ODAM: Open data and access to data mining

Deployment and User's Guide

Document version 1.0

Daniel J. Jacob

INRAE UMR 1332 BFP, Metabolomics Facility

France

Table of contents

1 - Introduction

In life sciences, (and particularly in plant sciences), each time an experimental design is implemented, we can, very schematically, represent the data flow according to 4 main steps, from raw data to publication of results.

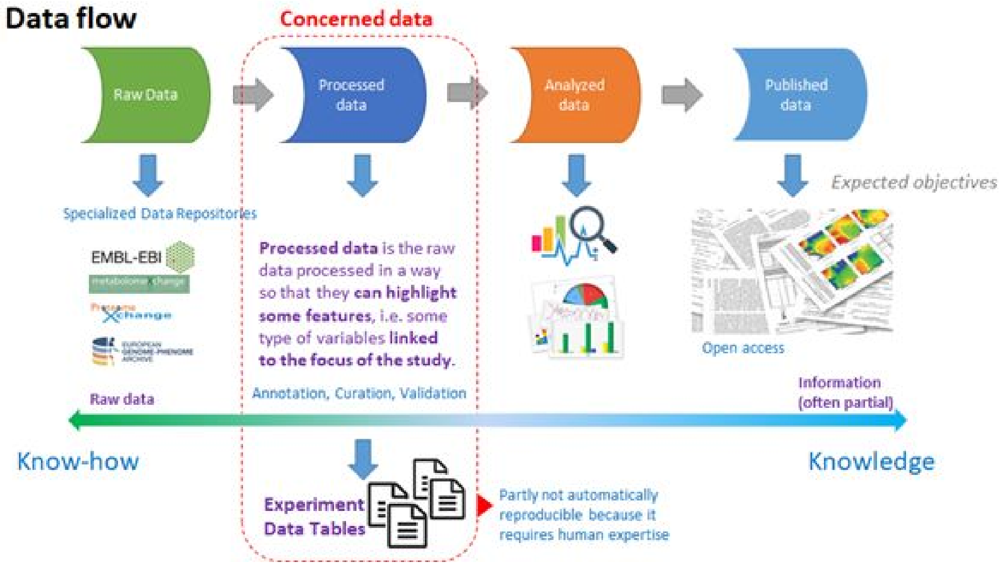

Data flow
Concerned data
Raw Data
Processed data
Analyzed data
Published data
Expected objectives
Specialized Data Repositories
EMBL-EBI
metabolomeXchange
ProteomeXchange
EUROPEAN GENOME-PHENOME ARCHIVE
Processed data is the raw data processed in a way so that they can highlight some features, i.e. some type of variables linked to the focus of the study.
Open access
Raw data
Annotation, Curation, Validation
Information (often partial)
Know-how
Knowledge
Experiment Data Tables
Partly not automatically reproducible because it requires human expertise

Figure 1.1 *: Very Schematically each time an experimental design is implemented one can represent the data flow according to 4 main steps, from raw data to publication of results.*

From a biological point of view, the data integrating the maximum amount of relevant information are those resulting from the pre-processing of so-called "raw" data (resulting from analytical techniques) and including annotations with curation then validated by one or more experts; i.e. those involving a transformation of analytical variables (peaks or resonances on spectra, locus on a DNA / RNA / Protein sequence, ...) into biological variables (metabolites, proteins, genes, ...). At this stage, because they are not synthesized, they still have all their variabilities (technological and biological replicas on all factorial levels) and therefore have more potential for reuse. Moreover, these data are not automatically reproducible (e.g. via workflows) because they require human expertise (i.e. know-how). This is why we have focused our data management on these experimental data tables.

The ODAM software proposes a simple way to make research data broadly accessible and fully available for reuse, including by a script language such as R. The main purpose is to make a dataset accessible online with minimal effort from the data provider, and to allow any scientists or bioinformaticians to be able to explore the dataset and then extract a subpart or the totality of the data according to their needs.

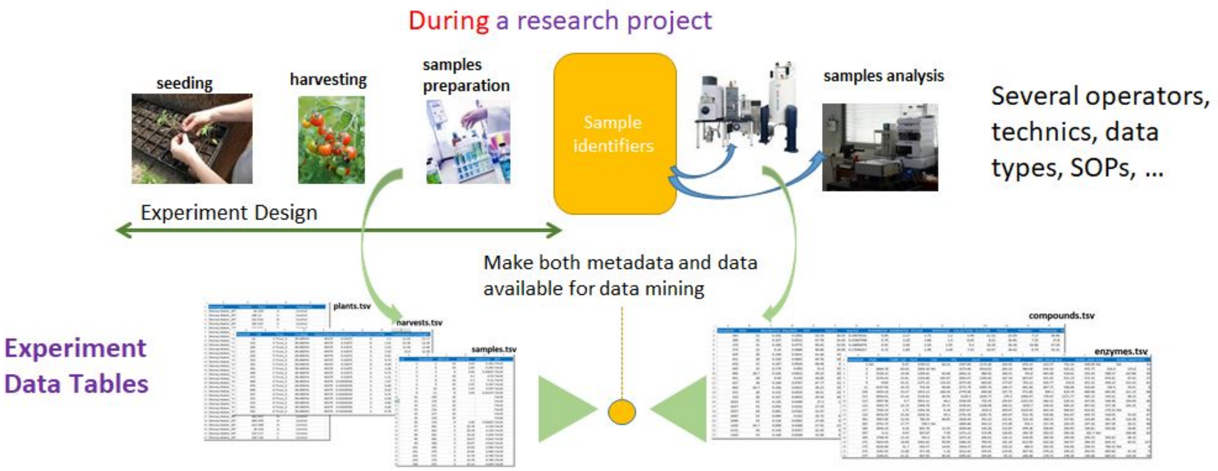


Figure 1.2 *When generating data in an experiment involving several types of data from several analytical techniques, and this for the same samples, the task of being able to easily link these different data on the basis of sample identifiers is crucial. This is because the consistency of the data must be ensured throughout the experiment, so that it becomes unnecessary for each member to conduct a laborious investigation to find out who has the correct identifiers. These operations are classically managed by a LIMS, a software which is generally very expensive.*

Properly managing your data as early as possible in a scientific study undoubtedly 1) saves time later on, 2) allows for greater consistency in data handling and is therefore more efficient. But organizing and structuring can be complex without tools and methods and sometimes requires above-average computer skills.

The ODAM software aims at being able to manage experimental data tables well without requiring high IT skills (developing a data model for example). Indeed, ODAM proposes to meet certain needs typically encountered during the implementation of an experimental design in life science including several different analyses of the same sample.

1. <u>Data collecting and preparation</u>
    - The formatting of all the data and matching the data from the different analyses with their experimental context can be a long step. Tasks such as collecting and preparing data in order to combine several data sources require a lot of long, repetitive and tedious manipulations. Similarly, when modeling, subsets must be selected and then many scenarios with different parameters must be tested.

2. <u>Data sharing</u>
    - Enabling centralized management of identifiers (e.g. plants, crops, samples, etc.) so that they are unique and shared by all project members. Indeed, as each biological sample is most often aliquoted and then sent for analysis by different techniques, the data returned in tabular form must be able to be linked to the other data according to the identifiers of the samples (Fig 1.2)

  - Giving access to data for rapid use by each project member and this throughout the development phase, from the implementation of the experimental design to the acquisition of data from the various sample analyses, so that all data are readily available as soon as they are generated.

3. Data publishing
   - To be able to publish one's data without a colossal effort of formatting, and without the need for data archaeology.
   - To be able to publish one's data according to the FAIR principles, at least the essentials
   - To facilitate the reuse of data by providing structural metadata, thus avoiding that data consumers spend a disproportionate amount of time trying to understand the digital resources they need and devising specific ways to combine them [3].

For this work, we made the choice to keep the good old way of scientists to use worksheets, thus using the same tool for both data files and metadata files. Moreover, our approach gives data access through web services thus providing a good way to connect distributed data.
This approach has to be regarded as complementary with publication of the data online within an institutional data repository as described in re3data.org for instance, associated or not with a scientific paper (see *Data publishing*).

An ODAM entry has been registered on bio.tools (ELIXIR, Europe’s leading life science organisations) in order to find all the links to the necessary tools and information. See https://bio.tools/ODAM

1.1 A concrete example:

- In order to illustrate each of the steps in this guide in concrete terms, we have chosen a dataset from an experiment on tomato fruits grown in greenhouses [1] and managed using ODAM software. The aim of this study was to build a model of fruit growth.
  - Data explorer https://pmb-bordeaux.fr/dataexplorer/?ds=frim1
  - Dataverse : https://doi.org/10.15454/95JUTK
  - Both repositories are supported by INRAE (https://www.inrae.fr/en, France) for a minimum period of 10 years (until 2030).
  - See *Appendix 1: a dataset example*

2 - Installation

The ODAM software can be deployed at multiple scales (local, intranet, internet), depending on the need and the target community.

2.1 Local installation

2.1.1 Deployment of ODAM with full functionality

This type of deployment is reserved more specifically for computers running under MS Windows 10 and Apple MacOS 10.x.

All the steps involved in the installation of the virtual machine have been published and are available online on the INRAE Data website.

> "ODAM Virtual Disk Image (VDI) for Oracle VM VirtualBox - Zipped with 7zip.", https://doi.org/10.15454/C9LAEF, Portail Data INRAE, V1

The three main steps are shown below:

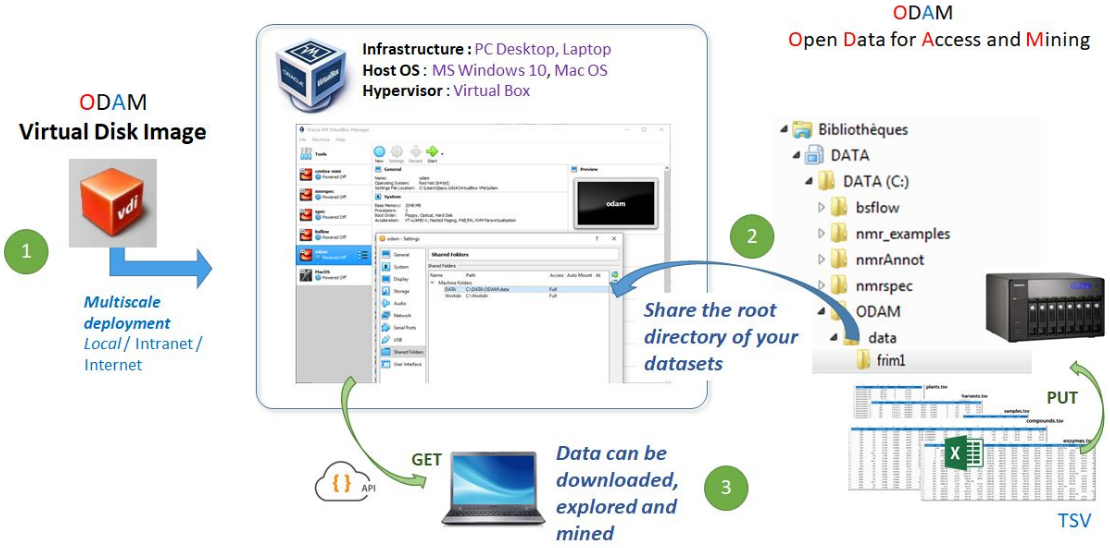

ODAM
Open Data for Access and Mining
Infrastructure : PC Desktop, Laptop
Host OS : MS Windows 10, Mac OS
Hypervisor : Virtual Box
ODAM
Virtual Disk Image
vdi
1
Multiscale deployment
Local / Intranet / Internet
2
Share the root directory of your datasets
Bibliothèques
DATA
DATA (C:)
bsflow
nmr_examples
nmrAnnot
nmrspec
ODAM
data
frim1
PUT
TSV
GET
API
Data can be downloaded, explored and mined
3

Figure 2.1: *Overview of the ODAM software deployment on your PC (Windows 10, MacOS 10;x)*

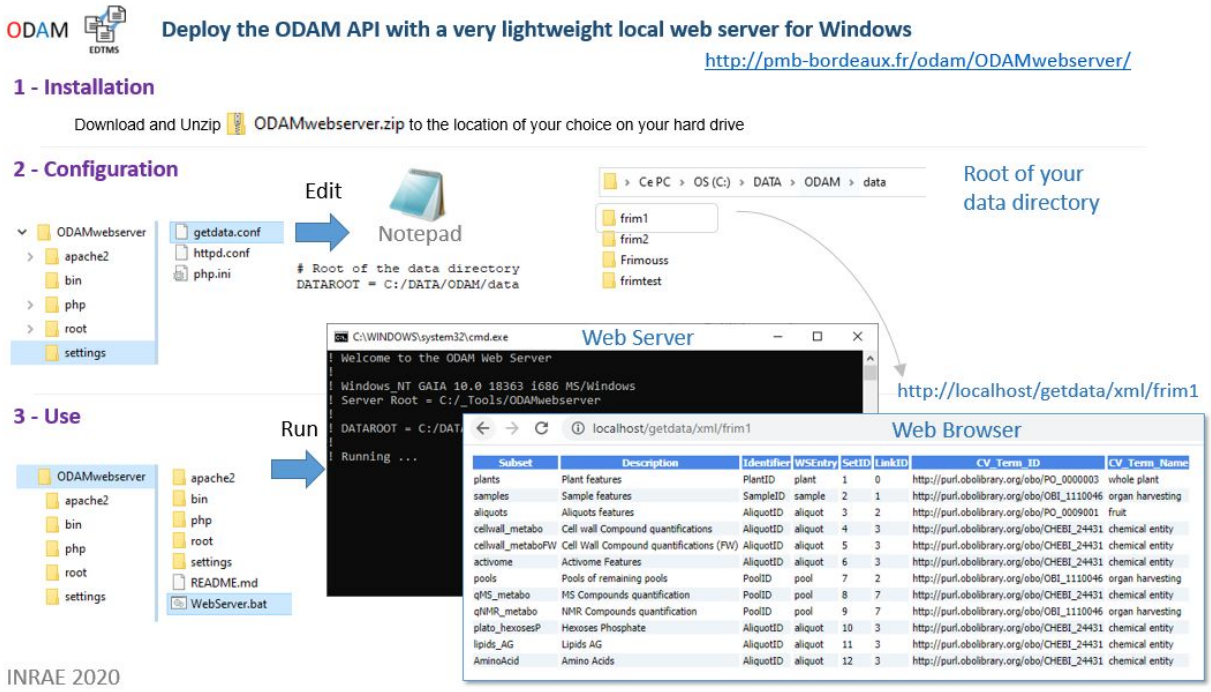
1. Download the virtual machine file, to be unzipped with 7zip. This requires prior installation of Oracle VM VirtualBox software
2. Data collection and preparation (see below)
3. Using ODAM (see below)

2.1.2 ODAM API on Windows

Deploy the ODAM API with a very lightweight local web server for Windows.

For Windows users, ODAMwebserver offers a simple, fast and efficient way to deploy the ODAM API layer locally in (almost) one click.

