Epidemiological Exploration

The exploration of variables is done through their representation in a circle-pack design. We will call it the MIP Variable Space.

User may also search for a variable by typing its name directly in the search box above the MIP Variable Space (see screen shot below, searching for APOE4 as an example)

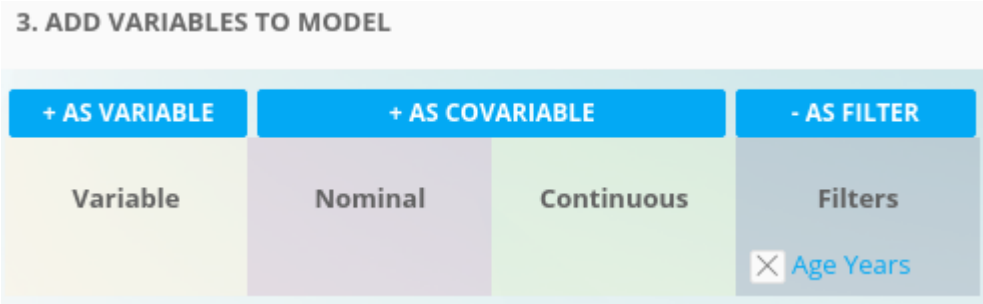


Some variables have descriptive information associated with it. They are displayed in the right-sided panel of the MIP Variable Space (see next screenshot).



User may stop variables exploration here, or continue, to define an analysis model by selecting variables from the MIP Variable Space, as variable (dependent variable), co-variables (independent variables) and filters. It is possible to select a group of variables in one step. See screenshot below.

Variables used for setting up grouping and conditions should be selected as filter variables.



2.2.1.1 Define an Analysis Model

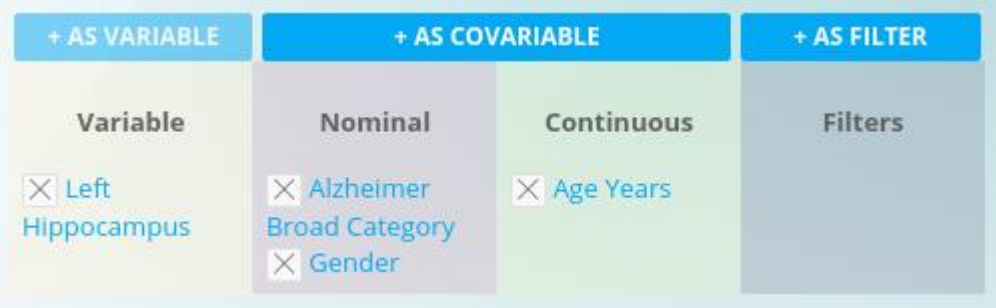
To define an analysis model, user needs to:

1. Search for the variables of interest in the MIP Variable Space, and click on the desired variable, or a group of variables.

Let's take an example: to predict changes in the volume of the Hippocampus in Alzheimer's disease, with respect to age and gender:

- Select the variable Left Hippocampus in the MIP Variable Space and then click on "+ AS VARIABLE" in the model table to define it as a response variable.
- Then select Age Years and Gender variables in EE and click "+ AS COVARIABLES", to define them as explanatory variables. Then do the same for Alzheimer Broad Category. Variables will be automatically classified as "NOMINAL" or "CONTINUOUS" depending on their respective types.

