XML import manual for the European Pollutant Release and Transfer Register

Documentation

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **JOB NUMBER:** | | | **DOCUMENT REF:** | | | |
| **Version** | **Purpose of issue** | **Originated** | **Checked** | **Reviewed** | **Authorised** | **Date** |
| 1.0 | Initial Draft | Martin Holleufer-Sørensen |  |  |  | 25.05 2009 |
| 1.1 | Access import modified | Wolfgang Starzer |  |  |  | 03.05.2011 |
| 1.2 | Kml process implemented | Morten Hjelmsmark |  |  |  | 03.05.2011 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **ATKINS DANMARK** | | | | | | |

Table of Contents

[1 Summary 1](#_Toc266978299)

[2 Setup of the xml import environment 2](#_Toc266978300)

[3 Prepare the import of xml files 3](#_Toc266978301)

[4 Import xml file to eprtrxml database and validate data 5](#_Toc266978302)

[5 Copy data to eprtrmaster database 7](#_Toc266978303)

[6 Postprocessing imported data 8](#_Toc266978304)

# Summary

This document explains all steps needed to import xml files uploaded by the member states to the E-PRTR system. The xml files containing E-PRTR data are uploaded to the Central Data Register (CDR) at EEA by the member states. Initially these data will be imported and validated in a temporary database (EPRTRxml) and afterwards the data will be moved to the E-PRTR master database making the data ready for review at the Review site. Atkins will handle the import and validation and will supply the member states with a feedback document in the member states envelope at CDR.

### Prerequisites:

* SQL Server 2008 Native Client
* Microsoft .NET Framework 3.5 SP1
* XML\_Import program

### Parameters:

Global parameters:

|  |  |
| --- | --- |
| **Name** | **Value** |
| {SVN} | http://svn.eionet.europa.eu/repositories/ EPRTR /trunk |
| {SVN-Deploy} | http://svn.eionet.europa.eu/repositories/EPRTR/trunk/Deployment/ |
| {EPRTR} | C:\ |
| {SQLScripts} | C:\EIONET\SQLScripts |
| {MAPFORCE} | C:\XML-Import\MAPFORCE |
| {XML\_DOWNLOAD} | C:\ XML-Import \XML\_DOWNLOAD |
| {XML\_VALIDATE} | C:\ XML-Import \XML\_VALIDATE |
| {XML\_IMPORT} | C:\ XML-Import \XML\_IMPORT |

Environment specific parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Build (Atkins)** | **Staging (Atkins)** | **Test/Prod (EEA)** |
| {DB-SERVER} | Sdkcga6302 | Sdkcga6306 | TETRASQL |
| {DB-SERVER-PHYSICAL} | Sdkcga6302 | Sdkcga6306 | TETRASQL |
| {DB-USER} | sa | gis | gis |
|  |  |  |  |

# Setup of the xml import environment

Download the file {SVN-Deploy}/XML-import/EPRTR.zip and unzip it in the {EPRTR}-folder. Download the files in {SVN}/SQLScripts and place them in the {SQLScripts}-folder.

# Prepare the import of xml files

All member states are asked to upload their data to the CDR. At the CDR the member states create envelopes for each one of their obligations. For some obligations (e.g. the Pollutant Release and Transfer Report) it is possible to create several envelopes/upload data several times. In order to get hold of the right version of data Atkins need to log into CDR and manually seek out the latest envelope with regards to the Pollutant Release and Transfer Report obligation. The web site address is:

<http://cdr.eionet.europa.eu/recent_etc?RA_ID=538>

When the right envelopes have been identified please make sure that all the validations have been passed. The XML schema validation and the E-PRTR Compliance validation need to be passed without errors of any kind. In case of errors the data will not be imported and the member state need to resubmit a new version of the E-PRTR report. The Additional validation is just additional information to the member states about special data in their report that they need to be aware of. This additional information is not preventing the data from being imported. When the validations have been checked to see that they were passed please follow the below steps to prepare the import.

