



## Download Manager User Manual



**HORIZON 2020**

[sdn-userdesk@seadatanet.org](mailto:sdn-userdesk@seadatanet.org) – [www.seadatanet.org](http://www.seadatanet.org)

SeaDataCloud - Further developing the pan-European infrastructure for marine and ocean data management

Grant Agreement Number: 730960

Deliverable number	Short title
	Download Manager User Manual
Long title	
Download Manager User Manual, Management at Data Centre	
Short description	
<p>This document describes how to prepare your data that will be downloaded by the Download Manager, it also gives information about how to check the conformity of the coupling table and the consistency between the coupling table and the CDI metadata catalogue. Troubleshooting is also detailed.</p>	
Author	Working group
See History	WP9
Dissemination	Copyright terms
Public	

## History

Version	Authors	Date	Comments
1.2.0	S. Guyomarch A. Briand	22/03/2012	<ul style="list-style-type: none"> <li>• PostgreSQL database</li> <li>• RESTful web service for modus 1</li> <li>• ZIP files and dynamicConfig.xml</li> <li>• SEG Y format</li> <li>• Nagios monitoring</li> <li>• Tnsnames</li> </ul>
1.2.0.a	S. Guyomarch	26/03/2012	<ul style="list-style-type: none"> <li>• Guide to replace JDBC drivers library</li> <li>• Troubleshooting: DATE column with Oracle, time information and modus 2</li> </ul>
1.3.0	S. Brégent	31/10/2012	Add Seismic data visualization for GeoSeas project
1.4.0	S. Brégent	14/11/2012	Add Seismic data visualization for GeoSeas
1.4.1	S. Brégent	04/02/2013	Added CAS parameters in DM_Servletconfig.properties file
1.4.2	S. Brégent	19/08/2013	Use BODC V2 (Splitter) New checker functionality: compare local coupling table with CDI metadata catalogue
1.4.3	A. Briand	18/10/2013	Update of database drivers



	S. Brégent		Management of duplicate entries in zip file
1.4.4	S. Brégent	13/05/2014	Refactoring
1.4.5	A. Briand S. Brégent	23/01/2015	New configuration options on Modus 2 mapping files, according to updates on ODV format (cf. SeadatanetDataTransport Formats v.1.13)  coupling table configuration: allow windows authentication using JTDS driver
1.4.6	S. Brégent	21/07/2015	Update version number  No change in the document
1.4.7	S. Brégent	05/04/2017	Use standard ISO format for date format in ODV mapping files (§ 1.4.3.2)

***The current document can be found at:***

<http://www.seadatanet.org/Standards-Software/Software/Download-Manager>

***References documents***

[1] *SeaDataNet Datafile Formats, version 1.1.1*

<http://www.seadatanet.org/Standards-Software/Data-Transport-Formats>

## Table of contents

1.	Data preparation .....	5
1.1.	Data storage.....	5
1.2.	Data conversion .....	5
1.3.	Coupling table configuration.....	6
1.3.1.	Coupling table in a configuration file .....	6
1.3.2.	Coupling table configuration in a database.....	10
1.4.	ODV mapping XML file configuration.....	11
1.4.1.	Purpose of the file .....	11
1.4.2.	Attributes of <code> node .....	11
1.4.3.	ODV metadata columns .....	12
1.4.4.	Optional Metadata SDN references .....	13
1.4.5.	ODV data columns.....	13
2.	Check the Download Manager .....	15
2.1.	The DM_Checker tool .....	15
2.1.1.	Coupling consistency check.....	15
2.1.2.	Central/coupling coherence check.....	16
2.2.	Nagios monitoring.....	17
3.	Troubleshooting .....	17
3.1.	Order status issue .....	17
3.2.	Result files issue .....	17
3.3.	Logs files issue.....	18
3.4.	Software version issue .....	19
3.5.	Note about DM_Checker memory allocation .....	19

# 1. Data preparation

This chapter describes how to prepare your data before installing the Download Manager.

## 1.1. Data storage

The Download Manager can manage data distributed by your organization in different modus to disseminate datasets to the client:

Modus	Description
1	Data are already in <b>files that are stored in a directory, or can be downloaded from a web service.</b> Each file corresponds to <b>one</b> Local CDI ID.
2	<b>Data are stored in a database.</b> The file is created on the fly by the Download Manager. See §1.4.
3	Data are already in <b>files that are stored in a directory, or can be downloaded from a web service.</b> One file contains <b>one or several</b> Local CDI ID.
4	Visualization service
5	Data linked to a visualization service, stored in files.

Data in files (modus 1, 3, 5) must be stored in a directory that must be accessible to the Download Manager i.e. this directory can be on the same machine or accessible via the network or a webservice. It is recommended that this directory's contents is backedup after a change in at least 2 different other locations.

This directory is considered as the 'root' directory; it is possible to divide the data files over multiple sub-directories under the root. The latter must be specified as a relative path in the data set file names in the coupling table (see §1.3).