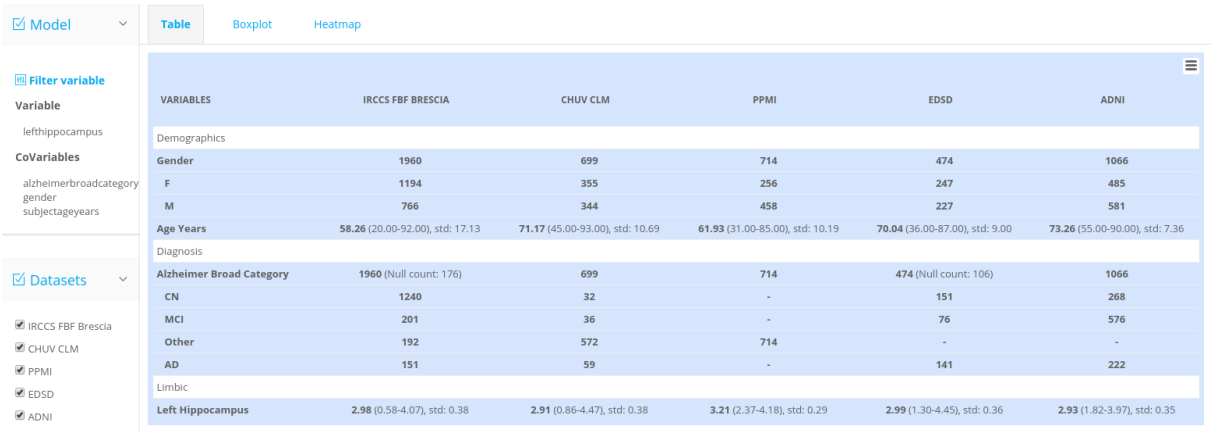
User may at any time remove variables from the Configuration table by clicking the "X" sign. Similarly, user may restore any previously searched variable from the "History" line.



2. When finished with analysis model definition, just click on the Review Model button at the bottom of the screen.

2.2.2 IA: Interactive Analysis

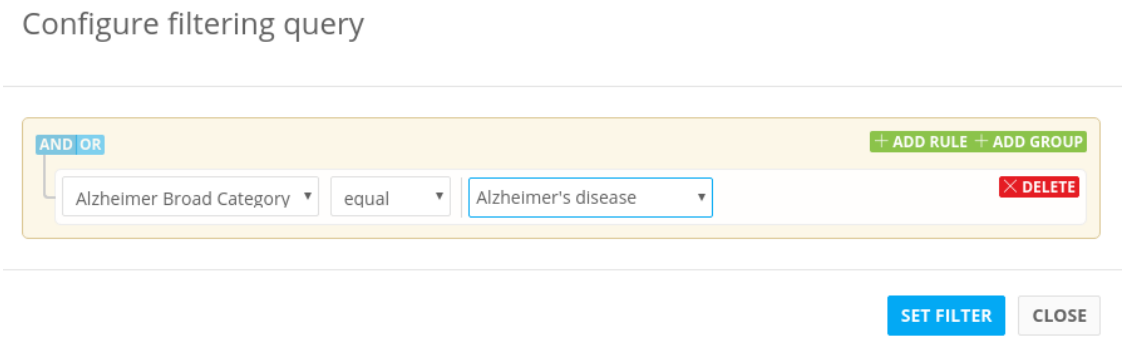
The statistical analysis of the model is run in the MIP. The results are summarized in appropriate tables and visualizations. Statistical description of the defined analysis model is provided in a table and in appropriate visualisations.



VARIABLES	IRCCS FBF BRESCIA	CHUV CLM	PPMI	EDSD	ADNI
Demographics					
Gender	1960	699	714	474	1066
F	1194	355	256	247	485
M	766	344	458	227	581
Age Years	58.26 (20.00-92.00), std: 17.13	71.17 (45.00-93.00), std: 10.69	61.93 (31.00-85.00), std: 10.19	70.04 (36.00-87.00), std: 9.00	73.26 (55.00-90.00), std: 7.36
Diagnosis					
Alzheimer Broad Category	1960 (Null count: 176)	699	714	474 (Null count: 106)	1066
CN	1240	32	-	151	268
MCI	201	36	-	76	576
Other	192	572	714	-	-
AD	151	59	-	141	222
Limbic					
Left Hippocampus	2.98 (0.58-4.07), std: 0.38	2.91 (0.86-4.47), std: 0.38	3.21 (2.37-4.18), std: 0.29	2.99 (1.30-4.45), std: 0.36	2.93 (1.82-3.97), std: 0.35

User can explore the defined analysis model by selecting/deselecting datasets, adding conditions (filters), and selecting visualisations.

To add conditions and grouping, click on the "filter variable" link on the left. That opens a pop-up window for configuring individual rules, their grouping and for selecting their combinations using "and/or" buttons. Conditional and grouping rules are applicable only to variables selected as "filter variables" in the EE functionality. See screenshot below.



