4 Rules

Rules provide a flexible and powerful way for doing advanced types of validation within the OpenClinica EDC system. OpenClinica's Rules module has been designed to be flexible, and makes it easy to define, run, and reuse Rules.

This document explains how these Rules are created, imported, run, and how actions associated with rules are executed. There is also an explanation of OpenClinica OIDs and how they are used with Rules.

Approved for publication by Ben Baumann. Signed on 2014-03-24 9:54AM

Not valid unless obtained from the OpenClinica document management system on the day of use.

4.1 Definitions and Acronyms

ActionType = Actions include DiscrepancyNoteAction, EmailAction, ShowAction, HideAction, InsertAction, and EventAction. See descriptions of each, below:

DiscrepancyNoteAction logs a discrepancy note on the target item if the rule expression evaluates to true or false.

EmailAction sends an email to a specified email address if the rule expression evaluates to true or false.

ShowAction shows a particular field(s) that is defined in the DestinationProperty

HideAction hides a particular field(s) that is defined in the DestinationProperty

InsertAction inserts a value into a particular field that is defined in the DestinationProperty

EventAction schedules a future event (or events) based on the date of another event

DestinationPropertyOID = The item where the Show, Hide, or Insert action will be applied.

EventDestination = Currently defaulted to STARTDATE, the only property that can be updated as the result of an EventAction rule.

OID = Please see OIDs and Clinical Data Keys

RuleDef = The rule definition, which includes the rule OID, name, description, and expression. This defines the rule, which is then assigned (referenced) in the RuleRef.

RuleRef = The rule definition/expression being used by the assignment is referred to in the RuleRef. This is an OID for the particular RuleDef. Each RuleRef may have multiple ActionTypes.

RunOn = For all actions except EventAction, the parameter that defines when the ActionType will execute. The current phases include InitialDataEntry, DoubleDataEntry, AdministrativeEditing, and Batch.

RunOnStatus = For EventAction, the parameter that defines when the EventAction will execute. The current status values include not_scheduled, scheduled, data_entry_started, completed, skipped, and stopped.

Target = The target is the item where an action will be fired. This is a single item in a CRF. When this item is encountered in the CRF and the user selects the Save button, the system will execute the actions associated with a rule.

ValueExpression = A calculation or other expression that defines what will be populated in the DestinationProperty (for an InsertAction) or EventDestination (for an EventAction).

4.2 OID Overview

What are OpenClinica Object Identifiers (OIDs)?

OIDs are identifiers used in OpenClinica to uniquely identify various entities. These identifiers are generated by OpenClinica and made available to Users on multiple screens. OIDs can be found by navigating the user interface (UI) or by downloading the study metadata file. Data import/export and Rules utilize these OIDs.

OpenClinica generates OIDs for the following entities

- Studies
- Sites
- Study Event Definitions
- Study Subjects
- Item Groups
- Items
- CRFs and CRF Versions
- Rule Definitions

When building Rules, we will focus on the OIDs and universal objects that are specific to rules creation which are:

- Study Event Definition OIDs
- CRF and CRF Version OIDs
- Item Group OIDs
- Item OIDs

In addition to the above OIDs, OpenClinica 3.3 introduced two additional universal objects for use with Rules using an EventAction - STARTDATE and STATUS. These two objects are available in all studies, and the object name does not change, so every study has a STARTDATE object and a STATUS object available for use.

How OIDs are generated :

When OpenClinica generates OIDs, there is a formula that is used for each of the different entities. All OIDs of the system must be alpha numeric, they must be in all uppercase, they may not contain whitespaces but may use underscores. Often, the OID will reflect the entity name and may contain additional identifiers to ensure uniqueness within the study.

