OpenClinica Data Importer
Software Design Document and User Guide

Document History

Version	Date	Description	Who
1.0	21/05/2010	Initial version	C. Parlayan, J.A.M. Beliën
1.1	21/04/2012	Production version	C. Parlayan
2.0	08/11/2012	Production 2.0	C. Parlayan
2.0.1	01/02/2013	Added Repeating events and groups	C. Parlayan
2.0.2	26/03/2013	The GRID now keeps the previously matched items when a new CRF or Group is chosen for more matching.	C.Parlayan
2.1.1	09/09/2013	Introduced label-oid file.	C. Parlayan, J. Rousseau, R. Voorham
2.2	30/11/2013	Input file allows EVENT_INDEX and GROUP_INDEX to be defined to accept Repeating events and items in rows.	C. Parlayan, J. Rousseau
3.0	01/11/2013	Type and range validations.	C. Parlayan, J. Rousseau
4.0	29-01-2014	Better UI	C. Parlayan
4.3.1	03-04-2014	Allow date of year instead of date of birth	C. Parlayan
4.3.1	03-04-2014	SITE_OID column introduced to allow defining site for subjects	C. Parlayan, J. Rousseau
4.3.1	04-04-2014	Names is dropdows's beside OID's	C. Parlayan, S. de Ridder
4.3.5	23-04-2014	culture info added for comparing floats	C. Parlayan, J. Rousseau
4.3.10	25-07-2014	Split factor = 75 instead of 0	C. Parlayan
4.3.10	23-07-2014	Create records in TDS database for web services	C. Parlayan
4.4	08-09-2014	Web services integration with TrialDataSolution's web database	C. Parlayan, G.R.Visser

Contents

1 INTRODUCTION

1.1 Product Identification

This document contains the Software Design Specifications for the OpenClinica generic data importer.

1.2 Purpose of the Document

The purpose of this document is to provide the detailed software design and the user guide.

1.3 Scope of the Document

The scope of the body of this document is to describe the implementation of "OCDataImporter" program. Other aspects related to SPSS, MS Excel and OpenClinica can be found in SPSS, Excel and OpenClinica documents.

1.4 Intended Audience

Open Clinica users, data managers.

1.5 References

Microsoft .NET documentation, OpenClinica documents, PostGreSQL documents.

2 GENERAL DESCRIPTION

2.1 Product Perspective

This product will make it possible to convert a study data file into an OpenClinica study.

2.2 Product Availability

The product is available in May 2010.

2.3 Principle Product Functionality

This application is based on reading a study file and converting it to an SQL and XML based set of files which can be used as input to Postgres and OpenClinica to create subjects, events and perform ODM upload.

2.4 General Constraints

This product requires Microsoft .NET Framework version 3.5 distributable Package.
It is included in Windows Vista and higher versions of Windows.

Otherwise, it is possible to download this from the following site:
http:msdn.microsoft.com/downloads/

The program is not yet available under other operating systems.

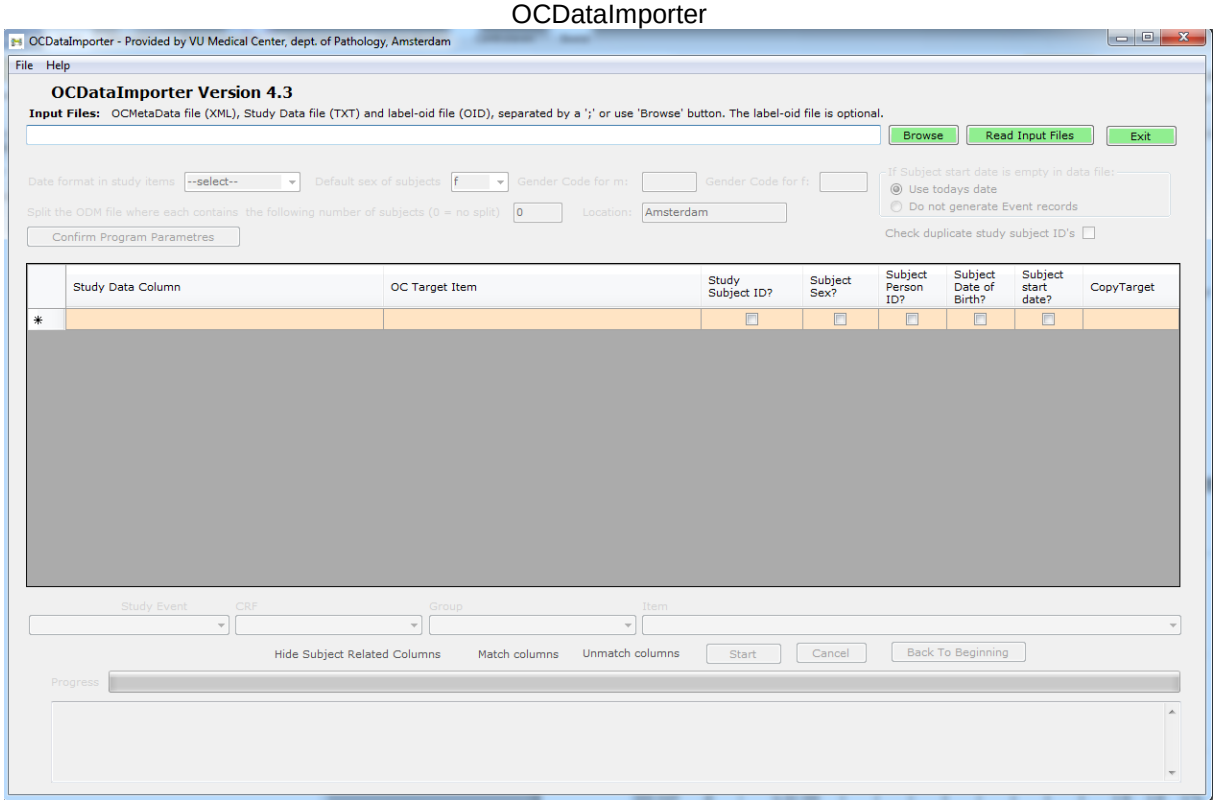
CAUTION: This program introduces a complex procedure of importing legacy data in Open Clinica. Usage of this program requires expertise level knowledge of Open Clinica and the Postgres database. Furthermore administrator privileges will be needed at the database server side in order to run the database scripts to create subjects and events. Please make sure you read this document carefully before you start using the application

3 USER GUIDE

This section describes how to use this application.

3.1 Installation

To install this program, use OCDataImporter.msi. Follow the instructions on screen.
When run, the following form should be seen:



On this screen, there are 3 steps defined: Read input files, define parameters and start conversion. The buttons of the stage that active is are green.

3.2 First step: Convert study to a tab delimited text file.

1- Create a directory, for example C:\DataImporter30_test\pr1.

2- Start SPSS, Excel or another program to convert the study file to a tab delimited text file, named for example “test-set-v01-UTF8.txt”.

3- Copy this file to C:\DataImporter30_test\pr1.

3.3 Create the Study, events and CRF(s) in OpenClinica.

To do this, start OpenClinica, login as Datamanager, Create a study, create the necessary events and CRF’s with items which matches with the items for the file generated in section 3.2. For more on using OpenClinica, refer to OpenClinica documents.

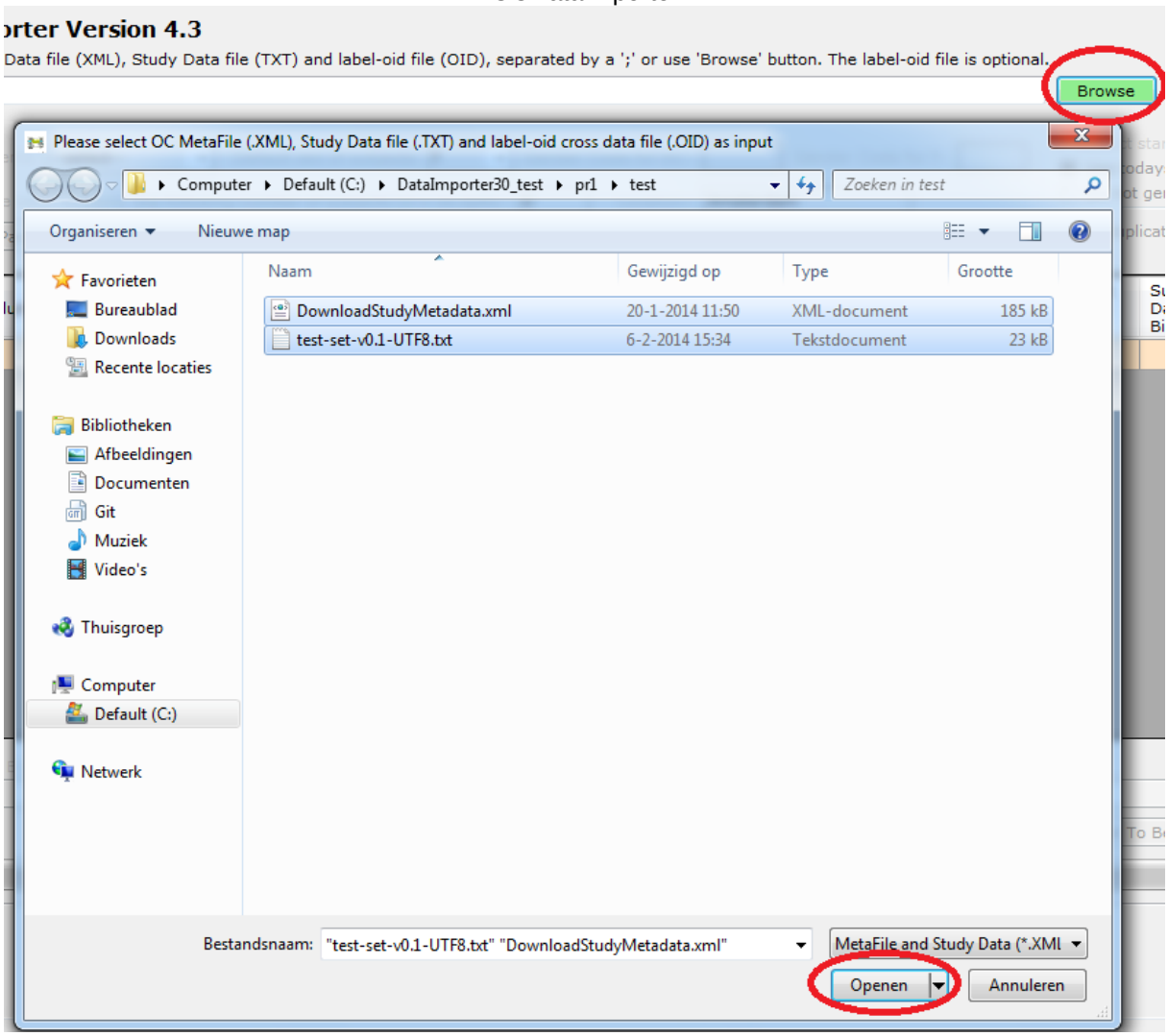
3.4 Create OpenClinica Meta Data File using OpenClinica.

Once the OpenClinica objects are created, generate the meta data file.
View the study and Click “here” of the sentence “Download all of the OID’s needed for data import and rules”. Give the file a name, for example “DownloadStudyMetadata.xml” and save the file in C:\DataImporter30_test\pr1.

3.5 Read the input files in “OCDataImporter”

Hit the “browse” button and select the files that are generated: The meta file and the tab delimited data file.

You must select both files, by pressing the ctrl key and clicking on both files.



Now hit the “read input files” button on OCDataImporter screen.

The program will check if the number of columns in the data file is the same with the items in the Meta file. If so, an automatic matching is done, but mostly there are more than one CRF’s defined for one patient data file so the column matching will be mostly done manually.

The ideal situation is, OC data items and the source file columns all have same names. For example if the source data column name is “Gender” and the OC item is also named “Gender” an automatic matching can be made be made.

3.5.1 Usage of label-oid translation file

Normally OpenClinica makes standard study subject oid’s like “SS_<subjectID>, for example “SS_1200” will be the study subject oid for a subject with ID = 1200. OCDataImporter assumes this procedure is followed and generates the insert statements (explained in later chapters) accordingly. If the inserts will be used for creating the study subjects, you will not need the label-oid translation table and skip reading this section.

This is however not the case if the study subjects are made with OC web services. If the subject id is longer than a certain number of characters, OC generates a study subject oid with some part of the subject id + sometimes a 4 digit number to make the id unique. Example:

Subject: COCOS_22100	OC generates the study subject id as: SS_COCOS_22_9563
Subject: COCOS_22101	OC generates the study subject id as: SS_COCOS_22_8260
Subject: 20220	OC generates the study subject id as: SS_20220

Etc.