After setting up grouping and conditional rules, the analysis model's data is updated. Corresponding tabular and visual summary statistics reflect the analysis model changes.

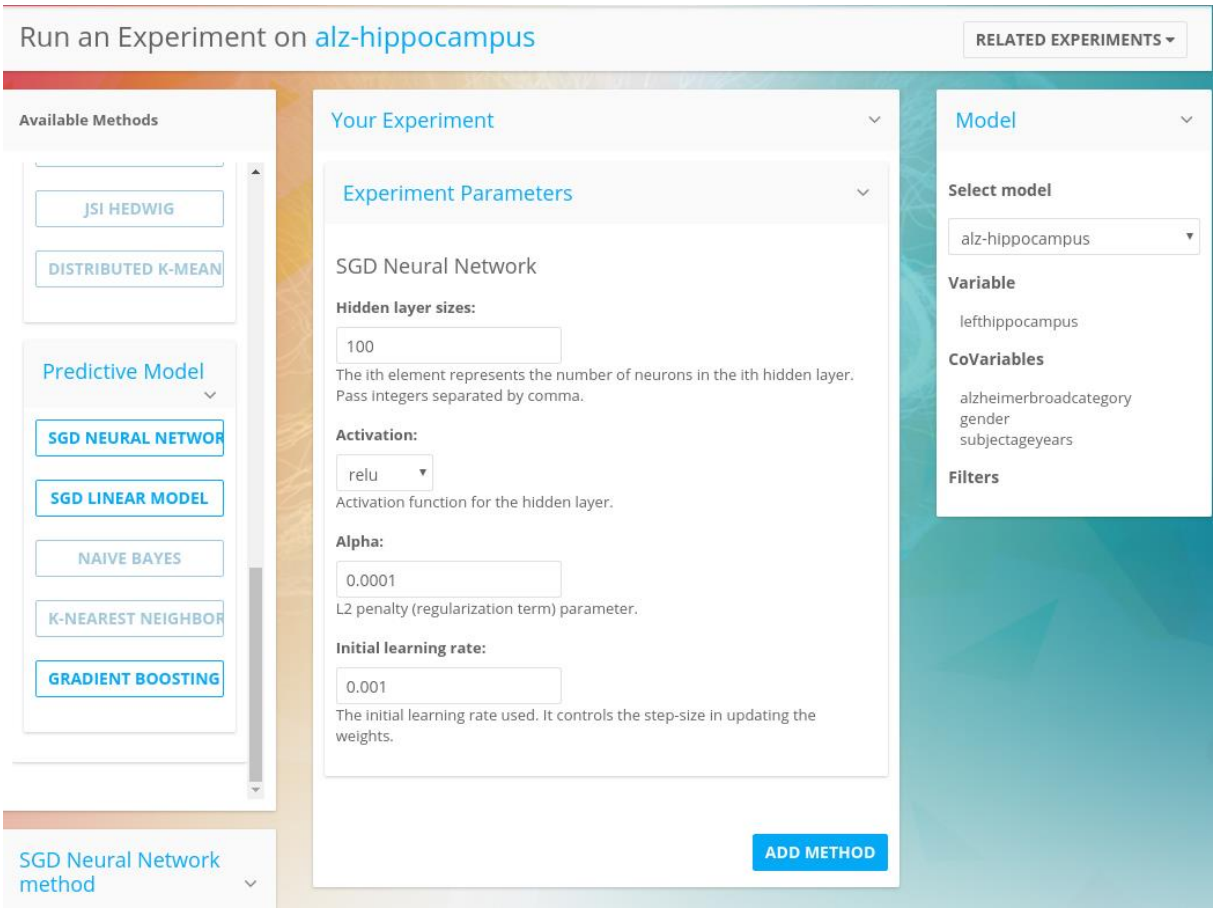
To perform an experiment, user needs to save the analysis model by giving it a name. He can also share it with the MIP community.

2.2.3 BSD: Biological Signatures of Diseases

To configure the experiment:

1- Click "Run Machine Learning Experiment", on the IA screen.

