D8.4 - Report on the interface of AIDA and external databases

Deliverable No.:	D8.4
Deliverable Name:	Report on the interface of AIDA and external databases
Contractual Submission Date:	31/12/2022
Actual Submission Date:	28/02/2022
Version:	V1.0

This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under grant agreement No 776262.

COVER AND CONTROL PAGE OF DOCUMENT	
Project Acronym:	**AIDA**
Project Full Name:	Artificial Intelligence Data Analysis
Deliverable No.:	D8.4
Document name:	Report on the interface of AIDA and external databases
Nature (R, P, D, O):	R
Dissemination Level (PU, PP, RE, CO):	PU
Version:	V1.0
Actual Submission Date:	28/02/2021
Author, Institution: E-Mail:	Alessandro Retino (CNRS) alessandro.retino@lpp.polytechnique.fr Jorge Amaya (KU Leuven) jorge.amaya@kuleuven.be Sofoklis Katakis (IRIDA) katakis@iridalabs.gr Leonidas Liakopoulos (IRIDA) liakopoulos@iridalabs.gr
Other contributors	Giorgio Pedrazzi (CINECA) g.pedrazzi@cineca.it

ABSTRACT: In this document we report on the definition and implementation of the interface between AIDApy and AIDAdb, which is based on the iRODS open-source tool. We provide examples on how to use iRODS to retrieve AIDAdb products as well as their metadata from Cineca's disks. We discuss possible improvements which may be done in the future outside of the boundaries of the AIDA project.

KEYWORD LIST:
Heliosphere, Magnetosphere, Spacecraft Data, Machine Learning, Artificial Intelligence, Database, Python

The author is solely responsible for its content, it does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of data appearing therein.

TABLE OF CONTENTS

1 Executive Summary

This document includes those activities of the Task 3 of the AIDA's WP8 ("*Interface with AIDA database and external databases*"), which are related to the definition and implementation of the interface between of AIDApy with the AIDAdb and to the interface with external databases.

It should be noted that the initial Task 3 was to develop the interface between the "*Mission tool*" and the "*Event search tool*" and the AIDAdb only. This task has evolved during the AIDA project towards the definition of a general interface between AIDApy, which includes the two tools above, and the AIDAdb. This evolution has led to a close collaboration between CNRS and KULeuven, IRIDA and CINECA, which have made major contributions to this task. The interface between AIDApy and AIDAdb is also described in the deliverable D2.4 .

It should also be noted that the initial Task 3 included an added-value activity to provide an interface between the AIDAdb and external databases, such as e.g. ESA Heliophysics Science Archives. Due to the fact that the implementation of the interface between AIDApy and AIDAdb took longer than expected, this additional activity has not been carried out due to lack of time.

Section 2 provides a general explanation on the interface with the AIDAdb, namely on the use of the iRODS open-source software to connect the AIDA user to the AIDAdb. Section 3 provides examples of how to connect to the AIDAdb to retrieve AIDAdb products as well as their metadata from the Cineca's disks. This includes directly executing a list of iRODS commands (option 1) as well as using a python API integrated into AIDApy (option 2). Section 4 provides information on the current status of the documentation related to the interface between AIDApy and AIDAdb. Finally Section 5 summarises the results and identifies possible future work that may be done outside the boundaries of the AIDA project, e.g. in a follow-up project.

2 Introduction

The AIDAdb has been designed to include all products issued from the different AIDA tools: numerical simulation and spacecraft low-level data as well as processed, higher level data such as data from virtual spacecraft launched through simulations' boxes and list of spacecraft data events and summary plots. See here for the current list of products. All these products are stored on Cineca high capacity disks and can be accessed with the help of the iRODS tool.

The open-source Integrated Rule-Oriented Data System iRODS is an open-source data management tool which allows users to openly access data across any type of storage systems located anywhere, providing much flexibility. This system has been installed in multiple High Performance Computing centres across Europe, and is compliant with the FAIR principles of scientific data management.