> "ODAM API using Apache Web Server
> http://pmb-bordeaux.fr/odam/ODAMwebserver/



Deploy the ODAM API with a very lightweight local web server for Windows

http://pmb-bordeaux.fr/odam/ODAMwebserver/

1 - Installation

Download and Unzip ODAMwebserver.zip to the location of your choice on your hard drive

2 - Configuration

```
# Root of the data directory
DATAROOT = C:/DATA/ODAM/data
```

Root of your data directory

3 - Use

```
! Welcome to the ODAM Web Server
!
! Windows_NT GAIA 10.0 18363 i686 MS/Windows
! Server Root = C:/_Tools/ODAMwebserver
!
! DATAROOT = C:/DAT
!
! Running ...
```

http://localhost/getdata/xml/frim1

Subset	Description	Identifier	WSEntry	SetID	LinkID	CV_Term_ID	CV_Term_Name
plants	Plant features	PlantID	plant	1	0	http://purl.obolibrary.org/obo/PO_0000003	whole plant
samples	Sample features	SampleID	sample	2	1	http://purl.obolibrary.org/obo/OBI_1110046	organ harvesting
aliquots	Aliquots features	AliquotID	aliquot	3	2	http://purl.obolibrary.org/obo/PO_0009001	fruit
cellwall_metabo	Cell wall Compound quantifications	AliquotID	aliquot	4	3	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity
cellwall_metaboFW	Cell Wall Compound quantifications (FW)	AliquotID	aliquot	5	3	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity
activome	Activome Features	AliquotID	aliquot	6	3	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity
pools	Pools of remaining pools	PoolID	pool	7	2	http://purl.obolibrary.org/obo/OBI_1110046	organ harvesting
qMS_metabo	MS Compounds quantification	PoolID	pool	8	7	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity
qNMR_metabo	NMR Compounds quantification	PoolID	pool	9	7	http://purl.obolibrary.org/obo/OBI_1110046	organ harvesting
plato_hexosesP	Hexoses Phosphate	AliquotID	aliquot	10	3	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity
lipids_AG	Lipids AG	AliquotID	aliquot	11	3	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity
AminoAcid	Amino Acids	AliquotID	aliquot	12	3	http://purl.obolibrary.org/obo/CHEBI_24431	chemical entity

INRAE 2020

__Figure 2.2__: Overview of the ODAM API deployment (Windows 10)

Just follow the instructions given in the README file : after downloading the ZIP file, just unzip it to your workspace, specify in the appropriate configuration file the full path to your data on disk, launch the WebServer application and you're done.

Note: For MacOS (10.13 or newer), you can easily install Docker, then proceed as described in the next section "*Intranet/Internet installation*".

2.2 Intranet/Internet installation

This type of deployment is reserved more specifically for computers or IT infrastructures running Linux, and allows the deployment of Docker containers. See for more details on “What is Docker” on opensource.com.

Note: Depending on the scale you want to deploy, your intranet can be local or via a VPN (Virtual Private Network) and for the internet, a machine on an institutional data center (e.g. university) is better but a machine on a cloud computing is a perfectly feasible solution.

All the steps to deploy the Docker containers corresponding to the ODAM software have been put online :

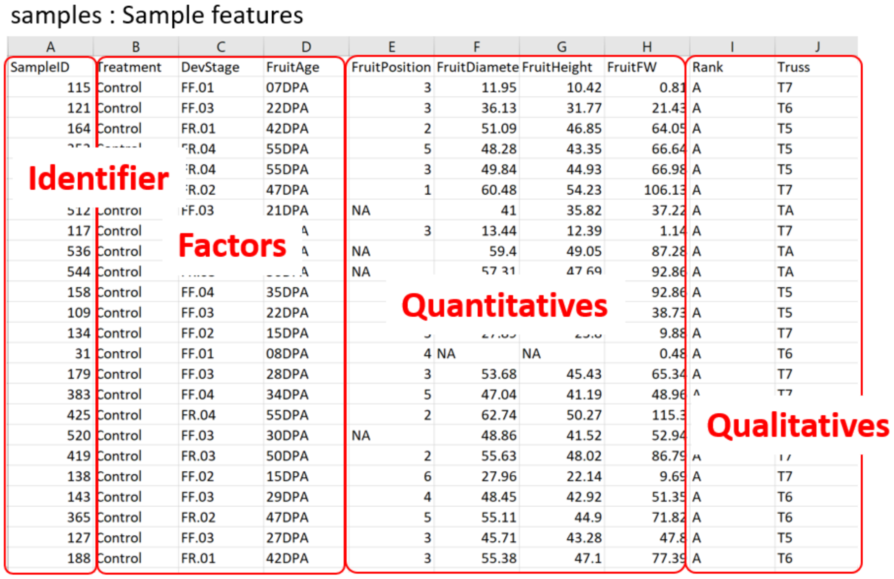
> Github : https://github.com/inrae/ODAM
> DockerHub : https://hub.docker.com/r/odam/getdata/

Remark : Installing one's own ODAM instance on an institutional server for example allows in this way experimental data tables to be widely accessible and fully reusable including through scripting languages such as R or Python, and this with minimal effort on the part of the data provider. Thus, the web of data could be seen as a data network on the web, based on appropriate technologies (Web API), and using standard data formats (TSV, JSON). Web applications, each with a clearly defined objective, then operate this network. A data can therefore be used for several applications and vice versa. The data management system becomes completely independent of its operation. The data is thus “decompartmentalized", a sine qua non condition for the Web of Data.

3 - Data collection and preparation

3.1 Data Type

Whatever the kind of experiment, this assumes a design of experiment (DoE) involving individuals, samples or whatever things, as the main objects of study and producing several experimental data tables. This also assumes the observation of dependent variables resulting from effects of some controlled independent variables (*factors*). Moreover, the objects of study usually have an *identifier* for each of them, and the variables can be *quantitative* or *qualitative*.



samples : Sample features

A	B	C	D	E	F	G	H	I	J
SampleID	Treatment	DevStage	FruitAge	FruitPosition	FruitDiamete	FruitHeight	FruitFW	Rank	Truss
115	Control	FF.01	07DPA	3	11.95	10.42	0.81	A	T7
121	Control	FF.03	22DPA	3	36.13	31.77	21.43	A	T6
164	Control	FR.01	42DPA	2	51.09	46.85	64.05	A	T5
[illegible]	[illegible]	FR.04	55DPA	5	48.28	43.35	66.64	A	T5
[illegible]	[illegible]	FR.04	55DPA	3	49.84	44.93	66.98	A	T5
[illegible]	[illegible]	FR.02	47DPA	1	60.48	54.23	106.13	A	T7
512	Control	FF.03	21DPA	NA	41	35.82	37.22	A	TA
117	Control	[illegible]	[illegible]	3	13.44	12.39	1.14	A	T7
536	Control	[illegible]	[illegible]	NA	59.4	49.05	87.28	A	TA
544	Control	[illegible]	[illegible]	NA	57.31	47.69	92.86	A	TA
158	Control	FF.04	35DPA	[illegible]	[illegible]	[illegible]	92.86	A	T5
109	Control	FF.03	22DPA	[illegible]	[illegible]	[illegible]	38.73	A	T5
134	Control	FF.02	15DPA	[illegible]	[illegible]	[illegible]	9.88	A	T7
31	Control	FF.01	08DPA	4	NA	NA	0.48	A	T6
179	Control	FF.03	28DPA	3	53.68	45.43	65.34	A	T7
383	Control	FF.04	34DPA	5	47.04	41.19	48.96	A	T7
425	Control	FR.04	55DPA	2	62.74	50.27	115.3	[illegible]	[illegible]
520	Control	FF.03	30DPA	NA	48.86	41.52	52.94	[illegible]	[illegible]
419	Control	FR.03	50DPA	2	55.63	48.02	86.79	A	T7
138	Control	FF.02	15DPA	6	27.96	22.14	9.69	A	T7
143	Control	FF.03	29DPA	4	48.45	42.92	51.35	A	T6
365	Control	FR.02	47DPA	5	55.11	44.9	71.82	A	T6
127	Control	FF.03	27DPA	3	45.71	43.28	47.8	A	T5
188	Control	FR.01	42DPA	3	55.38	47.1	77.39	A	T6

Figure 3.1: *Example of an experiment data table viewed according to the repartition by category as introduced in the text*

3.2 Data management : promote good practices

First and foremost, it is important to have well-organized data. The files generated during data collection have to be organized according to the entity-attribute relational model. Indeed, each entity corresponds to a type of collected data (samples, compounds, ...) for which is associated a set of attributes, i.e. observed or measured variables. Well organized data means that each variable forms a column, each observation forms a line, and each type of "unit observational" forms a table, i.e a file. Then, a *link* is established for each subset with the subset from which it was obtained, so that the links can be interpreted as "*obtained from*", since each column of each subset of data must be associated with a type of experimental data (called a *category*), especially those corresponding to identifiers that the links are based on.

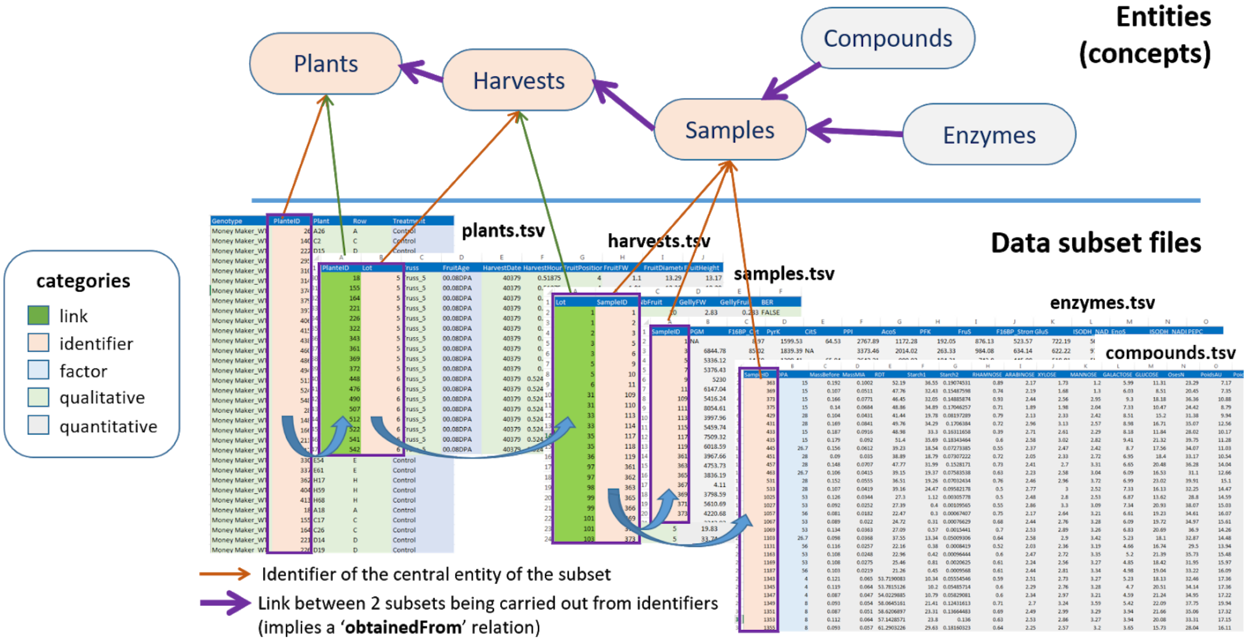


Figure 3.2*: A link is established for each subset with the subset from which it was obtained, so that the links can be interpreted as "obtained from", since each column of each subset of data must be associated with a type of experimental data (called a category), especially those corresponding to identifiers that the links are based on.*

Another good practice is to promote non-proprietary formats such as TSV, which is a necessary and indispensable step towards "open linked data" (5 gold stars principle). So an ODAM dataset is a bundle that contains a set of TSV files. The TSV files are simple tables containing the data of the dataset.

All steps concerning the collection and preparation of data have been published and are available online at protocols.io

> Data Preparation Protocol for ODAM Compliance. protocols.io
> https://dx.doi.org/10.17504/protocols.io.betcjeiw

The purpose of this protocol is to describe all the steps involved in collecting, preparing and annotating the data from an experiment associated with an experimental design (DoE) that will then allow the user to benefit from the services offered by ODAM. The overall approach is based on good data management practices concerning data structuring and the description of structural metadata.

Indeed, ODAM allows to put metadata in depth, i.e. at the level of the data itself (i.e. metadata at column-level such as factors, variables,...) and not only as a "hat" on the data set. Thus, having the actual data elements also machine-readable makes the dataset of a higher level of interoperability and makes functional interlinking and analysis in broader context much easier.

4 - Services provides by ODAM

The overall usage scheme of ODAM is shown in the following figure:

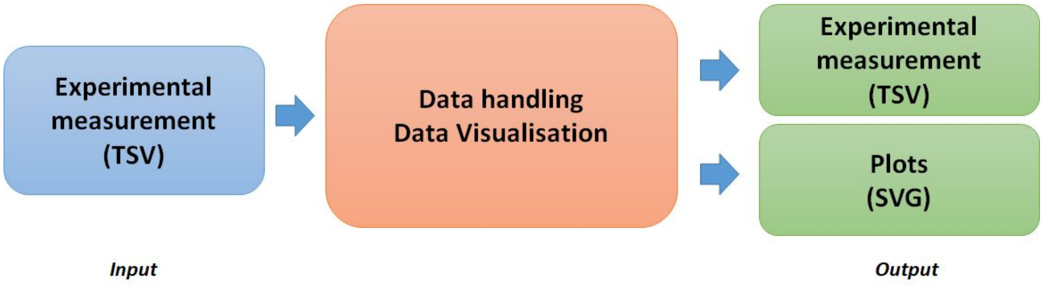

Experimental measurement (TSV)
Data handling
Data Visualisation
Experimental measurement (TSV)
Plots (SVG)
Input
Output

***Figure 4.1** : The overall usage scheme of ODAM*

As input, we have the various data files from the experiment. You can view the data graphically (see *Explore your data* section) and you can also merge certain files based on common identifiers and then select a sub-part. The output is the file resulting from the query.

The central idea which has been the founding idea of ODAM, is that data producers "just" have to drag and drop their data tables onto a storage space, which depending on the chosen infrastructure can be local (i.e. their PC, or a NAS) or remote (virtual disk space). See *Installation* section.

