XML4Pharma's ODM Study Designer
New features of version 2009-R1

Author: Jozef Aerts, XML4Pharma
Last Version: May 27, 2009

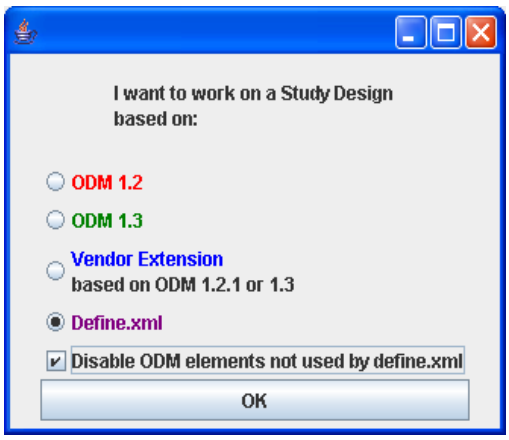
New features of version 2009-R1

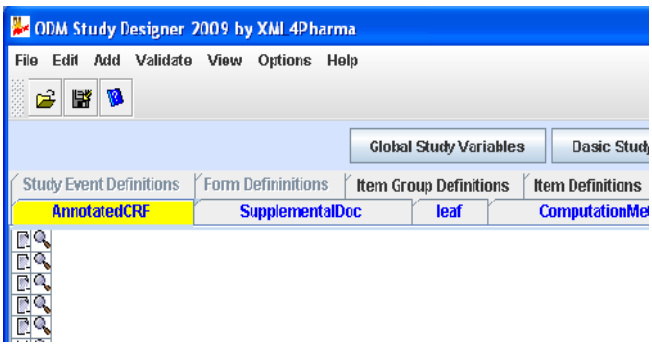
Version 2009-R1 of the ODM Study Designer has a number of new features. These include:

- extended support for define.xml
- vendor extensions: loading reference-definition pairs for extension elements
- CDASH forms included
- Schedule of Visits Views
- cleaning up functionality

1. Extended support for define.xml

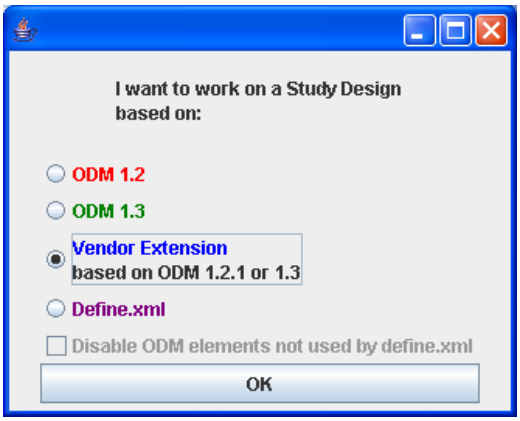
   At startup time, when choosing for "define.xml", an additional checkbox shows up asking whether ODM elements not used by define.xml should be disabled.
   When checking this box, no GUI elements for ODM elements that are defined by the ODM 1.2 standard, but that are not used by define.xml (such as Protocol, StudyEventDef, FormDef, will be created.





   This functionality is especially useful when working on standard define.xml files which will be used for submission to the FDA only.

2. Vendor extensions, loading reference-definition pairs for extension elements



Many ODM Vendor Extensions (based on the vendor extension mechanism as of ODM XML-Schema 1.2.1) use the, similarly to the ODM itself, so-named "reference-definition" or "Ref-Def" mechanims.
For example in the ODM, a form is defined using the FormDef element, whereas it is referenced by the StudyEventDef element by use of the FormRef element.

Similarly, when an extension element has been defined that has an OID, it can be referenced by another element/attribute combination.

For example, suppose we have defined a vendor extension element (in its own namespace) my:ArmDef with an attribute "OID" and an attribute "Name".
Suppose that in the StudyEventRef element, we have added an extra attribute my:ArmOID, referencing the ArmDef with the same OID.
Now, the relation between "ArmOID" and "ArmDef – OID" cannot be expressed in an XML-Schema, so we need to express it explicitely, just by providing is as a pair:

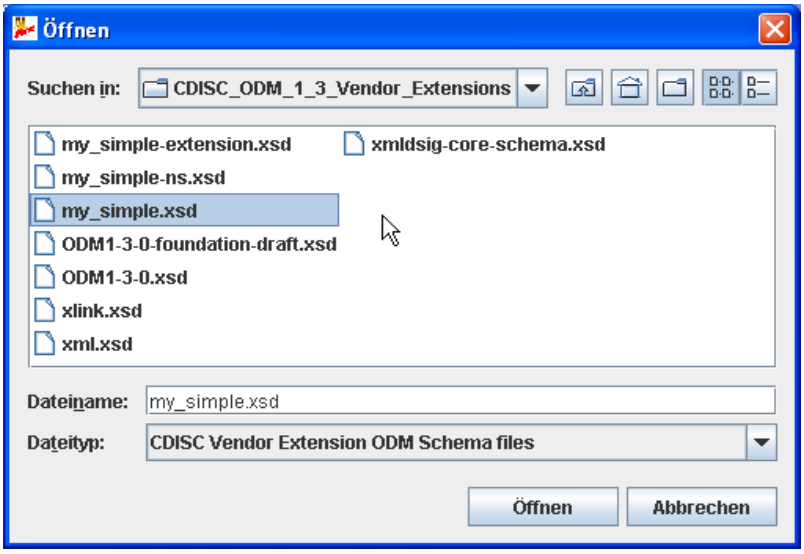
my:ArmOID my:ArmDef

A number of such pairs (one pair per line) can be added to a simple text file for a specific vendor extension. This file is the file that can then be loaded after the system has asked for the XML-Schema of the vendor extension.