Figure 1 shows how the AIDA user can connect to the AIDAdb, including the interface between AIDApy and AIDAdb (API). Two options are possible:

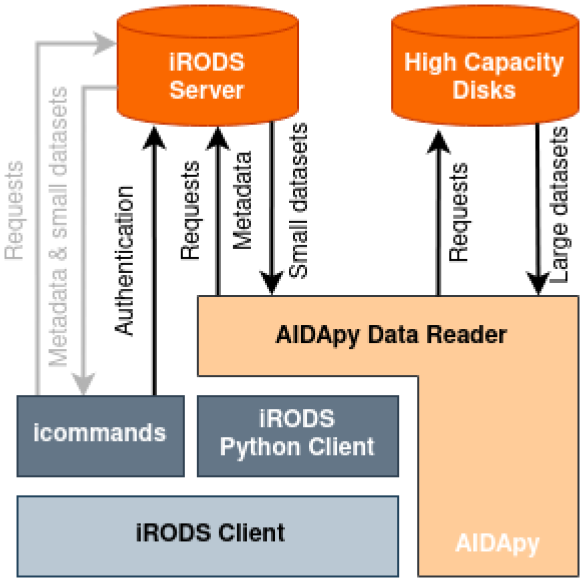


Figure 1. Schematic diagram showing the how to connect AIDA user to the AIDAdb

1. the user connects to AIDAdb by directly executing iRODS commands from his/her own computer using an iRODS client,
2. the user connects to AIDAdb through the AIDApy Data Reader API which is part of the AIDApy package

In all of the above cases, the first step for the user is to install the open-source iRODS client on his/her own computer and to login at Cineca's Marconi disk. The login (username and password) can be required at https://userdb.hpc.cineca.it . Technical support for this login can be obtained by contacting the Cineca's AIDAdb support .

Due to the large size of some of the AIDAdb products, e.g. full Particle-In-Cell numerical simulation raw data, only relatively small datasets (e.g. high-level products such as virtual spacecraft datasets and spacecraft data lists) are stored in the iRODS server at Cineca. This server also stores metadata information of all AIDAdb products, which include the locations of the full files.

Independently from which of the above options is chosen, the AIDA user can request such metadata and download them on his/her own computer. If only small datasets are needed, they can be downloaded directly for the iRODS server. If full files are needed, they can be downloaded via SCP to Marconi disk using the information on their location included in the metadata.

Section 3 below provides examples on how to connect to AIDAdb for each of the above options.

3 Examples of connecting to AIDAdb

The first step in the use of the AIDAdb is the authentication of the user in the iRODS and High Capacity Disk services from Cineca. After the coordinated hacker attacks of May 2020 to the European supercomputer centres, security had to be increased and we are now unable to propose completely free and anonymous access to our data storage servers.

Once the user has been granted access with the steps presented in the previous chapter, they need to authenticate their access from their own computer. The user then needs to download the *chem.pem* certificate file from Cineca, which can be found in point 2 of the iRODS documentation of Cineca:

https://wiki.u-gov.it/confluence/display/SCAIUS/iRODS-based+REPO#iRODSbasedREPO-iRODScommands

In the same link the user can see the contents of the second file that they need to download in their local computer. The file *irods_environment.json* contains the identification information of the iRODS server in Cineca and the information about the user. The user needs to modify the following entries in the .json file with their own information: irods_user_name, irods_ssl_certificate_chain_file, irods_ssl_ca_certificate_file. None of the other entries need to be changed.

We reproduce in the table below an exact *irods_environment.json* configuration file:

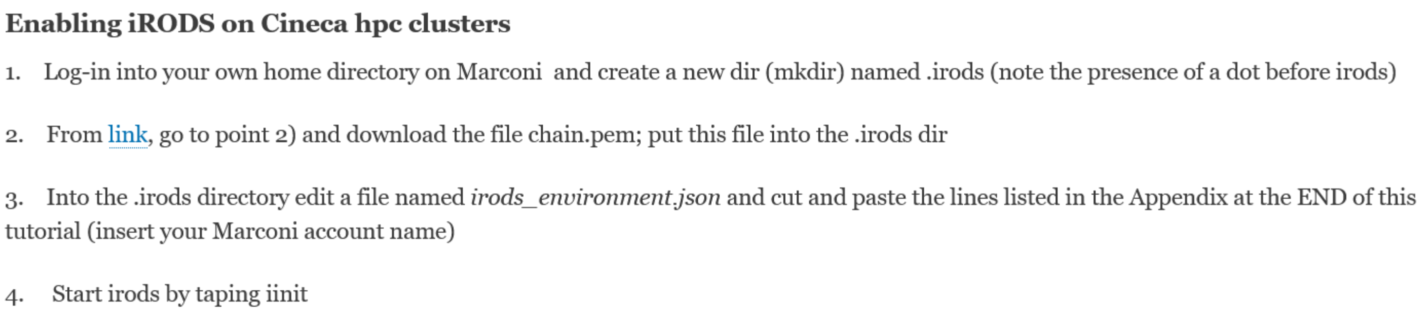
```
{
 "irods_host": "data.repo.cineca.it",
 "irods_port": 1247,
 "irods_default_resource": "cinecaRes1",
 "irods_home": "/CINECA01/home/DRES_AIDA_REP",
 "irods_cwd": "/CINECA01/home/DRES_AIDA_REP",
 "irods_user_name": "<USERNAME-TO-CHANGE>",
 "irods_zone_name": "CINECA01",
 "irods_client_server_negotiation": "request_server_negotiation",
 "irods_client_server_policy": "CS_NEG_REFUSE",
 "irods_encryption_key_size": 32,
 "irods_encryption_salt_size": 8,
 "irods_encryption_num_hash_rounds": 16,
 "irods_encryption_algorithm": "AES-256-CBC",
 "irods_default_hash_scheme": "MD5",
 "irods_match_hash_policy": "compatible",
 "irods_server_control_plane_port": 1248,
 "irods_server_control_plane_key": "TEMPORARY__32byte_ctrl_plane_key",
 "irods_server_control_plane_encryption_num_hash_rounds": 16,
 "irods_server_control_plane_encryption_algorithm": "AES-256-CBC",
 "irods_maximum_size_for_single_buffer_in_megabytes": 32,
 "irods_default_number_of_transfer_threads": 4,
 "irods_transfer_buffer_size_for_parallel_transfer_in_megabytes": 4,
 "irods_authentication_scheme": "PAM",
 "irods_ssl_certificate_chain_file": "<PATH-TO-CHANGE>/chain.pem",
 "irods_ssl_ca_certificate_file": "<PATH-TO-CHANGE>/chain.pem",
 "irods_ssl_verify_server": "cert"
}
```

With these two files in place, using the command ***iinit*** in the local computer will verify the user identity and allow their access to the AIDAdb. The user can now use the two options described in the previous section.

Option1

The first option for the AIDA user to connect to the AIDA iRODS server on Cineca's Marconi disk is to directly use "*iCommands*" from the iRODS clients installed on his/her computer (see Figure 1). iCommands are Unix utilities allowing the iRODS user to make use of a command-line interface to operate on data in the iRODS system. All the required steps are explained at this link.

As described in the previous section, the user first sets the iRODS environment information, then initialises access to the iRODS server through the ***iinit*** icommand from the iRODS client.



Enabling iRODS on Cineca hpc clusters

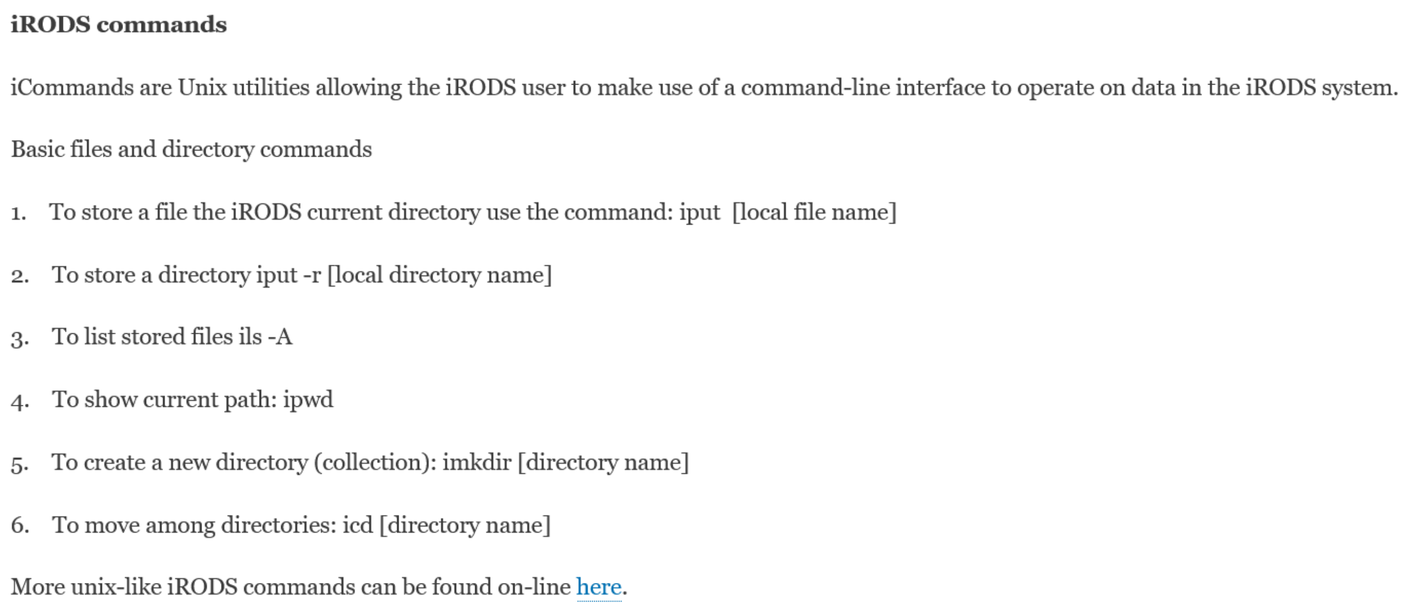
1. Log-in into your own home directory on Marconi and create a new dir (mkdir) named .irods (note the presence of a dot before irods)