In order to use OCDataImporter's CRF data ODM files with the study subjects generated by OC web services, a cross data file is needed to translate the subject id to study subject oid. This file should only contain the subject id's having a study subject oid **other than** SS_<subject id>, like the following:

Label	oid
COCOS_22100	SS_COCOS_22_9563
COCOS_22101	SS_COCOS_22_8260

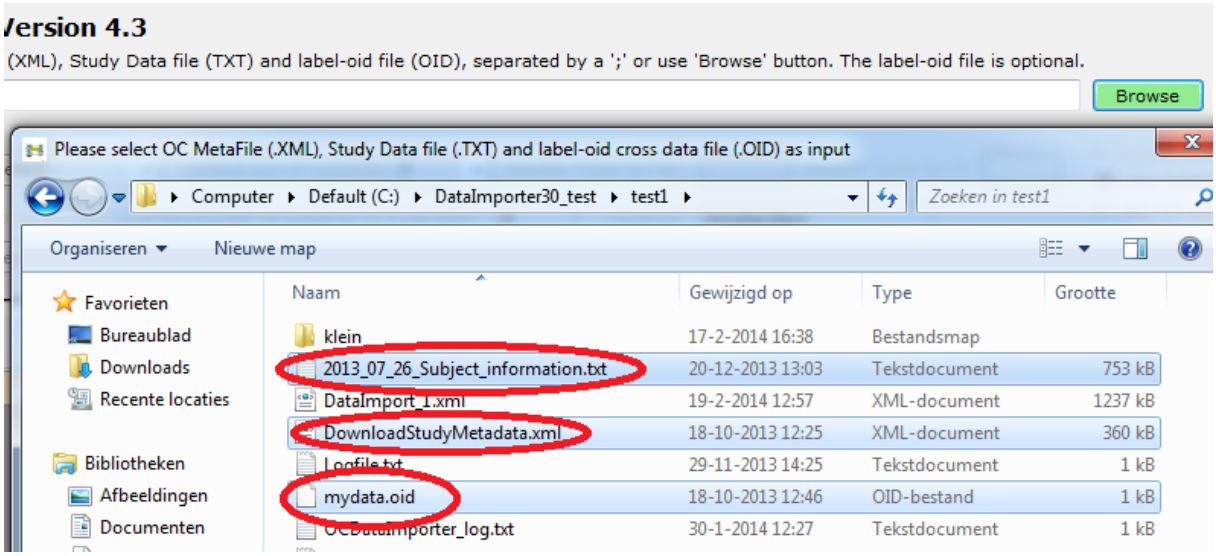
Note that 20220 is not included because this oid is conform the structure SS_<subject id>.

This file can be generated easily by using a postgres-query, similar to this:

```
psql -c "select label, oc_oid  from study_subject
where study_id in (5,6,7,8) and (oc_oid NOT LIKE 'SS_'  label)" -d openclinica
```

5,6,7,8 are the studies to include and oc_oid's are selected to be other than SS_<label>.

The label-oid file should be in the working directory, must have a .oid extention and can be selected as shown below:



3.5.2 Usage of SITE_OID column in data file

It is possible to indicate the site, if any, in the data file by adding a column called "SITE_OID" like in the following example:

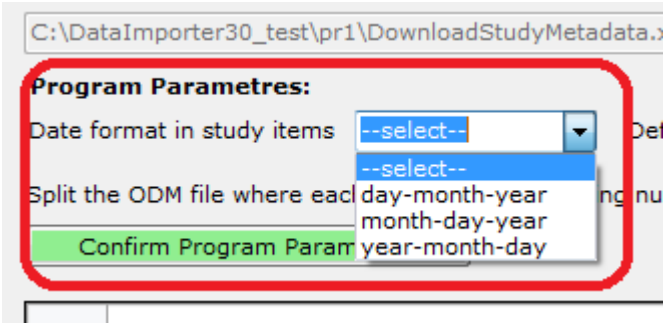
	A	B	C	D
1	Subj ID	DOB	Gender	SITE_OID
2	1	1957	1	S_MM13
3	2	1958	2	
4	3	1966	1	S_NSNSNS

In this case the subjects will be dedicated to the indicated site oid. If blank, study_oid will be assumed. If the site_oid doedn't exist, an error will be issued.

3.6 Program Parameters

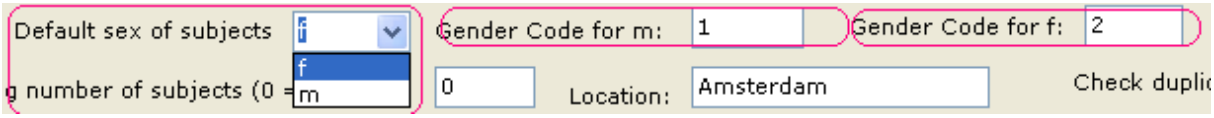
3.6.1 Specifying date format

You can optionally specify the format of dates in the data file by using the “Date format in study items” combo box. The dates in OC Import XML file ***must*** be in YYYY-MM-DD format. The program takes care of converting the dates from the format of your choice to OC format. If you do not want any conversion to take place, leave the selected item as “—select—“, but in this case either you must be sure that all dates are in ODM format YYYY-MM-DD or there are no dates at all in your data file.



3.6.2 Specifying gender codes

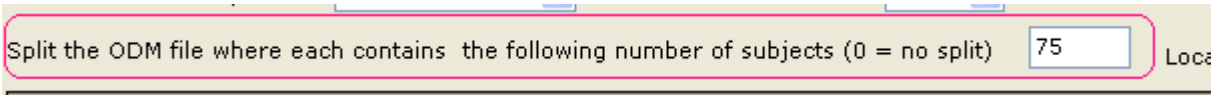
The codes for male and female in OpenClinica has to be “m” and “f” respectively. If this appears otherwise in your data file you can indicate this by using the appropriate textboxes as indicated below. If male is coded as 1 and female as 2 in your data file, below coding will make the proper translation.



If there is no gender code in your file and all participants are female, then you can leave the Gender code fields blank and select “f” as default sex of subjects. If the gender code fields are filled, default sex will be ignored.

3.6.3 Splitting the ODM file

Open Clinica may not be able to handle ODM files with more than 75 subject-data in one ODM file when the user interface is used for uploading. It is recommended to split the ODM file into pieces which will contain no more than a specified number of subject data. In version 3.1.2 it was able to handle 75 subject data in one file. When you enter 75, several files will be generated with 75 subject-data each, named “DataImport_1, …_2, …_3, etc.
Enter 0 if no splitting is desired. (We hope that there will be no splitting needed in the future versions)

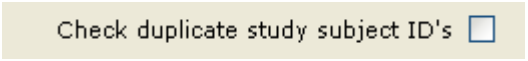


3.6.4 Specifying the location

It is possible to give the location name to be used in event subjects.

3.6.5 Specifying if duplicate key check is needed

If your data file contains only one row for each subject and the subject id's must be unique, this program can check that and issue an error message if that fails. Use the checkbox below, to perform this check.

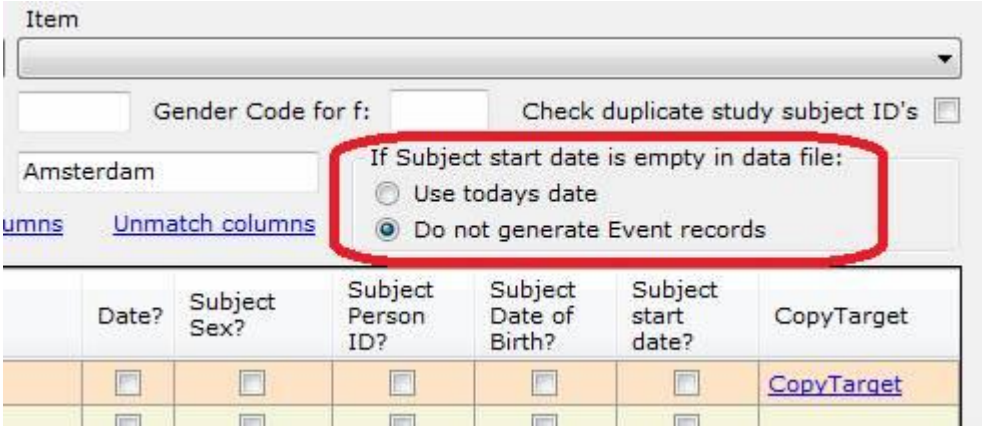


3.6.6 Options for events without staring dates

Use todays date: The event start date will be assumed todays date. This way it is possible to add data for that event.

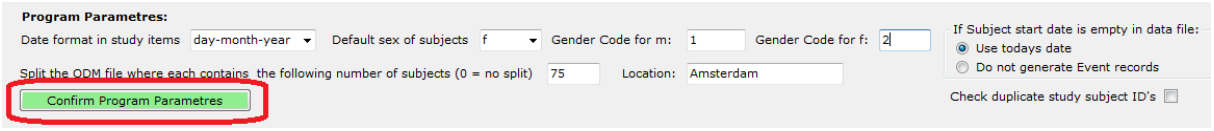
Do not generate Event records: This is the logical choice, since data addition to an event without start date is of no use.

But if the event has to be generated and the data must be added, choose "use todays date".



3.6.7 Confirming Program Parameters

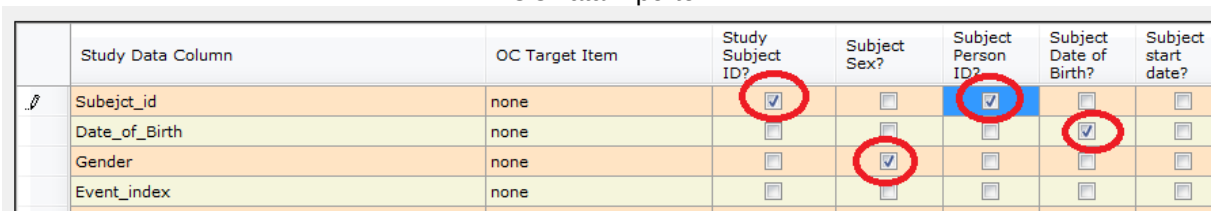
When all of the parameters are defined as explained in section 3.6, hit the "Confirm" button to proceed to next step.



3.7 Indicating study subject id, Subject sex, subject person's id, subject start date

The program needs to know which data column is the subject id. Without a subject id, the process can't be made. So there has to be one (and only one) subject id checked in the whole grid. This will be used to create subject records and relate the CRF data with this subject.

There are also other items needed to create the subject like subject sex, subject person's id, subject date of birth and subject start date. These can be indicated with the related checkboxes as shown below.

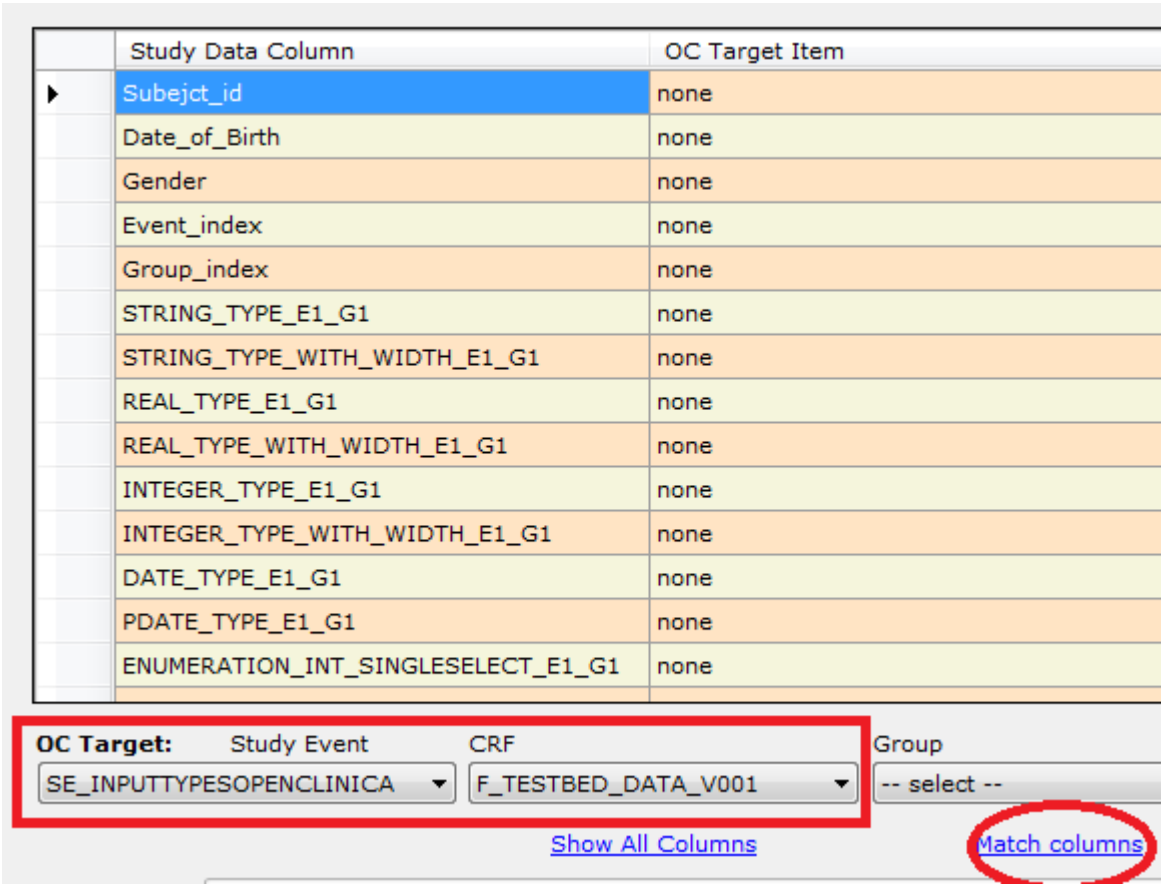


This will be used in insert statements to create study, event_study and event_crf records and is described later in this manual.