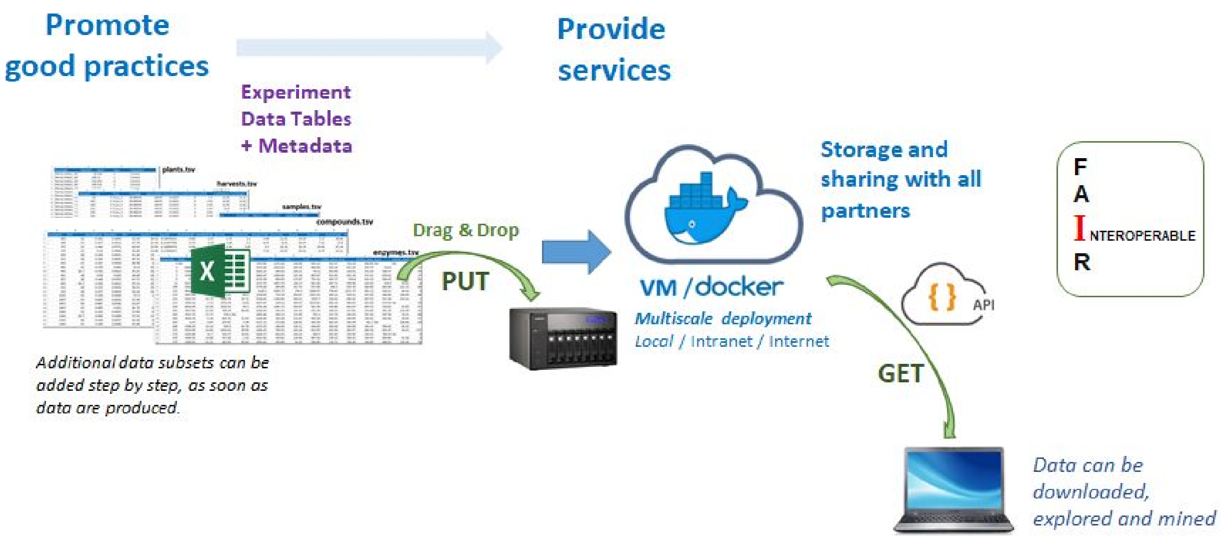


***Figure 4.2:** The central idea is that data producers "just" have to drag and drop their data tables onto a storage space depending on the chosen infrastructure (local, intranet, internet). An API (Application Programming Interface) layer is implemented which allows data handling (data selection and merging)*

So simply dropping data files (along with two additional metadata files) on the storage space allows users to access them through web services. This means there is no need for additional configuration on the server (Fig 4.2).

The dataset must first be previously formatted according to precise rules which requires that some good practices be followed and adhered to. See *Data collection and preparation section*.

Note: In case of having a remote storage space, the Syncany solution (Dropbox-like) might be a good choice. Several other tools or approaches can be used as: WinSCP, SAMBA, rsync, ...

Advantages of this approach

From a user's point of view, data collection should be done as early as possible in a project. There are several reasons:

- First, experimental design is defined very early, so it is possible to share in the form of a spreadsheet the list of identifiers of individuals with their unique identifiers (plant identifiers) associated with the particular features defining the factorial group (metadata) such as: genotype, the type of treatment, its position in the greenhouse or in the field, … Thus, the array of the "plants" may be created even before planting the seeds. So too, the array of the "harvesting" can be done as soon as harvested, before any analysis. Then, each analysis comes complementing the dataset as soon as they produce their data subset.
- Whether the data are acquired in the field or in the laboratory, their acquisition is carried out gradually, and is modified according to the first statistical treatments and preliminary results. This iterative aspect of production requires traceability and management tools. Typically, an experiment design often implies if followed as planned plenty of samples due to the multiplication of several numbers: number of factors X the number of factor levels X number of biological replication X different types of analysis X number of technical replication for each analysis type. Giving the possibility to explore the preliminary data (e.g. the distributions of some features of the samples), will allow scientists to eliminate outliers, thus avoiding subsequent analysis of these samples.
- Data have to be accessible to everyone as soon as they are produced, allowing other data producers in the data string to link the identifiers of their analytical samples within the information system and subsequently allowing all partners to gather data with the metadata (factors, dates, ...) of the Design of Experiment. By proceeding in this way, we ensure data coherence all the experiment time, and thus it becomes useless for each member to proceed a laborious investigation to find out who possesses the right identifiers.

From a technical point of view, the great advantage is to avoid the implementation of a complex data management system (requiring a data model) given that many changes can occur during the project. (possibility of new analysis, new measures or renouncing some others, ...). It also facilitates the subsequent publication of data: either the data can serve to fill in an existing database or the data can be broadcast through a web-service approach with the associated metadata. See *Data publishing.*

Tasks such as combining several data files are time-consuming and require repetitive and tedious handling. Similarly, when modeling, subsets have to be selected and many scenarios tested with different parameters. If performed manually (i.e., multiple copy-paste in a spreadsheet) these tasks are not only tedious but prone to handling errors. ODAM software is designed to perform precisely these tasks by providing several services for greater flexibility in data handling.

The ODAM software embeds an API (Application Programming Interface) layer which offers data selection and merging services. It is this API layer that allows interoperability between the different tables and the applications that will be able to use them (See *Appendix 3*).

With the help of this layer, it opens up a whole ecosystem of potential applications, depending on your needs but also on your skills in the proposed tools.

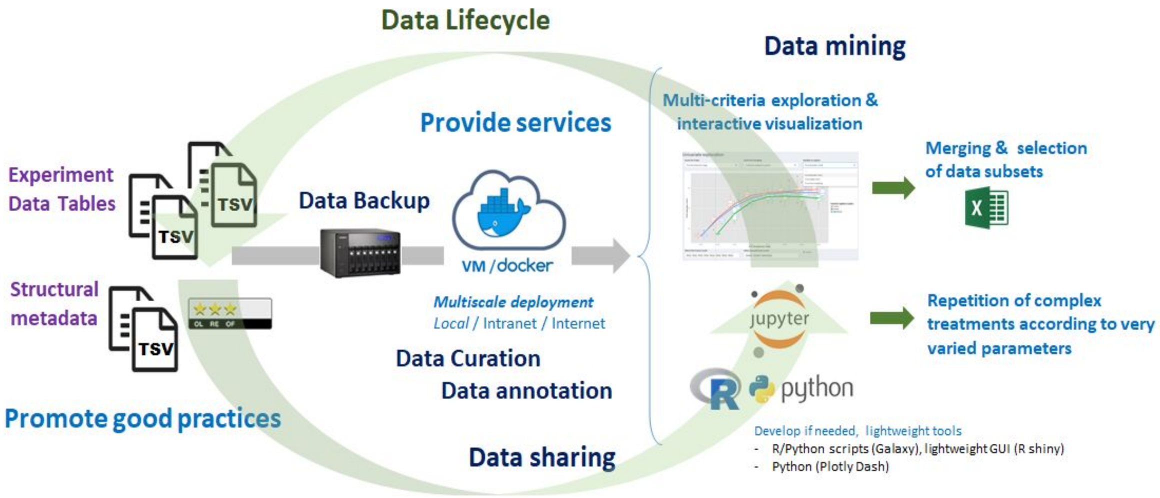
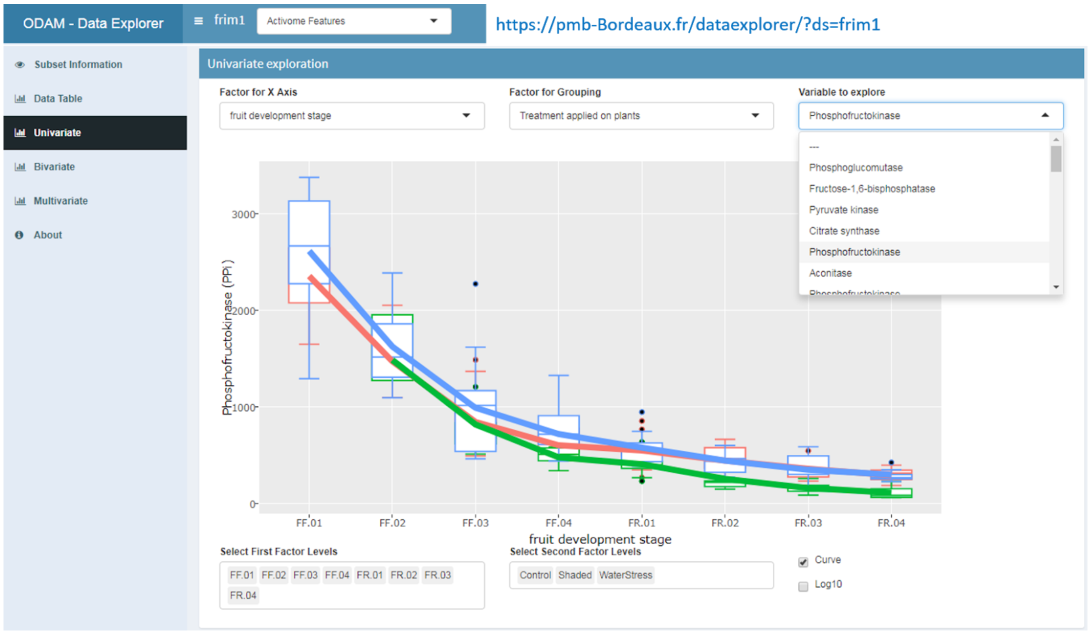


Figure 4.3: *From the set of data files (which are uncombined tables, each corresponding to a particular observational unit that we name an entity), the user can: 1) Visualize the data associated with their metadata according to several criteria and in a completely interactive way with the help of the data explorer. 2) Export in tabular form subsets selected according to his criteria with combined, merged data. 3) Build and test its models more easily using a scripting language such as R, which allows it to repeat different scenarios according to a variety of parameters. An R package allows to perform extractions according to the same criteria as those proposed in the data explorer. All this is possible thanks to the API layer, which opens up a whole ecosystem of potential applications.*

4.1 Explore your data

The Data Explorer makes data easy to explore, visualize, and subsequently to better understand the data as a whole. Explore your data in several ways according to your concerns by interacting with the graphs. For instance, univariate, bivariate and multivariate approaches have been implemented so that they are very easy to be interactively used. This is very useful in order to have a first glimpse of the data that can show trends and this allows the data to be well characterized, which is necessary to then choose how to analyze it later on.



https://pmb-Bordeaux.fr/dataexplorer/?ds=frim1

Activome Features

Univariate exploration

Factor for X Axis: fruit development stage

Factor for Grouping: Treatment applied on plants

Variable to explore: Phosphofructokinase

Select First Factor Levels: FF.01 FF.02 FF.03 FF.04 FR.01 FR.02 FR.03 FR.04

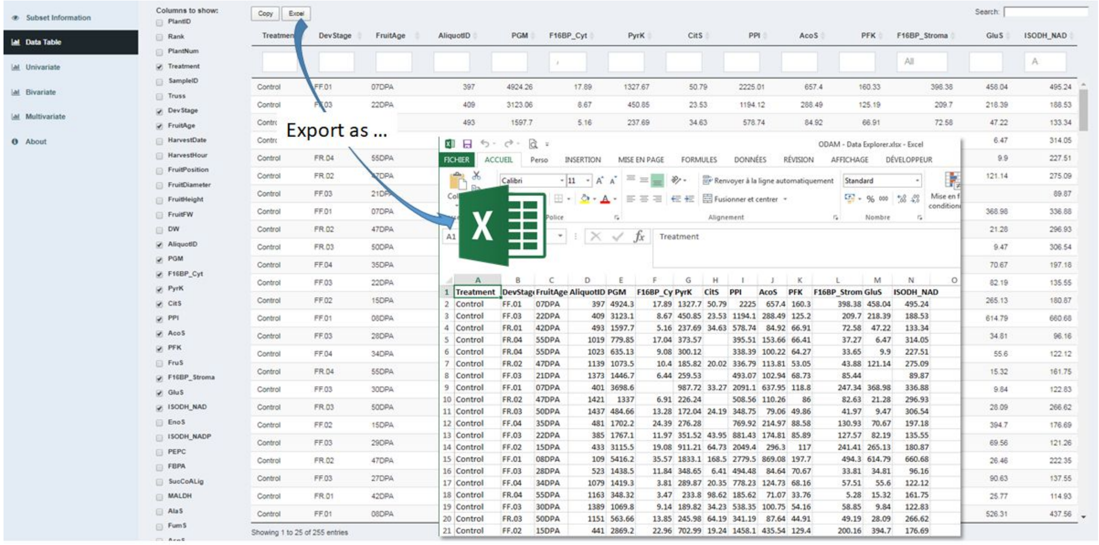
Select Second Factor Levels: Control Shaded WaterStress

☑ Curve
☐ Log10

***Figure 4.3**: Example of graphical output produced by the Data Explorer from* https://pmb-bordeaux.fr/dataexplorer/?ds=frim1

For each analysis, several possibilities can be explored by selecting many parameters and interacting with the graphs. Graphics can be easily exported as PNG files.

An interesting possibility is that you can make a selection of a subset of data and then export it directly into your spreadsheet.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Treatment	DevStage	FruitAge	AliquotID	PGM	F16BP_Cy	PyrK	CitS	PPI	AcoS	PFK	F16BP_Strom	GluS	ISODH_NAD	
2	Control	FF.01	07DPA	397	4924.3	17.89	1327.7	50.79	2225	657.4	160.3	398.38	458.04	495.24	
3	Control	FF.03	22DPA	409	3123.1	8.67	450.85	23.53	1194.1	288.49	125.2	209.7	218.39	188.53	
4	Control	FR.01	42DPA	493	1597.7	5.16	237.69	34.63	578.74	84.92	66.91	72.58	47.22	133.34	
5	Control	FR.04	55DPA	1019	779.85	17.04	373.57		395.51	153.66	66.41	37.27	6.47	314.05	
6	Control	FR.04	55DPA	1023	635.13	9.08	300.12		338.39	100.22	64.27	33.65	9.9	227.51	
7	Control	FR.02	47DPA	1139	1073.5	10.4	185.82	20.02	336.79	113.81	53.05	43.88	121.14	275.09	
8	Control	FF.03	21DPA	1373	1446.7	6.44	259.53		493.07	102.94	68.73	85.44		89.87	
9	Control	FF.01	07DPA	401	3698.6		987.72	33.27	2091.1	637.95	118.8	247.34	368.98	336.88	
10	Control	FR.02	47DPA	1421	1337	6.91	226.24		508.56	110.26	86	82.63	21.28	296.93	
11	Control	FR.03	50DPA	1437	484.66	13.28	172.04	24.19	348.75	79.06	49.86	41.97	9.47	306.54	
12	Control	FF.04	35DPA	481	1702.2	24.39	276.28		769.92	214.97	88.58	130.93	70.67	197.18	
13	Control	FF.03	22DPA	385	1767.1	11.97	351.52	43.95	881.43	174.81	85.89	127.57	82.19	135.55	
14	Control	FF.02	15DPA	433	3115.5	19.08	911.21	64.73	2049.4	296.3	117	241.41	265.13	180.87	
15	Control	FF.01	08DPA	109	5416.2	35.57	1833.1	168.5	2779.5	869.08	197.7	494.3	614.79	660.68	
16	Control	FF.03	28DPA	523	1438.5	11.84	348.65	6.41	494.48	84.64	70.67	33.81	34.81	96.16	
17	Control	FF.04	34DPA	1079	1419.3	3.81	289.87	20.35	778.23	124.73	68.16	57.51	55.6	122.12	
18	Control	FR.04	55DPA	1163	348.32	3.47	233.8	98.62	185.62	71.07	33.76	5.28	15.32	161.75	
19	Control	FF.03	30DPA	1389	1069.8	9.14	189.82	34.23	538.35	100.75	54.16	58.85	9.84	122.83	
20	Control	FR.03	50DPA	1151	563.66	13.85	245.98	64.19	341.19	87.64	44.91	49.19	28.09	266.62	
21	Control	FF.02	15DPA	441	2869.2	22.96	702.99	19.24	1458.1	435.54	129.4	200.16	394.7	176.69	

***Figure 4.4** : Export a previously selected data subset.*

4.2 Export data from your web browser

You can retrieve data directly from your web browser based on the API. To understand the syntax, see *Appendix 2 : Web Services.*

> You can also test from the API Documentation on SwaggerHub :
> https://app.swaggerhub.com/apis-docs/INRA-PMB/ODAM/1.0.1-oas3/

Example :

- First, view the subset list of a dataset along with the metadata
  https://pmb-bordeaux.fr/getdata/xml/frim1
- *Then,* retrieve a data table by merging "Activome Features" data (enzymes) and "NMR Compounds quantification" data
  https://pmb-bordeaux.fr/getdata/tsv/frim1/(activome,qNMR_metabo)

  If you have associated the files having the extension '.txt' (tab as separator) with e.g. MS Excel, then just click on it to open it.