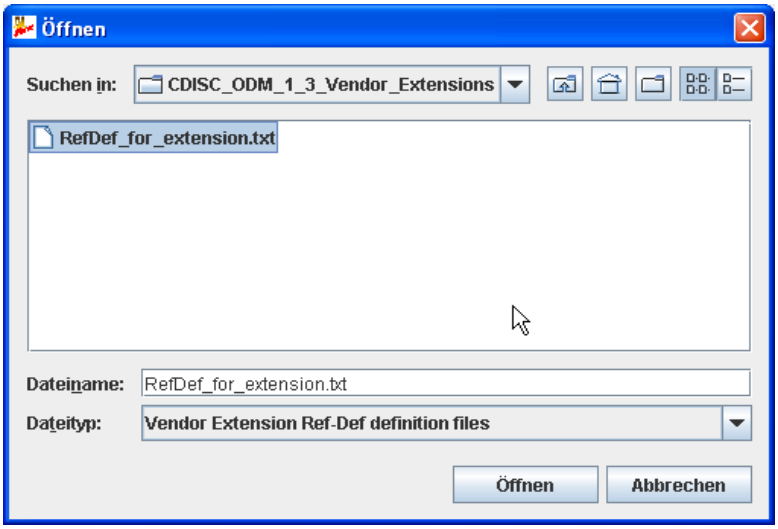
The distribution comes with a sample directory "CDISC_ODM_1_3_Vendor_Extensions" which contains a very simple vendor extension example, and for which the schema file is "my-simple.xsd".
The directory also contains a file "RefDef_for_extension.txt" containing the above shown line defining the "ref-def" relationship between "ArmOID" and "ArmDef"[1].

So when starting working with this vendor extension, one will have the following sequence:

1 Remark that no prefix should be given in the case a regular ODM element is referenced, e.g. when the vendor extension attribute "my:PreviousStudyEventOID" referenced the StudyEventDef element, the line in the file is simply:
my:PreviousStudyEventOID StudyEventDef

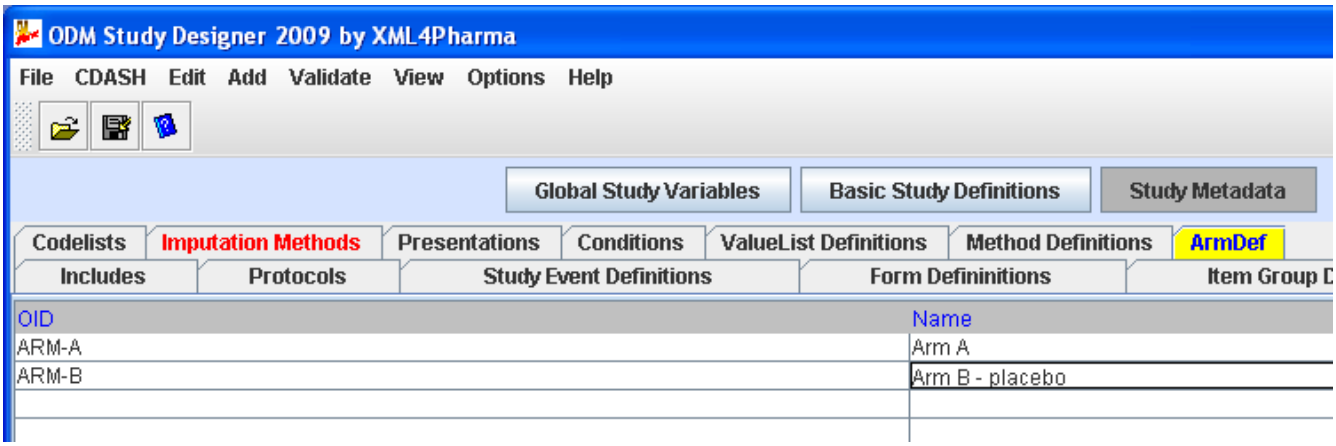

Öffnen
Suchen in:
CDISC_ODM_1_3_Vendor_Extensions
my_simple-extension.xsd
xmldsig-core-schema.xsd
my_simple-ns.xsd
my_simple.xsd
ODM1-3-0-foundation-draft.xsd
ODM1-3-0.xsd
xlink.xsd
xml.xsd
Dateiname:
my_simple.xsd
Dateityp:
CDISC Vendor Extension ODM Schema files
Öffnen
Abbrechen


Vendor Extension Ref/Def
Do you want to load a file
with Vendor Extension Ref-Def definitions ?
Ja
Nein