## 1.2. Data conversion

The table below summarizes the type of possible conversions:

**mono** means that the file contains only one LOCAL\_CDI\_ID,

**multi** is for files with 1 to n LOCAL\_CDI\_ID(s).

**input format** corresponds to the format of the files as they are stored at the data centre.

**output format** is the format of the files that the download manager will deliver to the user.

**modus** is the type of file described in the coupling table.

Input Format		Modus in coupling	Output format (always mono station)
name	station		
Database	-	2	CFPOINT
	-	2	ODV
Medatlas	mono	1	CFPOINT
		3	ODV
	multi	3	CFPOINT
			Medatlas
ODV	mono	1	CFPOINT
			ODV
	multi	3	CFPOINT
			ODV
CFPOINT	mono	1	CFPOINT
	multi	3	CFPOINT

## 1.3. Coupling table configuration

The next step is to configure the ‘coupling table’ with the relations between the Local\_CDI\_ID identifier, as included in the central CDI Directory, and local available data.

These mappings (relations) can be defined in two places: in a configuration file called coupling.txt or in a database table.

Coupling table selection is performed according to the value of the property coupling\_table\_dbmsin <dm\_datacentre\_dir>/DM\_batches/config/config.properties.

Reminder: <dm\_datacentre\_dir> is the directory where the DM Batches are installed.

Checks can be performed on the coupling table to raise some configuration issues. See §2.1.1 for additional information.

### 1.3.1. Coupling table in a configuration file

When a configuration file is used for defining the above relations, they must be declared in the file: <dm\_datacentre\_dir>/DM\_batches/config/coupling.txt.

Each line in this file defines a relation between a Local CDI ID of the data centre and either:

- a data file or RESTful web service;
- a database query returning the data;
- or one record within a Medatlas multi file
- a visualization service

The number and identity of the Local CDI ID records in the coupling table must be identical to the CDI records in the central CDI Directory at the SeaDataNet portal.

Furthermore, each Local CDI ID must lead to a unique local data set, either ‘from the shelf’ as pre-processed file or from a local database query (with consecutive conversion to SeaDataNet ODV format) or from a Medatlas multi file ‘from the shelf’.

Each line of the coupling table (*coupling.txt*) consists of the following fields, separated by a semicolon (;):

Field Number	Description
1. Local_CDI_Id	The CDI local identifier, as included in the central CDI Directory at the SeaDataNet portal.
2.Management modus	<p>Value equals to :</p> <p>1: if it concerns a pre-processed data file (retrieved from the shelf or from a RESTful web service) See §1.2 for possible conversion on the fly</p> <p>2: if data file has to be retrieved from a local database.</p> <p>3: if data file is an ODV, Medatlasor CFPOINT multi format and should be converted to profiles using the “splitter” application. If data file is a Medatlas mono format to be converted in ODV. See §1.2 for possible conversion on the fly</p> <p>4: if data is a visualization service accessing data on disk according to a predefined organization</p> <p>5: only if data is associated to a visualization service (modus4) which requires a preliminary extraction/processing from the archives eg when data are on tapes.</p> <p>For each a “modus4” local_cdi_id, at least 4 lines with modus5 MUST exist, with the same local_cdi_idie :</p> <ul style="list-style-type: none"> <li>✓ 1 to n lines for the seismic data files</li> <li>✓ 1 for the UKOOA navigation file</li> <li>✓ 1 for O&amp;M file</li> <li>✓ 1 to n lines for the SML lines</li> </ul> <p>Modus5 files must be pre-processed data file (retrieved from the shelf or from a RESTful web service, if the URL ends with a filename (*.*)</p>

3.Data format name	<p>Value equals depends on the available data files. Supported formats depending on modus are given in the table below.</p> <p><b>Users can choose a file format in their shopping request, based upon the indicated formats in the CDI metadata record. The data format name then is used to connect the format request to the specific data set file. See §1.2 for possible conversion on the fly.</b></p> <p>In case that one Local_CDI_ID corresponds to 2 files with different formats then the coupling table should contain 2 identical mapping records with different filenames and format names. E.g.:</p> <ul style="list-style-type: none"> <li>• RNODC_Bottle_14828_1;1;ODV;RNODC_Bottle_14828.txt</li> <li>• RNODC_Bottle_14828_1;1;MEDATLAS;RNODC_Bottle_14828.medatlas</li> </ul> <p>This example shows that CDI record RNODC_Bottle_14828_1 corresponds to pre-processed data files both in ODV format and MedAtlas format.</p> <p>In management modus 4, the format is the visualization service name. E.g.:</p> <ul style="list-style-type: none"> <li>• SISM10_FI091999010140_86864;4;HRSVS</li> </ul> <p>In management modus 5, 1 record in modus 4 and n records with different seismic format names for n seismic filenames. E.g.:</p> <ul style="list-style-type: none"> <li>• SISM10_FI091999010140_86865;4;HRSVS</li> <li>• SISM10_FI091999010140_86865;5;OEM;SISM10_FI091999010140_86865_oem.xml</li> <li>• SISM10_FI091999010140_86865;5;SML;SISM10_FI091999010140_86865_sml.xml</li> <li>• SISM10_FI091999010140_86865;5;UKO;SISM10_FI091999010140_86865_ukooa.uko</li> <li>• SISM10_FI091999010140_86865;5;TRA;99010140_AEA1002.TRA</li> </ul>
--------------------	---

**Table1 Supported formats depending on modus**

Modus format	1	2	3	4	5
<visualization service name>	x	x	x	✓	x
ODV	✓	✓	✓	x	x
MedAtlas	✓	x	✓	x	x
CFPOINT	✓	✓	✓	x	x
SEGY	✓	x	x	x	✓
SML	x	x	x	x	✓
UKO	x	x	x	x	✓
OEM	x	x	x	x	✓
Any format	✓	x	x	x	✓

The OEM files are used only for modus 5 to describe additional information about the measurement they must obey the following rules:

OEM files must match the pattern “\*\_oem.xml”.