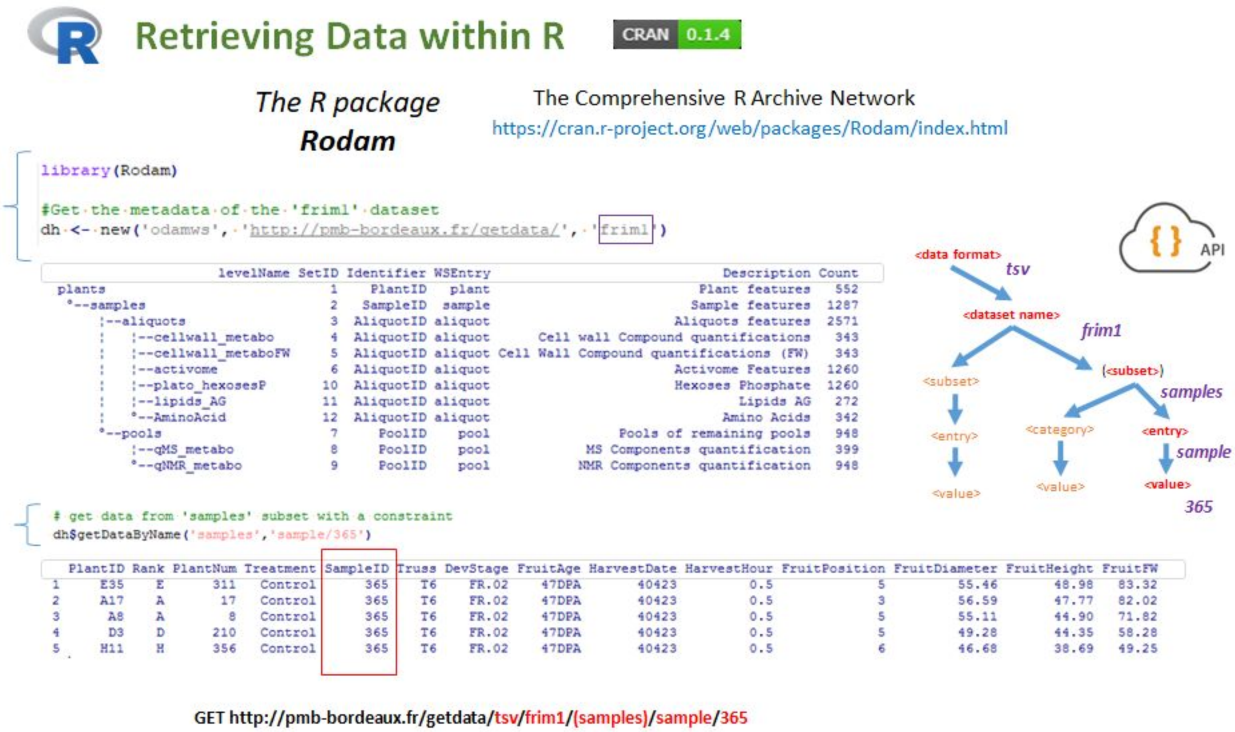
See *Appendix 3* for more details on how API requests work.

4.3 Using the API with R

It is possible to adopt more efficient approaches to analyze data, e.g for automating various processing operations, or for allowing users to select subsets of data and then carry out numerous repetitions of complex processing operations according to a wide variety of parameters (scenarios).
For this purpose we have developed an R (Rodam) package that allows data extraction according to the API schema (see *Appendix 2*)

The Rodam package provides a set of functions to retrieve data and their metadata from data sets formatted according to the ODAM data structuring approach.

> See the vignette "Demonstration of the functionalities of the R ODAM package"
> https://cran.r-project.org/web/packages/Rodam/vignettes/Rodam.html



Retrieving Data within R

The R package
Rodam

The Comprehensive R Archive Network
https://cran.r-project.org/web/packages/Rodam/index.html

```
library(Rodam)

#Get the metadata of the 'frim1' dataset
dh <- new('odamws', 'http://pmb-bordeaux.fr/getdata/', 'frim1')
```

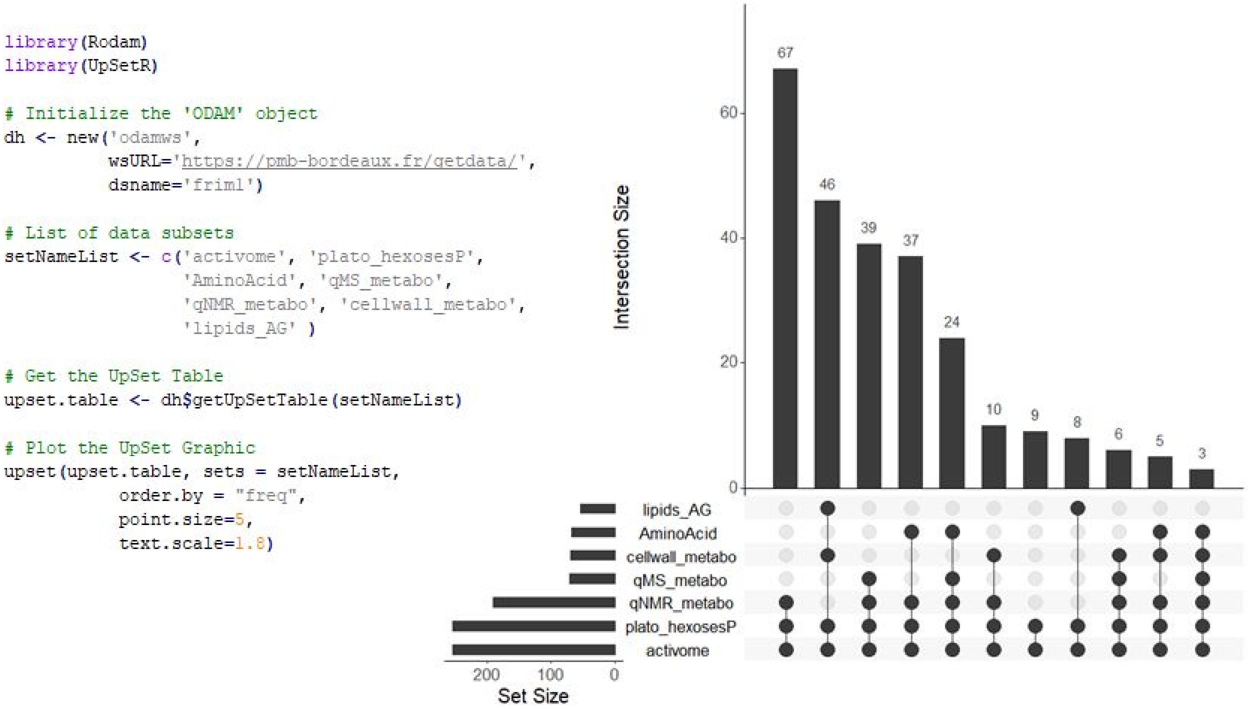
```
                     levelName SetID Identifier WSEntry                               Description Count
plants                             1    PlantID   plant                            Plant features   552
 °--samples                        2   SampleID  sample                           Sample features  1287
     ¦--aliquots                   3  AliquotID aliquot                          Aliquots features  2571
     ¦   ¦--cellwall_metabo        4  AliquotID aliquot        Cell wall Compound quantifications   343
     ¦   ¦--cellwall_metaboFW      5  AliquotID aliquot Cell Wall Compound quantifications (FW)     343
     ¦   ¦--activome               6  AliquotID aliquot                          Activome Features  1260
     ¦   ¦--plato_hexosesP        10  AliquotID aliquot                          Hexoses Phosphate  1260
     ¦   ¦--lipids_AG             11  AliquotID aliquot                                  Lipids AG   272
     ¦   °--AminoAcid             12  AliquotID aliquot                               Amino Acids   342
     °--pools                      7     PoolID    pool                  Pools of remaining pools   948
         ¦--qMS_metabo             8     PoolID    pool             MS Components quantification   399
         °--qNMR_metabo            9     PoolID    pool            NMR Components quantification   948
```

```
# get data from 'samples' subset with a constraint
dh$getDataByName('samples','sample/365')
```

```
  PlantID Rank PlantNum Treatment SampleID Truss DevStage FruitAge HarvestDate HarvestHour FruitPosition FruitDiameter FruitHeight FruitFW
1     E35    E      311   Control      365    T6    FR.02    47DPA       40423         0.5             5         55.46       48.98   83.32
2     A17    A       17   Control      365    T6    FR.02    47DPA       40423         0.5             3         56.59       47.77   82.02
3      A8    A        8   Control      365    T6    FR.02    47DPA       40423         0.5             5         55.11       44.90   71.82
4      D3    D      210   Control      365    T6    FR.02    47DPA       40423         0.5             5         49.28       44.35   58.28
5     H11    H      356   Control      365    T6    FR.02    47DPA       40423         0.5             6         46.68       38.69   49.25
```

GET http://pmb-bordeaux.fr/getdata/tsv/frim1/(samples)/sample/365

***Figure 4.5** : Illustration of the use of the API under R using the Rodam package. First of all (top) one can easily view the global structure of the data set. Then (bottom), one can apply a query for a single sample (here id 365) by returning a merge of several subsets (here plant+samples).*



```
library(Rodam)
library(UpSetR)

# Initialize the 'ODAM' object
dh <- new('odamws',
          wsURL='https://pmb-bordeaux.fr/getdata/',
          dsname='frim1')

# List of data subsets
setNameList <- c('activome', 'plato_hexosesP',
                 'AminoAcid', 'qMS_metabo',
                 'qNMR_metabo', 'cellwall_metabo',
                 'lipids_AG' )

# Get the UpSet Table
upset.table <- dh$getUpSetTable(setNameList)

# Plot the UpSet Graphic
upset(upset.table, sets = setNameList,
           order.by = "freq",
           point.size=5,
           text.scale=1.8)
```

***Figure 4.6** : Illustration of the use of the API under R using the Rodam package. Visualization of set intersections between data subset based on the sample identifier and using the* UpSetR *package*

4.3 Reproducible research with Jupyter notebooks

In order to illustrate a reproducible research process, we provide examples of jupyter notebooks (R & Python) based on the ODAM Web API, as well as links to view them (Jupyter nbviewer, CodeOcean) and re-run them (MyBinder, CodeOcean)

> GitHub :
>
> - https://github.com/djacob65/binder_odam
>
> CodeOcean
>
> - Daniel Jacob (2020) Rodam API Demo [Source Code]. https://doi.org/10.24433/CO.8981049.v1
> - Daniel Jacob (2020) PyODAM API Demo [Source Code]. https://doi.org/10.24433/CO.0011270.v1

Here is a simple tutorial on how to Install Jupyter Notebook linked to your R version

- https://github.com/inrae/ODAM/files/4318705/R_Jupyter_Install.pdf

Note : A good way to share your notebooks with colleagues or project members is to install the "Littlest JupyterHub" in a datacenter (institutional or in the cloud). It is a simple JupyterHub distribution for a small (0-100) number of users on a single server.

5 - Data publishing

Publishing data according to the FAIR principles implies well-described, accessible data that complies with community standards. In the report "Turning FAIR into reality" produced by the EU [4], the authors propose as a basic minimum standard for FAIRification: discovery metadata, persistent identifiers and access to the data or metadata. Therefore, the fact of depositing one's data in a data repository (e.g. Dataverse) with descriptive metadata is sufficient to meet these criteria. However, according to Annika Jacobsen et al [3], the FAIR principles can be seen as a consolidation of good data management practices to extend management with the notion of machine-driven data reuse. In others words, the FAIR principles define the attributes that data must have to enable and enhance its reuse, by humans and machines. Regarding implementation, the authors recommend that structural metadata (e.g. links between data tables) should also be provided along with unambiguous definitions of all internal elements (e.g. column definitions, units of measurement), through links to accessible (standard) definitions. This is precisely what is targeted in the implementation of data management by the ODAM software suite.

Therefore, not only descriptive but also structural metadata should be provided.

Descriptive metadata and Structural metadata

Descriptive metadata is descriptive information about a resource. It is used for discovery and identification. It includes elements such as title, abstract, author, and keywords. They make it easier for both humans and machines to find them.

Structural metadata is metadata about containers of data and indicates how compound objects are put together. It describes the types, versions, relationships and other characteristics of digital materials. They facilitate the reuse of data by providing structural metadata, thus avoiding that data consumers spend a disproportionate amount of time trying to understand the digital resources they need and devising specific ways to combine them.

Making data available online can be understood in different ways according to their nature and complexity, and the desired granularity. The simple approach is to deposit the data in flat files (generally a set of TSV/CSV files) in dedicated repositories, with only constraint to describe the data set as a whole with minimal metadata. But without structural metadata it is difficult if not impossible to understand the relationships between different data. Moreover, a dictionary describing each file (entity) as well as all the columns of the tables (attributes) offers a better guarantee in the correct (re)use of the data. Precisely, these last two points have been implemented in the ODAM software.

Unlike data sharing, which is simply making data available to a community, data publishing also requires that the data could be referenceable (i.e. a stable identification system) and contextualizable (i.e. the who and what). These points and many others are precisely what the FAIR principles seek to establish as a frame of reference.

5.1 Based on an ODAM repository

Because ODAM is primarily an Experimental Data Table Management System (EDTMS) for data sharing, it must be associated with a suitable data repository in order to support data publishing.

So the ODAM approach has to be regarded as complementary with publication of the data online within an institutional data repository as described in re3data.org (e.g. INRAE Data Portal) associated or not with a scientific paper.