For Study Event Definition OIDs:

- All Study Event Defnitions OIDs begin with SE_
- Open Clinica will use the characters of the Study Event Definition name (removing any whitespaces or special characters), up to 28 characters
- If the first 28 characters of the Study Event Definition name match the first 28 characters of another Study Event Definition Name, the system will append the OID with an underscore and a 3 digit, randomly generated number. The system does this to ensure uniqueness
- *Example:* Study Event Definition Name = Baseline Visit Study Event Definition OID = SE_BASELINEVISIT

For CRF and CRF Version OIDs:

- CRF OIDs:

Object	3.0.2 and beyond
CRF	12
CRF Version	10
Study Event Definition	28

As mentioned above users can view these OIDs on various screens. Heres a guide to find them.

- Study Event Definitions
  - View OIDs by going to Tasks > Build Study > Event Definitions > View
- Study Subjects
  - View OIDs by going to Subject Matrix and clicking on the View icon for a specific subject.
- Item Groups & Items
  - View OIDs by going to Tasks > Manage and Monitor Data> CRFs click on the View icon of the (original) CRF and then click on the Metadata icon of a specific version.
- CRFs and CRF Versions
  - View OIDs by going to the > Manage and Monitor Data> CRFs> click on the View icon of the (original) CRF
- Rule Definitions
  - View OIDs by going to Tasks > Study Setup> Manage and Monitor Data> Rules >click on the View icon of a specific rule Assignment

Functional approval by Laura Keita. Signed on 2014-10-03 11:10AM

Approved for publication by Ben Baumann. Signed on 2014-10-11 12:48PM

4.3 Rules Applicability

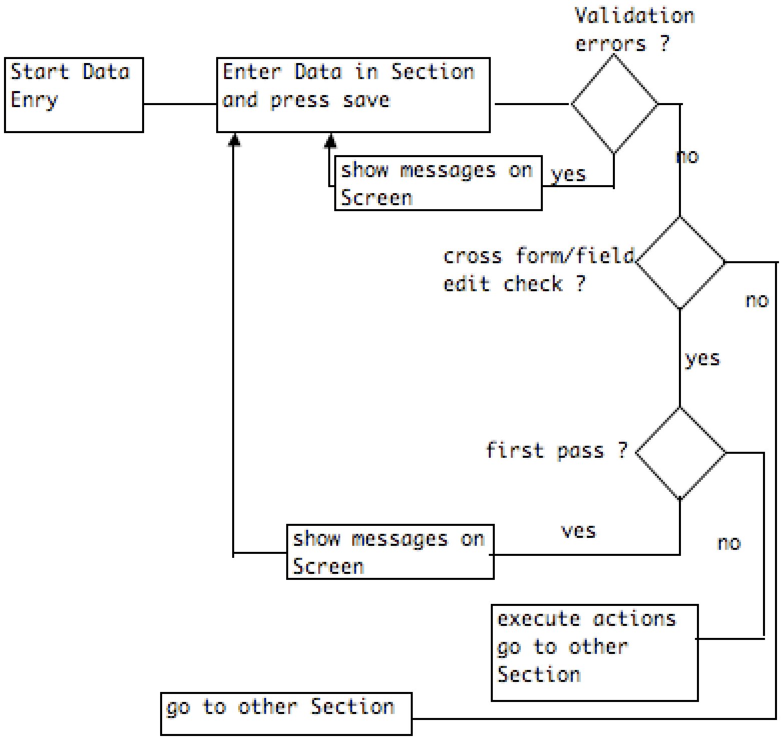
Where do Rules for Cross Field/Form Edit Checks Run?

OpenClinica provides several ways to run Rules for Edit Checks

- Data Entry

When rules for edit checks are configured correctly, they will run automatically when Data Entry is happening. OpenClinica supports multiple forms of data entry which include Initial Data Entry, Double Data Entry and Administrative Data Entry. When building your rules, you can specify during which phase or phases of data entry the rules should execute.