There can be only one O&M file for one Local CDI ID.

Then, others parameters depends on the modus value:

In case of modus 1 and 3 – Retrieving pre-processed data files from the shelf	
Field Number	Description
4.Local data filename	Dataset filename in ODV / NetCDF CFPOINT / MedAtlas / SEGYP format. Note: in case the data sets are divided over multiple subdirectories under the ‘root’ (see <i>config.properties</i> file), then each data set name should be preceded by its subdirectory name(s):



	<p>e.g.: chemistry/ RNODC_Bottle_14828.txt</p> <p>The 'root' should not be included in the data set file name, because that is already defined as parameter "data_path" in the <i>config.properties</i> file.</p> <p>Remark: always start the local data filename without a slash (/). This is sufficient for modus 1 to enable the Download Manager to find the correspondence between Local_CDI_Id and the data filename in the requested data format.</p> <p>If the management modus equals to 3 then the "splitter" application will start automatically to retrieve requested records from the MedAtlas, ODV or CFPOINT multi file, producing these as single MedAtlas, ODV or CFPOINT files. Splitter can also convert an ODV or MedAtlas multi file to produce a single CFPOINT file. The produced format (single MedAtlas, CFPOINT or ODV) depends on the format requested by the user during ordering. Note that the user can only choose between the formats that have been made available by the data centre through the CDI XML records. In this case, the data centre should list both ODV, CFPOINT and MedAtlas as available data formats in the CDI XML files.</p> <p><b>Note :</b> when modus=1, it is possible to package in a single archive file like "rar" or "zip" several files associated with a same Local_CDI_ID. The extensions that must be kept for the output files have to be defined in the dynamicConfig.xml file (see section "Configuration of dynamicConfig.xml" in installation manual)</p> <p><b>Note :</b> when modus=1 and format=SEGY, you must provide an archive file (ie .zip file) that can contains only one SEGY file with data, or the SEGY file with data and the UKOAP190 file with navigation. The extension of the archive must be defined in the dynamicConfig.xml file (see section "Configuration of dynamicConfig.xml" in installation manual)</p>
--	---

**An example of a line in coupling.txt for a pre-processed file retrieved from the shelf (modus 1):**

*RNODC\_Bottle\_14828\_1;1;ODV;RNODC\_Bottle\_14828.txt*

**Examples of a line in coupling.txt for a record within a MedAtlas multi file (modus 3):**

*RNODC\_Bottle\_14828\_1;3;MEDATLAS;RNODC\_Bottles.med*

*RNODC\_Bottle\_14828\_1;3;ODV;RNODC\_Bottles.med*

In case of modus 1 or 5 – Retrieving pre-processed data files from a RESTful web service	
Field Number	Description
4. Local data filename	The URL where is located the file to retrieve from a RESTful web service. It must start with http:// It must end with a filename (*.*) if modus is 5
5. Proxy (optional)	Proxy URI uses by the Download Manager when contacting the URL. This parameter is optional.
Login (optional)	Login / username to access the web service This parameter is optional.
Password (optional)	Password to access the web service This parameter is mandatory when login / username is specified ; otherwise, it is optional

The Download Manager makes an HTTP request toward the URL (field n°4). When the request succeeds (i.e. it receives HTTP status 200/OK), it copies the content of the response into an output file.

A proxy can be specified with field n°5 when necessary.

A HTTP-authentication may be required to access the RESTful web service. Currently, BASIC and DIGEST HTTP-authentication have been successfully tested. In this case, the Download Manager uses fields' n°6 and n°7 as username and password.

**Examples of a line in coupling.txt for a pre-processed file retrieved from a RESTful web service (modus 1):**

No proxy and no HTTP-authentication:



sdn-userdesk@seadatanet.org – www.seadatanet.org

SeaDataCloud - Further developing the pan-European infrastructure for marine and ocean data management

Grant Agreement Number: 730960

*Bottle\_1547;1;ODV;http://www.azerty.com/test\_ws/odv/Bottle\_1547*

Use a proxy (<http://proxy.mydomain.com:3128>) and no HTTP-authentication:

*Bottle\_1547;1;ODV;http://www.azerty.com/test\_ws/odv/Bottle\_1547;http://proxy.mydomain.com:3128*

Use a proxy (<http://proxy.mydomain.com:3128>) and HTTP-authentication (myUsername / myPassword):

*Bottle\_1547;1;ODV;http://www.azerty.com/test\_ws/odv/Bottle\_1547;http://proxy.mydomain.com:3128;myUsername;myPassword*

No proxy (i.e. the 5<sup>th</sup> field is empty) and HTTP-authentication (myUsername / myPassword):

*Bottle\_1547;1;ODV;http://localhost:8183/test\_ws/odv/Bottle\_1547;;myUsername;myPassword*