2- In the next BSD screen, choose a method among the "Available Methods", see screenshot below.



3- For some methods user can configure parameters of the algorithm.

4- Click on “Add method” to add it to the methods selected for the experiment. Several methods can be added to an experiment.

5- User can choose to train and validate the experiment on various datasets. See screenshot below.

6- When ready to run the experiment, user needs to give it a name and then to click on RUN EXPERIMENT.

The results of the experiment are presented in textual, tabular or visual format depending on the type of methods chosen and their implementation.

2.2.4 Online Resources

A video demoing the MIP is available on YouTube:
https://www.youtube.com/watch?v=MNWExzouMJw&t=61s

3. Other MIP Functionalities

3.1.1 Writing Articles

1- Go to main MIP Dashboard (click the HBP icon on the top left-hand side);
2- Click on "Write an Article";
3- Write the article: use the editor to add a title, abstract and content to your article;
4- You can also drag and drop results of your models (or others' if are shared) from the left-hand side into the content of your article;
5- Give a name to your article, save it and, optionally, share it.

3.1.2 Accessing my Saved Articles and Models

User can at any time access the already written articles and saved analysis models, via:

- My Data – all own work, shared or unshared;
- My Community – all work shared within the MIP community;

User can preview articles, save them to a file system accessible from his computer or open them for re-editing.

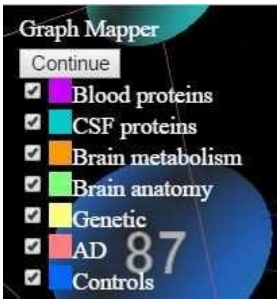
3.1.3 Third Party Applications

3rd Party Application are made by users to provide some insights on specific models or visualisations.

3.1.3.1 3D Biological Rules:

Navigate in a 3D world of variables.

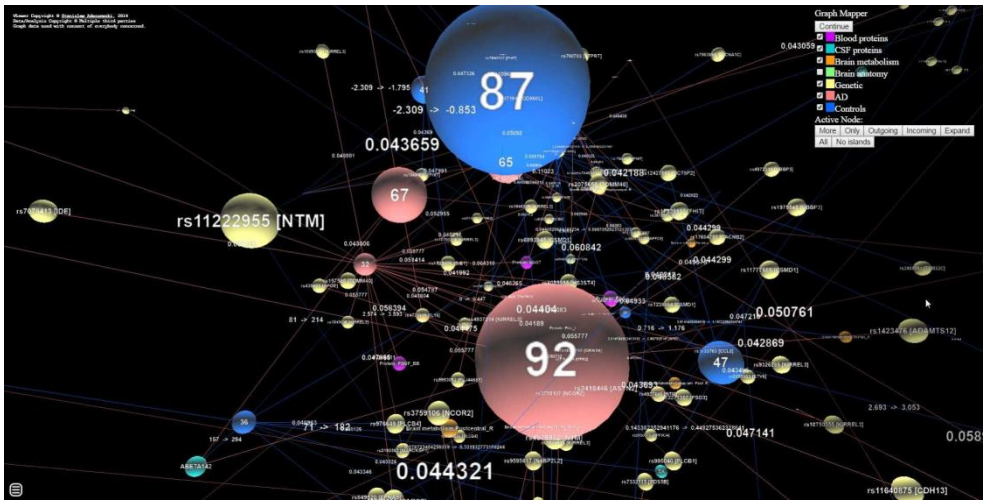
The 3D application shows the results of a rule-based clustering algorithm applied to Alzheimer's disease patients to identify homogeneous subgroups of patients. The hypothesis is that patients in each subgroup have the same underlying cause of the disease. The rule-based algorithm aims to explain the variability between individuals and describes a population by a group of "local over-densities".



These are defined as subspaces over combinations of variables: blood proteins (magenta), CSF proteins (aqua-blue), Brain metabolism (orange), brain anatomy (green)and genetic (yellow).

The red spheres represent AD subgroups and the blue ones healthy controls. The number in a sphere indicates the number of subjects belonging to the subgroup. The edges show rules between the spheres and variables that define each subgroup.

User can also use the left mouse button for rotation, the middle button for zoom and the right mouse button for translation.