As it is generally considered good practice in FAIR when the resource and its metadata are stored independently, but persistently linked, the FAIR principles can be applied to a data ecosystem where each component contributes to meeting one or more of the criteria [3] [4]

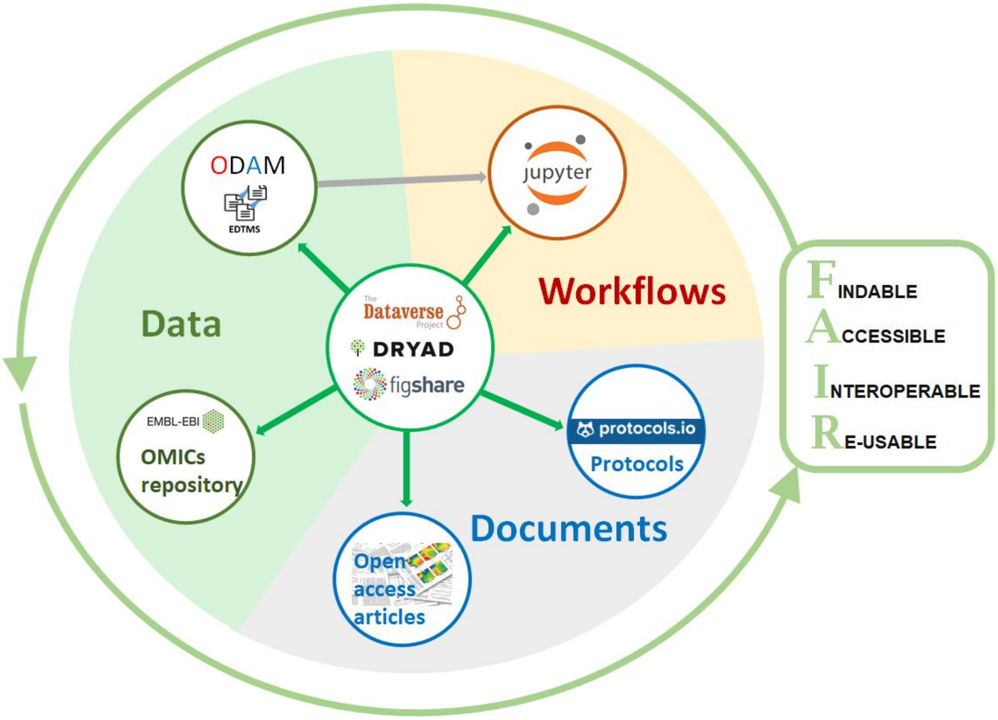
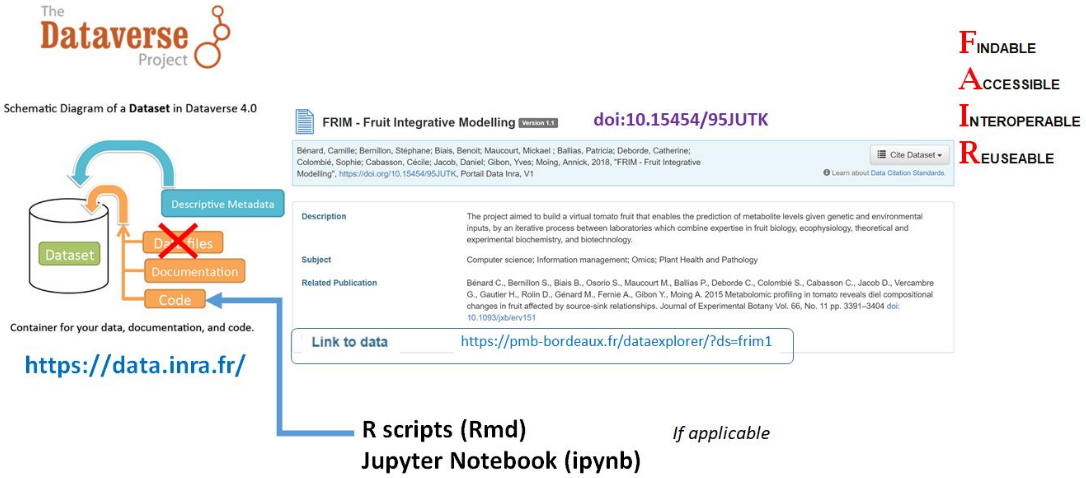


Figure 5.1*: Data publishing - Not all data, documents, workflows and other tools need to be located in a single system, but from a central repository, it is the set of links that constitutes the true information management system. It must be able to be traversed by a human being as well as by machines.*

Whereas institutional data repositories focus on the experiment description with the corresponding descriptive metadata, the ODAM approach, by adjoining some minimal but relevant structural metadata, gives access to the data themselves with the possibility to explore and mine them.



FRIM - Fruit Integrative Modelling Version 1.1

Bénard, Camille; Bernillon, Stéphane; Biais, Benoit; Maucourt, Mickael ; Ballias, Patricia; Deborde, Catherine; Colombié, Sophie; Cabasson, Cécile; Jacob, Daniel; Gibon, Yves; Moing, Annick, 2018, "FRIM - Fruit Integrative Modelling", https://doi.org/10.15454/95JUTK, Portail Data Inra, V1

Learn about Data Citation Standards.

Description	The project aimed to build a virtual tomato fruit that enables the prediction of metabolite levels given genetic and environmental inputs, by an iterative process between laboratories which combine expertise in fruit biology, ecophysiology, theoretical and experimental biochemistry, and biotechnology.
Subject	Computer science; Information management; Omics; Plant Health and Pathology
Related Publication	Bénard C., Bernillon S., Biais B., Osorio S., Maucourt M., Ballias P., Deborde C., Colombié S., Cabasson C., Jacob D., Vercambre G., Gautier H., Rolin D., Génard M., Fernie A., Gibon Y., Moing A. 2015 Metabolomic profiling in tomato reveals diel compositional changes in fruit affected by source-sink relationships. Journal of Experimental Botany Vol. 66, No. 11 pp. 3391–3404 doi: 10.1093/jxb/erv151
Link to data	https://pmb-bordeaux.fr/dataexplorer/?ds=frim1

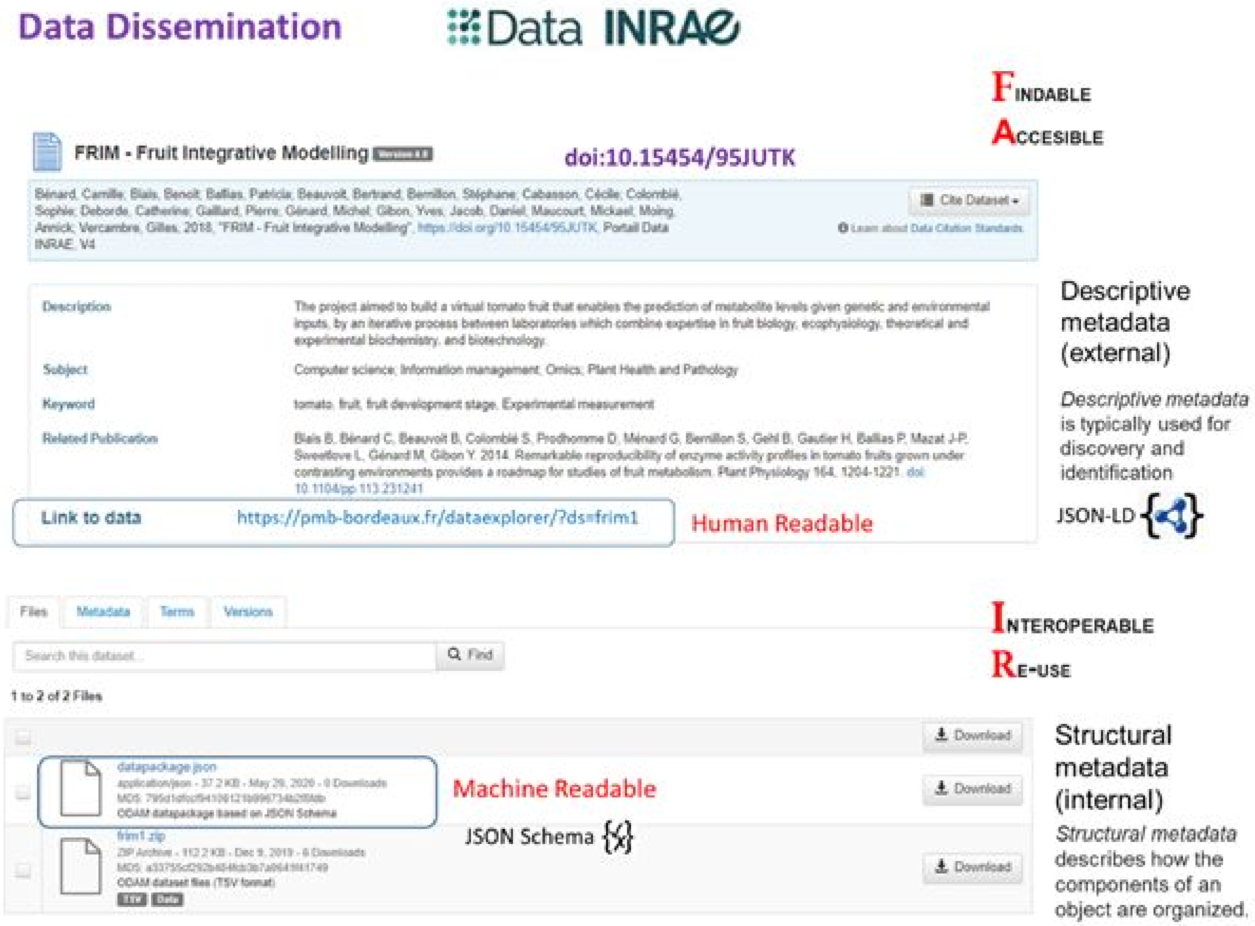
Figure 5.2 *: Example of a data publication using Dataverse for descriptive metadata and ODAM for both structural metadata and the data itself. Dataverse also provides the ability to associate other documents, codes and even other related data.*

While ODAM ensuring data sharing function allows direct access to subsets of data by API request, an institutional data repository ensuring data publishing function has to support an important part of the FAIR criteria. Thus, the combination of the two systems makes it possible to cover all the essential FAIR criteria. See *Appendix 5*.

5.2 Without ODAM repository

If you do not wish to set up your own ODAM repository, you can at least simply upload your data files with associated structural metadata to an institutional repository. Add a datapackage.json file within your collection of data files (see *Appendix 4*). Because, when disseminating data, defining an explicit schema for structural metadata along with unambiguous definitions of all internal elements (e.g. column definitions, units of measurement), through links to accessible (standard) definitions allows machines to better interpret the data for reuse. This will result in better annotated and more easily usable data that meets effortlessly the FAIR criteria for reusability. Indeed, concerning the FAIRification of data, this has a positive impact on the FAIR criteria 'Interoperable' and 'Reusable', encouraging structured data using a discoverable, community-endorsed schema or data model.

In addition, for easy reuse, simply indicate that the data has been formatted and structured to be compatible with the ODAM software, so that users of the data can take advantage of all associated services and tools by installing it on their desktop or laptop.



Data Dissemination

Data INRAE

FINDABLE

ACCESIBLE

FRIM - Fruit Integrative Modelling

doi:10.15454/95JUTK

Bénard, Camille; Blais, Benoit; Ballias, Patricia; Beauvoit, Bertrand; Bernillon, Stéphane; Cabasson, Cécile; Colombié, Sophie; Deborde, Catherine; Gaillard, Pierre; Génard, Michel; Gibon, Yves; Jacob, Daniel; Maucourt, Mickael; Moing, Annick; Vercambre, Gilles, 2018, "FRIM - Fruit Integrative Modelling", https://doi.org/10.15454/95JUTK, Portail Data INRAE, V4

Learn about Data Citation Standards.

Description	The project aimed to build a virtual tomato fruit that enables the prediction of metabolite levels given genetic and environmental inputs, by an iterative process between laboratories which combine expertise in fruit biology, ecophysiology, theoretical and experimental biochemistry, and biotechnology.
Subject	Computer science; Information management; Omics; Plant Health and Pathology
Keyword	tomato, fruit, fruit development stage, Experimental measurement
Related Publication	Blais B, Bénard C, Beauvoit B, Colombié S, Prodhomme D, Ménard G, Bernillon S, Gehl B, Gautier H, Ballias P, Mazat J-P, Sweetlove L, Génard M, Gibon Y. 2014. Remarkable reproducibility of enzyme activity profiles in tomato fruits grown under contrasting environments provides a roadmap for studies of fruit metabolism. Plant Physiology 164, 1204-1221. doi 10.1104/pp.113.231241
Link to data	https://pmb-bordeaux.fr/dataexplorer/?ds=frim1

Human Readable

Descriptive metadata (external)

Descriptive metadata is typically used for discovery and identification

JSON-LD

INTEROPERABLE

RE-USE

1 to 2 of 2 Files

datapackage.json
application/json - 37.2 KB - May 29, 2020 - 0 Downloads
MD5: 795d1dfccf94106121b996734b2f8db
ODAM datapackage based on JSON Schema

Machine Readable

JSON Schema

frim1.zip
ZIP Archive - 112.2 KB - Dec 9, 2019 - 6 Downloads
MD5: a33755c0292b4048cb3b7a0641f41749
ODAM dataset files (TSV format)

Structural metadata (internal)

Structural metadata describes how the components of an object are organized.

Figure 5.3 - *Data INRAE repository as a hub (based on Dataverse)*

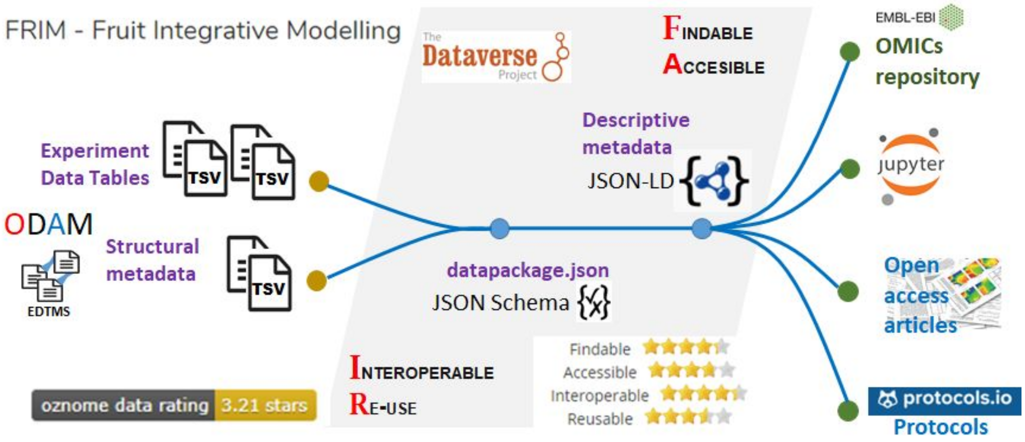


Figure 5.4 - *Interconnection of the different elements of the FRIM dataset from the Data INRAE repository as a hub (based on Dataverse). By relying on explicit schemas (JSON-LD, JSON Schema) for both metadata and data, it becomes possible to reuse the data without friction, both by humans and machines.*

References

1. Bénard, Camille; Biais, Benoit; Ballias, Patricia; Beauvoit, Bertrand; Bernillon, Stéphane; Cabasson, Cécile; Colombié, Sophie; Deborde, Catherine; Gaillard, Pierre; Génard, Michel; Gibon, Yves; Jacob, Daniel; Maucourt, Mickael; Moing, Annick; Vercambre, Gilles, (2018), "FRIM - Fruit Integrative Modelling", doi:10.15454/95JUTK, Portail Data INRAE, V3
2. Wilkinson, M., Dumontier, M., Aalbersberg, I. *et al.* (2016) The FAIR Guiding Principles for scientific data management and stewardship. *Sci Data* 3, 160018. doi:10.1038/sdata.2016.18
3. Annika Jacobsen, Ricardo de Miranda Azevedo, Nick Juty, Dominique Batista *et al.* (2020) Data Intelligence 2:1-2, 10-29. doi:/10.1162/dint_r_00024
4. Athanasios Karalopoulos *et al.* (2018), Turning FAIR into reality, Final Report and Action Plan from the European Commission Expert Group on FAIR Data, EU publications https://ec.europa.eu/info/sites/info/files/turning_fair_into_reality_1.pdf