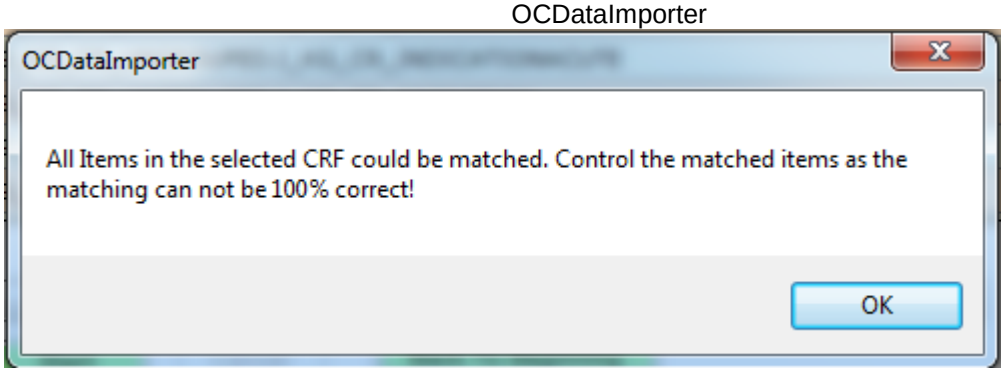
Once defined, the columns “Study Subject ID?”, “Subject Sex?”, “Subject Person ID?”, “Subject Date of Birth?” and “Subject Start Date?” can be hidden (in order to make space in grid to display other data) by using link button:

3.8 Matching data columns and OpenClinica Items

To match the columns in case there are more than one study event and/or CRF’s, you can choose the event and the CRF and then hit “Match columns” link. The Group and Items combobox should be left as –select--. This will check for similar names and match them.

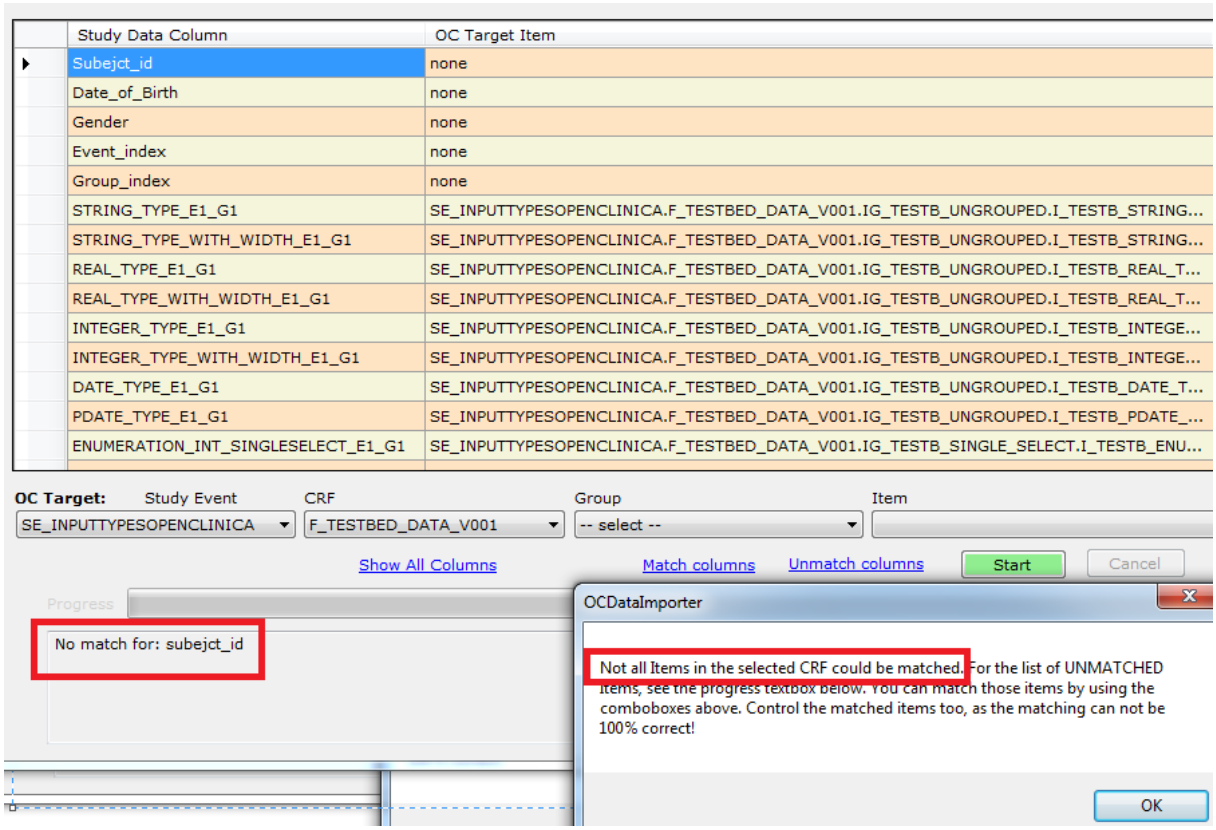


This can be done for all CRF’s separately, so gradually most of the items will be matched. If all columns are matched, program displays:



Even if this is the case, it is recommended to control if the matching is correct.

If the names are not similar or the OC item does not exist in the selected CRF (because it exists in another CRF or even doesn't exist at all) matching can also be done manually. The program will issue a message indicating this. To see the items that are NOT matched, see the Progress textbox as shown below:

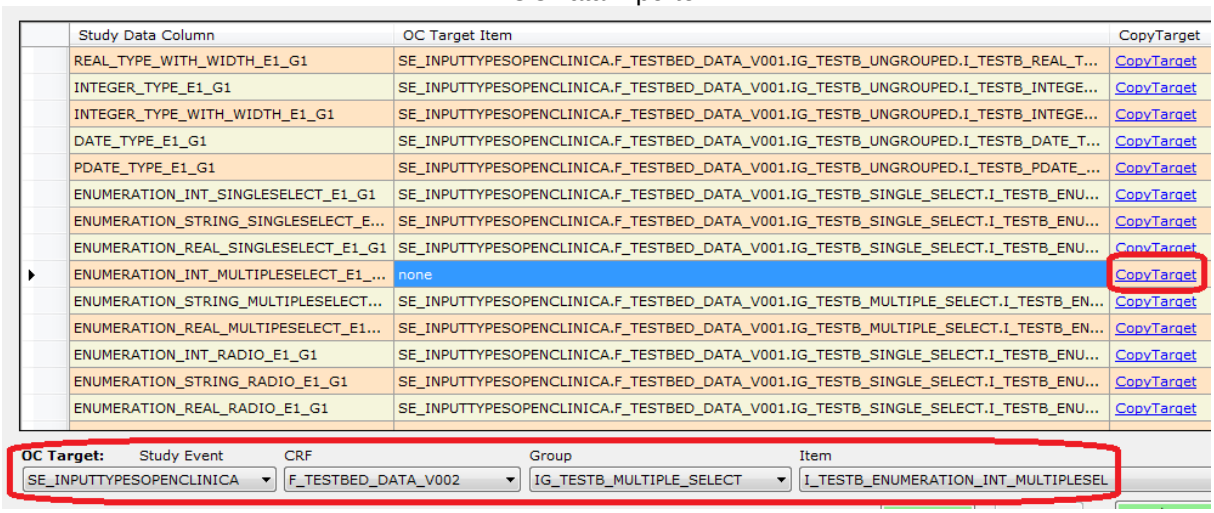


When no matching could be made, the program will display the "study data columns" of the grid, but it will leave the "OC Target Item" blank or fill it as "none". This can be filled in by using the "CopyTarget" link. To use this link, first chose the target item using the Study event, CRF, Group and Items combo boxes, then hit the CopyTarget of the row which has to be matched with that item.

Suppose you want to match "ENUMERATION_INT_MULTIPLESELECT" with

"SE_INPUTTYPESOPENCLINICA.F_TESTBED_DATA_V002.IG_TESTB_MULTIPLE_SELECT.I_TESTB_ENUMERATION_INT_MULTIPLESEL"

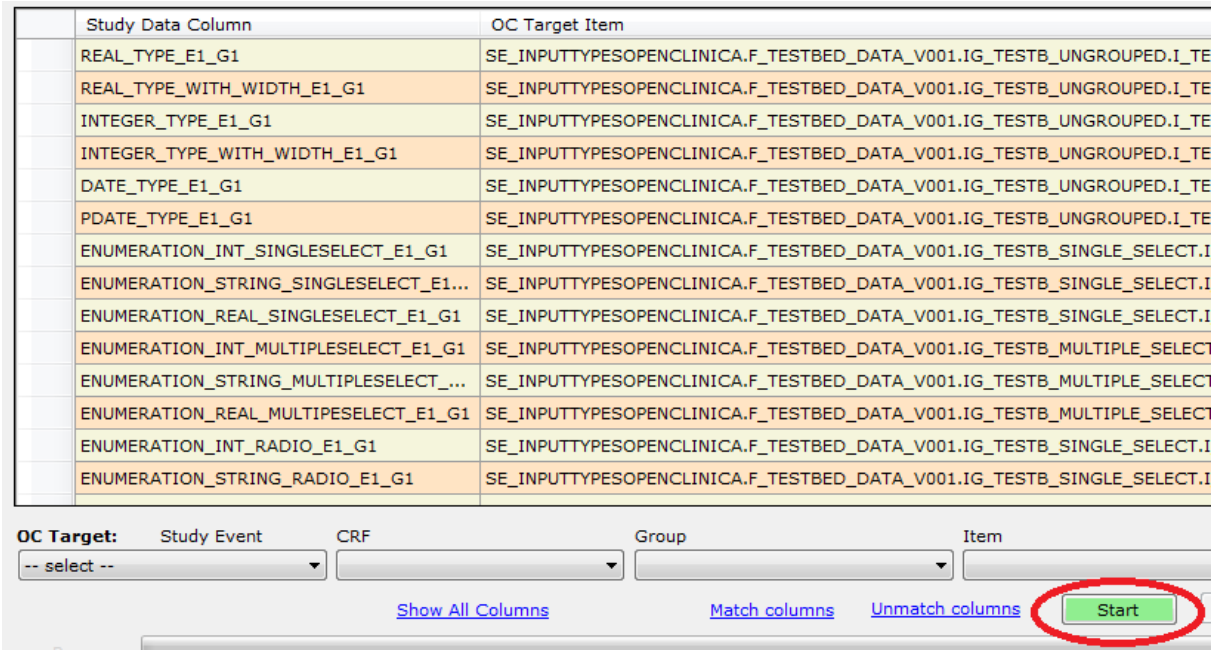
Use the combo boxes so that the above item appears in the combo boxes and hit "CopyTarget" of the row of "ENUMERATION_INT_MULTIPLESELECT".



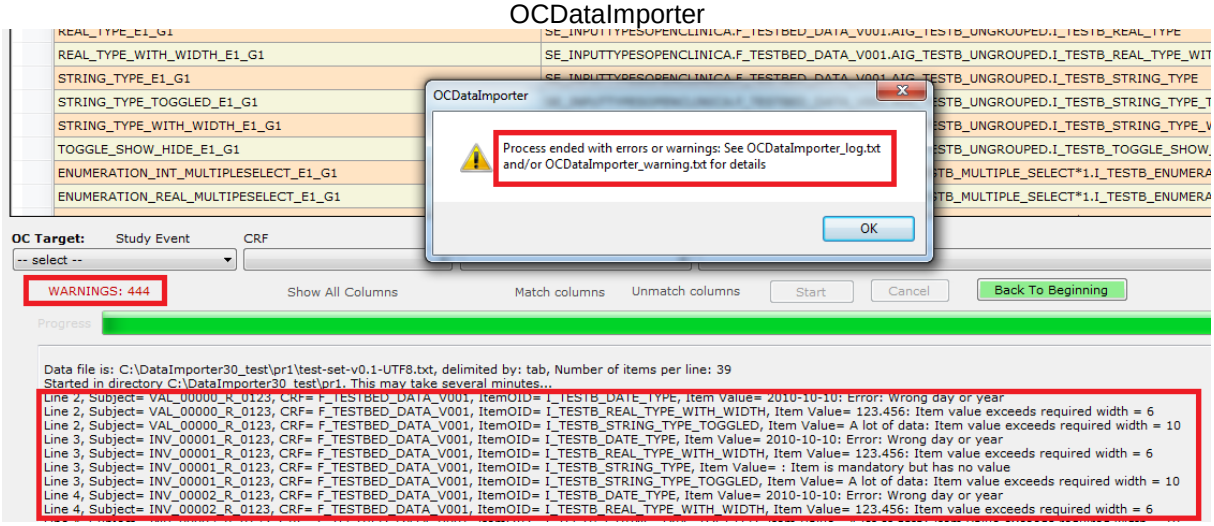
The program will copy the target item into the “OC target Item” field.

3.9 Starting the conversion

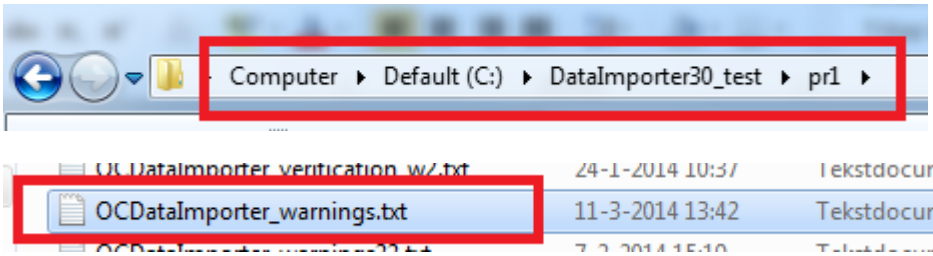
After matching columns, the program is ready to process the data and generate the files needed to upload to OpenClinica. Hit the start button, as shown below.