User can select and deselect the different variables clicking on the variables tick to make them appear or disappear in the 3D world.

For the genetic variable click "More" to get additional information from the http://www.ensembl.org/ database.

3.1.3.2 2D Biological Rules

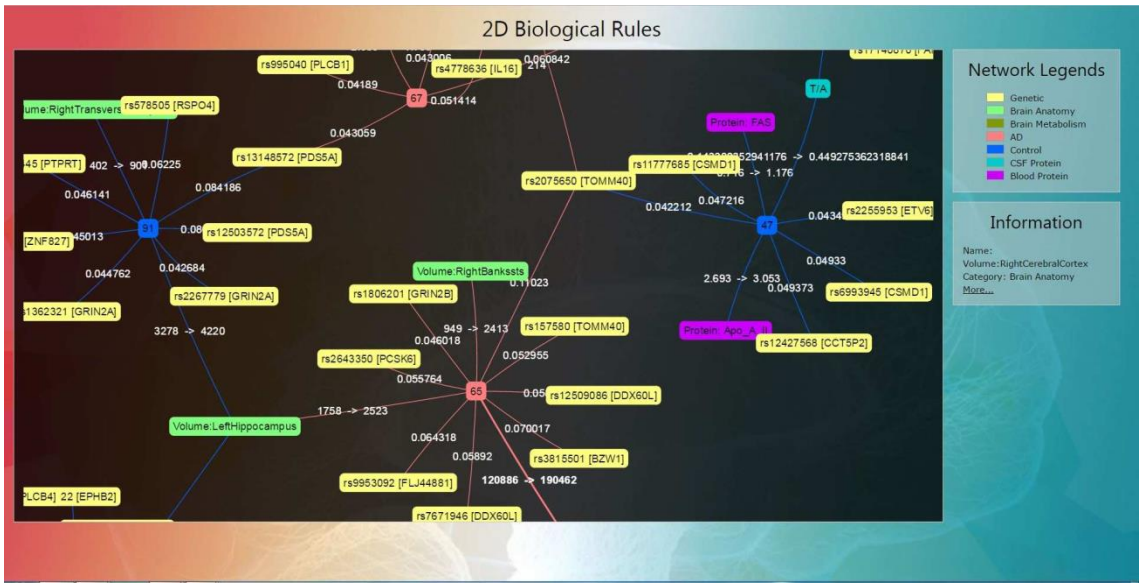
A graphical view of biological rules.

Like the 3D application, 2D application shows the results of a rule-based clustering algorithm applied to Alzheimer's disease patients to identify homogeneous subgroups of patients. The hypothesis is that patients in each subgroup have the same underlying cause of the disease. The rule-based algorithm aims to explain the variability between individuals and describes a population by a group of "local over-densities".

These are defined as subspaces over combinations of variables: genetic (yellow), brain anatomy (green), brain metabolism (orange), CSF proteins (aqua-blue) and blood proteins (magenta).

In this application user can select a variable on the 2D Map and get information on the right side.

The red spheres represent AD subgroups and the blue ones healthy controls. The number in a sphere indicates the number of subjects belonging to the subgroup. The edges show the rules between the spheres and the variables that define each subgroup.



4. Glossary

Model: analysis models: Set of variables, co-variables, filters, training and validation datasets.

Epidemiological Exploration (EE): Exploration of variables, their distribution related to datasets, genders, age groups and diagnosis.

Interactive Analyses (IA): Summary statistics of the selected model by dataset.

Biological Signatures of Diseases (BSD): Statistical and machine learning algorithms.

Variable: Dependent or target variable.

Co-variable : Independent variable, or predictor.

- **Nominal**: Categorical type.
- **Continuous**: Numerical (real, integer).

Filter: Include or exclude specifics subjects.

Dataset: Set of patients or subjects related data provided by specific medical institutes.

Federation: Set of algorithms designed to aggregate intermediate results from different datasets.

Distributed: Set of algorithms performed either on different datasets and outputting parallel results, or predictive models trained iteratively on different datasets.

Experiment: Set of algorithms applied on a model.

Methods: Algorithms.