2. From link, go to point 2) and download the file chain.pem; put this file into the .irods dir

3. Into the .irods directory edit a file named *irods_environment.json* and cut and paste the lines listed in the Appendix at the END of this tutorial (insert your Marconi account name)

4. Start irods by taping iinit

The link above also includes the explanation of the most important iCommands that are needed.



iRODS commands

iCommands are Unix utilities allowing the iRODS user to make use of a command-line interface to operate on data in the iRODS system.

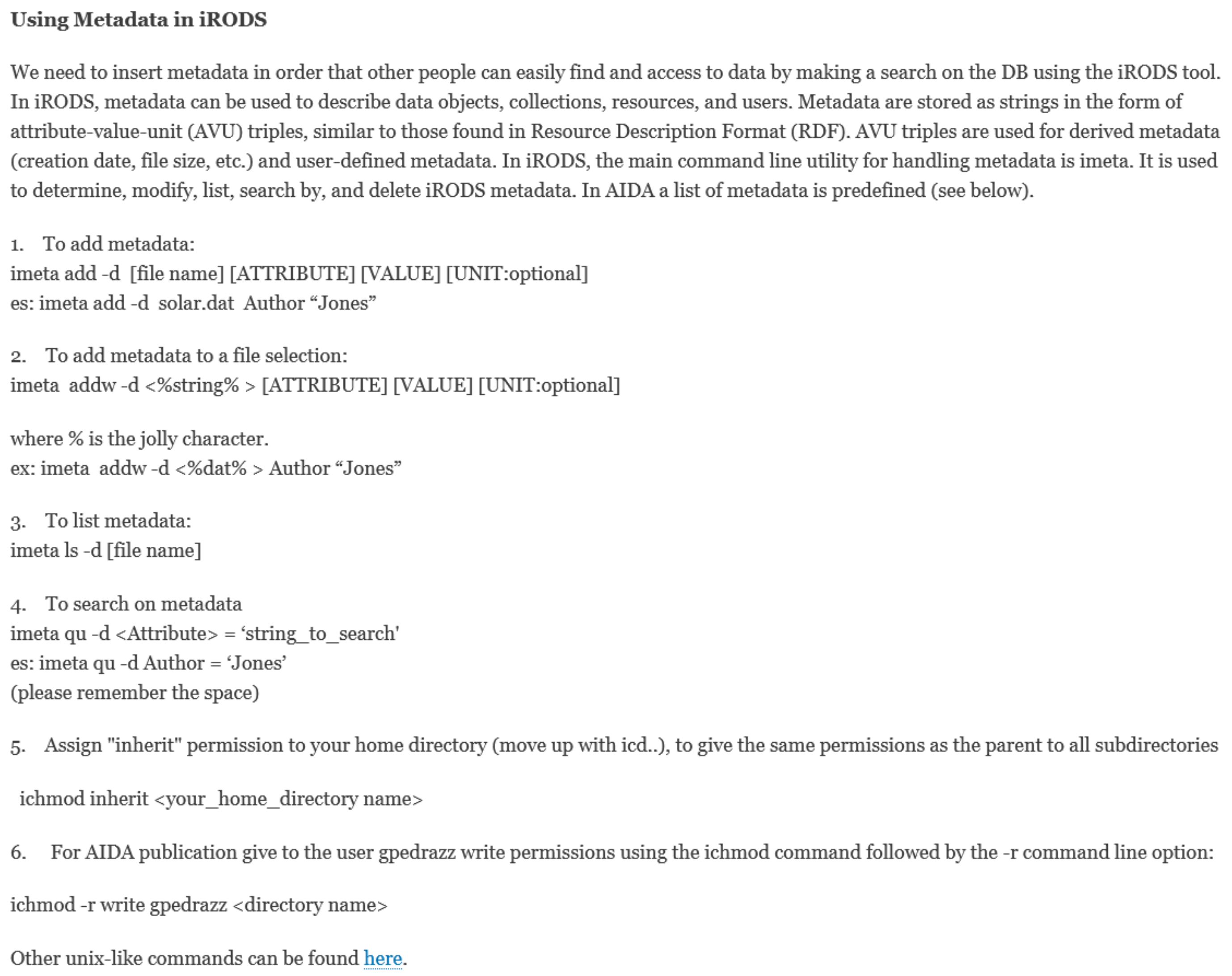
Basic files and directory commands

1. To store a file the iRODS current directory use the command: iput [local file name]
2. To store a directory iput -r [local directory name]
3. To list stored files ils -A
4. To show current path: ipwd
5. To create a new directory (collection): imkdir [directory name]
6. To move among directories: icd [directory name]

More unix-like iRODS commands can be found on-line here.

The icommands are very similar to traditional Linux commands, but with an “i” in front. Certain operations to move from one directory to the next or to print the current directory examined in the iRODS database are very similar to Linux: icd, ipwd, imkdir, etc.

The link above also includes an explanation on how to read and write metadata in iRODS using the icommands, and includes the current list of AIDA metadata entries defined by the AIDA Data Management Plan.