This action can result like the following:

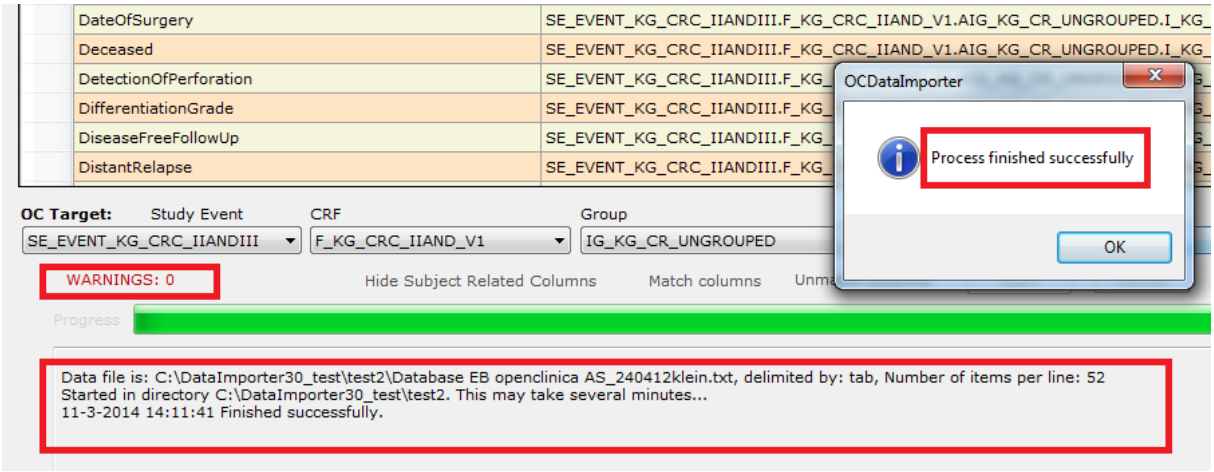


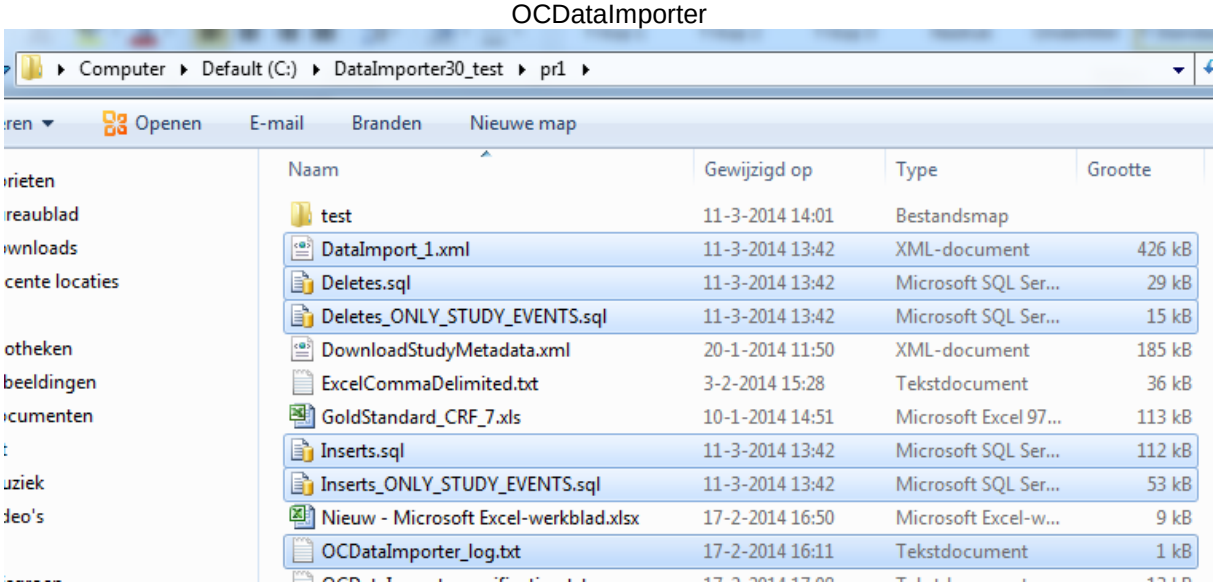
In this run, there are 444 errors detected, for example “Item Value= 123.456: Item value exceeds required width = 6”. These warnings can be found in the file called “OCDataImporter_warnings.txt” in the working directory.



Each warning/error should be corrected before upload as they will cause errors in OpenClinica. See appendix for the list of possible things that can go wrong.

When the process ends with no errors or warnings, the program will issue a “success” message and the output files will be seen in the working directory:





3.10 Creating subjects, study_subjects and study_events in PostGreSQL

3.10.1 Using "Inserts.sql".

The file "Inserts.sql" will be used to create subject, study_subject and sudy_event rows in PostGres. It contains insert statements for those three tables.

For example, for the first line of the study data:

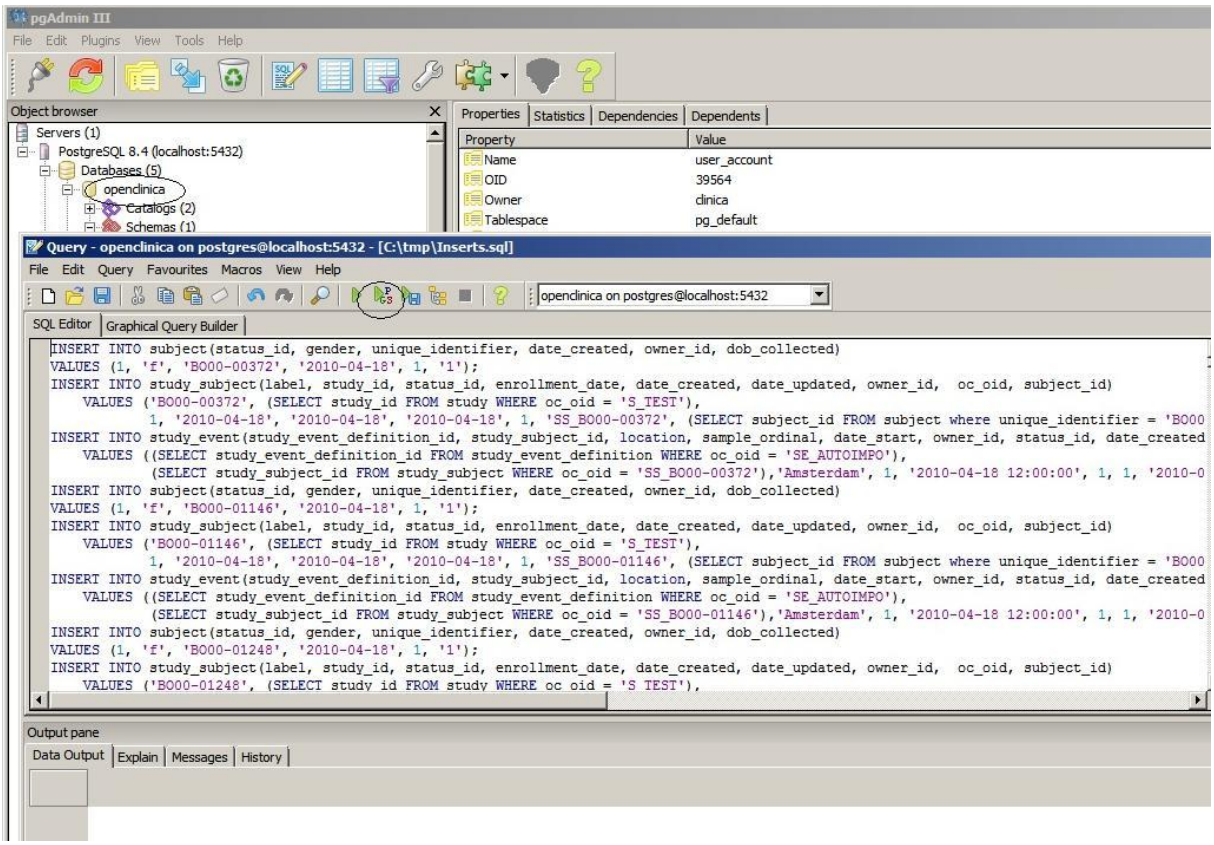
```
NIJM_CAIRO_CRC_002          0                                   No distant metastasis, number of affected
organs is 0 (array CGH data is available)                       0       5-mrt-1942      1       1       1
        13-jan-2003     60              1       4       -1              -1      1       -1
        -1              -1      0       1       3       0       -1      -1      -1      -1      0
        -3      -1      -1      -1      0               1       1       0       18              -1
        1       435             -1      1       0               -1      -1      435             -1
        1       0               -1      -1      435             -1      1               1454    1
        1       1       0
```

The following will be created:

```
INSERT INTO subject(status_id, gender, unique_identifier, date_created, owner_id, dob_collected,
date_of_birth)
   VALUES (1, 'm', 'NIJM_CAIRO_CRC_002', '2012-11-05', 1, '1', '1942-03-05');
INSERT INTO study_subject(label, study_id, status_id, enrollment_date, date_created, date_updated,
owner_id,  oc_oid, subject_id)
   VALUES ('NIJM_CAIRO_CRC_002', (SELECT study_id FROM study WHERE oc_oid =
'S_CAIRO'),
          1, '2012-11-05', '2012-11-05', '2012-11-05', 1, 'SS_NIJM_CAIRO_CRC_002', (SELECT
subject_id FROM subject where unique_identifier = 'NIJM_CAIRO_CRC_002'));
INSERT INTO study_event(study_event_definition_id, study_subject_id, location, sample_ordinal,
date_start, owner_id, status_id, date_created, subject_event_status_id, start_time_flag,
end_time_flag)
   VALUES ((SELECT study_event_definition_id FROM study_event_definition WHERE oc_oid =
'SE_CAIRO_EVENT'),
            (SELECT study_subject_id FROM study_subject WHERE oc_oid =
'SS_NIJM_CAIRO_CRC_002'),'Amsterdam', 1, '2012-11-05 12:00:00', 1, 1, '2012-11-05', 3, '0', '0');
```

To run this SQL file, start PGAdmin at the OpenClinica database server, select the openclinica database, start Query tool from "Tools" pulldown menu, read "Inserts.sql" file and run this with "execute pgscript". See Postgres user manuals for more information on how this is made, if

necessary. **CAUTION: Make sure the OpenClinica instance is NOT running at this time and don't forget to make a full backup of the database before you start!**



When this is done, proceed with the following section.
Note: Deletes.sql can be used to undo the above operation.

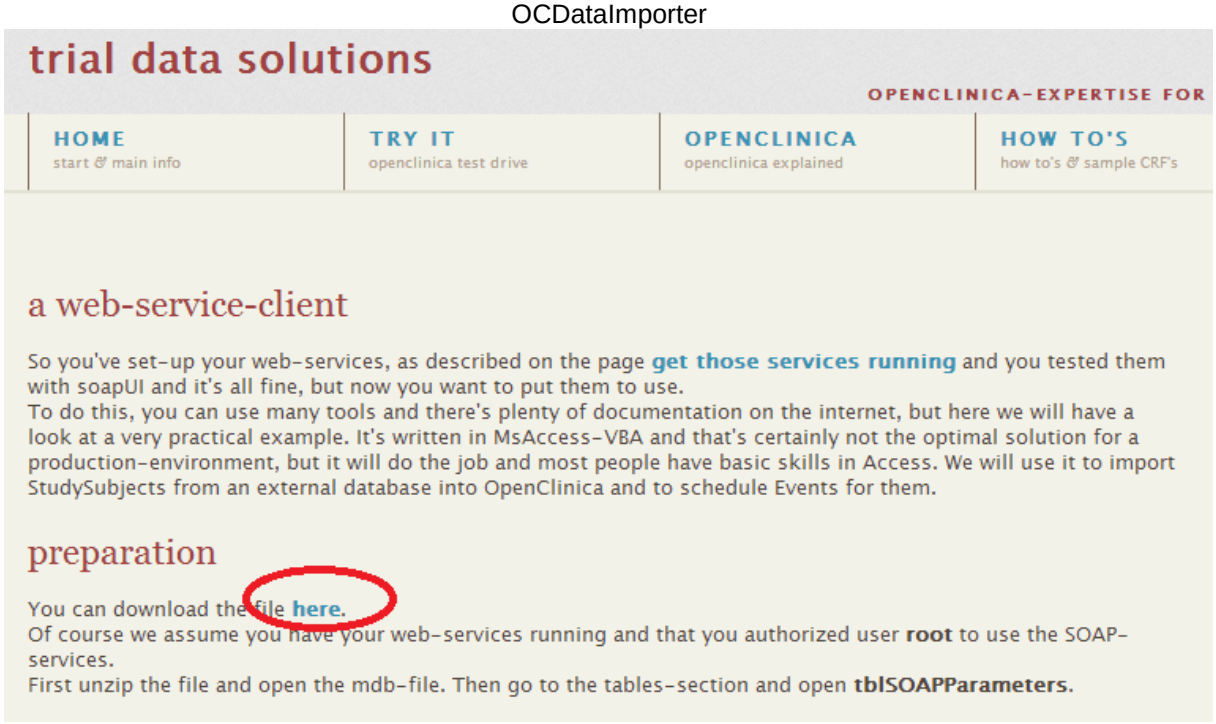
3.10.2 Using Trial Data Solutions' database in combination with OC web services.