In case of modus 2 – Retrieving data from a local database	
Field Number	Description
4.SQL query	<p>SQL query to retrieve from local database all metadata and data, necessary for creating the ODV or CFPOINT data file.</p> <p>Note: the length of the sql_query in the coupling table can be expanded up to 4000 characters.</p> <p>Note: it must include an ORDER clause according to the ODV format specifications (see document [1]) A “row_group” is identified in the mentioned document as “Data rows with exactly the same metadata parameters and that are grouped together”:</p> <ul style="list-style-type: none"><li>• For profile data, “rows within the row_group are ordered by increasing depth or pressure”</li><li>• For point time series, “rows within the row_group are ordered by increasing time”</li><li>• For trajectories, “rows within the row_group are ordered by increasing time”</li></ul> <p>As a conclusion, the ORDER clause should begin with METADATA columns (to group rows with same metadata) and then with the DEPTH/PRESSURE or the TIME column according to the shape of the generated ODV file (to sort lines inside a row_group according to ODV format).</p> <p>See §1.4.3 about XML mapping files, METADATA columns name should be specified in these files.</p> <p>See document [1] for additional information about ODV specifications.</p>
5.Database connection	<p>For a description of the string format, see in batch config.properties file.</p> <p>Note for MS SQL Servers:If you use Microsoft Windows authentication instead of set the login/passord in the config file, you can use the jtds driver: coupling_table_connection=jdbc:jtds:sqlserver://&lt;server&gt;:&lt;port&gt;/&lt;database_name&gt; In this case, coupling_table_user and coupling_table_password will be ignored</p>
6.Login	Login for database user. It cannot be empty except if MS authentication is used.
7.Password	Password for database user. It cannot be empty except if MS authentication is used.
8.Mapping file	usedfor conversion to the SeaDataNet ODV Format. This field contains only the filename without path to its directory. For more information about ODV mapping xml files see §1.4.

**NOTE:** If a parameter value contains a semicolon (typically, JDBC connection string for a MS-SQL Server database), this value must be declared between two double quotes (“”).

#### An example of a line in coupling.txt for database retrieval (modus 2):

*RNODC\_Bottle\_14828\_10;2;ODV;select \* from table1 where ... order by metadataColumn1, ..., metadataColumnN,pressureColumn;jdbc:mysql://195.178.224.89/dm\_test;root;root;odv\_mapping\_2.xml*

The Download Manager will find the requested Local\_CDI\_Id in the coupling table, identify the management modus and pass control to the “read-method” with such input parameters as connection string, login and password. The “read-method” will connect to the indicated database and retrieve the



[sdn-userdesk@seadatanet.org](mailto:sdn-userdesk@seadatanet.org) – [www.seadatanet.org](http://www.seadatanet.org)

SeaDataCloud - Further developing the pan-European infrastructure for marine and ocean data management

Grant Agreement Number: 730960

data using the SQL query from the coupling table. The query result will be put to the Memory buffer and then converted to result data file with output format ODV. For converting data from the database to ODV file “read-method” uses the mapping configuration file between system parameter names and local database column names which must be configured at the next step.

**An example of a line in coupling.txt for database retrieval (modus 2) – Case of a MS-SQL Server database:**

```
RNODC_Bottle_14828_10;2;ODV;select * from table1 where ... order by metadataColumn1, ...,
metadataColumnN,presureColumn; "jdbc:sqlserver://195.178.224.89;databaseName=dm_test";root;root;odv_mapping_2.xml
```

JDBC connection string value is set between two double quotes (") because of the semicolon (;) char.

**An example of a line in coupling.txt for a service (modus 4):**

```
SISM10_FI091999010140_86864;4;HRSVS
```

**Examples of a line in coupling.txt for a service (modus 5):**

```
SISM10_FI091999010140_86865;4;HRSVS
```

```
SISM10_FI091999010140_86865;5;OEM;SISM10_FI091999010140_86865_oem.xml
```

```
SISM10_FI091999010140_86865;5;SML;SISM10_FI091999010140_86865_sml.xml
```

```
SISM10_FI091999010140_86865;5;UKO;SISM10_FI091999010140_86865_ukooa.uko
```

```
SISM10_FI091999010140_86865;5;TRA;99010140_AEA1002.TRA
```

### 1.3.2. Coupling table configuration in a database

When the choice is made to put the coupling table definition in a database table, the DM\_Batch should have access to a table with the structure as defined below. The table name and database settings must be defined in <dm\_datacentre\_dir>/DM\_batches/config/config.properties as described in The Download Manager Installation Manual

The advantage of storing the coupling table in a database is that the contents of the table can easily be generated using SQL. This is especially useful if the data itself is also stored in a database.

The table structure of coupling table must be as follows:

Field name	Type	Can be Null?	Mandatory column	Comment
id	int	No	Yes	Auto increment, primary key
local_cdi_id	varchar(64)	No	Yes	Index on this column is recommended for performance
modus	int	No	Yes	Can be one of the following:1, 2, 3, 4, 5
format	varchar(8)	No	Yes	depends on available data files and supported formats, see <b>Coupling table</b> in a configuration file
filename	varchar(128)	Yes	Yes	When using a RESTful web service with modus 1, use this field to specify the URL With modus 4, use this field to specify the service name
sql_query	varchar(512)	Yes	Yes	The length of this field <b>can be expanded up to 4000 characters.</b>
connection_string	varchar(256)	Yes	Yes	For a description of the string format, see in batch config.properties file. Note for MS SQL Servers:If you use Microsoft Windows authentication instead of set the login/passord in the config file, you can use the jtds driver: coupling_table_connection=jdbc:jtds:sqlserver://<server>:<port>/<database_name>



				In this case, coupling_table_user and coupling_table_password will be ignored
login	varchar(64)	Yes	Yes	Login for database user. It cannot be empty except if MS authentication is used.
password	varchar(64)	Yes	Yes	Password for database user. It cannot be empty except if MS authentication is used.
mapping_file	varchar(128)	Yes	Yes	used for the conversion to the SeaDataNet ODV Format. This field contains only the filename without path to its directory. For more information see §1.4.
proxy	varchar(256)	Yes	No	If you plan to use modus 1 and RESTful web service behind a proxy, you have to create this column. Otherwise, it is optional.