1. Download the xml file containing the PRTR-report from the CDR system to {XML\_DOWNLOAD}. Prefix the name of the file with the member state initials. E.g. the xml file from Denmark will be prefixed with "DK\_". This is done in order to keep track of which member state the file belongs to.
2. Identify the following information from the CDR system (some of the information can be obtained by adding “/xml” to the URL of the envelope:
   * CdrURL for the envelope (e.g. http://cdrtest.eionet.europa.eu/az/eu/colsglhra/envshfmmq)
   * CdrUploaded (Date and time the file was uploaded e.g 2009-05-15 12:05:00)
   * CdrReleased (Date and time the envelope was released e.g 2009-05-18 13:56:11)
   * Resubmit reason (in some cases the member states attach documents (.doc/.txt) describing why the report was resubmitted).
3. Prepare the above information in a batch script named e.g. Slovenia\_2007\_imp.bat The batch script need to look like the script below:

echo off

echo Importing Slovenia

echo -----------------------------------------------

SET basedir=C:\EIONET\SQLScripts

echo Recreating EPRTRxml database

call %basedir%\recreate\_EPRTRxml.bat

echo Importing xml file into EPRTRxml database

call C:\XML-import\MAPFORCE\EPRTR\_Import\_CMD.exe "C:\XML-import\XML\_DOWNLOAD\SI\_2009-9-03\_PRTR\_Report\_re\_subimission.xml" %SQLCMDSERVER% %SQLCMDUSER% %SQLCMDPASSWORD%

echo ----------------------------------------------

echo Validating data

SET SQLCMDDBNAME=EPRTRxml

sqlcmd -Q "EXEC EPRTRxml.dbo.validate\_xml\_data"

echo -----------------------------------------------

echo Copying data from xml to master

rem sqlcmd -i %basedir%\reset\_cols\_4\_xmlimport.sql

rem sqlcmd -Q "EXEC EPRTRxml.dbo.import\_xml @pCDRURL = N'http://cdr.eionet.europa.eu/si/eu/eprtrdat/envsp95uq',@pCDRUploaded = N'2009-09-03 10:08:00', @pCDRReleased=N'2009-09-03 10:09:00', @pResubmitReason=''Data has been corrected;"

echo -----------------------------------------------

1. Place the batch script in the folder {XML\_VALIDATE}

# Import xml file to eprtrxml database and validate data

Besides the validation executed in CDR it is also important that the downloaded xml-files are validated manually with regards to which reporting year the xml file concerns and which xsd schema the member state have been using when creating the xml file.

The reporting year <rsm:ReportingYear>2007</rsm:ReportingYear> in the PollutantReleaseAndTransferReport element of the xml file need to correspond to the reporting year which is handled. No other reporting years are accepted at the moment. Later on previous reporting years will be accepted but no future reporting years.

The xsd schema version is at the moment 2.0 so the header of the xml file needs to read:

<rsm:PollutantReleaseAndTransferReport xsi:schemaLocation="urn:eu:com:env:prtr:data:standard:2 http://www.eionet.europa.eu/schemas/eprtr/**PollutantReleaseAndTransferReport\_2p0.xsd**" xmlns:wgs84="urn:convertKKJ" xmlns:rsm="urn:eu:com:env:prtr:data:standard:2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

If this is not true the report is not accepted and the member state will be notified about the error in the feedback report.

When the above manual validation has been carried out a database validation will be executed as described below.

When all member states batch files are in place in the {XML\_VALIDATE} folder please execute {XML\_VALIDATE}\1\_Validate\_XML\_Files.bat

This script executes all batch files and generates log files which tell about the import to the temporary xml database and the validation of data (e.g. slovenia\_2007\_imp.bat.log). The information from the import and validation log needs to be given as feedback to the member state in the CDR system. Furthermore the following spreadsheet needs to be updated with status of the import and validation on the member states. The spreadsheet gives an overview of the quality of the delivered reports from the member states.