Using Metadata in iRODS

We need to insert metadata in order that other people can easily find and access to data by making a search on the DB using the iRODS tool. In iRODS, metadata can be used to describe data objects, collections, resources, and users. Metadata are stored as strings in the form of attribute-value-unit (AVU) triples, similar to those found in Resource Description Format (RDF). AVU triples are used for derived metadata (creation date, file size, etc.) and user-defined metadata. In iRODS, the main command line utility for handling metadata is imeta. It is used to determine, modify, list, search by, and delete iRODS metadata. In AIDA a list of metadata is predefined (see below).

1. To add metadata:

```
imeta add -d  [file name] [ATTRIBUTE] [VALUE] [UNIT:optional]
es: imeta add -d  solar.dat  Author “Jones”
```

2. To add metadata to a file selection:

```
imeta  addw -d <%string% > [ATTRIBUTE] [VALUE] [UNIT:optional]
```

where % is the jolly character.

```
ex: imeta  addw -d <%dat% > Author “Jones”
```

3. To list metadata:

```
imeta ls -d [file name]
```

4. To search on metadata

```
imeta qu -d <Attribute> = ‘string_to_search'
es: imeta qu -d Author = ‘Jones’
```

(please remember the space)

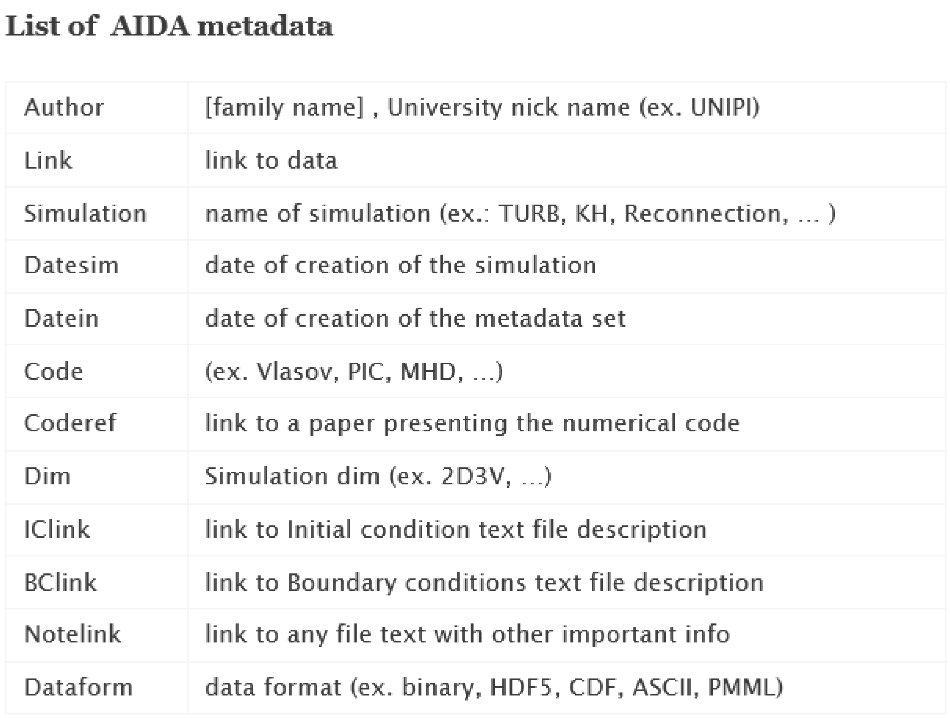
5. Assign "inherit" permission to your home directory (move up with icd..), to give the same permissions as the parent to all subdirectories

```
 ichmod inherit <your_home_directory name>
```