**Warning: It is very important that the field names are the same as in the table. Length of string types may differ from the values shown in the table.**

**Updating:** Each Data Centre will regularly update its CDI entries in the central CDI Directory at the SeaDataNet portal. This implicates that the Data Centre also has to update the Coupling Table for synchronization.

## 1.4. ODV mapping XML file configuration

### 1.4.1. Purpose of the file

If the data retrieved by the Download Manager is in a database (“modus 2” in the coupling table), the DM will generate a file in ODV format from the result of the query defined in the coupling table.

To do this, a mapping between field names in the database and columns in the ODV file needs to be defined. This is done in a mapping configuration XML file named *odv\_mapping.xml* (or any other name), placed under `<dm_datacentre_dir>/DM_batches/config/mapping/` (DM\_Batch will always use this path). An example file named *odv\_mapping\_example.xml* is included in the installation package.

**A mapping configuration XML file includes mappings between parameters and units from SeaDataNet Vocabularies P01 and P06 and the SeaDataNet Quality Flags per parameter from the SeaDataNet Vocabulary L20, column names of ODV data format and database table column names and follows a standard format. However depending on the database or table there can be several mapping configuration XML files in use.**

The format provides information about the file update date and time, data format name and combinations of full names of parameters from P01 Vocabulary, units of measurement from P06 Vocabulary and **SeaDataNet Quality Flags from L20 Vocabulary** which are mapped to combinations of name, units of measurement and quality flags for the ODV data format. Each parameter is mapped to a table column name in the database. Column names are described by a “local” attribute in the mapping.

For information about the ODV format specifications, see document [1]

Checks can be performed on the XML mapping files declared in the coupling table to raise some configuration issues. See §2.1.1 for additional information.

### 1.4.2. Attributes of <code> node

Each metadata column and each pair of data value/quality check flag from the generated ODV file are declared by a <code>node in the XML mapping file. Each <code> node can have the following attributes:



- Attribute “from” contains a parameter name from the P01 vocabulary and units from the P06 vocabulary. It is used only for ODV data columns and must be left empty for ODV metadata columns.
- Attribute “to” contains the ODV column name and units name separated by comma. Name and units must correspond to name and units in “from” attribute.
- Attribute “local” contains the table column name in the local database.
- The “read-method” uses this configuration file to retrieve data values from columns described in ‘local’ attributes and put these values to ODV columns described in ‘to’ attributes.
- Attribute “qflag” contains table column name in local database for quality check flag for current parameter. This must be based upon the SeaDataNet quality flag scale as included in the L20 vocabulary.

These attributes are described more in details below.

### 1.4.3. ODV metadata columns

ODV files must contain 9 mandatory metadata columns:

- Some of them can be retrieved by the Download Manager and can be omitted in the XML mapping file,
- Some of them cannot be retrieved by the Download Manager and have to be declared in the XML mapping file,
- Some of them can have a default value assigned by the Download Manager. If they are omitted in the XML mapping file, the default value is assigned. Otherwise, value is retrieved from the database column defined in the XML mapping file.

Mapping on a metadata column are identified in the XML mapping file by `<code>` nodes with an empty value for the attribute “from” (from=””).

Moreover, the “qflag” attribute is not used and can be omitted for these nodes.

It will be added automatically by the batch component. For more details, see §1.4.5.4

#### 1.4.3.1. Unneeded metadata columns in XML mapping file

2 metadata columns can be omitted in the XML mapping file because the Download Manager can retrieve their values from its process.

- **EDMO code is retrieved from the DownloadManager configuration file.**
- **Local\_CDI\_ID is retrieved from the value of the current Local\_CDI\_ID that is being processed by the DownloadManager.**

#### 1.4.3.2. Mandatory metadata columns in XML mapping file

5 metadata columns have to be declared in the XML mapping file because the Download Manager cannot retrieve their values.

They have to be declared like this. Just replace the value of each local attribute by the name of the associated columns in database.

Please note that the attributes “to” have to be exactly as in this example, including casing and spaces, except for the date column.

The pattern must be a valid ISO8601 format mask appropriate for the precision to which the date and time will be written in the output ODV file.

```

<!-- Mandatory Metadata columns that can not be retrieved from batch -->
<code from="" to="SDN:LOCAL:Cruise" local="CRUISE_COLUMN_NAME" />
<code from="" to="SDN:LOCAL:Station" local="STATION_COLUMN_NAME" />
<code from="" to="SDN:LOCAL:YYYY-MM-DDThh:mm:ss.sss" local="DATE_COLUMN_NAME" />
<code from="" to="SDN:LOCAL:Longitude, degrees east" local="LONGITUDE_COLUMN_NAME"/>

```



```
<code from="" to="SDN:LOCAL:Latitude, degrees north" local="LATITUDE_COLUMN_NAME" />
```

### 1.4.3.3. Optional metadata columns in XML mapping file

2 metadata columns are optional in the XML mapping file. If they are omitted in the XML mapping file, a default value is assigned. Otherwise, value is retrieved from the database column defined in the XML mapping file.