The access database of trial data solutions introduce an easy way of using web services in OpenClinica. The documentation and the software can be found here:

http:www.trialdatasolutions.com/tds/howto/wsclient.jsp

As of version 4.3.10, it is possible to create study subject records and schedule events by letting the OCDataImporter populating 2 of the tables of this database. This is a safer and easier approach than using "inserts" statements directly at postgres server. This section explains how this can be done.

1- Read the information given in the above link and download the web service access database.



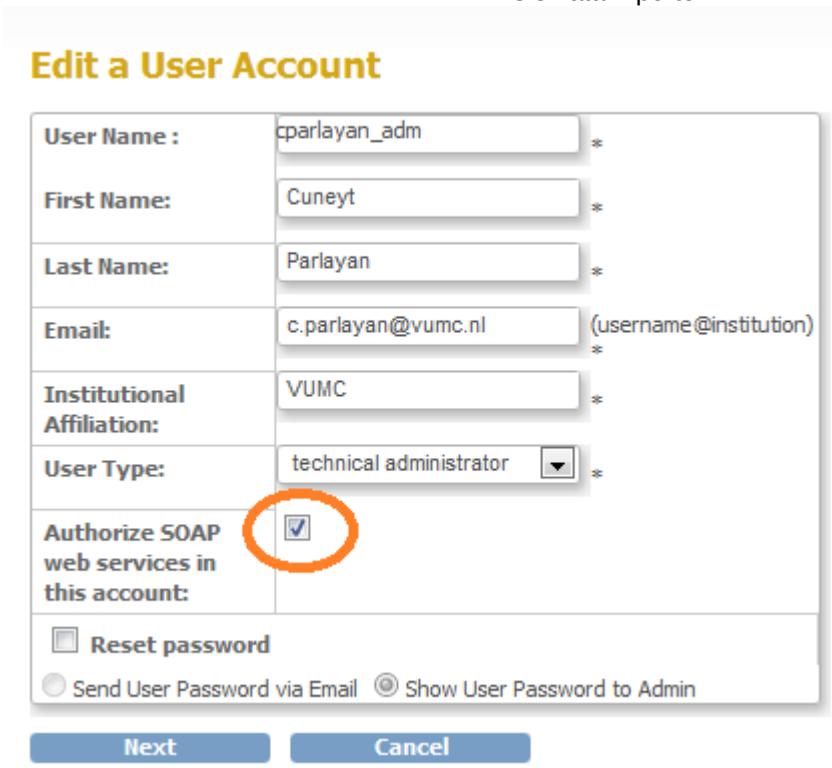
Make a backup of the OC postgres database (or ask your system administrator to do so) and make a copy of the clean ws access database you have just downloaded.

2- After downloading the access database, move or copy it to OCDataImporter working directory and make sure you have write permissions on the database.

3- Open table “tblSOAPParameters” and change the URL to OpenClinica and the path to temporary directory, as desired:

sysURL	https://www-acc.openclinica.nl/OpenClinica-ws/
tempdir	c:\temp

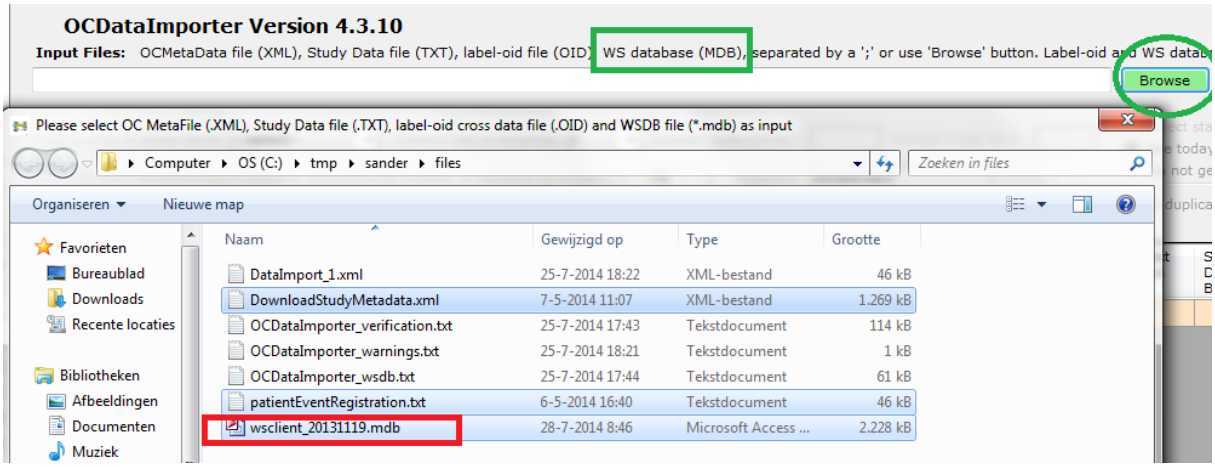
4- Make sure that OpenClinica web services is installed and the user who is going to run the forms is authorized to use the SOAP services, in OpenClinica. If you can’t do this yourself, ask your system administrator.



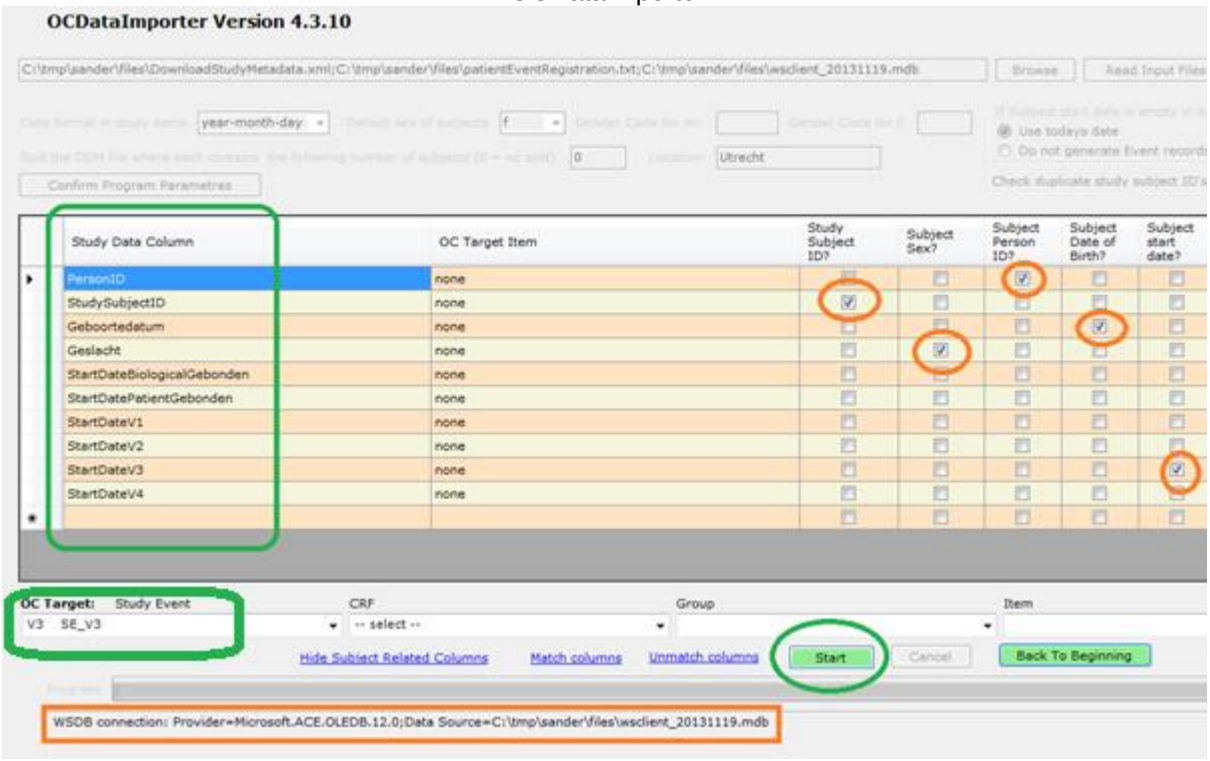
5- Get the studies using “frmGetStudies”. This should ask for your OpenClinica credentials and fill the “tblOCStudies” table with the available studies.

6- Get the study metadata by using “frmGetStudyMetadata”. This should ask for your OpenClinica credentials and the study name you will use to register subjects and fill the “tblStudyEventDef” table with the events of the selected study, among other metadata tables. This can take a long time depending the size of your metadata.

7- Run OCDataImporter (again) and select the database beside the input files.



8- Do the mappings, make sure that the proper event is selected in the “OC Target: Study Event” dropdown box and generate the XML files as explained in other chapters of this document. The database objects that are important for study subject and study event generation has to be marked as shown with orange circles.



9- After hitting the “start” button, OCDataImporter will confirm the connection (orange rectangle) and populate the following tables of the Trial Data Solutions’ database, as shown below:

tblNewSubjectsToCreate
tblNewEventsToCreate

tblNewSubjectsToCreate

StudySubjectID	SecondaryID	EnrollmentDate	PersonID	Gende	DateOfBirth
BIO-20-10002_0		25-7-2014	TR-20-10002_0	f	1-1-1969
BIO-20-10002_1		25-7-2014	TR-20-10002_1	f	1-1-1969
BIO-20-10002_2		25-7-2014	TR-20-10002_2	f	1-1-1969

tblNewEventsToCreate

StudySubjectID	EventOID	Location	StartDate
BIO-20-10002_0	SE_V3	Utrecht	25-7-2014
BIO-20-10002_1	SE_V3	Utrecht	25-7-2014
BIO-20-10002_2	SE_V3	Utrecht	25-7-2014
BIO-20-10003_0	SE_V3	Utrecht	25-7-2014
BIO-20-10003_1	SE_V3	Utrecht	11-2-2010
BIO-20-10003_2	SE_V3	Utrecht	25-7-2014
BIO-20-10003_3	SE_V3	Utrecht	11-11-2011
BIO-20-10003_4	SE_V3	Utrecht	5-11-2013
BIO-20-10003_5	SE_V3	Utrecht	25-7-2014
BIO-20-10004_0	SE_V3	Utrecht	25-7-2014
BIO-20-10004_1	SE_V3	Utrecht	24-2-2010
BIO-20-10005_0	SE_V3	Utrecht	25-7-2014
BIO-20-10005_1	SE_V3	Utrecht	11-2-2010
BIO-20-10006_0	SE_V3	Utrecht	25-7-2014
BIO-20-10006_1	SE_V3	Utrecht	25-7-2014

10- Run form “frmCeateStudySubjects” to create study-subjects. After the form finishes, open the table “tblNewStudySubjectsToCreate”, check if the “Response” column is filled with results, including whether the action was successful. Make sure that all records have a “success” result, before going into next step. If not, delete all records in this table, restore a backup of the postgres OC database and go back to step 7.

tblNewSubjectsToCreate

Stud	Seco	Enrc	Perso	Gende	DateOfBirth	YearOfBirth	Response
OY001	OY001	1-1-1980	OY001	m		1985	<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"><SOAP-ENV:Header/><SOAP-ENV:Body><createResponse xmlns="http://openclinica.org/ws/studySubject/v1"><result xmlns="http://openclinica.org/ws/studySubject/v1">Success</result><label xmlns="http://openclinica.org/ws/studySubject/v1">OY001</label></createResponse></SOAP-ENV:Body></SOAP-ENV:Envelope>

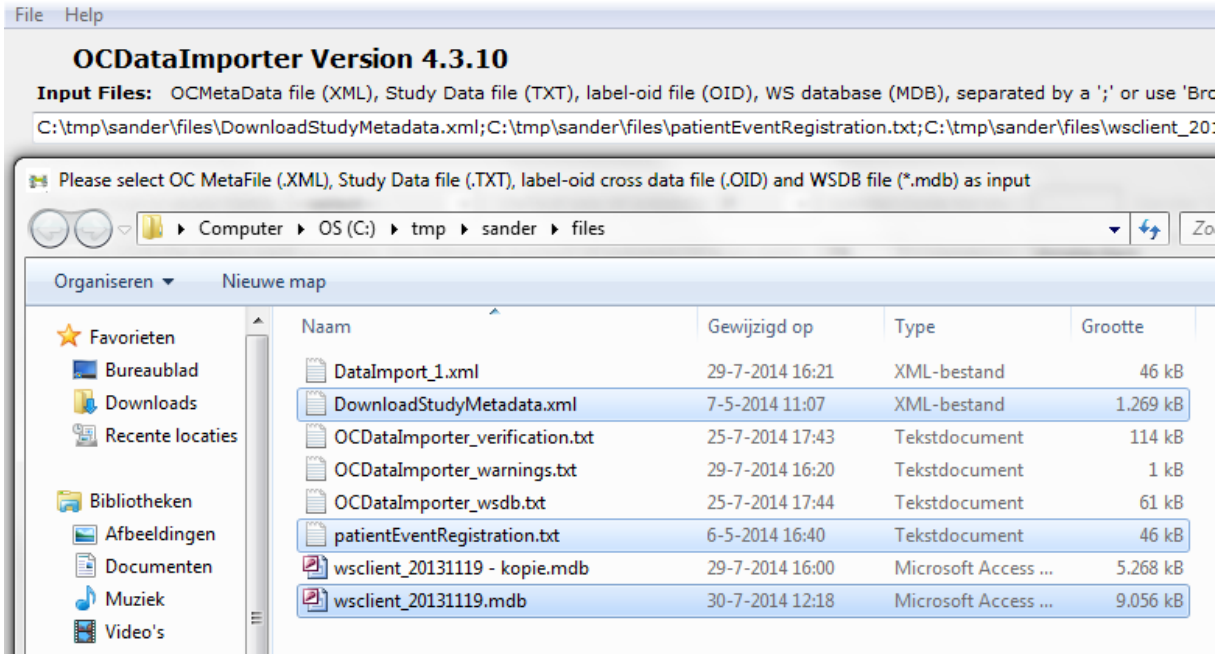
11- Run form "**frmCeateEvents**" to create event scheduling in OpenClinica using web services. After the form finishes, open the table "tblNewEventsToCreate", check if the "Response" column is filled with results, including whether the action was successful and the studySubjectOID's are all present.

