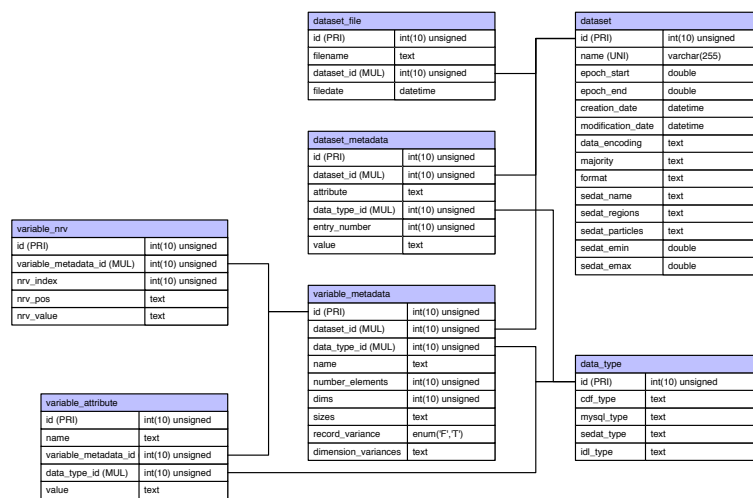


ODI User Guide

ESTEC/Contract No. 21964/08/NL/AT

September 25, 2009



Peter Wintoft, Swedish Institute of Space Physics

Daniel Heynderickx, DH Consultancy



DC_H

Document status sheet

Version	Date	Comment
1.0	2009-09-25	First version.

Contents

1	Summary	4
2	Prerequisites	4
3	The ODI database	4
4	ODI tools	4
4.1	Exploring the database	4
4.2	Interface for reading data	5
4.2.1	PHP implementation	5
4.3	Exporting data to text file	6
4.4	Exporting data to CDF file	6
A	Metadata	7

1 Summary

The Open Data Interface (ODI) database builds on the MySQL database server to store space physics data compliant with the CDF/ISTP/PRBEM guide lines. In this manual we describe how to explore the database and pull data out of the system.

2 Prerequisites

The ODI system must have been properly installed and set up. This is described in the *ODI Administrator Guide*. It is assumed that it is installed under `$ODI_HOME`.

3 The ODI database

The ODI system builds on a MySQL database. For an end user it is only possible to read data and metadata from the system, it is not possible to modify the database. As the ODI database is a MySQL database the data can be directly accessed in many different ways, e.g. from the `mysql` command line, PHP, Java, IDL, Matlab, and so on.

ODI contains two categories of tables: tables that hold *metadata* and tables holding *data*. The metadata tables are shown in Appendix A. These tables contain all metadata for all datasets stored in ODI. Each table contains a unique key (`id`) that is used to tie e.g. the metadata to a specific variable, for a specific dataset. In this way queries can be constructed to find the right metadata.

To each dataset there is an associated table that holds the data. The names of the datatables always start with `dataset_` followed by a unique name following the syntax `dataset_<platform>_<instrument>` like `dataset_xmm_rm`. The data are organised in the tables with the following column names:

`cdf_epoch`, `epoch`, `millisec`, `<var 1>`, `<var 2>`, ..., `dataset_file_id`

where `cdf_epoch` is the CDF epoch, `epoch` is the epoch in YYYY-MM-DD hh:mm:ss format, `millisec` is the millisecond part. Then follows the variables, one column for each variable, and finally there is a `dataset_file_id` column that is used internally by the ODI system to keep track of ingested datasets.

[More text will be added.]

4 ODI tools

4.1 Exploring the database

The command

`show_datasets.php [<dataset>]`

shows the SEDAT name, ODI name, number of records, and epoch range for the selected dataset, where `<dataset>` is the SEDAT name. If `<dataset>` is omitted all datasets are listed. The `<dataset>` may contain the MySQL wild card character %.

All metadata of a dataset and its variables are listed with the command

```
show_metadata.php <dataset>
```

where `<dataset>` is the SEDAT name of the dataset.