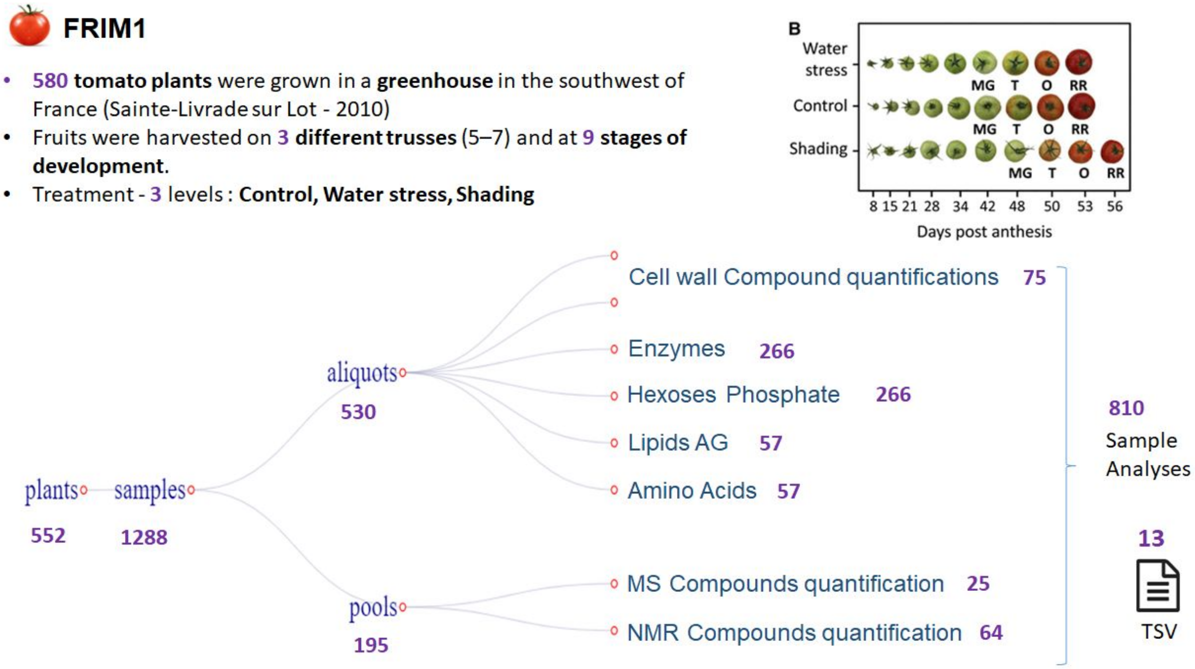
Appendix 1 : A dataset example

Fruit Integrative Modelling, an ERASysBio+ project : Yves Gibon (Coordinator) , Jean-Pierre Mazat , Michel Génard , Lee Sweetlove , David Fell, Alisdair Fernie, Johann Rohwer

The project aimed to build a virtual tomato fruit that enables the prediction of metabolite levels given genetic and environmental inputs, by an iterative process between laboratories which combine expertise in fruit biology, ecophysiology, theoretical and experimental biochemistry, and biotechnology.

- To build a kinetic model encompassing the routes carbon takes, once imported into the fruit cells from the source organs of the mother plant.
- To integrate the kinetic model with a phenomenological model predicting sugar and organic acid contents as functions of time, light intensity, temperature and water availability.
- To obtain large-scale experimental measures of the consequences of altered environmental conditions.

To assess the influence of the environment on fruit metabolism, tomato (Solanum lycopersicum 'Moneymaker') plants were grown under contrasting conditions (optimal for commercial, shaded production) and locations. Samples were harvested at nine stages of development, and 36 enzyme activities of central metabolism were measured as well as protein, starch, and major metabolites, such as hexoses, sucrose, organic acids, and amino acids.



FRIM1

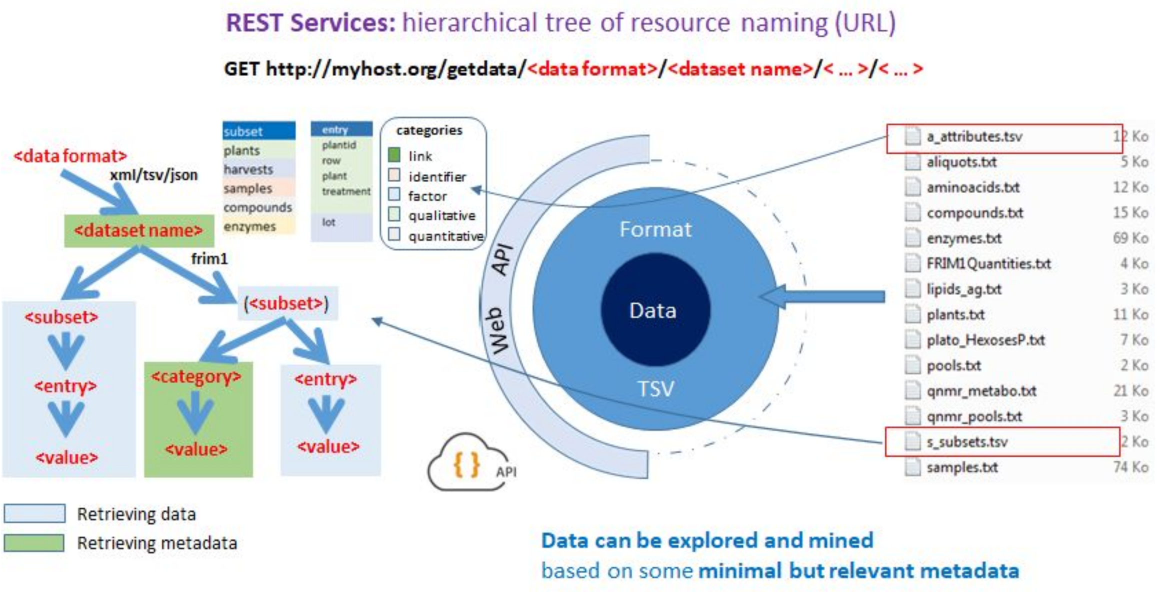
- 580 **tomato plants** were grown in a **greenhouse** in the southwest of France (Sainte-Livrade sur Lot - 2010)
- Fruits were harvested on 3 **different trusses** (5–7) and at 9 **stages of development**.
- Treatment - 3 levels : **Control, Water stress, Shading**

About 580 tomato plants were grown in a greenhouse in the southwest of France (Sainte-Livrade sur Lot) during the summer of 2010 according to usual production practices.

Biais B, Bénard C, Beauvoit B, Colombié S, Prodhomme D, Ménard G, Bernillon S, Gehl B, Gautier H, Ballias P, Mazat J-P, Sweetlove L, Génard M, Gibon Y. 2014. Remarkable reproducibility of enzyme activity profiles in tomato fruits grown under contrasting environments provides a roadmap for studies of fruit metabolism. Plant Physiology 164, 1204-1221. doi: 10.1104/pp.113.231241

Appendix 2 : Web Services

Based on REST services using a Resource Naming convention: an understandable resource naming leading to an easily leveraged Web service API (Identification/querying of resources) and easy to implement within R. Output formats: TSV, JSON and XML. Even if the WS outputs are not dedicated to human readers (the script languages as R are the typical clients), the XML outputs can be human readable in a web browser, made possible by using a XSL transformation mechanism which converts the XML outputs to HTML format.

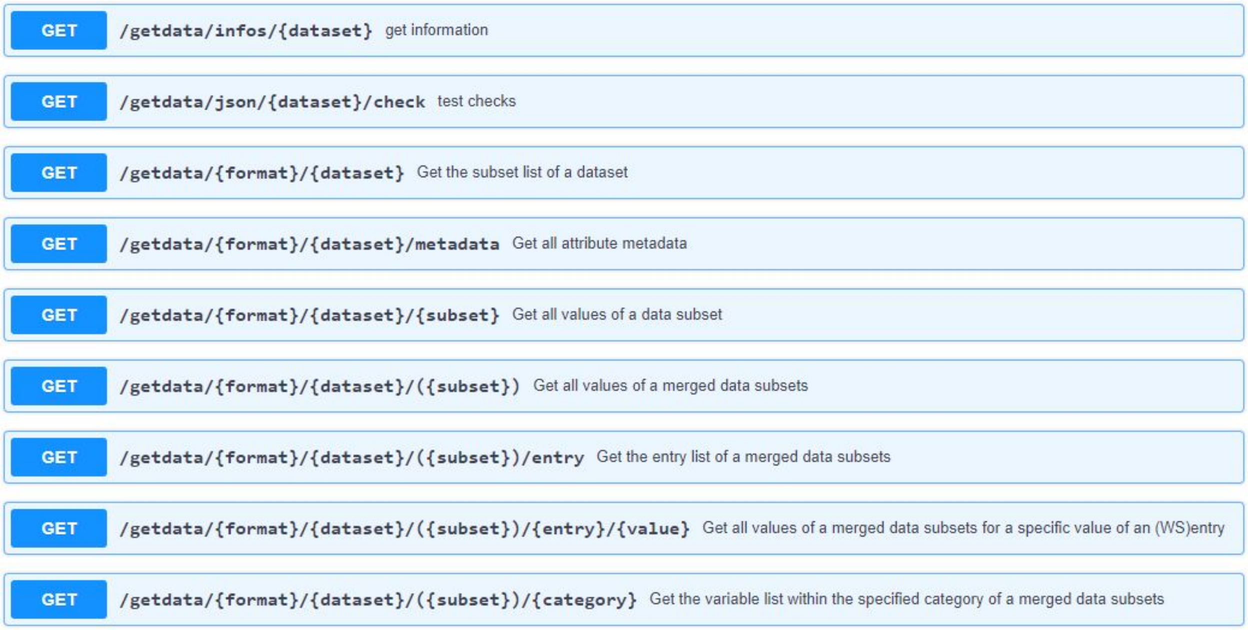


Using the two metadata files, it is possible to build a tree structure from which the data files can be queried to extract a subset. The tree structure is built on the Entity-Attribute-Value scheme.

Field	Description	Examples
<data format>	Format of the retrieved data; possible values are: 'xml', 'json' or 'tsv'	*tsv*
<dataset name>	Short name (tag) of your dataset	*frim1*
<subset>	Short name of a data subset	*samples*
<entry>	Name of an attribute entry (defined by the user in the a_attribute file (column 'entry')	*sampleid*

<category>	Name of the attribute category; (assigned by the user in the a_attribute file (column 'category') possible values are: 'identifier', 'factor', 'qualitative', 'quantitative'	*quantitative*
(**<subset>**)	Set of data subsets by merging all the subsets with lower rank than the specified subset and following the pathway defined by the "is_part_of" links.	*(samples) <=> plants + samples*
<value>	Exact value of the desired entry or category	*1 (subset)* *Factor (category)*

API Documentation on SwaggerHub : INRA-PMB/ODAM/1.0.1-oas3



GET `/getdata/infos/{dataset}` get information

GET `/getdata/json/{dataset}/check` test checks

GET `/getdata/{format}/{dataset}` Get the subset list of a dataset

GET `/getdata/{format}/{dataset}/metadata` Get all attribute metadata

GET `/getdata/{format}/{dataset}/{subset}` Get all values of a data subset

GET `/getdata/{format}/{dataset}/({subset})` Get all values of a merged data subsets

GET `/getdata/{format}/{dataset}/({subset})/entry` Get the entry list of a merged data subsets

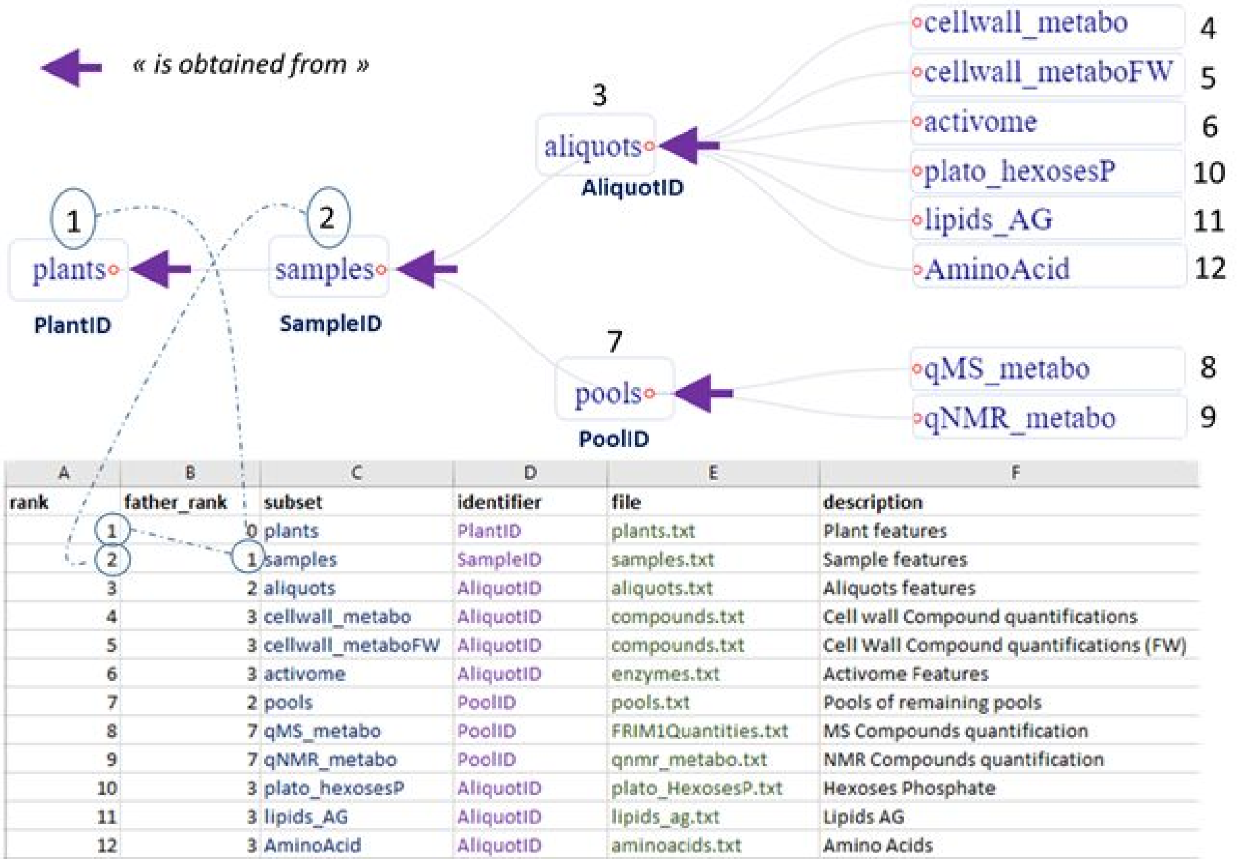
GET `/getdata/{format}/{dataset}/({subset})/{entry}/{value}` Get all values of a merged data subsets for a specific value of an (WS)entry

GET `/getdata/{format}/{dataset}/({subset})/{category}` Get the variable list within the specified category of a merged data subsets

Appendix 3 : Example of a very detailed API request

Since all the experimental data tables were generated as part of an experiment associated with a Design of Experiment (DoE), each file thus contains data acquired sequentially as the experiment progressed. There must therefore be a link between each file, i.e. information that connects them together. In most cases (if not all), this information corresponds to identifiers that make it possible to precisely reference within the experiment each of the elements belonging to the same observation entity forming a coherent observation unit. For example, each plant, each sample has its own identifier, and each of these entities corresponds to a separate data file.