For those familiar with how our data entry pages are setup, you must hit Save on a section to complete the CRF, or move onto another section. Rules for Edit Checks will run when you hit save and there is an item in that section set as a target for a specific Rule. The rule will run and provide a result; the result of the rule will dictate the number of actions to be executed. If the number of actions is greater than zero the user will see the Action's message on the screen. For the first pass the rules have run but the actions have not been executed. Only the messages are displayed on the screen. We also refer to this as a dryRun When the user hits save again, the actions will be executed automatically and the user will be moved to the next section if applicable. See the document below to see the workflow.



- Bulk Execution (Rules in Manage Study Module)

You will see a table, which contains all of the rule assignments specific to the Study you are in. Each row will show the target, which can be made up of (Study Event definition, CRF Name, Item Groups and Items), or just the Item, and the rules/action combination attached to that target. In addition to managing the Rule Assignments / Rules you can also execute them. You can run one RuleDef at a time, but all the actions associated with the RuleDef will execute. Once you have pressed the execute button a dryRun will occur and you will see a summary page previewing what actions will be executed. If you are happy with those results, press the Submit button in which case the actions will be executed.

- Bulk Execution (CRFs in Manage Study Module)
  You can also view and run rules from the Manage CRFs Screen. By choosing to view a specific CRF you will also see a link to Run All the Rules, or View All the Rules for the CRF in that study. Once you have pressed the run button a dryRun will occur and you will see a summary page previewing what actions will be executed. If you are happy with those results, press the Submit button in which case the actions will be executed.

Rule expressions are designed to evaluate to true or false, However, if any of the inputs to a rule expression are missing (null or not entered) or has a missing code (like UNK or NPE), the rull expression will evaluate to 'FAIL' and no actions willl be taken. This treatment of unknown (aka null) values is consistent with how nulls are treated in most databases, where A + NULL = NULL See for example here and here.

Consider a rule such as (a + b + c) gt 100 where items a,b, and c are integers and one or more is missing (null or not entered) or has a missing code (like UNK or NPE). You can confirm using the 'Test Rule' interface (OC 3.0.4 or later) for an expression like the example with 'NPE' or a blank as an input to the test will produce the following results... indicating the rule evaluates to FAIL when there is any non-integer value as an input to the expression.

```
====================================================

Rule Validation: Valid

Expression Evaluates To: FAIL OCRERR_0001 : Logic Error, a and b cannot be used with the PLUS operator.

Actions Fired: N Ran in: 0.022 seconds

Action Summary: No actions will be executed (0017083)

================================================
```

Approved for publication by Ben Baumann. Signed on 2014-03-24 8:43AM

Not valid unless obtained from the OpenClinica document management system on the day of use.

4.4 Creating Rules

Simple single-field validation checks can be created within the CRF Template or by writing a Rule. More complex edit checks that compare values across fields, across forms, or across Events can only be addressed by writing a Rule. Rules are written in XML and then uploaded into OpenClinica. You can also use OpenClinica's Rule Designer, an XML code generator, to create your Rules.

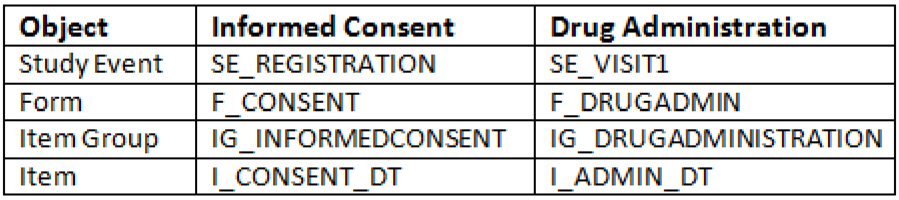
Before writing a Rule, you should have an idea of what you want that Rule to do. For example, if date of drug administration is less than the date of informed consent, you'd like the Rule to create a Discrepancy Note.

We'll use this example to step through the process of writing a Rule.

To create a Rule using XML:

1. As a user with a Data Manager or Study Director role, Go to **Tasks > Build Study** and look up the OIDs for the Study Events, Forms, Item Groups, and Items that will be used in the Rule.