4.2 Interface for reading data

To access data from the ODI database a standard interface is defined. This interface can then be implemented in different languages. To read data from ODI involves the following steps:

1. Connect to the database.
2. Select a dataset.
3. Specify variables.
4. Specify an epoch range or a record range.
5. Read data.
6. Read metadata.

The following generic parameters must be specified:

```
[user],[password],[host],[port],[socket]  
odi_dataset_name | sedat_dataset_name  
epoch_start,epoch_end | record_start,record_end  
variable_array
```

where “[]” indicates optional arguments, and “|” means that just one should be specified. Whether the `user`, `password`, etc. must be specified depends on how the system is set up.

To connect to ODI and read data and metadata the following generic functions are needed:

```
connectToOdi(user,password,host,port,socket)  
readData(dataset,range,variables)  
readMetadata(dataset,variables)
```

where the argument and return parameters depends on the language.

4.2.1 PHP implementation

In PHP the interface is implemented as

```
$conn = connect_to_odi($conn_params)
$data = read_data($conn,$dataset,$range,$variables)
$meta = read_metadata($conn,$dataset,$variables)
```

where `$dataset`, `$range`, and `$variables` are arrays according to

```
$dataset = array("odi_name"=>"<dataset name>") |
    array("sedat_name"=>"<dataset name>")
$range = array("epoch_start"=>"<start epoch>", "epoch_end"=>"<end epoch>") |
    array("record_start"=>"<start record>", "record_end"=>"<end record>")
$variables = array("<var 1>", "<var 2>", ...)
```

The return parameter `$data` is a 2-dimensional array, where the records vary over the first dimension and the variables over the second dimension. The return parameter `$meta` is an array containing the global metadata and the variables metadata.

4.3 Exporting data to text file

To export a dataset the epoch range and variables must be specified. This information is given in an XML settings file. A template settings file can be generated with the command

```
export_build_xml.php <dataset>
```

which will create the file `<dataset>.set.xml`. This file may then be edited. Here one can change the name of the output file, the epoch range, what variables to export and the order of the variables. They are given within the XML tags `<export_file></export_file>`, `<epoch_start></epoch_start>`, `<epoch_end></epoch_end>`, and `<variable></variable>`, respectively. The next step is to run

```
export_to_xml.php <settings file>
```

which will produce the export file, which is also an XML file. The file contains a header section with the field names and a data section with each field separated by commas.

