

Deliverable 6.6

Project ID	654241
Project Title	A comprehensive and standardised e-infrastructure for analysing medical metabolic phenotype data
Project Acronym	PhenoMeNal
Start Date of the Project	1st September 2015
Duration of the Project	36 Months
Work Package Number	6
Work Package Title	PhenoMeNal Virtual Research Community Gateway
Deliverable Title	D6.6 Report on available processes, methods and SOPs deployed in the grid/cloud
Delivery Date	M30
Work Package leader	EMBL-EBI
Contributing Partners	EMBL-EBI
Authors	Ken Haug (EMBL-EBI), Luca Pireddu (CRS4)
Abstract: This deliverable provides a report on the standard processes developed to ensure consistency within the VRE and VRE Portal. All tools and workflows follow standardised templates for inclusion in the VRE and respective workflows.	



Table of Contents:

1	Executive Summary	3
2	Contribution towards the project objectives	3
3	Detailed report on the deliverable	3
3.1	Templates - Tool integration and Documentation	3
3.2	GitHub readme and VRE Service Catalogue / App Library	3
3.3	GitHub wiki and VRE technical documentation	5
3.4	Galaxy and Docker templates	6
3.5	Release process	6
3.6	Standardised automated testing procedure	6
3.7	Standardised tool and container versioning	7
4	Delivery and Schedule	7
5	Conclusion	7



1 Executive Summary

This deliverable provides a report on the standard processes developed to ensure consistency within the VRE and VRE Portal. All tools and workflows follow standardised templates for inclusion in the VRE and available workflows. The VRE Service Catalogue/App Library require a pre-defined structure for tools to be included in the VRE Portal.

Consistent and reproducible workflow execution is of course essential for VRE users. Each workflow requests on-demand inclusion of various Docker containers to enable execution of the respective tools included. To enable this seamless on-demand architecture, each tool and process has to follow a strict methodology. Tool inclusion requires pre-defined Galaxy and container definitions (Docker) to work in the VRE. PhenoMeNal does not enforce restrictions in regards to scientific standard operating procedures (SOPs), but rather technical SOPs to ensure successful VRE workflow inclusion and execution.

2 Contribution towards the project objectives

The deliverable has contributed towards the project objective:

Objective 6.1: Establishment of the PhenoMeNal Virtual Research Community (VRC) portal

3 Detailed report on the deliverable

3.1 Templates - Tool integration and Documentation

This section describes the different tools integration points and standardised documentation templates required for automated inclusion in the PhenoMeNal Portal and any deployed VRE.

3.2 GitHub readme and VRE Service Catalogue / App Library

The VRE App Library is dynamically populated from the respective repositories under the PhenoMeNal organisation on GitHub. Each tool is accompanied by a set template (readme.md file) that enable automatic inclusion in the App Library. The same template file is also used in the respective GitHub repository to technically describe the tool functionality. See **figure 1** for an example from the VRE App Library and **figure 2** for the same tool in the GitHub repository



W4M LCMS matching

Annotation of MS peaks using matching on a spectra database.

created by Pierrick Roger (CEA).
containerised by Pierrick Roger (CEA).



Website	http://workflow4metabolomics.org
Version	Version: 3.1.1
Git Repository	https://github.com/phnmnl/container-lcmsmatching.git

Features

LCMS annotation.

Description

This tool performs LC/MS matching on an input list of MZ/RT values, using either a provided in-house single file database or a connection to Peakforest database.

Installation

For local individual installation:

```
docker pull docker-registry.phenomenal-h2020.eu/phnmnl/lcmsmatching
```

Usage Instruction

For direct docker usage:

```
docker run docker-registry.phenomenal-h2020.eu/phnmnl/lcmsmatching ...
```

Figure 1: Example of workflow tool included in the PhenoMeNa VRE



README.md



W4M LCMS matching

Version: 3.1.1

Short description

Annotation of MS peaks using matching on a spectra database.

Description

This tool performs LC/MS matching on an input list of MZ/RT values, using either a provided in-house single file database or a connection to Peakforest database.

Key features

- LCMS annotation.