If values have to be retrieve from the database, they have to be declared like this. Just replace the value of each local attribute by the name of the associated column in database.

```
<!--  
Optional Metadata columns Bot. Depth [m]  
If not given, value set to 0  
-->  
<code from="" to="SDN:LOCAL:Bot. Depth, m" local="BOTTOM_DEPTH_COLUMN_NAME" />  
<!--  
Optional Metadata columns Type  
If not given, value set to *  
-->  
<code from="" to="SDN:LOCAL:Type " local="TYPE_COLUMN_NAME" />
```

If “Bot. Depth, m” node is omitted, default value for Bot. Depth [m] column in ODV file will be “0”.

If “Type” node is omitted, default value for Type column in ODV file will be “\*”.

### 1.4.4. Optional Metadata SDN references

This element can be used to create SDN references (see Seadatanet Formats Documentation 1.13)

```
<!--  
Optional Metadata SDN references -->  
<code from="" to="sdn_reference" local="SDN_REF_COLUMN_NAME" />
```

The column SDN\_REF\_COLUMN\_NAME shall contain all sdn references (sdn\_reference XML tags) as a string without separator

### 1.4.5. ODV data columns

After metadata columns definition, the data columns mapping has to be declared.

Each mapping for data column is declared like this:

```
<code from="OBJECT, UNITS" to="SUBJECT, UNITS_TITLE" local="VALUE_COLUMN_NAME"  
qflag="QUALITY_COLUMN_NAME" />
```

The columns order in the generated ODV file will be the same as the declaration of their mapping in the XML mapping file.

To be compliant with the ODV format, the first data declared in the XML mapping file must be the primary variable of the ODV document.

- For profile data, “the primary variable is the z-co-ordinate, which for SeaDataNet is either depth in meters or pressures in decibars”,
- For point time series, “the primary variable is time (UT)”,
- For trajectories, “the primary variable is the z-co-ordinate, which for SeaDataNet is standardized as depth in meters”.

#### 1.4.5.1. “from” attribute

“from” attribute must be declared like this :

from="OBJECT, UNITS"

Where:



sdn-userdesk@seadatanet.org – www.seadatanet.org

SeaDataCloud - Further developing the pan-European infrastructure for marine and ocean data management

Grant Agreement Number: 730960

**OBJECT:** A concept from the SeaDataNet P01 vocabulary.  
It must start by SDN:P01:: and contains only ASCII characters.

**UNITS:** This contains a URN for the storage units in the file for the data column in the SeaDataNet P06 vocabulary. It is NOT the code for units linked in the past versions of the P01 vocabulary.  
It must start by SDN:P06:: and contain only ASCII characters.

### 1.4.5.2. “to” attribute

“to” attribute must be declared like this:

to="SUBJECT, UNITS\_TITLE"

Where:

**SUBJECT:** The text user to label the column as it appears in the column row header without the units declaration.

It must start by SDN:LOCAL: and contain only ASCII characters.

**UNITS\_TITLE:** The storage units as it appears in the column row header.

It must not be empty and contain only ASCII characters (do not use characters like “°” for “°C”, “μ” for “μg/mol”, etc...).

**Note:** the SUBJECT and the UNITS\_TITLE must correspond to the OBJECT and UNITS declared in the “from” attribute.

### 1.4.5.3. “local” attribute

“local” attribute must be declared like this :

local="VALUE\_COLUMN\_NAME"

Where:

**VALUE\_COLUMN\_NAME:** The name of the column in the database that contains the value for the given parameter.

It must not be empty.

If there is no data value, then the data value column in the ODV file is left blank (value "") and the quality flag is set to “9” (it means “missing value” in L20).

### 1.4.5.4. “qflag” attribute

“qflag” attribute is optional and can be omitted in the XML mapping file.

“qflag” must be declared like this:

qflag="QUALITY\_COLUMN\_NAME"

Where:

**QUALITY\_COLUMN\_NAME:** The name of the column in the database that contains the SeaDataNet quality flag for the given data value.

This must be based upon the SeaDataNet quality flag scale as include in the L20 vocabulary.

**All measurement data published in an EU marine research project which uses the SeaDataNet Software Infrastructure must be quality tested and the “qflag” column must be defined for quality tested data.**

When the attribute “qflag” is absent or its value is empty (qflag=""), then the check value will be always ‘0’ in the resulting ODV file as default value. It means “no quality check” in L20.

If there is no data value for the value retrieved from the attribute “local”, then the data value column in the ODV file is left blank (value "") and the quality flag is set to “9”. It means “missing value” in L20.



### 1.4.5.5. “instrument” attribute

“instrument” attribute is optional and can be omitted in the XML mapping file.

“instrument” must be declared like this:

```
qflag="INSTRUMENT_COLUMN_NAME"
```

Where:

INSTRUMENT\_COLUMN\_NAME: The name of the column in the database that contains the BODC L22 instrument code to associate with this parameter.

### 1.4.5.6. “fall\_rate” attribute

“fall\_rate” attribute is optional and can be omitted in the XML mapping file.