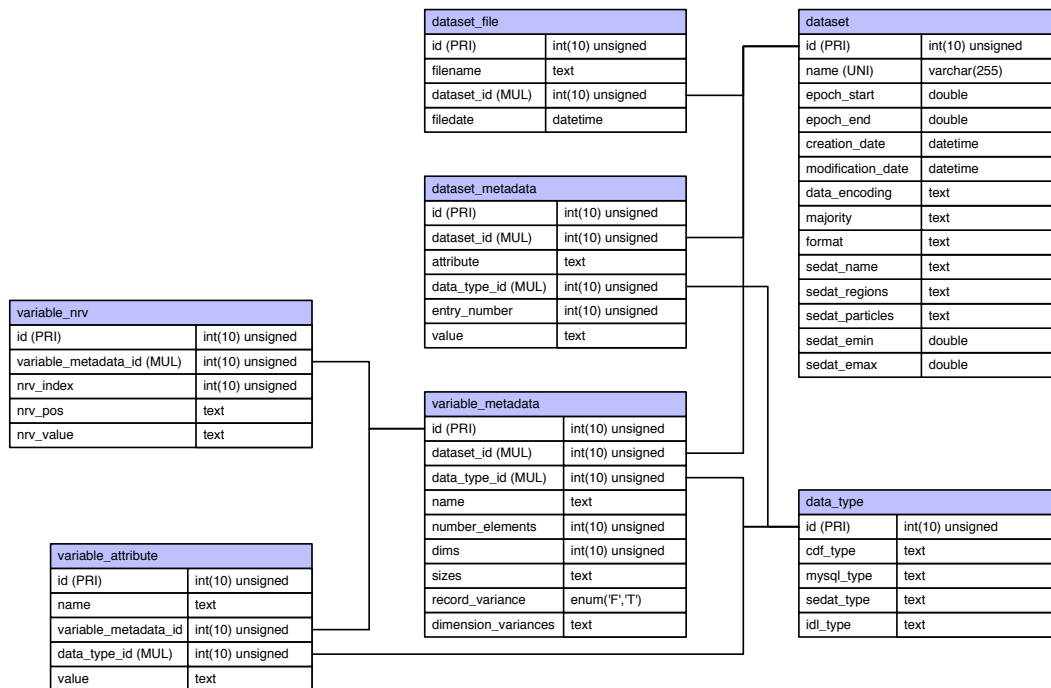
4.4 Exporting data to CDF file

The ODI data and metadata can be exported to a CDF file using the IDL program

```
odicdf, cdf_name, dataset_name, varnames=varnames, $  \\
    epochrange=epochrange, recordrange=recordrange
```

where the name for the cdf file, the SEDAT dataset name, the variables to be extracted (all variables if this keyword is not specified), and epoch or record range is given (one and only one of epoch or record range).

A Metadata



dataset	A table holding the names of each dataset_* table together with some key information.
id	A unique identifier.
name	The name of the dataset table in the ODI database.
epoch_start	The first epoch in the dataset.
epoch_end	The last epoch in the dataset.
creation_date	The date when the dataset_* table was created.
modification_date	The last date when the dataset was modified.
data_encoding	The CDF header attribute DATA ENCODING.
majority	The CDF header attribute MAJORITY.
format	The CDF header attribute FORMAT.
sedat_name	SEDAT dataset name without SYSTEM! prefix.
sedat_regions	SEDAT region code.
sedat_particles	SEDAT particle code.
sedat_emin	SEDAT min. energy (MeV).
sedat_emax	SEDAT max. energy (MeV).

dataset_file	A table to hold the file names of all ingested data files.
id	A unique identifier.
filename	The file name of the ingested file.
dataset_id	A key to the associated dataset in table dataset.
filedate	The date when the file was ingested.

dataset_metadata	Metadata for each dataset.
id	A unique identifier.
dataset_id	A key to the associated dataset in table dataset.
attribute	The attribute name. This corresponds to the CDF global attribute.
data_type_id	A key to the associated data type in table data_type.
entry_number	The CDF global parameter entry number.
value	The value (or contents) of the dataset attribute.

data_type	CDF data types together with associated MySQL and SEDAT data types.
id	A unique identifier.
cdf_type	The CDF data type.
mysql_type	The MySQL data type.
sedat_type	The SEDAT data type.

variable_attribute	The attributes for each variable.
id	A unique identifier.
name	The name of the variable attribute.
variable_metadata_id	A key to the associated variable in table variable_metadata.
data_type_id	A key to the associated data type in table data_type.
value	The value (or contents) of the variable attribute.

variable_metadata	The metadata for each variable.
id	A unique identifier.
dataset_id	A key to the associated dataset in table dataset.
data_type_id	A key to the associated data type in table data_type.
name	The name of the variable.
number_elements	The CDF variable parameter Number Elements.
dims	The CDF variable parameter Dims.
sizes	The CDF variable parameter Sizes.
record_variance	The CDF variable parameter Record Variance.
dimension_variances	The CDF variable parameter Dimension Variances.

<code>variable_nrv</code>	The values of the non-record-variant data.
<code>id</code>	A unique identifier.
<code>variable_metadata_id</code>	A key to the associated variable in table <code>variable_metadata</code> .
<code>nrv_index</code>	An index to the nrv variable. It goes from 1 to <code>dims</code> .
<code>nrv_value</code>	The value of the nrv variable.