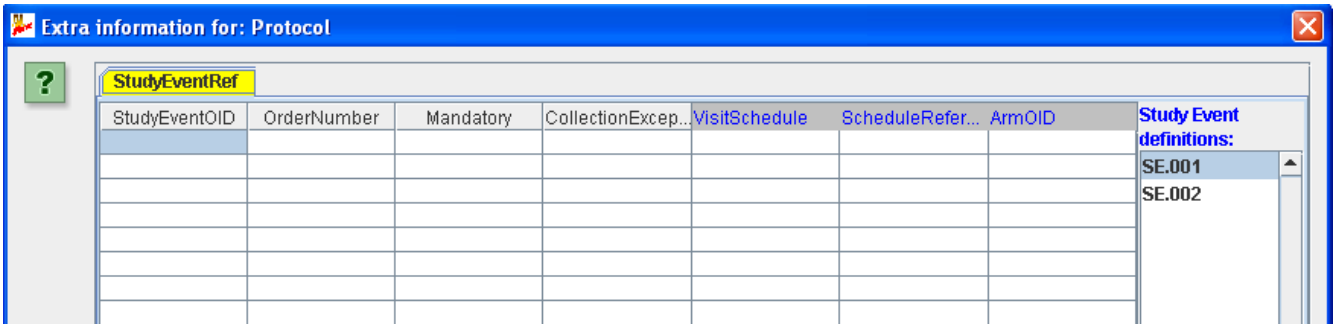

Öffnen
Suchen in:
CDISC_ODM_1_3_Vendor_Extensions
RefDef_for_extension.txt
Dateiname:
RefDef_for_extension.txt
Dateityp:
Vendor Extension Ref-Def definition files
Öffnen
Abbrechen

and

In the "Study MetaData" part, one now finds a new (vendor extension) element ArmDef, with two (mandatory) attributes "OID" and "Name". Once we have defined some arms, they can be referenced.

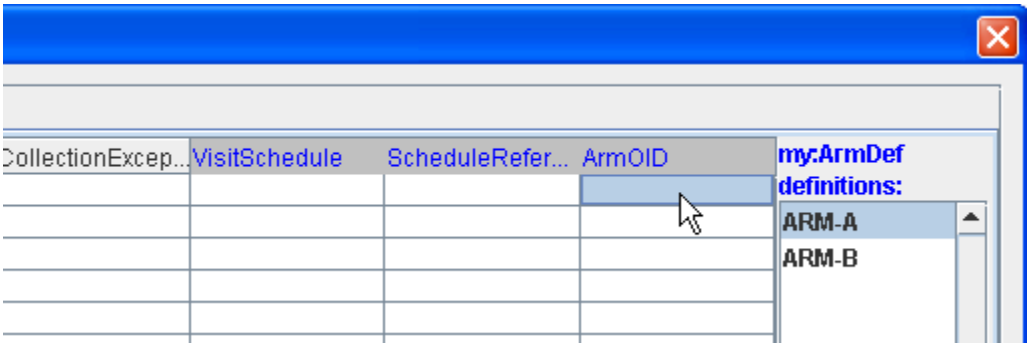


After having defined some StudyEvents, and when one now goes to the "Protocol" element, and clicks the "+" icon to add StudyEvents, one sees:

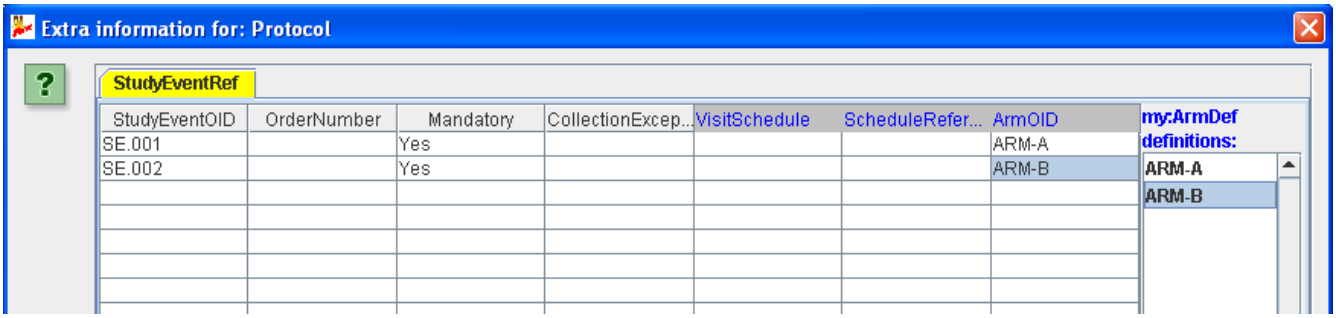


When clicking in the "ArmOID" cell[2], the right side of the window changes, and the previous defined Arms are shown:

2 Remark that the column title for this attribute is colored blue, reminding us that it is an extension attribute



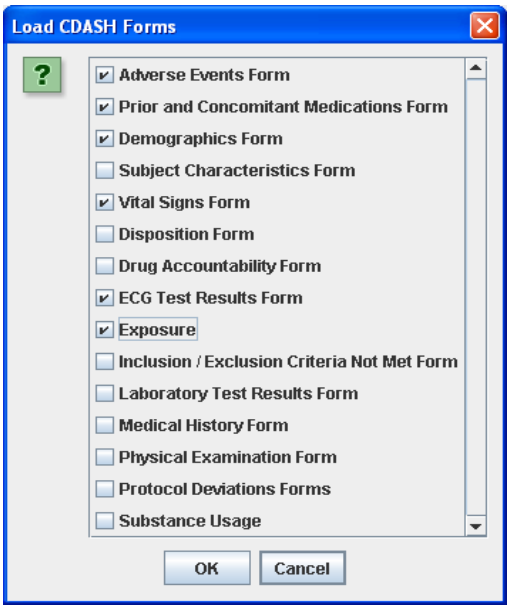
One can now drag-and-drop one of them into the "ArmOID" cell. The result is e.g.:



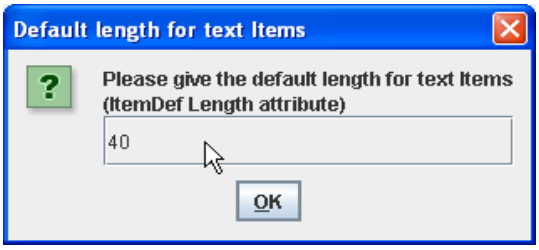
P.S.: In the sample extension, the "VisitSchedule" and "ScheduleReference" allow to add timeplanning of visits, e.g. "visit SE.005" should e.g. come 10 days after the previous visit, or e.g. 30 days after the first visit.

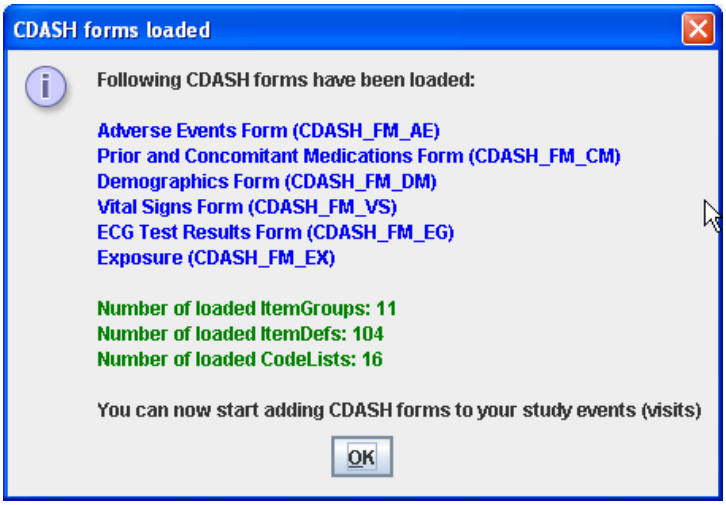
3. CDASH forms

Version 2009-R1 of the software implements the CDASH forms.
Using the menu "CDASH -> Load CDASH forms", one can select which CDASH forms one wants to add to the study design:



After having selected the desired forms, the software will ask for the default text “Length” for the ItemDefs that are loaded, as the CDASH standard does not define what these lengths should be:

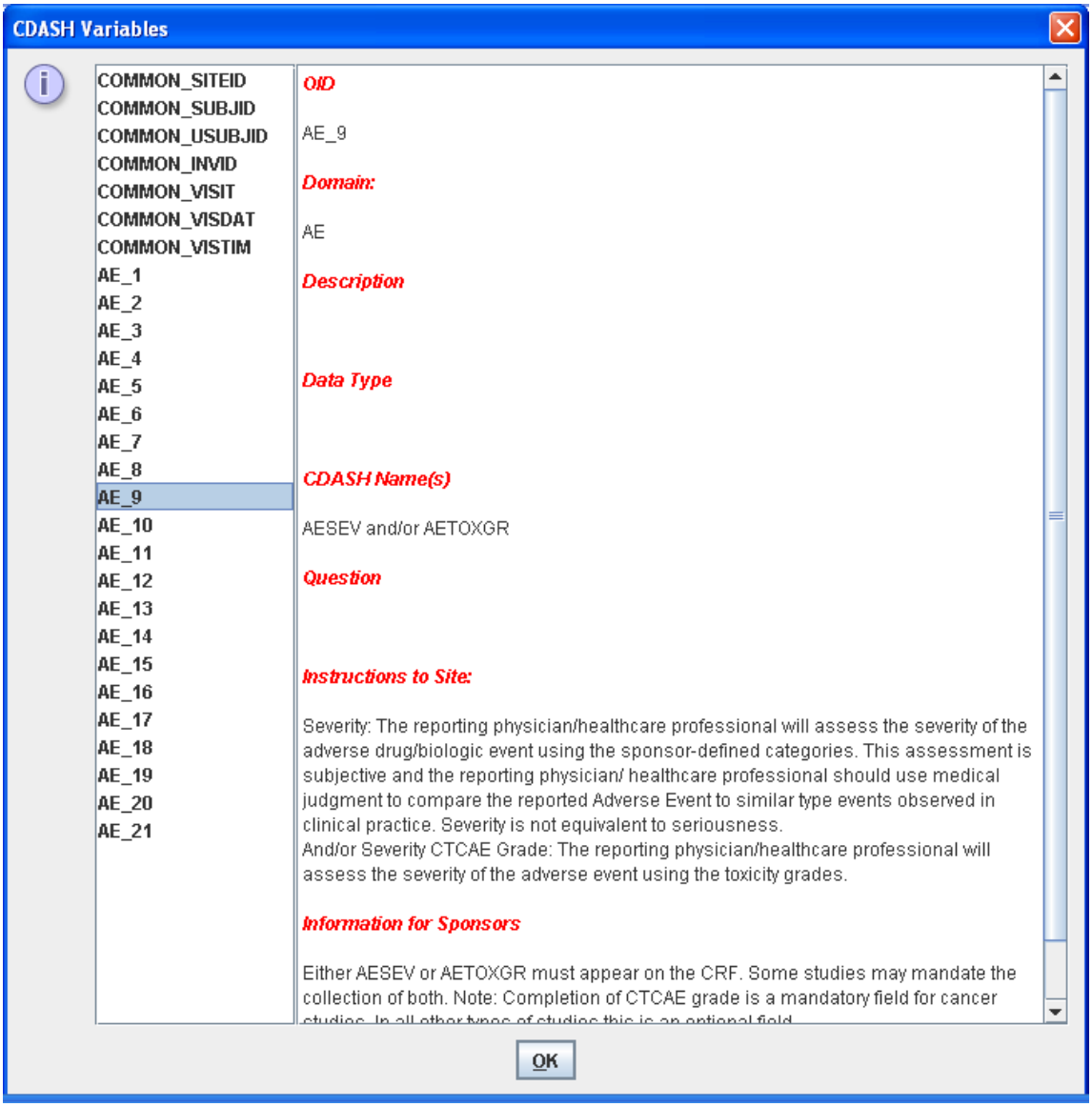




One can now immediately use the CDASH forms (e.g. in the eCRF view), but one is advised to first look at the individually loaded ItemDefs and ItemGroupDefs, and to tailor these to the specific study, this as the published CDASH forms have a lot of “defaults” in them, wich may not be applicable to your study. Also, you might want to add additional fields to the loaded CDASH forms.

Remark: in near future, one will be able to load different “implementations” (i.e. different “scenarios” as defined in the CDASH document) of the same CDASH form.

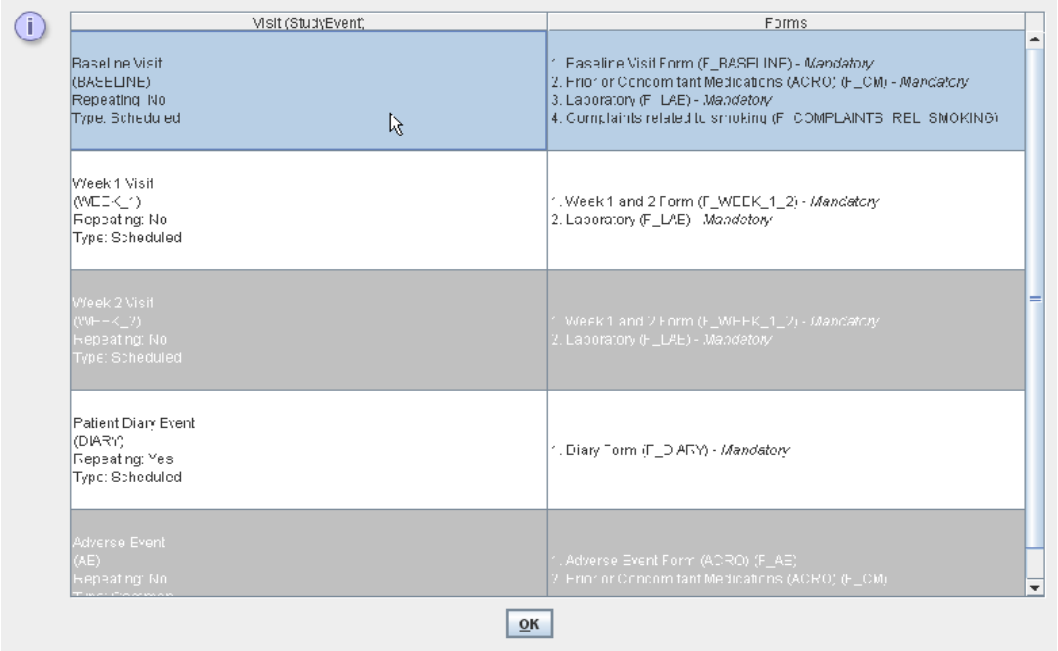
Another CDASH functionality is that one can inspect the properties of the loaded CDASH variables, using the menu “CDASH -> Show CDASH Variable Properties”. For example:



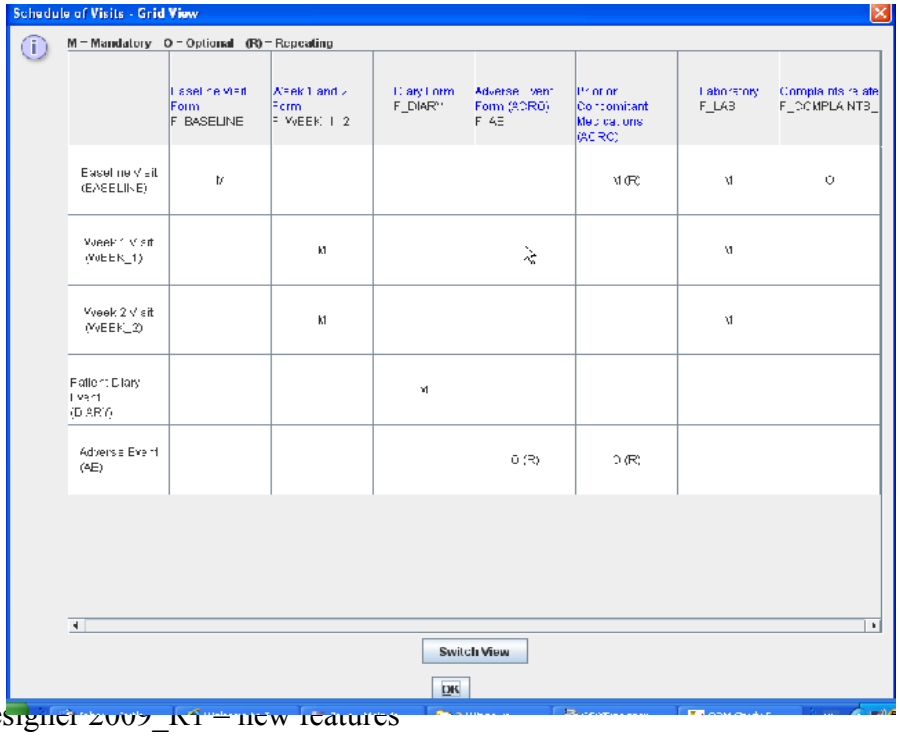
This essentially gives you the information from the documentation of the CDASH 1.0 standard.