tblNewEventsToCreate

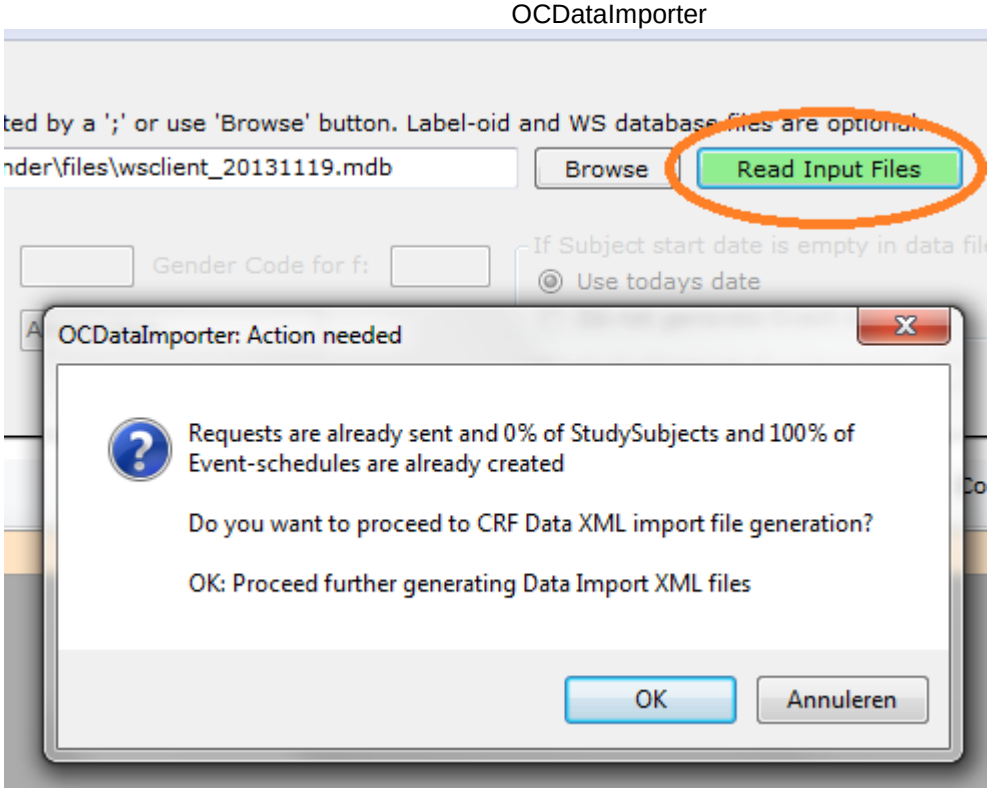
RecID	StudySubjec	EventOID	Locatior	StartDate	Response
1	OY001	SE_V1		7-8-2014	<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"><SOAP-ENV:Header/><SOAP-ENV:Body><scheduleResponse xmlns="http://openclinica.org/ws/event/v1"><result xmlns="http://openclinica.org/ws/event/v1">Success</result><eventDefinitionOID xmlns="http://openclinica.org/ws/event/v1">SE_V1</eventDefinitionOID><studySubjectOID xmlns="http://openclinica.org/ws/event/v1">SS_OY001</studySubjectOID><studyEventOrdinal xmlns="http://openclinica.org/ws/event/v1">1</studyEventOrdinal></scheduleResponse></SOAP-ENV:Body></SOAP-ENV:Envelope>

In order to make use of these studySubjectOID's, OCDataImporter has to be run for the second time. This is because these studySubjectOID's were *not known* during the first run. This is the only way to generate the correct data import XML files as the OID's are needed for subject identification in de CRF data.

12- To do this, run OCDataImporter again, with same parameters, just like in step 5:



13- When the button "Read Input files" is hit, OCDataImporter will now *behave differently* and this time ask the following:



When the OK button is hit, OCDataImporter will read all of the studySubjectOID's from the Response column and re-generate all data import XML files, according to these studySubjectOID's. Only after this, it will be possible to upload the resulting XML files containing CRF data, into OpenClinica, as explained in the next chapter 3.11. See also section 3.5.1.

Points to consider:

It is strongly advised to perform the above mentioned steps first in an acceptance or test environment. Before starting, make a backup of the OC postgres database (or ask your system administrator to do so) and make a copy of the clean ws access database.

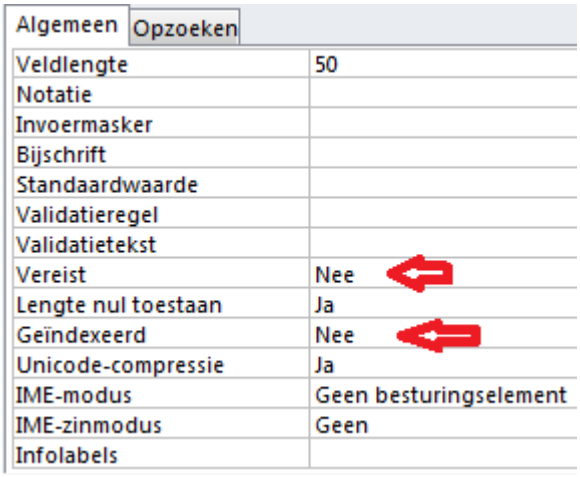
If the study configuration parameter "PersonID used" is set to "not used" the following modification has to be made to table tblNewSubjectsToCreate:

tblNewSubjectsToCreate

Veldnaam	Gegevenstype
StudySubjectID	Tekst
SecondaryID	Tekst
EnrollmentDate	Datum/tijd
PersonID	Tekst
Gender	Tekst
DateOfBirth	Datum/tijd
YearOfBirth	Numeriek
Response	Memo

Field "PersonID should be modified as:

Mandatory: No
Indexed: Yes, duplicates OK

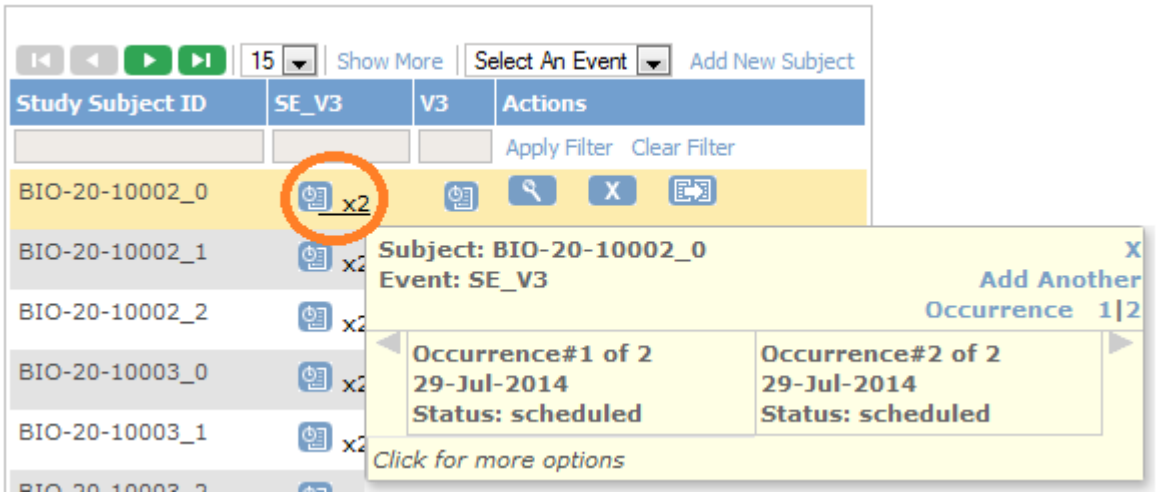


It is advised that when a certain step fails or has to be repeated, restore the OC database to its state before the forms "frmCreateStudySubjects" and "frmCreateEvents" were run, download or copy a clean ws access database to your working directory and start again from scratch. If you do this, think about the SOAP parameters (explained in step 3)

All this is needed because there is no "rollback" of running the forms. As a result of this, it is not possible to repeat the above steps, because once the events are scheduled, the Response columns will always return "fail" in the next runs of forms "frmCeateStudySubjects" and "frmCeateEvents" – except if the event is a repeating event - .This will make it impossible to return the studySubjectOID's because these are only returned if the request is successful. Without the studySubjectOID's the DataImport XML files will not work. In order to work around this problem, you must generate an OID file as explained in section 3.5.1 manually and use this file to generate proper studySubjectOIDs and run OCDataImporter for the 3rd time with that file as input – but, starting from scratch is easier.

If the event to schedule is a repeating event, each running of "frmCreateEvents" will schedule the next repeat-index; that is first run will generate the first repeat, second; second repeat etc.. Make sure that the "StartDate" is correct when you run the second, third, etc times.

Our example after the "second" run:



Note that if web services are used, the program will not generate the SQL files (inserts.sql, deletes.sql, etc.) needed to update the Postgres database, as these will be redundant.

3.11 Importing data to OpenClinica using "DataImport_(n).xml".

The file DataImport_(n).xml contains a fixed begin section + import statements for each line in the file created in section 3.2. + fixed end section.

```
<?xml version="1.0" encoding="UTF-8"?>
<ODM xmlns="http:www.cdisc.org/ns/odm/v1.3"
xmlns:xsi="http:www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http:www.cdisc.org/ns/odm/v1.3 ODM1-3.xsd"
ODMVersion="1.3" FileOID="1D20080412202420" FileType="Snapshot"
Description="First dataset for testing of ODM" CreationDateTime="2008-04-12T20:24:20" >
<ClinicalData StudyOID="S_VUMCDEPT" MetaDataVersionOID="v1.0.0">
…..
   <SubjectData SubjectKey="SS_NIJM_CAIRO_CRC_002">
      <StudyEventData StudyEventOID="SE_CAIRO_EVENT" StudyEventRepeatKey="1">
         <FormData FormOID="F_CAIRO_GEN_IN_V12">
            <ItemGroupData ItemGroupOID="IG_CAIRO_UNGROUPED" TransactionType="Insert" >
               <ItemData ItemOID="I_CAIRO_AGE" Value="60" />
               <ItemData ItemOID="I_CAIRO_DATE_OF_BIRTH" Value="1942-03-05" />
               <ItemData ItemOID="I_CAIRO_DATE_OF_RANDOMISATION" Value="2003-01-13" />
               <ItemData ItemOID="I_CAIRO_ELIGIBLE" Value="0" />
               <ItemData ItemOID="I_CAIRO_GENDER" Value="1" />
               <ItemData ItemOID="I_CAIRO_PATIENT_ID" Value="NIJM_CAIRO_CRC_002" />
               <ItemData ItemOID="I_CAIRO_WHO_PERFORMANCE_STATUS" Value="1" />
            </ItemGroupData>
         </FormData>
      </StudyEventData>
   </SubjectData>
….
….
</ClinicalData>
</ODM>
```

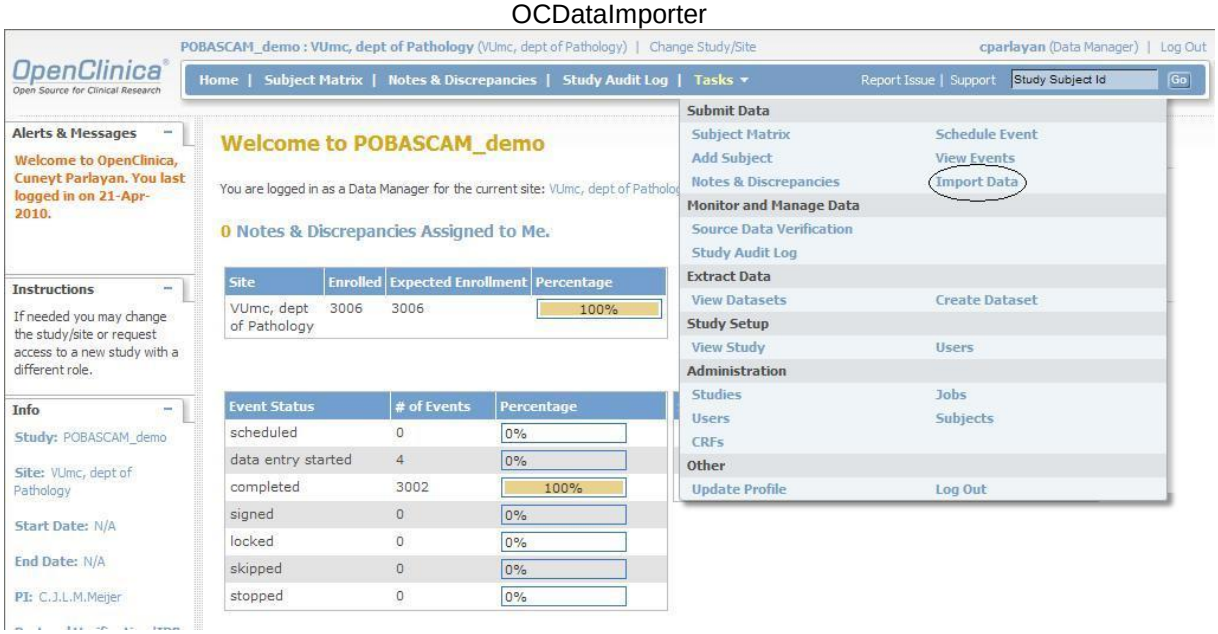
The above subject data corresponds with the first line of the data file:

```
NIJM_CAIRO_CRC_002          0                                   No distant metastasis, number of affected
organs is 0 (array CGH data is available)                       0       5-mrt-1942      1       1       1
        13-jan-2003     60              1       4       -1              -1      1       -1
        -1              -1      0       1       3       0       -1      -1      -1      -1      0
        -3      -1      -1      -1      0               1       1       0       18              -1
        1       435             -1      1       0               -1      -1      435             -1
        1       0               -1      -1      435             -1      1               1454    1
        1       1       0
```

Depending on the given split factor (section 3.11), a number of DataImport_(n).xml files will be created.