<http://svn.eionet.europa.eu/repositories/DotNet/Eprtr/trunk/Deployment/XML-import/Status_EPRTR_Data_Import.xls>

So far the validation routine validates data for the following issues:

1. Validating previous nationalids and previous reporting year given in xml file
2. Validating if a facility in a previous report is referenced more than once in the new report
3. Validating if nationalids and previous nationalids are the same for new facilities
4. Validating if previous nationalid return multiple records
5. Validating if reported coordinates are within country polygons
6. Validating if reported coordinates are set to zero or not reported

Based on the result of the import and validation log it needs to be decided whether or not data is going to be imported to the EPRTRMASTER database. Copy the batch scripts for the countries with valid data to the {XML\_IMPORT} folder and uncomment the following lines:

sqlcmd -i %basedir%\reset\_cols\_4\_xmlimport.sql

sqlcmd -Q "EXEC EPRTRxml.dbo.import\_xml @pCDRURL = N'http://cdr.eionet.europa.eu/si/eu/eprtrdat/envsp95uq',@pCDRUploaded = N'2009-09-03 10:08:00', @pCDRReleased=N'2009-09-03 10:09:00', @pResubmitReason=''Data has been corrected;"

# Copy data to eprtrmaster database

When all relevant member states batch files are in place and uncommented please execute {XML\_IMPORT}\1\_Import\_XML\_Files.bat

The following attributes are set automaticly during the import-procedure based on the information entered in the batch files:

* POLLUTANTRELEASEANDTRANSFERREPORT.CDRURL 🡪 URL of envelope in CDR
* POLLUTANTRELEASEANDTRANSFERREPORT.CDRUPLOADED 🡪 Date and time when the xml file was uploaded
* POLLUTANTRELEASEANDTRANSFERREPORT.CDRRELEASED 🡪 Date and time when the envelope in CDR was released
* POLLUTANTRELEASEANDTRANSFERREPORT.RESUBMITREASON 🡪 Voluntary description from the envelope in CDR
* FACILITYREPORT.LOV\_RIVERBASINDISTRICTID 🡪 RiverBasinDistrictID of RiverBasinDistrict based on spatial analysis
* FACILITYREPORT.LOV\_NUTSRegionID 🡪 NUTSRegionID of NUTSRegion based on spatial analysis
* FACILITYREPORT.LOV\_STATUS 🡪 Valid, Outside or Missing with regards to the coordinates of the facility

Besides the above attributes a couple of default values that are not required in the xml file are set during xml import:

* POLLUTANTRELEASE.LOV\_TOTALQUANTITYUNITID is set to 2 which refers to KGM
* POLLUTANTRELEASE.LOV\_ACCIDENTALQUANTITYUNITID is set to 2 which refers to KGM POLLUTANTTRANSFER.LOV\_QUANTITYUNITID is set to 2 which refers to KGM
* WASTETRANSFER.LOV\_QUANTITYUNITID is set to 3 which refers to TNE

# Postprocessing imported data

## Copy data to Review Site

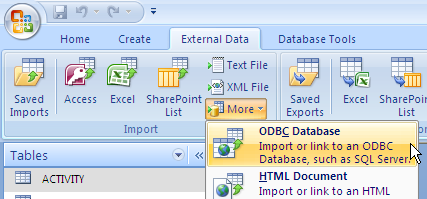
In order to copy the imported data to the Review Site download the file copyEPRTRmasterViews2EPRTRwebTables.sql from {SVN}/SQLScripts. Please replace every instance of the word EPRTRweb with EPRTRreview in the script. Also make sure that the "set Published date" part of the script is commented out and the "set ForReview date" part of the script will be executed. When everything is ready execute the script in a sql-editor.

## Copy data to Public Site

In order to copy the imported data to the Public Site download the file copyEPRTRmasterViews2EPRTRwebTables.sql from {SVN}/SQLScripts. Please replace every instance of the word EPRTRweb with EPRTRpublic in the script. Also make sure that the "set ForReview date" part of the script is commented out and the "set Published date" part of the script will be executed. When everything is ready execute the script in a sql-editor.