The files generated during data collection have to be organized according to the entity-relationship model similar to relational database management systems (RDBMS), as shown below with the FRIM1 dataset.



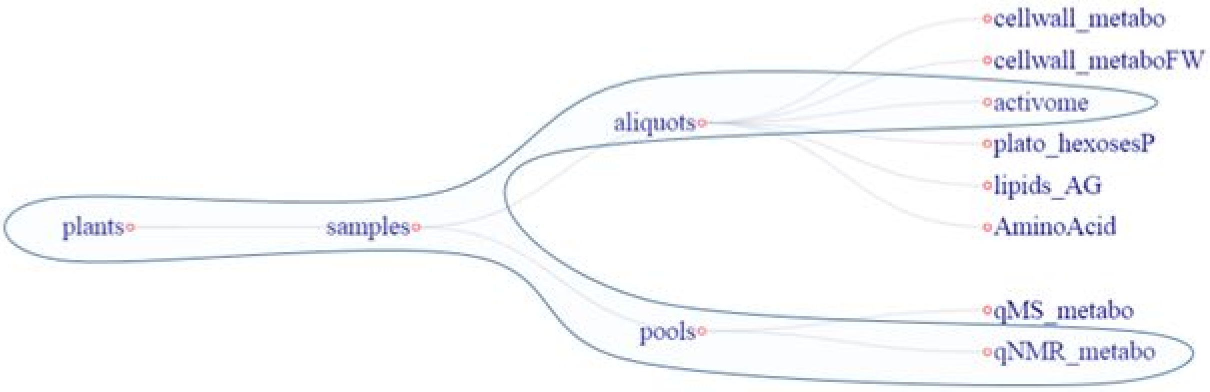
A	B	C	D	E	F
rank	father_rank	subset	identifier	file	description
1	0	plants	PlantID	plants.txt	Plant features
2	1	samples	SampleID	samples.txt	Sample features
3	2	aliquots	AliquotID	aliquots.txt	Aliquots features
4	3	cellwall_metabo	AliquotID	compounds.txt	Cell wall Compound quantifications
5	3	cellwall_metaboFW	AliquotID	compounds.txt	Cell Wall Compound quantifications (FW)
6	3	activome	AliquotID	enzymes.txt	Activome Features
7	2	pools	PoolID	pools.txt	Pools of remaining pools
8	7	qMS_metabo	PoolID	FRIM1Quantities.txt	MS Compounds quantification
9	7	qNMR_metabo	PoolID	qnmr_metabo.txt	NMR Compounds quantification
10	3	plato_hexosesP	AliquotID	plato_HexosesP.txt	Hexoses Phosphate
11	3	lipids_AG	AliquotID	lipids_ag.txt	Lipids AG
12	3	AminoAcid	AliquotID	aminoacids.txt	Amino Acids

Structural metadata (file s_subsets.tsv) for FRIM1 dataset.

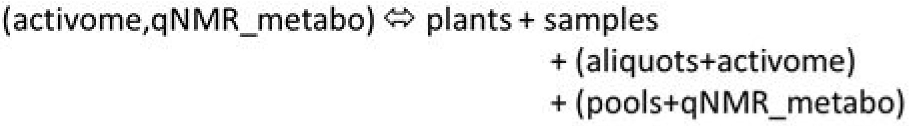
See "Data Preparation Protocol for ODAM Compliance" for more explanation.

Each file (entity) corresponds to a type of collected data (samples, compounds, ...) for which is associated a set of attributes, i.e. a set of variables that may include observed or measured variables (quantitative or qualitative), controlled independent variables (factors) and obviously an identifier.

Since there is a relationship between each of the files, it is therefore possible to combine them by following the paths of their mutual links. Suppose we want to retrieve activome data (enzyme expressions) combined with NMR observed metabolite data, including experimental metadata (factors and phenotypic data), as shown below:



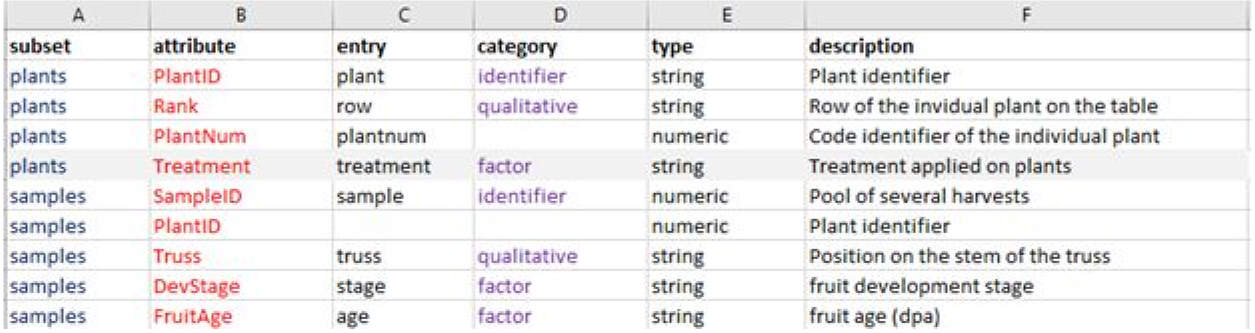
The ODAM API offers this possibility of combining data by following relational paths, using the '**(***{subset}***)**' operator in the API request. Simply specify the desired subsets separated by a comma, as shown below:



```
(activome,qNMR_metabo) ⇔ plants + samples
                                  + (aliquots+activome)
                                  + (pools+qNMR_metabo)
```

The '**(***{subset}***)**' operator allows you to get data subsets by merging all the subsets with lower rank than the specified subsets and following the pathway defined by the “obtainedFrom" links.

Let's further assume that we only want the data for the '*Control*' treatment. The ODAM API also offers the possibility of setting a selector called an '*entry*' provided that such an entry has been specified for the attribute to which one wants to apply a selection, as shown below:



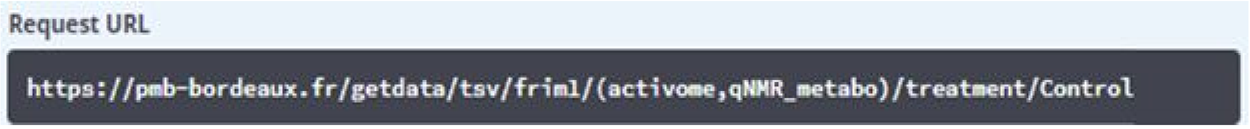
A	B	C	D	E	F
subset	**attribute**	**entry**	**category**	**type**	**description**
plants	PlantID	plant	identifier	string	Plant identifier
plants	Rank	row	qualitative	string	Row of the invidual plant on the table
plants	PlantNum	plantnum		numeric	Code identifier of the individual plant
plants	Treatment	treatment	factor	string	Treatment applied on plants
samples	SampleID	sample	identifier	numeric	Pool of several harvests
samples	PlantID			numeric	Plant identifier
samples	Truss	truss	qualitative	string	Position on the stem of the truss
samples	DevStage	stage	factor	string	fruit development stage
samples	FruitAge	age	factor	string	fruit age (dpa)

Extract of structural metadata (file a_attributes.tsv).

See "Data Preparation Protocol for ODAM Compliance" for more explanation.

In our case, the 'Treatment' factor has the entry (alias) 'treatment'. It is therefore possible to use it as a selector in the API request.

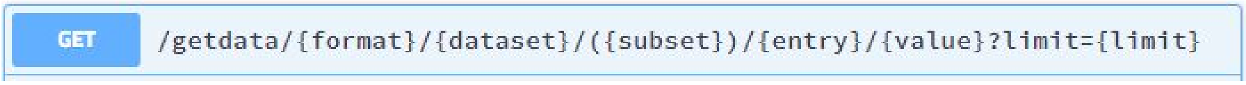
Thus, our complete API request would be this one:



Request URL

```
https://pmb-bordeaux.fr/getdata/tsv/frim1/(activome,qNMR_metabo)/treatment/Control
```

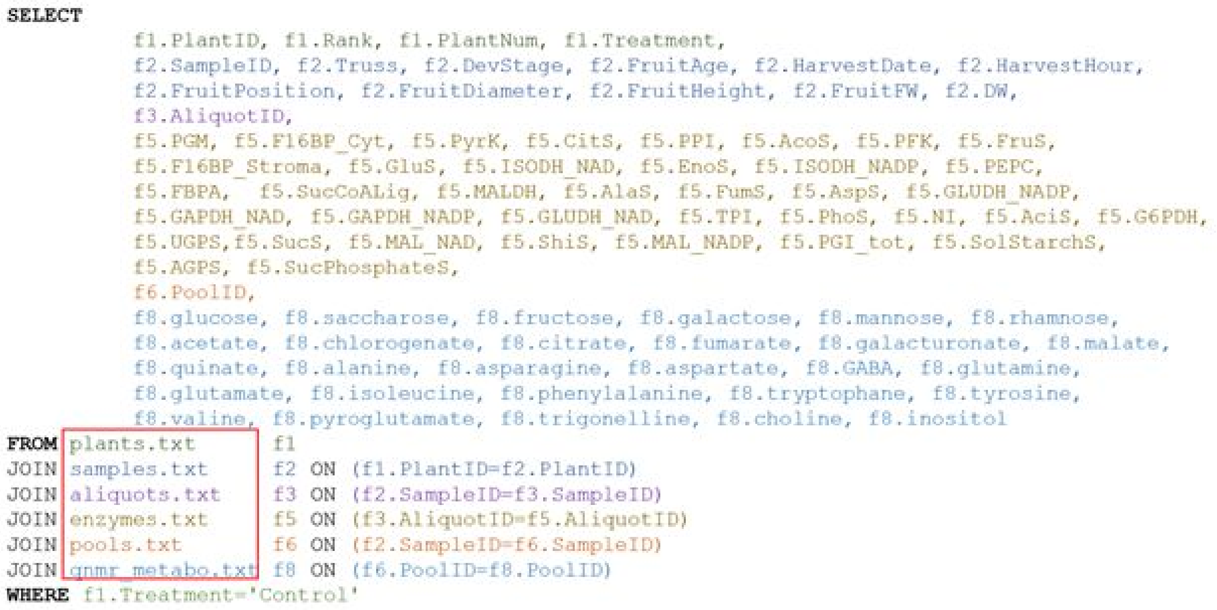
To test this request by yourself, you can go to ODAM's Swagger interface (https://pmb-bordeaux.fr/odamsw/) and play with the followed specific query :



```
/getdata/{format}/{dataset}/({subset})/{entry}/{value}?limit={limit}
```

Finally, how does it work internally? How are the different subsets of data combined in practice?

Since each data subset being a data file, and linked to one or more other data subsets by means of common identifiers, it is possible to apply a SQL query from the data files thanks to the powerful tool 'q - Text as Data' (http://harelba.github.io/q/) which allows to apply SQL queries directly to CSV and TSV files as shown below:



```
SELECT
         f1.PlantID, f1.Rank, f1.PlantNum, f1.Treatment,
         f2.SampleID, f2.Truss, f2.DevStage, f2.FruitAge, f2.HarvestDate, f2.HarvestHour,
         f2.FruitPosition, f2.FruitDiameter, f2.FruitHeight, f2.FruitFW, f2.DW,
         f3.AliquotID,
         f5.PGM, f5.F16BP_Cyt, f5.PyrK, f5.CitS, f5.PPI, f5.AcoS, f5.PFK, f5.FruS,
         f5.F16BP_Stroma, f5.GluS, f5.ISODH_NAD, f5.EnoS, f5.ISODH_NADP, f5.PEPC,
         f5.FBPA,  f5.SucCoALig, f5.MALDH, f5.AlaS, f5.FumS, f5.AspS, f5.GLUDH_NADP,
         f5.GAPDH_NAD, f5.GAPDH_NADP, f5.GLUDH_NAD, f5.TPI, f5.PhoS, f5.NI, f5.AciS, f5.G6PDH,
         f5.UGPS,f5.SucS, f5.MAL_NAD, f5.ShiS, f5.MAL_NADP, f5.PGI_tot, f5.SolStarchS,
         f5.AGPS, f5.SucPhosphateS,
         f6.PoolID,
         f8.glucose, f8.saccharose, f8.fructose, f8.galactose, f8.mannose, f8.rhamnose,
         f8.acetate, f8.chlorogenate, f8.citrate, f8.fumarate, f8.galacturonate, f8.malate,
         f8.quinate, f8.alanine, f8.asparagine, f8.aspartate, f8.GABA, f8.glutamine,
         f8.glutamate, f8.isoleucine, f8.phenylalanine, f8.tryptophane, f8.tyrosine,
         f8.valine, f8.pyroglutamate, f8.trigonelline, f8.choline, f8.inositol
FROM plants.txt        f1
JOIN samples.txt       f2 ON (f1.PlantID=f2.PlantID)
JOIN aliquots.txt      f3 ON (f2.SampleID=f3.SampleID)
JOIN enzymes.txt       f5 ON (f3.AliquotID=f5.AliquotID)
JOIN pools.txt         f6 ON (f2.SampleID=f6.SampleID)
JOIN qnmr_metabo.txt   f8 ON (f6.PoolID=f8.PoolID)
WHERE f1.Treatment='Control'
```

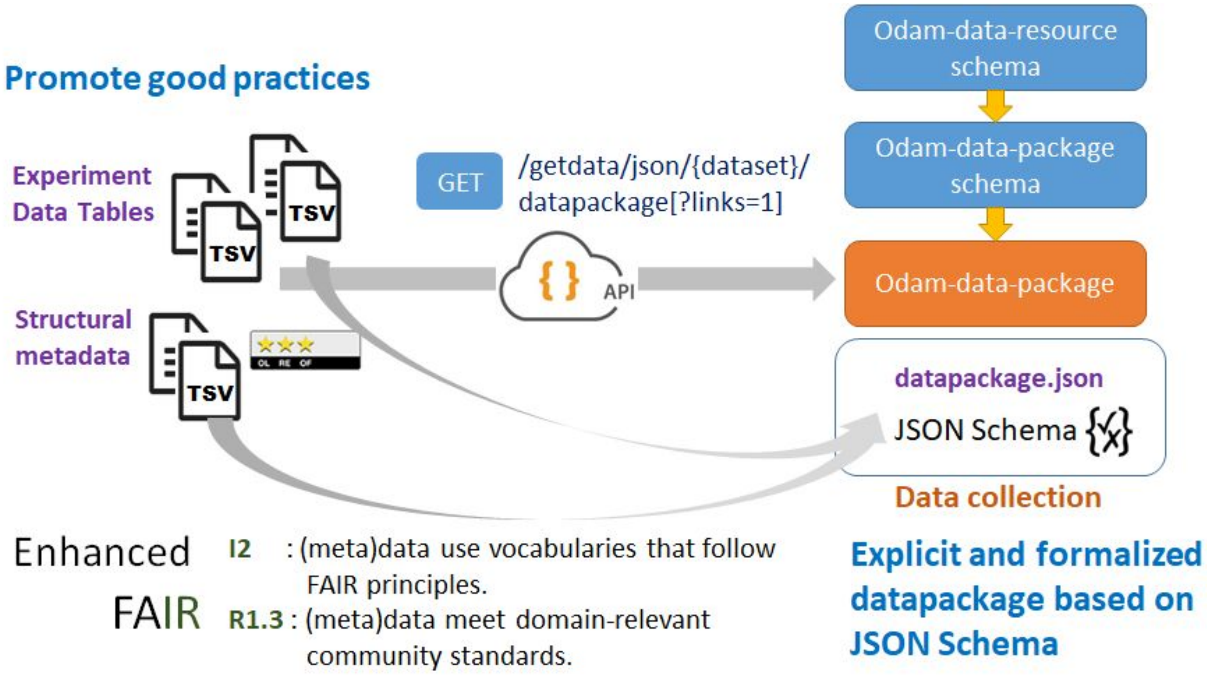
In addition, if you want to display information about the API request, just add '**/debug**' at the end of the request, as shown below:

http://pmb-bordeaux.fr/getdata/tsv/frim1/(activome,qNMR_metabo)/treatment/Control/debug

Appendix 4: ODAM datapackage based on JSON Schema

A datapackage is a simple container format based on JSON Schema specifications (a) used to describe a collection of data within a file named ‘datapackage.json’ by convention (b) which can be added to the collection of your data files. Once data disseminated, this ‘datapackage.json’ file therefore contains all structural metadata along with unambiguous definitions of all internal elements (e.g. column definitions, units of measurement), through links to accessible (standard) definitions, allowing machines to better interpret the data for reuse.