“fall\_rate” must be declared like this:

```
qflag="FALL_RATE_COLUMN_NAME"
```

Where:

FALL\_RATE\_COLUMN\_NAME: The name of the column in the database that contains the BODC L33 fall rate code to associate with this parameter.

## 2. Check the Download Manager

### 2.1. The DM\_Checker tool

The DM\_Checker tool helps Data Centres to perform several checks on:

- the dynamic configuration file `<dm_datacentre_dir>/DM_batches/config/dynamicConfig.xml`
- the coupling table declared in `<dm_datacentre_dir>/DM_batches/config/config.properties`
- the XML mapping files declared in this coupling table
- the availability of data
- the consistency between coupling table and CDI metadata catalogue

To launch the DM\_Checker, use the following command. Run the DM\_Checker with the same user who runs DM\_Batch :

```
java -Xmx256m -jar <dm_datacentre_dir>/DM_batches/DM_Checker.jar -  
config<dm_datacentre_dir>/DM_batches/config/config.properties
```

When no error is found, the log file ends with:

```
INFO - Local check result=OK
```

When errors have occurred, the log file ends with:

```
ERROR - Local check result=ERROR
```

To know which errors have occurred and solve the issues:

- Read consistency check results in the DM checker log file: DM\_checker.log in the logs directory (DT4 InstallSheet field)
- Read the CDI metadata catalogue/Coupling synchronisation check results in the DM\_Checker\_result\_<date>.csv file, where date is the creation date of the file, using the format yyyyMMdd\_HHmms. This file is created in the DM checker log directory. (DT4 InstallSheet field)

#### 2.1.1. Coupling consistency check

First, a test is performed on the coupling table and on the dynamic configuration file. Then, checks on XML mapping files are done only for rows that have passed the coupling table test.



Hereby some of the checks on coupling table:

- Each couple [Local\_cdi\_id, Format] must be unique, unless if modus is 5 and format is NOT OEM
- For rows with modus 1 or 3, data file must exist and be readable
- For rows with modus 2, XML mapping files must exist and be readable.

Hereby some of the checks on the dynamic configuration file:

- Root node <dynamicConfig> exists and is unique
- Node <extensionsListExceptions> exists, is unique and is a child of <dynamicConfig> node

Hereby some of the checks on the XML mapping files defined in coupling table:

- Each file must contain only ASCII characters
- All mandatory metadata mapping ODV column must be declared
- For each data mapping declaration, check if OBJECT, UNITS and SUBJECT starts with an allowed characters sequence (SDN:P01::, ...)

## 2.1.2. Central/coupling coherence check

DM\_Checker tool uses the URL given in the config.properties file (rsm\_cdi\_format\_map\_url\_prefix) to get the CDI metadata catalogue about the data centre.

The tool compares the catalogue and the coupling table: each couple CDI/format in the coupling table should be found in the CDI metadata catalogue, and each couple CDI/format in the CDI metadata catalogue should be found in the coupling table,

Since the DM can convert from MEDATLAS, to ODV:

If the format read in the coupling table is “MEDATLAS”, the expected formats in the CDI metadata catalogue are MEDATLAS and ODV.

If the format read in the CDI metadata catalogue is “ODV”, the expected format in the coupling table is MEDATLAS or ODV.

Since the DM can convert from MEDATLAS or ODV to CFPOINT:

If the format read in the CDI metadata catalogue is “CFPOINT”, the format in the coupling table can be MEDATLAS or ODV.

The csv file columns are:

Column name	description
CDI	Local CDI ID (read from CDI metadata catalogue or coupling table)
nb occur cdi+format in local DM	Number of occurrences where the combination cdi+format has been found in the coupling table (should be 1)
format	Format (read from CDI metadata catalogue or coupling table)
modus	Read from the coupling table
data status	Indicates if data has been found (file found for modus 1 or 3, expected properties found in the coupling table if modus 2) OK: data OK Not OK: data not OK NA: Not available (for example if the CDI/format has been read in the CDI metadata catalogue, but not found in the coupling table)
cdi sync status	OK: CDI has been found on both sides Local CDI ID missing in coupling: couple CDI/format has been read from CDI metadata catalogue, but has not been found in the coupling table Global CDI ID missing in catalogue: couple CDI/format has been read from the coupling table, but has not been found in the CDI metadata catalogue NA: cdi not readable (eg. Cdi value is empty in the coupling table)



format sync status	OK: couple CDI+format has been found on both sides NA: Not available if cdi sync status is not OK format missing in catalogue: couple CDI+format has been found on both sides, but format has not been found in the CDI metadata catalogue format missing in coupling: couple CDI+format has been found on both sides, but format has not been found in the coupling table
details	DM_Checker tool log warnings or errors

You can also search the strings “ERROR” in the log file.

## 2.2. Nagios monitoring

Nagios monitoring system for the installed Download Managers is required by the SeaDataNet II project and Geo-Seas project.

The monitoring task was given to Maris for Geo-Seas and HCMR for SeaDataNet.