   In our example Study, the following OIDs apply:

Object	Informed Consent	Drug Administration
Study Event	SE_REGISTRATION	SE_VISIT1
Form	F_CONSENT	F_DRUGADMIN
Item Group	IG_INFORMEDCONSENT	IG_DRUGADMINISTRATION
Item	I_CONSENT_DT	I_ADMIN_DT

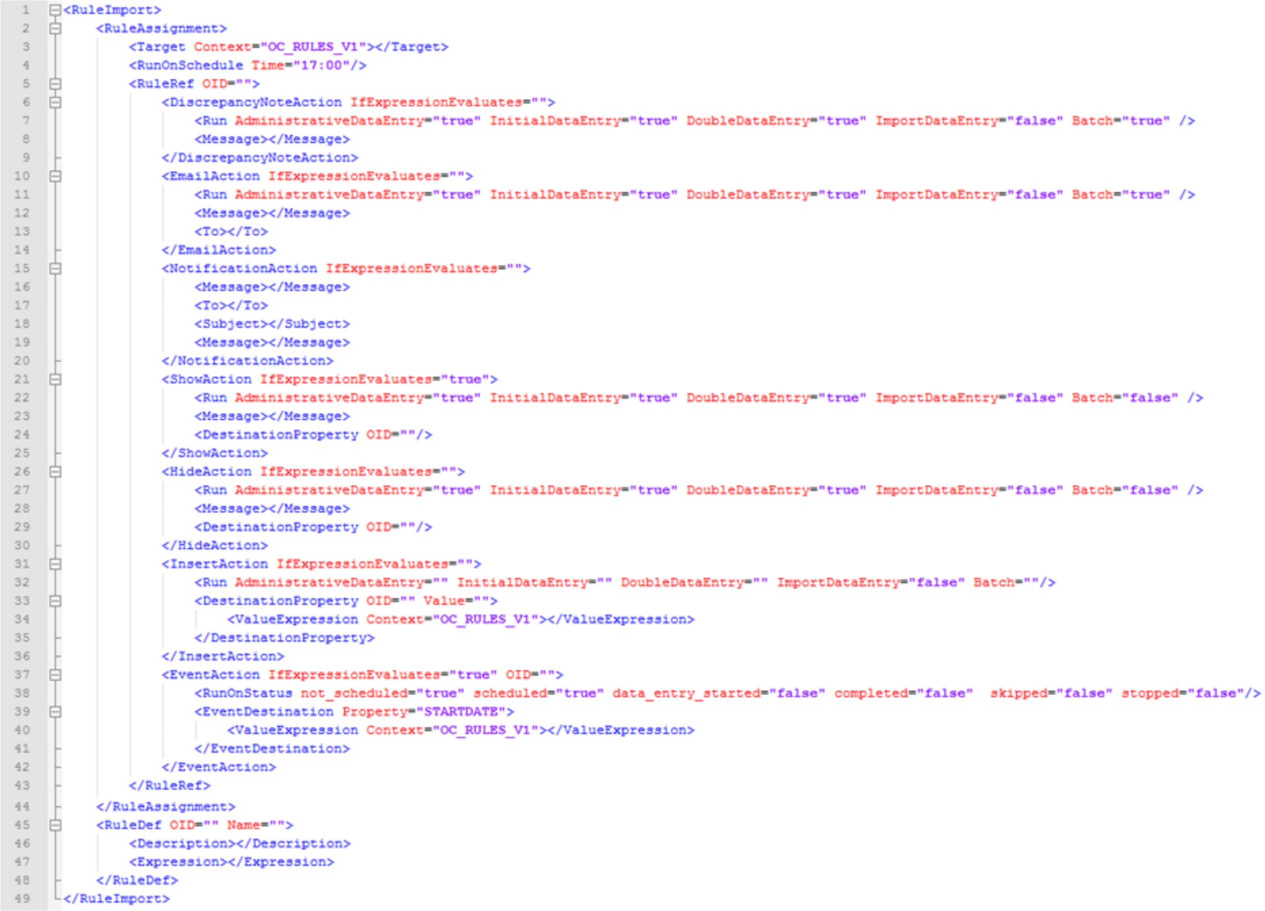
2. Go to **Tasks > Build Study**, and on the **Create Rules** row of the Tasks table, click ➕.