When this file is (these files are) uploaded in OpenClinica, the data conversion process will be completed.

To do this, start OpenClinica, click ok “Import Data” of the tasks menu.



In the next screen, you can enter the path to DataImport_(n).xml:

Click then “continue” to start data import. Repeat this for all DataImport files.

3.12 Importing CRF data containing repeating events and repeating groups

Importing CRF data containing repeating events and repeating groups are in general very similar to importing simple CRF data as explained in the above sections. The way of designing the column names for repeating events and/or repeating groups in the data file must be like one of the following formats:

3.12.1 Importing CRF data containing repeating events and repeating groups by using data file containing the repeating information on separate columns:

columnName_Ex_Gy Where E stands for Repeating Events, x = StudyEventRepeatKey,
G stands for Repeating groups, y = ItemGroupRepeatKey.

Therefore valid examples are: col_E1, col_E1_G1, col_G1, col_E2, etc.

Example data file:

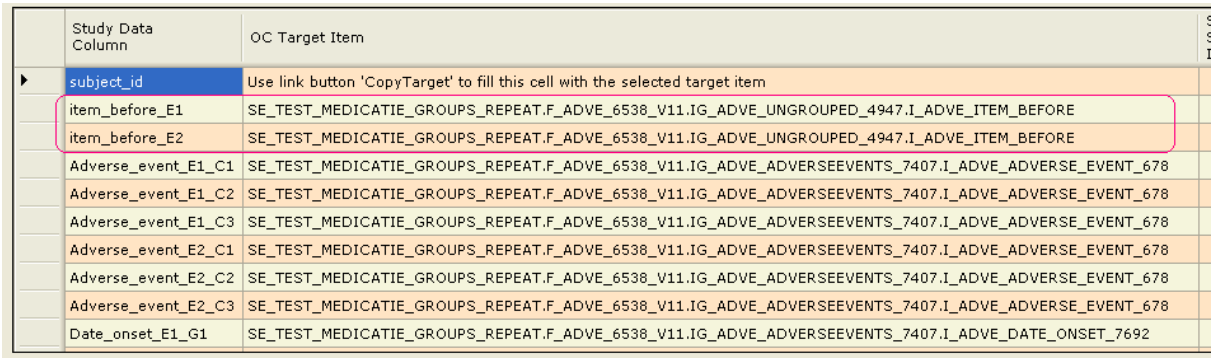
subject_id item_before_E1 item_before_E2 Adverse_event_E1_G1 Adverse_event_E1_G2 Adverse_event_E1_G3 Adverse_event_E2_G1 Adverse_event_E2_G2 Adverse_event_E2_G3 Date_onset_E1_G1 Date_onset_E1_G2 Date_onset_E1_G3 date_onset_E2_G1 date_onset_E2_G2 date_onset_E2_G3
KG_CRC_IIandIII_V2_002 bef1 bef2 ae11 ae12 ae13 ae21 ae22 ae23 1-1-1960 1-2-1960 1-3-1960 2-1-1960 2-2-1960 2-3-1960

Will result in:

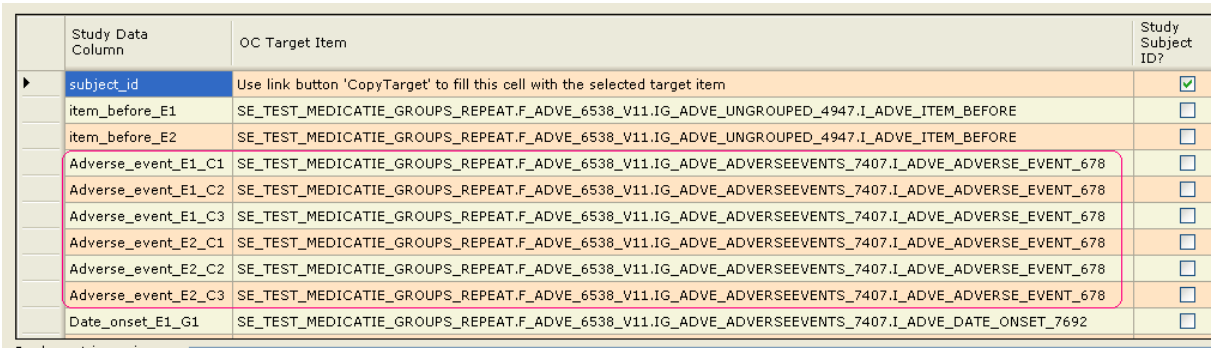
```
<?xml version="1.0" encoding="UTF-8"?>
<ODM xmlns="http:www.cdisc.org/ns/odm/v1.3"
xmlns:xsi="http:www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http:www.cdisc.org/ns/odm/v1.3 ODM1-3.xsd"
ODMVersion="1.3" FileOID="1D20080412202420" FileType="Snapshot"
Description="Dataset ODM" CreationDateTime="2012-11-01T10:00:00" >
<ClinicalData StudyOID="S_CUNEYTTE" MetaDataVersionOID="v1.0.0">
   <SubjectData SubjectKey="SS_KG_CRC_IIandIII_V2_007">
      <StudyEventData StudyEventOID="SE_TEST_MEDICATIE_GROUPS_REPEAT"
StudyEventRepeatKey="1">
         <FormData FormOID="F_ADVE_6538_V11">
            <ItemGroupData ItemGroupOID="IG_ADVE_UNGROUPED_4947"
TransactionType="Insert" >
         <ItemData ItemOID="I_ADVE_ITEM_BEFORE" Value="bef1" />
         </ItemGroupData>
          <ItemGroupData ItemGroupOID="IG_ADVE_ADVERSEEVENTS_7407"
ItemGroupRepeatKey="1" TransactionType="Insert" >
               <ItemData ItemOID="I_ADVE_ADVERSE_EVENT_678" Value="ae11" />
               <ItemData ItemOID="I_ADVE_DATE_ONSET_7692" Value="1960-01-01"
/>               </ItemGroupData>
            <ItemGroupData ItemGroupOID="IG_ADVE_ADVERSEEVENTS_7407"
ItemGroupRepeatKey="2" TransactionType="Insert" >
               <ItemData ItemOID="I_ADVE_ADVERSE_EVENT_678" Value="ae12" />
               <ItemData ItemOID="I_ADVE_DATE_ONSET_7692" Value="1960-02-01"
/>            </ItemGroupData>
            <ItemGroupData ItemGroupOID="IG_ADVE_ADVERSEEVENTS_7407"
ItemGroupRepeatKey="3" TransactionType="Insert" >
               <ItemData ItemOID="I_ADVE_ADVERSE_EVENT_678" Value="ae13" />
               <ItemData ItemOID="I_ADVE_DATE_ONSET_7692" Value="1960-03-01"
/>            </ItemGroupData>

      <StudyEventData StudyEventOID="SE_TEST_MEDICATIE_GROUPS_REPEAT"
StudyEventRepeatKey="2">
         <FormData FormOID="F_ADVE_6538_V11">
            <ItemGroupData ItemGroupOID="IG_ADVE_UNGROUPED_4947"
TransactionType="Insert" >
         <ItemData ItemOID="I_ADVE_ITEM_BEFORE" Value="bef2" />
         </ItemGroupData>
          <ItemGroupData ItemGroupOID="IG_ADVE_ADVERSEEVENTS_7407"
ItemGroupRepeatKey="1" TransactionType="Insert" >
               <ItemData ItemOID="I_ADVE_ADVERSE_EVENT_678" Value="ae21" />
               <ItemData ItemOID="I_ADVE_DATE_ONSET_7692" Value="1960-01-02"
/>               </ItemGroupData>
            <ItemGroupData ItemGroupOID="IG_ADVE_ADVERSEEVENTS_7407"
ItemGroupRepeatKey="2" TransactionType="Insert" >
               <ItemData ItemOID="I_ADVE_ADVERSE_EVENT_678" Value="ae22" />
               <ItemData ItemOID="I_ADVE_DATE_ONSET_7692" Value="1960-02-02"
/>            </ItemGroupData>
            <ItemGroupData ItemGroupOID="IG_ADVE_ADVERSEEVENTS_7407"
ItemGroupRepeatKey="3" TransactionType="Insert" >
               <ItemData ItemOID="I_ADVE_ADVERSE_EVENT_678" Value="ae23" />
               <ItemData ItemOID="I_ADVE_DATE_ONSET_7692" Value="1960-03-02"
/>            </ItemGroupData>
```

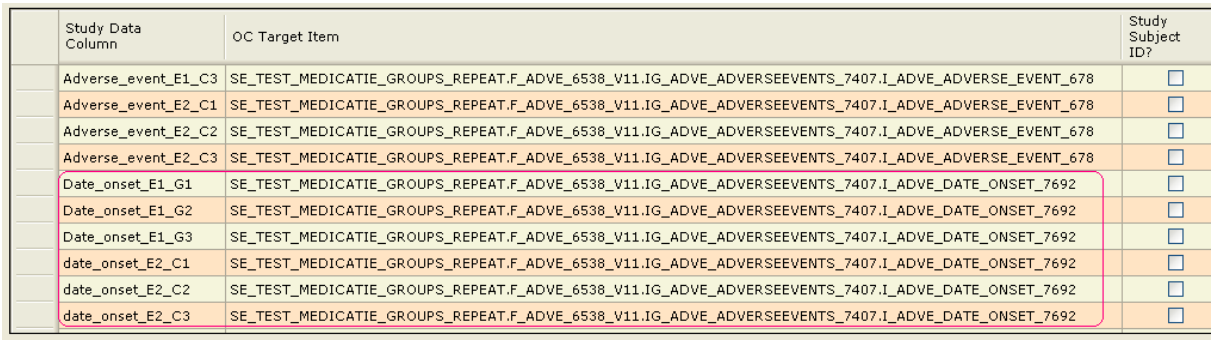
All repeating data columns should be mapped with the same OC item, as to be seen below.



Study Data Column	OC Target Item
subject_id	Use link button 'CopyTarget' to fill this cell with the selected target item
item_before_E1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_UNGROUPED_4947.I_ADVE_ITEM_BEFORE
item_before_E2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_UNGROUPED_4947.I_ADVE_ITEM_BEFORE
Adverse_event_E1_C1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678
Adverse_event_E1_C2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678
Adverse_event_E1_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678
Adverse_event_E2_C1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678
Adverse_event_E2_C2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678
Adverse_event_E2_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678
Date_onset_E1_G1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692



Study Data Column	OC Target Item	Study Subject ID?
subject_id	Use link button 'CopyTarget' to fill this cell with the selected target item	☑
item_before_E1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_UNGROUPED_4947.I_ADVE_ITEM_BEFORE	☐
item_before_E2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_UNGROUPED_4947.I_ADVE_ITEM_BEFORE	☐
Adverse_event_E1_C1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E1_C2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E1_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E2_C1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E2_C2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E2_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Date_onset_E1_G1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐



Study Data Column	OC Target Item	Study Subject ID?
Adverse_event_E1_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E2_C1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E2_C2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Adverse_event_E2_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_ADVERSE_EVENT_678	☐
Date_onset_E1_G1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐
Date_onset_E1_G2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐
Date_onset_E1_G3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐
date_onset_E2_C1	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐
date_onset_E2_C2	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐
date_onset_E2_C3	SE_TEST_MEDICATIE_GROUPS_REPEAT.F_ADVE_6538_V11.IG_ADVE_ADVERSEEVENTS_7407.I_ADVE_DATE_ONSET_7692	☐

Note: Please do not use data column names containing “_Ex” or “_Gx” where x is a digit; unless you are referring to a repeating event and/or group item, as this will not work and produce an error.