4. Schedule of Visits Views

Using the menus “View – Schedule of Visits (table)” and “View – Schedule of Visits (grid)”, the user can now inspect the schedule of vists.
Using the first menu, a table of all the visits (as listed in the “Protocol”) is given together with the list of Forms used in each visit.



Using the menu “View – Schedule of Visits (grid)”, the user obtains a grid view of the schedule of visits. The button “Switch View” allows to switch between a view where the visits (StudyEvents) are on the vertical axis, and a view where the visits are on the horizontal axis.



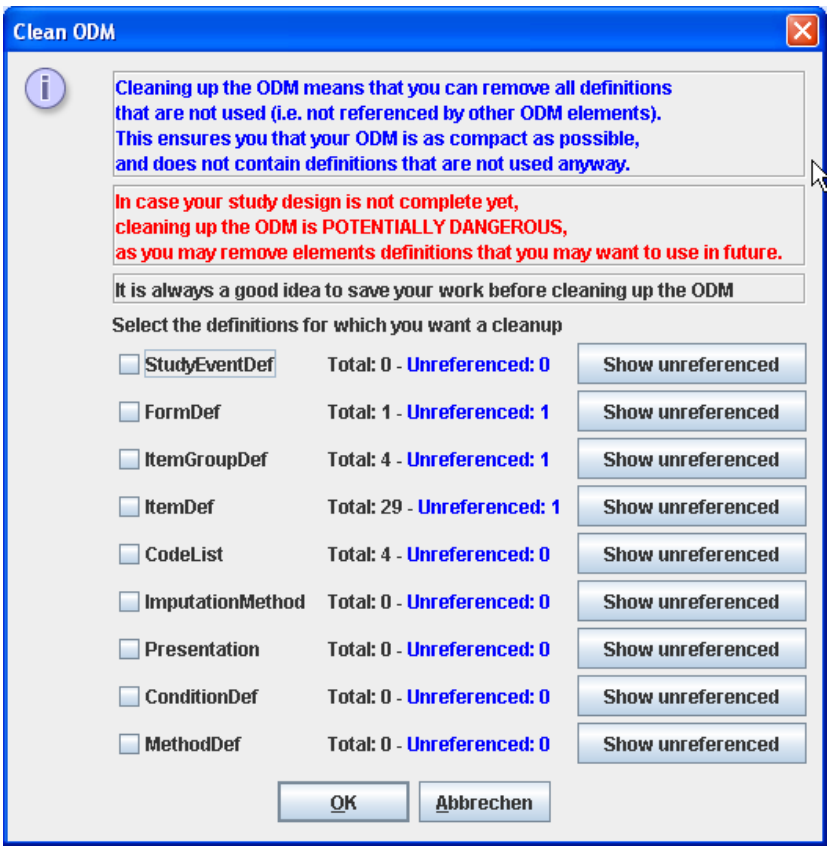
In the table, it is shown whether the used form is optional (O) or mandatory (M), and whether it is repeating ((R)).

5. Cleaning up facilities

Companies will typically use libraries of CodeLists, ItemDefs, ItemGroupDefs, etc.., which can be loaded using the "Load from Library" buttons. This may mean that one has e.g. much more ItemDefs loaded than are actually used (i.e. referenced from ItemGroupDef elements).

All these loaded ItemDefs will also appear in the ODM file when one saves the study design using "File -> Save Study".

In order to have a smaller ODM file, one can use the "Edit -> Clean" menu.
The following window will appear:



Using this wizard, one can first view e.g. which StudyEvents are not referenced (by the Protocol element), and then remove them from the study design if desired.
Similarly for CodeLists and ItemDefs, one can inspect which ones are not used, and remove them if desired.

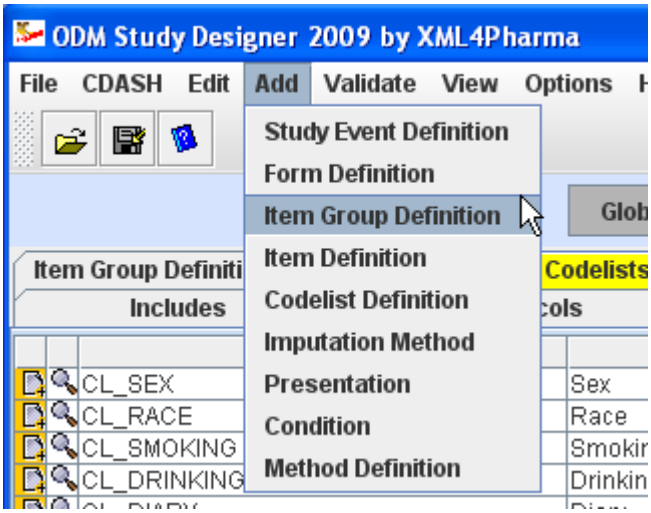
Remark that "cleaning up" is potentially dangerous when you are not ready yet with your study

design. So it is always wise to first make a “save” of everything, before cleaning up the study design.