Nagios server can perform a request and sees “OK” message at URL below:

[https://<DM\\_server\\_address>/dm/](https://<DM_server_address>/dm/)

**Nagios server can perform a request at the URL below and sees the version of the Download Manager installed at data center.**

**[https://<DM\\_server\\_address>/dm/status?output=version](https://<DM_server_address>/dm/status?output=version)**

Data centers can be informed about the status and logs of their own services on monitoring web site: <http://seadatanetnm.hcmr.gr/portal/> . To add and check your Download Manager on monitoring portal, you need an extranet account (given by [sdn-userdesk@seadatanet.org](mailto:sdn-userdesk@seadatanet.org)).

## 3. Troubleshooting

This section contains several problems that are frequently encountered when installing/running the Download Manager, with possible solutions.

### 3.1. Order status issue

***Problem:** When I place an order for my own CDI records, the status of the order stays “Approval pending” and does not change to “Ready for download”.*

**Solution:**

- Check if your firewall allows incoming and outgoing requests over port 80 to/from your Download Manager. Please note that the DM can only communicate with the MARIS server over port 80. If you require the DM/Tomcat to run on a different port, it should run behind a proxy. See **§Erreur ! Source du renvoi introuvable.** of this manual.
- Check DM servlets log file (*dm\_servlets.log*) for error messages. Check if the RSM has made a request toward the DM servlet Controller and if *DM\_Batch.start* file has been created. Search the message “INFO - Trigger activation succeed” in logs.
- Check DM\_Batch log file (*dm\_batch.log*) for error message. Maybe the request has been reported with an error status toward the RSM.

### 3.2. Result files issue

***Problem:** I can order my own CDI records and download a zip file, but the zip file is empty!*

**Solution:**



[sdn-userdesk@seadatanet.org](mailto:sdn-userdesk@seadatanet.org) – [www.seadatanet.org](http://www.seadatanet.org)

SeaDataCloud - Further developing the pan-European infrastructure for marine and ocean data management

Grant Agreement Number: 730960

- Check if all paths defined in *config.properties* are correct.
- Check if all filenames defined in coupling table are correct. Also check if there are no formatting errors in *coupling.txt* : make sure the separating character is always the semicolon (;) and that there are no unwanted spaces in the file (esp. at the end of lines).

**Note:** it is recommended that you always start using the Download Manager using static (pre-processed) files. This makes problems easier to diagnose and solve. If everything works with static files you can switch to a database, if desired.

**Problem:** *I'm using a database as my data source. When I order my data, the resulting ODV file only contains the line "SDN parameter mapping".*

**Solution:**

- Please check your SQL query and make sure it does not contain any errors.
- Check *dm\_batch.log* for any error messages related to the database query.

**Problem:** *I'm using a database as my data source. When I order my data, the resulting ODV file only contains the complete header including all parameter definitions and the ODV column headers, but no data.*

**Solution:**

- Please check your database SQL queries in *coupling.txt* and check if it really returns any results.
- Check *dm\_batch.log* for any error messages related to the database query.

**Problem:** *I want to generate ODV files from an Oracle database (modus 2 case). However, when the DM batch is retrieving a column of DATE type, the time information (i.e hours, minutes and seconds) is set to 00:00:00.*

**Solution:** *Instead of retrieving the column as a DATE field, ask Oracle to return it as a string and format it: SELECT to\_char(MY\_DATE\_COLUMN, 'yyyy-mm-dd"T"hh24:mi:ss') FROM...*

### 3.3. Logs files issue

**Problem:** *My log file (dm\_batch, dm\_tools.log or dm\_servlet.log) contains error messages such as:*

```
09/03/2009 11:33:18 - ERROR - java.io.FileNotFoundException:
e:\DownloadManager\DownloadPath\userRep\dailyFile.zip (Access is
denied)
```

**Solution:** Check the permissions of the directories. The Tomcat user (i.e. the user Apache Tomcat is running under) and the user that run the batch should have read and write permissions on this directory.

**Problem:** *My log file (dm\_batch.log) contains error messages that deal with an SSL certificate.*

**Solution:** *Try to update the jssecacerts certificate keystore file. See §Erreur ! Source du renvoi introuvable., « Update SdnAuthService SSL certificate »*

**Problem:** *My log file (dm\_batch.log) contains error messages such as:*



*ERROR - Exception uncaught properly by batch process !!!  
(dm.DM\_Batch.process(DM\_Batch.java:1521))*

*ERROR - javax.xml.ws.WebServiceException: Failed to access the WSDL at:  
https://www.ifremer.fr/SdnAuthService/SdnAuthService?WSDL. It failed with:*

*sun.security.validator.ValidatorException: PKIX path building failed:  
sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to  
requested target. (dm.DM\_Batch.process(DM\_Batch.java:1522))*

**Solution:** *Try to update the jssecacerts certificate keystore file. See §Erreur ! Source du renvoi introuvable., « Update SdnAuthService SSL certificate »*

### 3.4. Software version issue

*Problem: When calling the trigger page .../dm/controller, I get the error message:*

*java.lang.NoClassDefFoundError: Could not initialize class com.sun.xml.ws.spi.ProviderImpl*

*Solution: If you're using JDK 1.6, please make sure you're using version 1.7 or higher.*

### 3.5. Note about DM\_Checker memory allocation

The DM\_Checker tool has been tested successfully with a coupling table of 115890 rows (stored both in a file and in a database) and 256 Mb allocated for the JVM.

If an error of kind "java.OutOfMemory" occurs during the DM\_Checker execution, try to increase the number of Mb allocated for the Java process. Modify the value of the parameter -Xmx256m to -Xmx512m for example.