Example: Adding subjects, study subjects and repeating study events with start dates:

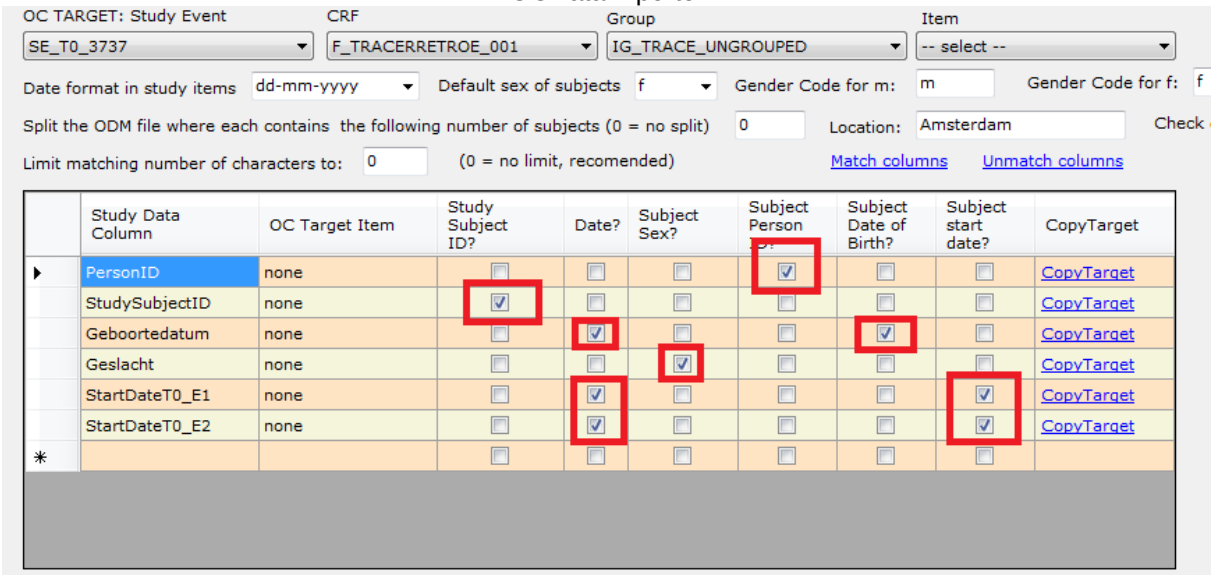
StartDateT0_E1 is the start date for event number 1 and StartDateT0_E2 for event 2. The suffixes E1 and E2 represents this.

Data file:

PersonID	StudySubjectID	Geboortedatum	Geslacht	StartDateT0_E1	StartDateT0_E2
TR-20-0086	ESRA-20-0112	02-07-1945	f	09-02-2012	
TR-20-0146	ESRA-20-0175	16-01-1953	f	17-02-2011	18-02-2011
TR-20-0050	ESRA-20-0068	24-10-1932	m	14-10-2010	
TR-20-0066	ESRA-20-0090	28-08-1976	f	19-05-2011	

OCDataImporter:

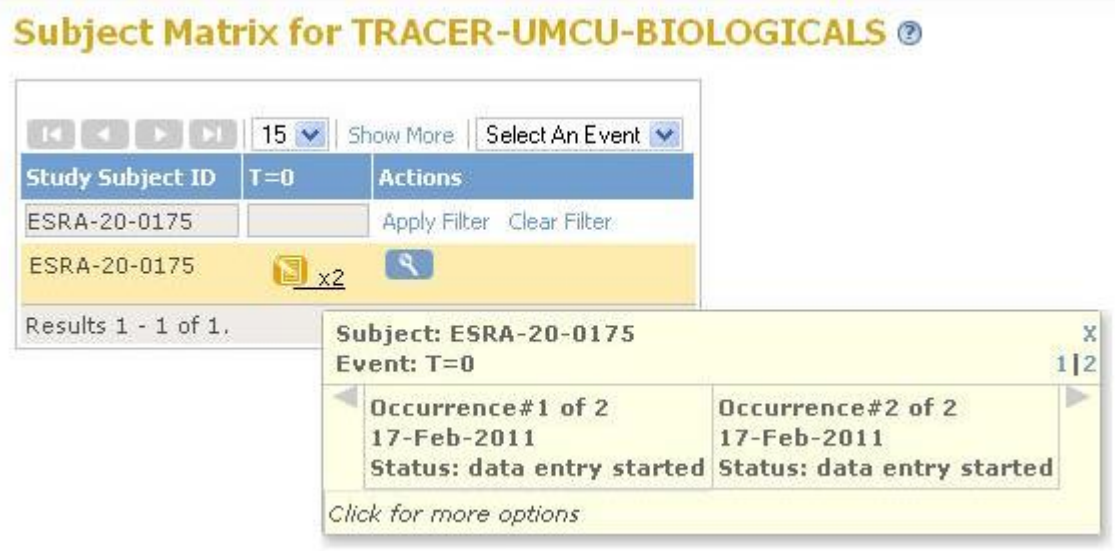
OCDataImporter



Insert statements generated for TR-20-0146:

INSERT INTO subject(status_id, gender, unique_identifier, date_created, owner_id, dob_collected, date_of_birth)
VALUES (1, 'f', 'TR-20-0146', '2013-08-28', 1, '1', '1953-01-16');
INSERT INTO study_subject(label, study_id, status_id, enrollment_date, date_created, date_updated, owner_id, oc_oid, subject_id)
VALUES ('ESRA-20-0175', (SELECT study_id FROM study WHERE oc_oid = 'S_TRACERUM'), 1, '2013-08-28', '2013-08-28', '2013-08-28', 1, 'SS_ESRA-20-0175', (SELECT subject_id FROM subject where unique_identifier = 'TR-20-0146'));
INSERT INTO study_event(study_event_definition_id, study_subject_id, location, sample_ordinal, date_start, owner_id, status_id, date_created, subject_event_status_id, start_time_flag, end_time_flag)
VALUES ((SELECT study_event_definition_id FROM study_event_definition WHERE oc_oid = 'SE_T0_3737'),
(SELECT study_subject_id FROM study_subject WHERE oc_oid = 'SS_ESRA-20-0175'),'Amsterdam', 1, '2011-02-17 12:00:00', 1, 1, '2013-08-28', 3, '0', '0');
INSERT INTO study_event(study_event_definition_id, study_subject_id, location, sample_ordinal, date_start, owner_id, status_id, date_created, subject_event_status_id, start_time_flag, end_time_flag)
VALUES ((SELECT study_event_definition_id FROM study_event_definition WHERE oc_oid = 'SE_T0_3737'),
(SELECT study_subject_id FROM study_subject WHERE oc_oid = 'SS_ESRA-20-0175'),'Amsterdam', 2, '2011-02-17 12:00:00', 1, 1, '2013-08-28', 3, '0', '0');

Result in OpenClinica after running inserts in PGAdmin:



3.12.2 Importing CRF data containing repeating events and repeating groups, by using data file containing the repeating information under columns EVENT_INDEX and GROUP_INDEX

In this approach a subject must be repeated in rows, rather than columns and that is the only difference between the method described in 3.18.1.

The file in this format which was described in 3.18.1 would now look like:

PersonID	StudySubjectID	Geboortedatum	Geslacht	StartDateT0	EVENT_INDEX
TR-20-0086	ESRA-20-0112	02-07-1945	f	09-02-2012	1
TR-20-0146	ESRA-20-0175	16-01-1953	f	17-02-2011	1
TR-20-0146	ESRA-20-0175	16-01-1953	f	18-02-2011	2
TR-20-0050	ESRA-20-0068	24-10-1932	m	14-10-2010	1
TR-20-0066	ESRA-20-0090	28-08-1976	f	19-05-2011	1

A more general example is:

subject_id	item_before	Adverse_event	EVENT_INDEX	GROUP_INDEX
1000	text1	event	1	1
1000	text1	event	2	1
1000	text1	event	1	2
1000	text1	event	2	2

Etc.

Column names EVENT_INDEX and GROUP_INDEX must be used only for above purpose.

Do not combine EVENT_INDEX, GROUP_INDEX and _E1, _G1 column names, use one approach for the entire input file.

Both methods have their advantages and disadvantages.

3.13 Data validation

The data in the input file is validated against the CRF definitions during the process. For each wrong data a warning is created and logged in file "`OCDataImporter_warnings.txt`" in the working directory.

List of possible warnings are:

1- Item is mandatory but has no value
2- Item is mandatory and hidden; and has no value. This might cause errors if the item gets shown by another condition
3- Item type is real but contains non numeric characters:
4- Item contains more numbers than allowed after the decimal point
5- Item type is integer but contains non integer characters
6- Item length <n> is too small
7- Value not in the specified code list
8- Range Check fail: Value <value> is not GT|GE|LT|LE|NE|EQ <value specified in CRF>
9- Input data file format incorrect at line = <n> Expecting: <m> columns, found <k>
10- "Duplicate key <value> at line <n>
11- Subject sex can be only 'f' or 'm'. You have <tekst> at line <n>
12- Invalid subject birth date <date> at line <n>
13- Invalid start date <date> at line <n>

It is recommended that the data file should be warning-free before data upload.

Appendix: Error messages

1- User Manual not found: User manual can't be found in the installation directory. See section 3.1
2- Can't open selected data file "thedatafile.txt ", can't continue. Delimiter = Items per line = 20: The data file can't be opened, it may not exist or you don't have enough privileges to open it.
3- Failed to start Acrobat reader: acrord32 doesn't exist to open the document file. Go to the installation directory and double click on the document file to open it manually.
4- Do you want to load your previous grid?: The program saves the last grid used; if you want to reload the last saved grid, click yes.
5- Can't generate grid dump file (see log file for details - Do you have enough permissions to write in target folder?): You probably have no write permissions; log file is created in the working directory.
6- DataImport_* files will be overwritten. Do you want to delete the old files?: If you run the program on a directory which older DataImport files exist, they will be overwritten. Click No if you don't want to lose your old files, save them elsewhere and try again.
7- Please select (only) one field as STUDY SUBJECT ID by using check box; You have 2 selected.: Only one and only one study subject id may be selected.
8- "Please select at most one field as STUDY SUBJECT SEX/PERSON ID/SUBJECT DATE OF BIRTH/STUDY START DATE by using check box; You have 2 selected.: These can only be checked for one row maximum.
9- Please enter location: Location name is mandatory.
10- Input data file format incorrect at line = 26 Expecting: 12; found: 13 items; this is the faulty line: At this line there is a mismatch with number of columns and number of data items. (12 columns and 13 items)
11- Duplicate key "thekeyvalue" at line = 34: If the duplicate key check is performed (section 3.6.5) and a duplicate key is detected, this message will be displayed.
12- Subject sex can be only 'f' or 'm'. You have '"MALE"' at line 12. Index: 5. Exiting...The generated files ARE INCOMPLETE AND CAN NOT BE USED : At the 5th column there should be the gender code and this must be conform section 3.6.2.
13- Invalid subject birth date '"12061998"' at line 23. Index: 7. Exiting...The generated files ARE INCOMPLETE AND CAN NOT BE USED: At the 7th column there should be the birth date but that can't be converted to ODM format. See section 3.6
14- Invalid subject start date '"12061998"' at line 23. Index: 7. Exiting...The generated files ARE INCOMPLETE AND CAN NOT BE USED: At the 7th column there should be the start date but that can't be converted to ODM format. See section 3.6
15- Exception while reading data file: Unexpected error; see the log file for details.
16- Error while getting STUDYEVENT Repeating Key: Cant resolve the DataItemColumnName Adverse_event_3 + ". The proper name should look like 'DataItem_E2_G3 Where E2 means Event repeating key = 2 and G3 means Group repeating key = 3. Exiting...The generated files

ARE INCOMPLETE AND CAN NOT BE USED: The name of the data item column for repeating events and groups has a format of DataItem_Ex_Cy where x=repeat key study event and y=repeating group. This is not the case with Adverse_Event_3. See section 3.12.

17- Error while getting GROUP Repeating Key: See 16.
18- Wrong index at: 67. Exiting...The generated files ARE INCOMPLETE AND CAN NOT BE USED"; The program is unable to get the location of the item at the specified line.
19- Can't get study event/group/event/CRF/item definitions; please check the format of the file: The meta file created by OpenClinica is probably corrupt. Use an XML editor to see what is wrong; eventually regenerate the file.
20- Please enter or select correct input files: Either type two file names separated by a semicolon (;) or use browse button.: Specify the 2 input files correctly.
21- Can't open selected data file, PROBABLY file doesn't belong to you or is read only. Please make sure you are the owner, then try again: This happens mostly when the data file is saved on another computer or by someone else than the user who is running the program now. Make a copy of the data file and use that one as input.
22- Can't open selected OC meta file, can't continue: See log file.
23- Not all Items in the selected CRF could be matched. For the list of UNMATCHED Items, see the progress textbox below. You can match those items by using the comboboxes above. Control the matched items too, as the matching can not be 100% correct!: See section 3.8
24- All Items in the selected CRF could be matched. Control the matched items as the matching can not be 100% correct!: See section 3.8
25- Process is not finished yet. Are you sure you want to stop this program?: This is issued when the cansel button is hit. The process has to be rerun after this.
26- Please use all of the comboboxes above to define a target item. (there are still -- select --'s up there): When the “copy target” link button is clicked, all of the target item comboboxes must be selected. See section 3.8
27- Please read input files first: Hit the “Read input files” button to do this.
28- Data file contains _E and/or _G columns while EVENT_INDEX and/or GROUP_INDEX columns also exist. Please stick to one method of file construction; having both is ambigious: See section 3.12