6. Other small improvements and new features

- validation is now possible when loading an existing ODM file with a study design.
  When the user loads an existing ODM file with study design information, the user is asked whether validation should be performed immediately.

- Faster navigation using the “Add” menu.
  A new menu “Add” has been added to allow faster navigation. It allows to quickly add a new StudyEvent, Form, ..., etc. without needing to select the corresponding tab.



When selected, the system immediately selects the corresponding tab, adds a new row to the table when necessary, and positions the cursor in the first cell of the row.

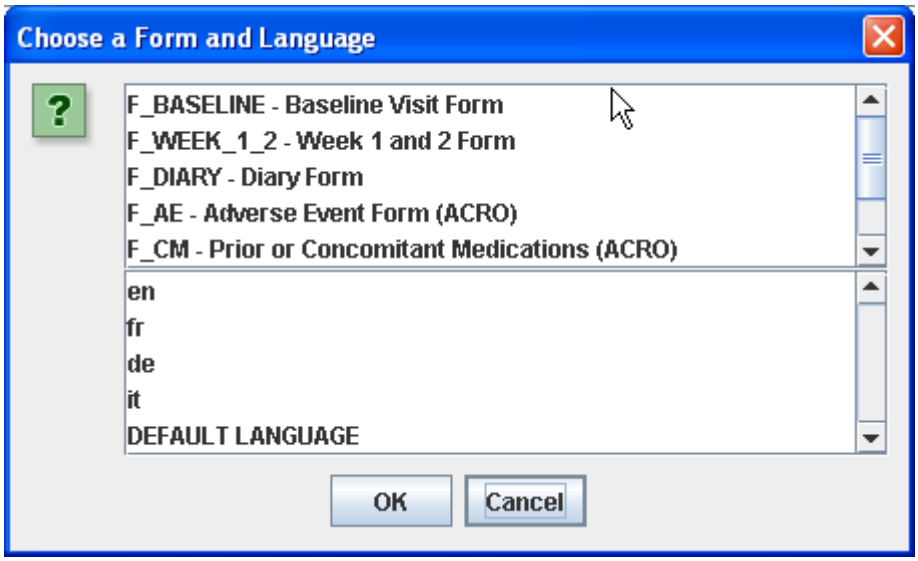
New features of version 2008-R1

Version 2008-R1 of the ODM Study Designer has a number of new features. These include:

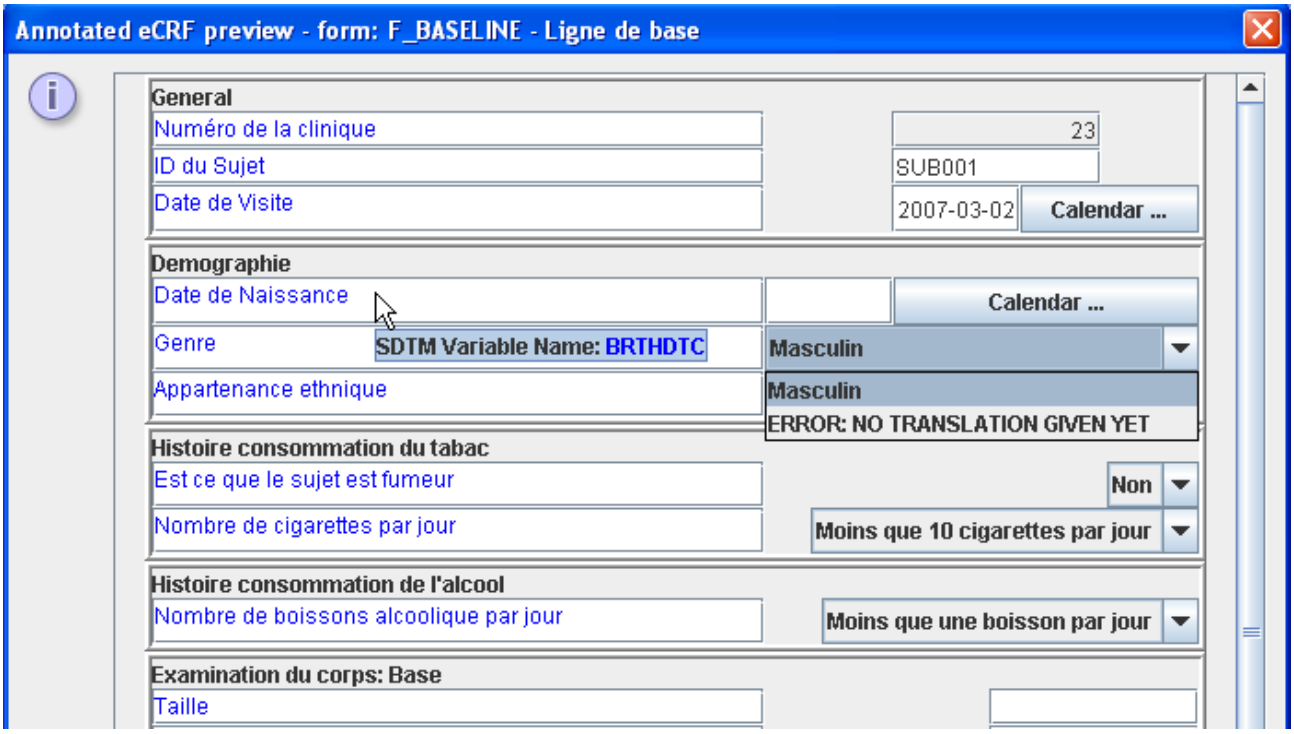
- Faster menu navigation through menu shortcuts
- Full support for ODM 1.3 and all vendor extensions based on ODM 1.3
- Improved validation against the ODM standard
- Create, view and test prototype Annotated CRFs