## Export data to Full and Public Access databases

In order to export E-PRTR data to Access databases used by ETC and EEA dataservice download the two Access databases found in {SVN}/Export2Access and follow the procedure given below:

1. Open up the Access database
2. Open up a ODBC connection to the EPRTRmaster database
3. Select “Import the source data into a new table.....” and click OK
4. Click New (Data Source), choose SQL Server Native Client 10.0 or SQL Server and click Next. The following steps 5-8 only has to be done once. Next time just select the Data Source already set up.
5. Enter a name for the new Data Source (e.g. tetrasql) and click Next and click Finish
6. Enter the appropriate Server (e.g. TETRASQL) and click Next
7. Choose SQL Server authentication and type in login ID (e.g. gis) and password (e.g XXXXXX) and click Next
8. Change the default database to EPRTRmaster and click Next, Finish, Test Data Source, and OK
9. Now select the Data Source that was just created and click OK. Type in password and click OK
10. If the Access database opened is the Full\_Database.mdb select all views prefixed with FULL (e.g. dbo.FULL\_ACTIVITY) (12 views in all) and click OK. If the Access database opened is the Public\_Database.mdb select all views prefixed with PUBLISH (e.g. dbo.PUBLISH\_ACTIVITY).
11. Run Macro FINISH\_DATABASE (calls macro RENAME\_TABLES, Add\_Primary\_Keys and Add\_Foreign\_Keys)
12. Change manually name of table dbo\_FULL\_UPLOADEDREPORTS to UPLOADEDREPORTS.
13. Save the database and exit Access
14. Change the name of the database to Full\_Database\_YYY-MM-DD, respective Public\_Database\_YYYY-MM-DD.
15. Zip the databases and put them on EEA ftp server (<ftp://ftp.eea.europa.eu/Ose3%20GIS/EPRTR/>) username: eeaftp

An outdated version of the documentation of the two databases can be found in {SVN}/documents/TechnicalSpecification/, a newer version is published by EEA on: <http://www.eea.europa.eu/data-and-maps/data/member-states-reporting-art-7-under-the-european-pollutant-release-and-transfer-register-e-prtr-regulation-3>

# Creating KMZ file

This is the procedure to follow when a kml file has to be recreated.

The scripts etc. can be found on *svn.eionet.europa.eu/repositories/EPRTR/trunk/Eprtr\_kml\_solution*, or in the *C:\GISLibrary\Python\EPRTR\_KML* folder on *EEAPC8662.*

If new data is issued for the same year as the previous run, it should be enough just to run the script again. The script can be run by double clicking the **Create\_EPRTR\_KML.bat** file located in the Script folder (on EEAPC8662 it located in C:\GISLibrary\Python\EPRTR\_KML\Scripts).

If the new data doesn’t show in GoogleEart or if data is covering a new year, you need to update the base layer file. Locate the Layer file EPRTR\_style\_basic\_reportid.lyr (on EEAPC8662 it located in C:\GISLibrary \Python\EPRTR\_KML\Data) and open it in ArcMap.

* Update the data source
* Make sure the joins still works
* In the definition query tab, update query to the correct reportingyear
* In the fields tab reset the list of implemeted fields – remember to **Apply**!

|  |  |
| --- | --- |
| **Layer name** | **Alias** |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.FacilityReportID | FacilityReportID |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.FacilityName | FacilityName |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.ReportingYear | ReportingYear |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.Address | Address |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.City | City |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.PostalCode | PostalCode |
| EPRTRPUBLIC.SDE.FACILITYSEARCH\_MAINACTIVITY\_GEOGRAPHICALCOORDINATE.IASectorCode | IASectorCode |
| EPRTRPublic.dbo.LOV\_COUNTRY.Name | Country |
| EPRTRPublic.dbo.LOV\_ANNEXIACTIVITY.Code | Sector code |
| EPRTRPublic.dbo.LOV\_ANNEXIACTIVITY.Name | Sector |

* When done, save the layer by using ’Save As Layer file..’ and save the lyr file in the data folder

After the layer file is updated, make sure the ..\scripts\ EPER\_fill\_mxd.py points at it.

#Path to the base lyr file

inpPth = solPth + "\Data\EPRTR\_style\_basic\_reportid.lyr"

Then run the **Create\_EPRTR\_KML.bat** file located in the Script folder (on EEAPC8662 it located in C:\GISLibrary\Python\EPRTR\_KML\Scripts).

For more information read the document **EPRTR kml solution documentation.docx**, which can be found in the svn repository *svn.eionet.europa.eu/repositories/EPRTR/trunk/Eprtr\_kml\_solution*