Functionality

- Annotation / MS / L2a Library Spectrum Match
- Post-processing

Approaches

- Metabolomics / Untargeted

Figure 2: Workflow tool from Figure 1, as it appears in the GitHub repository

The specific requirements for PhenoMeNal GitHub readme.md files are found [here](#)¹

3.3 GitHub wiki and VRE technical documentation

Following the same principles as described above, there are templates for PhenoMeNal technical documentation. The PhenoMeNal Wiki² contains all the technical documentation for each respective tool and workflow. To dynamically read and reuse this documentation, another template was designed. All tool descriptions follow the defined template, which is then automatically read to extract the data required to generate the VRE Help³ pages.

¹ <https://portal.phenomenal-h2020.eu/help/The-Guideline-for-Container-GitHub-Respository-README.md-Creation>

² <https://github.com/phnmnl/phenomenal-h2020/wiki>

³ <https://portal.phenomenal-h2020.eu/help>



3.4 Galaxy and Docker templates

Each tool included in the VRE has to follow a set of guidelines to enable seamless inclusion. Specifically, tools have to be documented through a ReadMe file that adheres to the standard format to provide required metadata, a Docker container configuration and, if necessary, a Galaxy wrapper.

The definition is available from the technical help documentation, as referenced above.

More specifically the details regarding:

- Docker file creation and details are found [here](#)⁴
- Creating a Galaxy wrapper are found [here](#)⁵

3.5 Release process

In PhenoMeNal we have adopted a process to organise the *release* of PhenoMeNal tools and services, which results in a snapshot of the software that is suitable for general usage. While the work done by the PhenoMeNal consortium is generally almost immediately available to the public through the software repositories visible on GitHub, software that is undergoing development is usually unstable, and thus unsuited for day-to-day usage.

The release process bi-annually labels snapshots of the tools and infrastructure that are considered production ready, and thus suitable for general usage. At this point, new tools also are evaluated to determine whether they are suitable for inclusion and obsolete ones are eliminated. The labelled snapshots together form the *release candidate*, which undergoes a period of testing. Finally, the official release occurs with the snapshot of the PhenoMeNal tools and infrastructure being named – for easy identification – and made public with the endorsement of the consortium.

The next PhenoMeNal release, *Cerebellin*, was published end of February 28th 2018. The full release schedule is available on the wiki⁶.

3.6 Standardised automated testing procedure

PhenoMeNal has standardised all testing by using the Jenkins Continuous Integration (CI) system⁷. Each time code is checked into any production or development repository belonging to the PhenoMeNal GitHub organisation, a thorough automated test is performed. The software including new changes is built and tested on data. Upon successful testing, the CI system makes the updated Docker image available to users through the PhenoMeNal Docker Container Registry⁸.

⁴ <https://portal.phenomenal-h2020.eu/help/Dockerfile-Guide>

⁵ <https://portal.phenomenal-h2020.eu/help/How-to-make-your-software-tool-available-through-PhenoMeNal>

⁶ <https://github.com/phnmnl/phenomenal-h2020/wiki/Release-information>

⁷ <https://portal.phenomenal-h2020.eu/help/Jenkins-Guide>

⁸ <https://container-registry.phenomenal-h2020.eu/>



3.7 Standardised tool and container versioning

A standardised versioning scheme for the tools and tool containers in PhenoMeNal has been designed to allow unequivocal identification of the specific tools versions included with a given PhenoMeNal release. Any given tool packaged in PhenoMeNal is composed of two main parts:

1. The software tool itself;
2. The Docker container, which wraps the tool.

Since variations in either component can result in a change in the overall system behaviour, they are each versioned individually, thus resulting in a composite version number for each packaged tool. For instance, the latest packaged snapshot of the *iso2flux* tool has version v0.2_cv1.1.36, which is to be interpreted as: iso2flux version 0.2; container version 1.1.36. Any change to either the tool code or its container will result in an increment to the relevant version number.

4 Delivery and Schedule

The deliverable has been submitted on time.

5 Conclusion

Having defined concise standardised templates and procedures for inclusion of tools and workflows, there is now an automated streamlined process to ensure the VRE Portal software, Service Catalogue & App Library is continuously up to date.