   The Import Rule Data screen displays.

3. Click the **All Actions Without Comments** link and open the file in an XML editor.

   Free XML editors (such as Notepad++ for Windows or TextWrangler for Mac) are available on the internet. Be sure to open the file in an XML editor; if you edit the file in a browser, the Rule will not upload properly.

   The Rule Template, which contains all possible Rule Actions, displays:



```
<RuleImport>
    <RuleAssignment>
        <Target Context="OC_RULES_V1"></Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="">
            <DiscrepancyNoteAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="true" />
                <Message></Message>
            </DiscrepancyNoteAction>
            <EmailAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="true" />
                <Message></Message>
                <To></To>
            </EmailAction>
            <NotificationAction IfExpressionEvaluates="">
                <Message></Message>
                <To></To>
                <Subject></Subject>
                <Message></Message>
            </NotificationAction>
            <ShowAction IfExpressionEvaluates="true">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="false" />
                <Message></Message>
                <DestinationProperty OID=""/>
            </ShowAction>
            <HideAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="false" />
                <Message></Message>
                <DestinationProperty OID=""/>
            </HideAction>
            <InsertAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="" InitialDataEntry="" DoubleDataEntry="" ImportDataEntry="false" Batch=""/>
                <DestinationProperty OID="" Value="">
                    <ValueExpression Context="OC_RULES_V1"></ValueExpression>
                </DestinationProperty>
            </InsertAction>
            <EventAction IfExpressionEvaluates="true" OID="">
                <RunOnStatus not_scheduled="true" scheduled="true" data_entry_started="false" completed="false"  skipped="false" stopped="false"/>
                <EventDestination Property="STARTDATE">
                    <ValueExpression Context="OC_RULES_V1"></ValueExpression>
                </EventDestination>
            </EventAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="" Name="">
        <Description></Description>
        <Expression></Expression>
    </RuleDef>
</RuleImport>
```

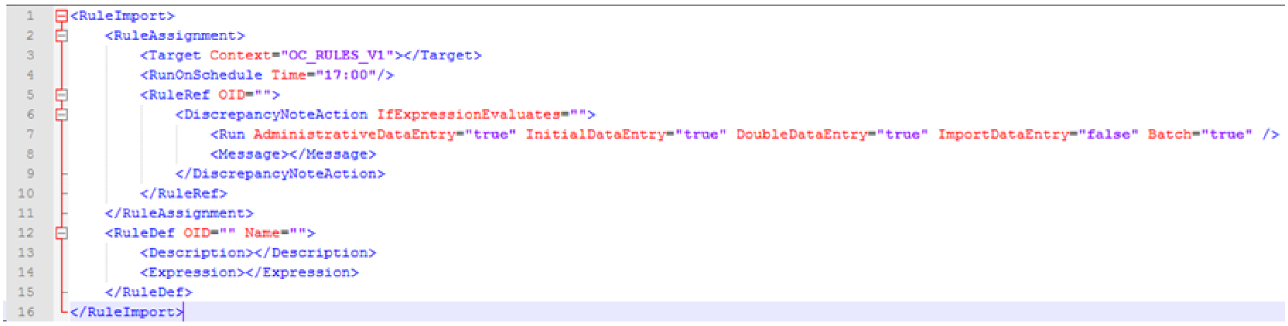
4. To save the template, click **File>Save As** and provide a file location and name for the Rule.