6. For AIDA publication give to the user gpedrazz write permissions using the ichmod command followed by the -r command line option:

```
ichmod -r write gpedrazz <directory name>
```

Other unix-like commands can be found here.



List of AIDA metadata

Author	[family name] , University nick name (ex. UNIPI)
Link	link to data
Simulation	name of simulation (ex.: TURB, KH, Reconnection, …)
Datesim	date of creation of the simulation
Datein	date of creation of the metadata set
Code	(ex. Vlasov, PIC, MHD, …)
Coderef	link to a paper presenting the numerical code
Dim	Simulation dim (ex. 2D3V, …)
IClink	link to Initial condition text file description
BClink	link to Boundary conditions text file description
Notelink	link to any file text with other important info
Dataform	data format (ex. binary, HDF5, CDF, ASCII, PMML)

An example of this first option is included in the following Google Colab notebook under " *#Option1: iRODS icommands*" :

https://colab.research.google.com/drive/1uinnEEqsBhEHJze5XMX9Jf3IuDKPMdrb?usp=sharing

Option 2.
The second option uses a new Data Reader included in the AIDApy Data Engine. After the authentication of the user using the methods mentioned before, the user can call an AIDApy module that performs the data access and data downloading. The AIDAdb Data Reader is available in to any Python developer by importing the following module:

from aidapy.data.irods_interface.irods_connectors import IRodsClient

A detailed description on how to use the AIDAdb Data Reader can be found in the AIDApy online documentation, in the examples directory, in our jupyter notebooks, and in the following Google Colab notebook under " *#Option2: AIDApy Data Reader API*" :

https://colab.research.google.com/drive/1uinnEEqsBhEHJze5XMX9Jf3IuDKPMdrb?usp=sharing

4 Documentation

The currently available information on the interface between AIDApy and AIDAdb and the connection of the AIDA user to AIDAdb is included in this report as well as in the deliverable D2.4 . Information is also available on the AIDA website .

The full information will be included in the AIDA online documentation, which will be maintained after the end of the AIDA project and will always include the latest documentation.

5 Conclusions and future work

In this report, we have described the definition and implementation of the interface between AIDApy and AIDAdb. The interface is based on the open-source iRODs software. In the current version, two options are possible for the AIDA user to connect to AIDAdb and retrieve the necessary AIDAdb products:

1. In the first option, the user connects to AIDAdb by directly executing iRODS commands from his/her own computer. The necessary iRODS commands are explained in Section 3 as well as in one example included in a Google Colab notebook.

2. In the second option, an AIDApy Data Reader API is used. This API is part of the AIDApy package. One example is included in a Google Colab notebook.

For both options, the user can directly obtain from the Cineca's iRODS server small datasets such as high-level products (virtual spacecraft data, spacecraft events' lists etc.). The user can also obtain the metadata of all AIDAdb products, which include the location of the full files on Cineca's Marconi high capacity disks. These full files can be downloaded by the user via SCP to Marconi.

The full documentation related to the AIDApy/AIDAdb interface will be included in the AIDA online documentation. Current information is included in this deliverable, in deliverable D2.4 as well as on the AIDA website .

Improvements of the AIDApy/AIDAdb interface may be done in future outside of the boundaries of the AIDA project, e.g. in a follow-up project. Similarly, the AIDAdb could be in the future interfaced to external databases such as e.g. ESA Heliophysics Science Archives, to broaden the impact of the AIDA project.