The ODAM datapackage schema is an explicit schema for structural metadata, very close to the 'Frictionless Data' specifications (c) on which it is based. It was developed within the ODAM project in order to fulfill the FAIR principles related to the (Re)usable criteria.


Promote good practices
Experiment Data Tables
TSV
TSV
Structural metadata
TSV
GET
/getdata/json/{dataset}/datapackage[?links=1]
{ }
API
Odam-data-resource schema
Odam-data-package schema
Odam-data-package
datapackage.json
JSON Schema
Data collection
Enhanced FAIR
I2 : (meta)data use vocabularies that follow FAIR principles.
R1.3 : (meta)data meet domain-relevant community standards.
Explicit and formalized datapackage based on JSON Schema

The ‘**datapackage.json**’ file can be generated directly from the API by specifying '**/datapackage**' at the end of the request. By default, the reference to the data files is relative. To have an URL as reference for the data files, it is necessary to add at the end of the request '**?links=1**'

> JSON Schema for ODAM data package
> https://github.com/djacob65/odam-datapackage

Concerning the FAIRification of data, this has a positive impact on the FAIR criteria 'Interoperable' and 'Resusable', encouraging structured data using a discoverable, community-endorsed schema. See OZNOME Data ratings (d), 'Usable' section.

Tip: To view JSON files in a more user-friendly way into your web browser, you should install a JSON Formatter extension/plugin. (e.g 'JSON Formatter' extension for Google Chrome from 'more tools/extensions' menu, 'JSON Formatter' plugin for MS Edge from the Windows store)

(a) https://json-schema.org/
(b) https://datahub.io/docs/data-packages
(c) https://specs.frictionlessdata.io/
(d) https://confluence.csiro.au/display/OZNOME/Data+ratings

Appendix 5 : FAIR data principles

Coverage example of the different FAIR criteria

- Applied to the dataset from an experiment on tomato fruits grown in greenhouses [1] for which data management and publication relies on Data INRAE and ODAM
- Using several evaluation grids

The Data INRAE institutional data repository uses Dataverse software whose coverage of FAIR criteria is discussed in Wilkinson et al. 2016 [2]

FAIR Evaluation Grids

We used three FAIR grids, very different from each other.

1. The first one, 5 Star Data Rating Tool from OZONOME aims to carry out an evaluation based on the FAIR principles as defined by Willkinson et al (2016). The main output is a global rating, indicating the global FAIRness of the dataset.
2. The second, the FDMM (Fair Data Maturity Model) (A) document describes a maturity model for FAIR assessment with assessment indicators, priorities and evaluation methods, useful for the normalisation of assessment approaches to enable comparison of their results.
3. The third, the FAIR SHARC (SHAring Rewards and Credit) (B) document allows assessing FAIRness of projects and related human processes by either external evaluators or the researchers themselves, implying to implement simple FAIRness assessment in various communities and identify procedures and training that must be deployed and adapted to their practices and level of understanding.

SHARC (Sharing Rewards and Credit) is a recognized and endorsed interest group within RDA (Research Data Alliance).

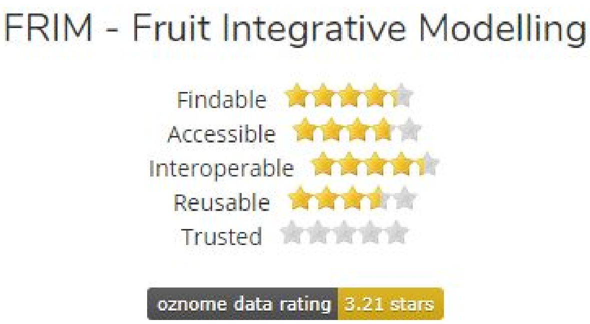
- https://drive.google.com/file/d/1uif-jy9QBno_WPnpGL14LFpDzL366tMH/view?usp=sharing

FDMM (FAIR Data Maturity Model) is a recognized and endorsed working group within RDA (Research Data Alliance). This FAIR Evaluation Grid describes a maturity model for FAIR assessment with assessment indicators, priorities and evaluation methods, useful for the normalisation of assessment approaches to enable comparison of their results.

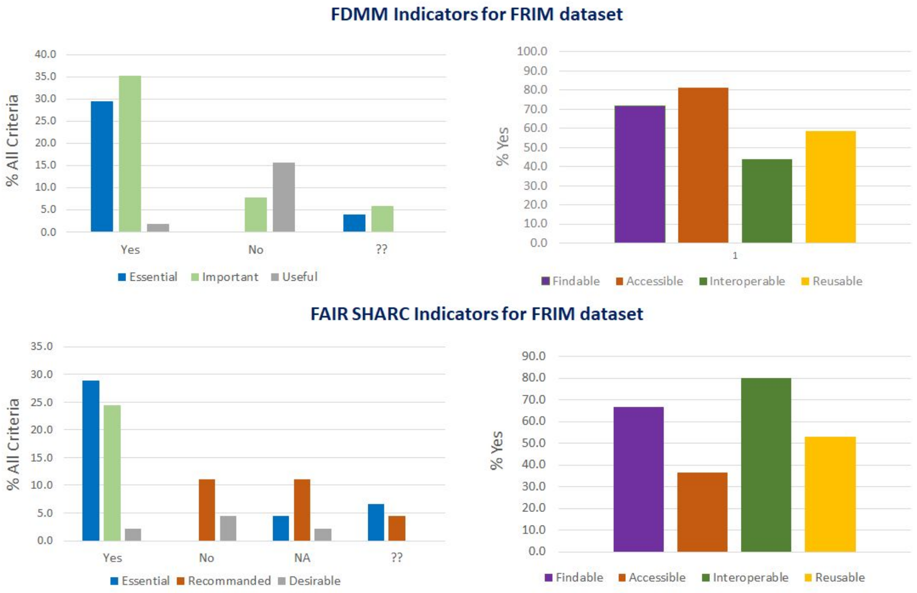
- https://drive.google.com/file/d/1a520Cbu8bryEeZIPI3h1l6zkaO7MZ39-/view?usp=sharing

5 ★ Data Rating Tool This tool provides implementations of the FORCE 11 FAIR data principles

- https://oznome.csiro.au/5star/?view=5ec2a9654d0983adde57a21e

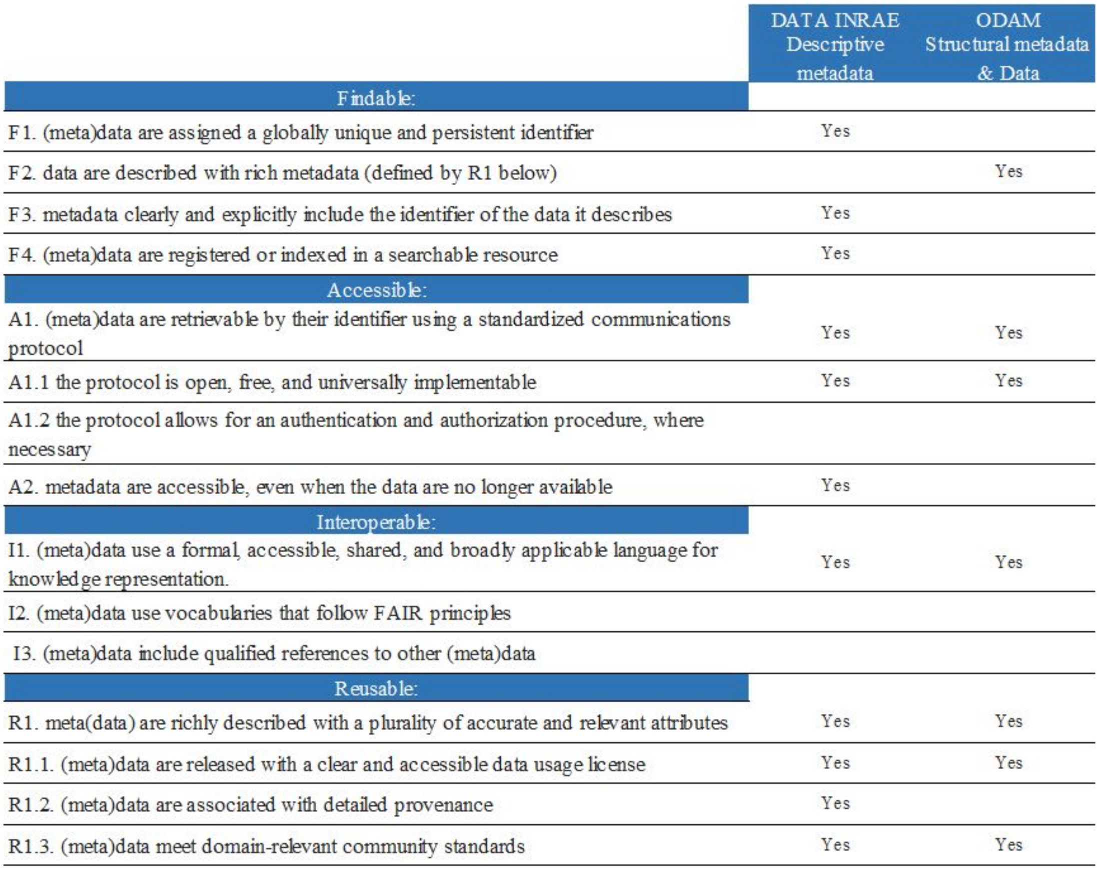


- See Data ratings
  - https://confluence.csiro.au/display/OZNOME/Data+ratings



Synthesis of FAIR evaluation grids applied to the Frim dataset

The Fair Data Maturity Model (FDMM) document (A) describes a maturity model for the FAIR assessment with indicators, priorities and assessment methods, which are useful for standardizing assessment approaches in order to allow comparison of their results. Whereas the FAIR SHARC (SHAring Rewards and Credit) (B) document allows the fairness of projects and associated human processes to be assessed, either by external evaluators or by the researchers themselves. Therefore, these grids cannot be compared with each other, but rather complement each other.



	DATA INRAE Descriptive metadata	ODAM Structural metadata & Data
Findable:		
F1. (meta)data are assigned a globally unique and persistent identifier	Yes	
F2. data are described with rich metadata (defined by R1 below)		Yes
F3. metadata clearly and explicitly include the identifier of the data it describes	Yes	
F4. (meta)data are registered or indexed in a searchable resource	Yes	
Accessible:		
A1. (meta)data are retrievable by their identifier using a standardized communications protocol	Yes	Yes
A1.1 the protocol is open, free, and universally implementable	Yes	Yes
A1.2 the protocol allows for an authentication and authorization procedure, where necessary		
A2. metadata are accessible, even when the data are no longer available	Yes	
Interoperable:		
I1. (meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.	Yes	Yes
I2. (meta)data use vocabularies that follow FAIR principles		
I3. (meta)data include qualified references to other (meta)data		
Reusable:		
R1. meta(data) are richly described with a plurality of accurate and relevant attributes	Yes	Yes
R1.1. (meta)data are released with a clear and accessible data usage license	Yes	Yes
R1.2. (meta)data are associated with detailed provenance	Yes	
R1.3. (meta)data meet domain-relevant community standards	Yes	Yes

Summary table of essential FAIR criteria based on force11.org, applied to the Frim dataset

References

A. RDA FAIR Data Maturity Model Working Group (2020). FAIR Data Maturity Model: specification and guidelines. Research Data Alliance. DOI: 10.15497/RDA00045

B. Romain David, Laurence Mabile, Alison Specht, Stryeck, Sarah, Mogens Thomsen, et al. FAIRness Literacy: the Achilles' Heel of applying FAIR Principles. 2020. https://hal.archives-ouvertes.fr/hal-02483307

Appendix 6 : All online resources related to ODAM software

Set of tools and protocols implemented in this work

Description	Type	Link
Data Preparation Protocol for ODAM Compliance	Documentation	doi:10.17504/protocols.io.betcjeiw
API Documentation based on Swagger	Web API Tool	https://app.swaggerhub.com/apis-docs/INRA-PMB/ODAM/1.0.1-oas3/
R ODAM package and How to use it	Package	https://cran.r-project.org/package=Rodam
Virtual Machine embedding the ODAM software on Oracle VM VirtualBox along with its installation guide	Virtual Machine + Documentation	https://doi.org/10.15454/C9LAEF
A very lightweight local web server for Windows to deploy the ODAM API	Web API Tool	http://pmb-bordeaux.fr/odam/ODAMwebserver/
Docker containers on DockerHub for installing on a Linux machine	Container	https://hub.docker.com/r/odam/getdata/ https://hub.docker.com/r/odam/dataexplorer/
ODAM Source code on GitHub	Source Code	https://github.com/inrae/ODAM
JSON Schema for ODAM data package	JSON Schema	https://github.com/djacob65/odam-datapackage
Examples of Jupyter notebooks (R & Python) based on the ODAM Web API	Notebooks	https://github.com/djacob65/binder_odam https://doi.org/10.24433/CO.8981049.v1 https://doi.org/10.24433/CO.0011270.v1
Modeling the growth of tomato fruits based on enzyme activity profiles (Example of data analysis interfaced by ODAM)	Notebook	https://hal.inrae.fr/hal-02611223