5. Once the Template is saved, there are 5 steps to writing the Rule:
    - Delete the actions you don't want the Rule to perform.
    - Identify the Target.
    - Determine if you want the Rule to run on a specified schedule once per day (currently recommended for Notification Action Rules only; for other Rule Actions, carefully consider the impact of scheduling multiple rules to run on a schedule - and managing that schedule. For example, if you set all your Rules to run at 17:00 hours, performance may be impacted. However, if you have a particularly large form with many Rules associated with it, it may be better to schedule those rules to run on a schedule to increase performance during data entry.)
    - Define the Rule (RuleDef)
    - Reference the Rule (RuleRef)

Delete the Actions You Don't Want the Rule to Perform

In the Rule Assignment portion of the template, delete the lines for the actions that you don't want. For information see Rule Action Types.

For our Discrepancy Note example, the resulting template would be as follows:

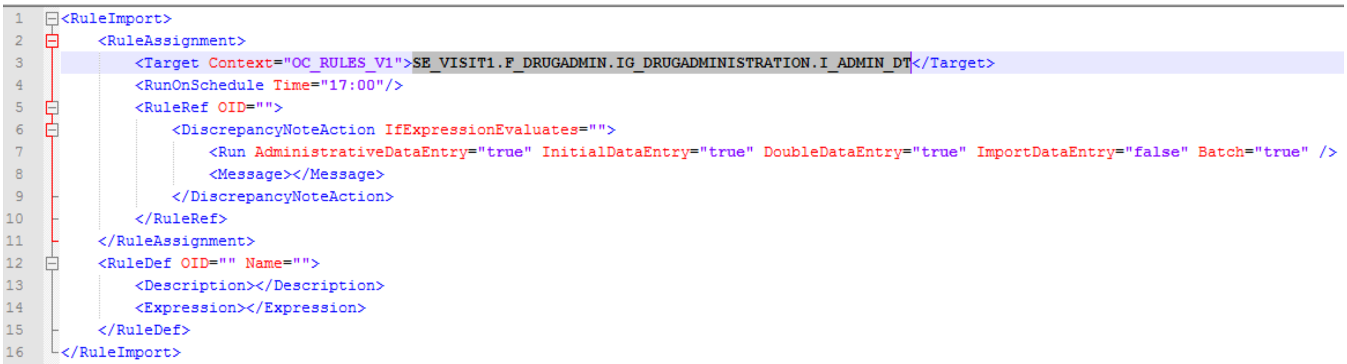


```xml
<RuleImport>
    <RuleAssignment>
        <Target Context="OC_RULES_V1"></Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="">
            <DiscrepancyNoteAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="true" />
                <Message></Message>
            </DiscrepancyNoteAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="" Name="">
        <Description></Description>
        <Expression></Expression>
    </RuleDef>
</RuleImport>
```

Identify the Target

Place the cursor between the opening and closing tags of the **Target** and provide the OID for the Target of the Rule. For a Discrepancy Note action, the Target is the Item that will be flagged with a Discrepancy Note if a specified condition is met.

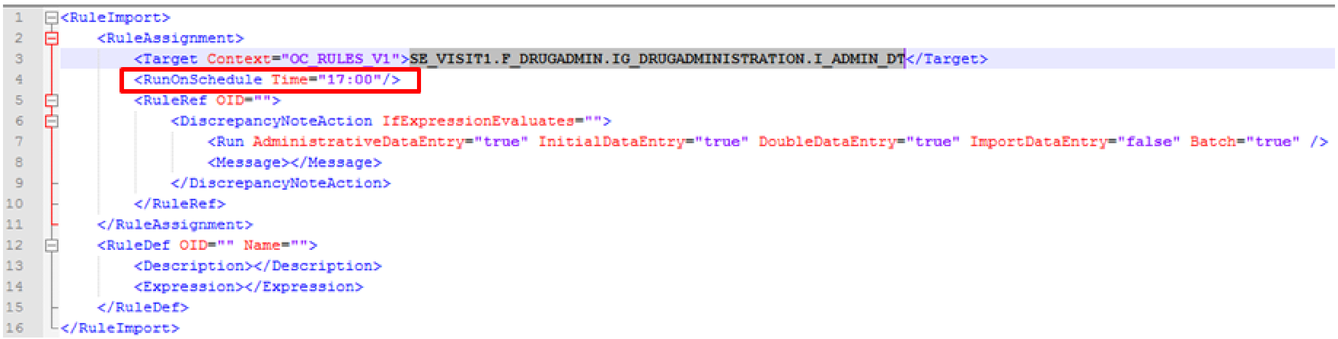
In our example of checking Drug Administration Date against Informed Consent Date, the Target is the Drug Administration Date because it is the second item value entered. The system would not know of an issue with these two items until that value is entered. Since these dates occur on different Events, the Target must begin with the Study Event OID and finish with the Item OID as follows:



```xml
<RuleImport>
    <RuleAssignment>
        <Target Context="OC_RULES_V1">SE_VISIT1.F_DRUGADMIN.IG_DRUGADMINISTRATION.I_ADMIN_DT</Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="">
            <DiscrepancyNoteAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="true" />
                <Message></Message>
            </DiscrepancyNoteAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="" Name="">
        <Description></Description>
        <Expression></Expression>
    </RuleDef>
</RuleImport>
```