In version 1.1, it is possible to create and test prototype annotated CRFs.
In order to do so, use the menu "View – Annotated CRFs" (Ctrl-E).

You will be presented a list of forms you have created sofar, and a list of all the languages for which translations of items have been given.



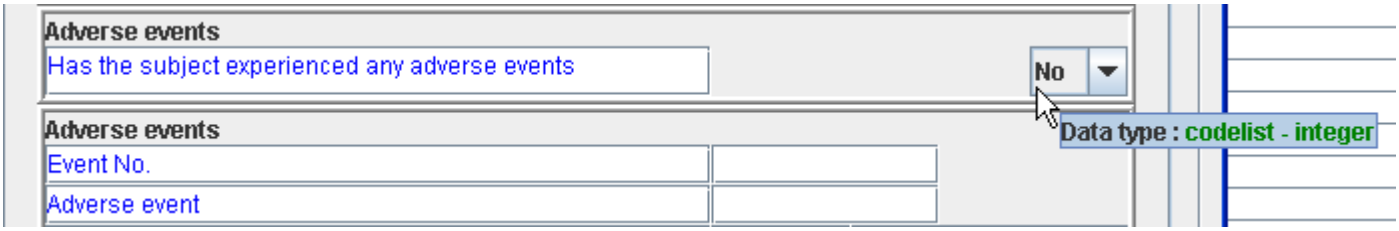
Choose the form of your choice, and the language in wich the selected form should appear. Then click "OK". An prototype annotated version of the eCRF is shown in a separate window. For example:



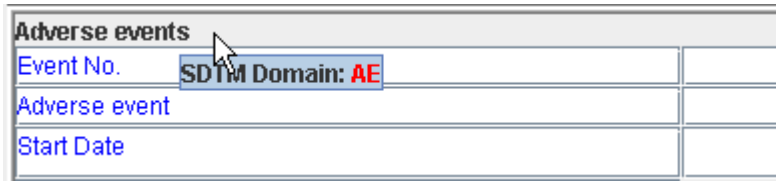
Now you can try out the eCRF. It allows you to check whether you have set the correct data types, whether your codelists are complete, but also whether you provided all necessary translations for the questions, group labels, and codelist items. For example, if you did not provide the french translation for the Item "F" (Female) in the codelist, and you try out the french version of the form, you will see that when you try the use the combobox for "Gender", one of the Items is given as "ERROR: NO TRANSLATION" given yet.

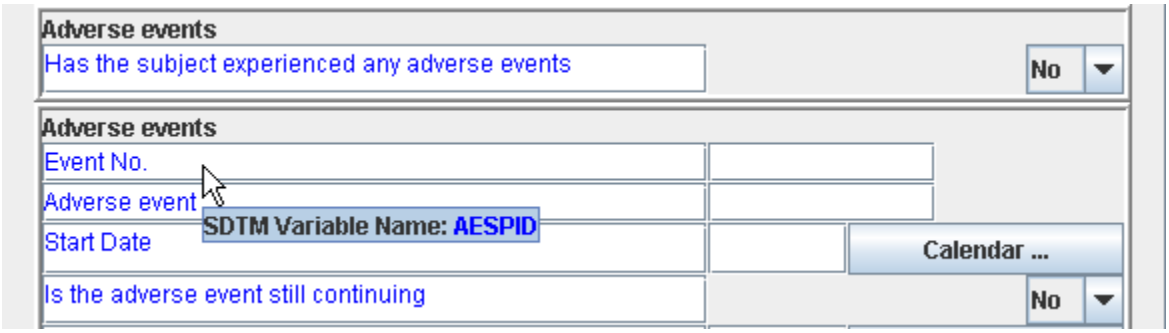
Furthermore, you can test the data types. If for example, you have defined "Site Number" as an "integer", than it should only be possible to enter an integer in the corresponding text field.
For a good number of data types (like dates, times, datetimes, durations), special widgets are provided.

The annotations become visible when one holds the mouse over a label or widget.
When one holds the mouse over a widget, the data type is shown, and whether a codelist is involved.



When holding the mouse over a group label, the associated SDTM Domain is made visible (if one has been defined). Similarly, when one holds the mouse over a question label, the SDTM Variable Name is shown, if one has been defined in the study design (ODM SDSVarName attribute).





P.S. Conditional occurrence of questions (ODM 1.3) has not been implemented yet in the eCRF prototype viewer.