Determine if You Want the Rule to Run on a Specified Schedule

Rules will run automatically during data entry when the user clicks Save. If you want your Rule to run at a specified time each day, you can specify that time in the RunOnSchedule tag.



```xml
<RuleImport>
    <RuleAssignment>
        <Target Context="OC_RULES_V1">SE_VISIT1.F_DRUGADMIN.IG_DRUGADMINISTRATION.I_ADMIN_DT</Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="">
            <DiscrepancyNoteAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="true" />
                <Message></Message>
            </DiscrepancyNoteAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="" Name="">
        <Description></Description>
        <Expression></Expression>
    </RuleDef>
</RuleImport>
```

With the setting highlighted above, the Rule will run at 17:00 hours (5:00pm) every day. The time must be entered in 24-hour format (00:00 to 23:59), though OpenClinica will only run the Rule based on the hour values provided. For example, if you specify 17:30, the Rule will run at 17:00.

If the time is left blank, the Rule will run at 20:00. If you remove the parameter completely, the Rule will run based on its run parameters only and will not run on a schedule.

RunOnSchedule is particularly useful for running a Rule based on OpenClinica Participate data since edit checks are limited for Patient Reported Outcome data during data entry. OpenClinica uses the Participant local date-time for RunOnSchedule execution when the Participant local date-time is available. Otherwise, the server date-time is used.

Define the Rule (RuleDef)

In the RuleDef portion of the Template, define the Rule Expression. This is the condition that will trigger the Rule to fire. Mathematic operators and logical operators can be used in the Rule Expression. For more information, see Expression Element.

The Rule Expression is always relative to the Target. Once the full Target is defined, you only need to provide the Item-level reference for the Target, but when referencing any Item outside of that Form and/or Event, the full path for the Item must be provided.

For our example, the Expression is:

I_ADMIN_DT lt SE_REGISTRATION.F_CONSENT.IG_INFORMEDCONSENT.I_CONSENT_DT

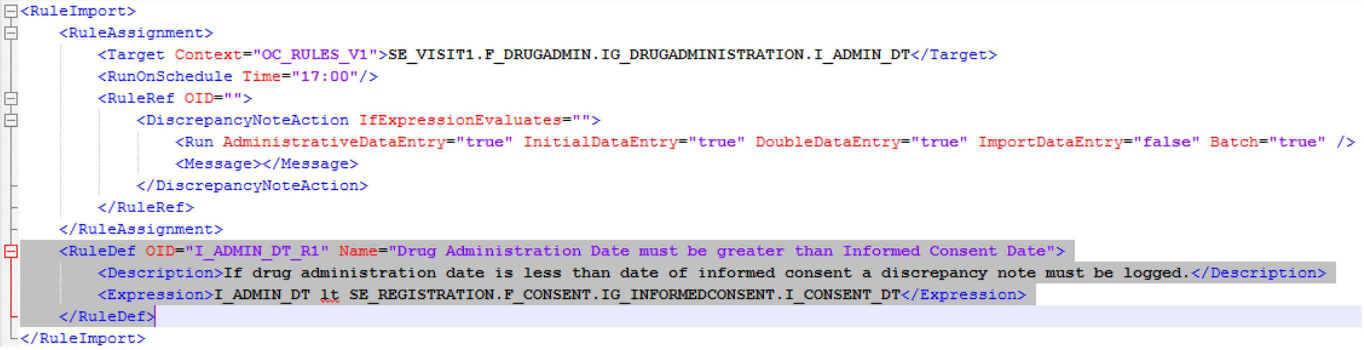
Once the Expression is written, the rest of the RuleDef is a matter of entering text to help identify and describe the Rule, as follows:

Description: A clear, concise explanation of what the Rule is doing. This is presented to users who view Rules using Task>Monitor and Manage Data>Rules, and is only displayed if the user clicks Show More.

Name: A very brief explanation of what the Rule is doing. This is presented to users who view Rules using Task>Monitor and Manage Data>Rules.

OID: The user-defined unique identifier for the Rule. This must be entered in all capital letters and can only contain letters, numbers, and underscore. It must be unique within the Study and can be no more than 40 characters.

For our example, the resulting Rule would be as follows:



```
<RuleImport>
    <RuleAssignment>
        <Target Context="OC_RULES_V1">SE_VISIT1.F_DRUGADMIN.IG_DRUGADMINISTRATION.I_ADMIN_DT</Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="">
            <DiscrepancyNoteAction IfExpressionEvaluates="">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="false" Batch="true" />
                <Message></Message>
            </DiscrepancyNoteAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="I_ADMIN_DT_R1" Name="Drug Administration Date must be greater than Informed Consent Date">
        <Description>If drug administration date is less than date of informed consent a discrepancy note must be logged.</Description>
        <Expression>I_ADMIN_DT lt SE_REGISTRATION.F_CONSENT.IG_INFORMEDCONSENT.I_CONSENT_DT</Expression>
    </RuleDef>
</RuleImport>
```

Reference the Rule (RuleRef)

In the RuleRef portion of the Template, reference the Expression you wrote in the RuleDef and execute the Rule.

Copy the OID of the Rule that you created in the RuleDef portion of the Template into the **RuleRef OID**.

In our example it would appear as follows: <RuleRef OID="I_ADMIN_DT_R1">

Identify when the Rule should run as follows:

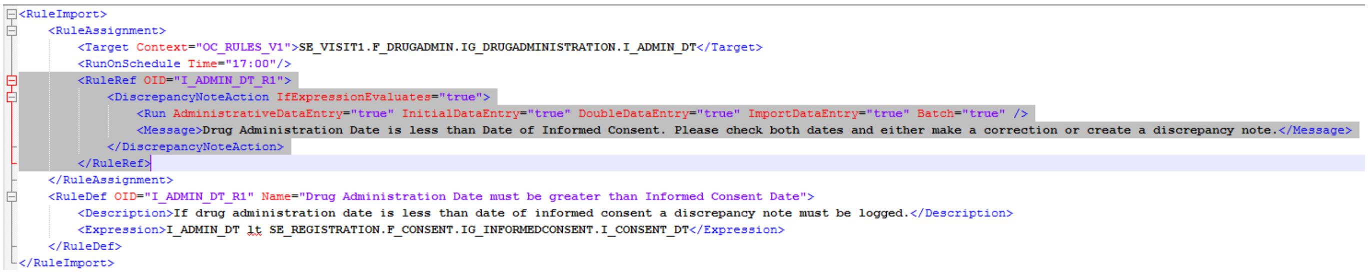
A Rule Expression can either evaluate to true or false. It is best practice to write your Rules to trigger when the expression evaluates to true. There will be times when you will need to write Rules to evaluate to false, but that should be the exception.

For our example, if the Expression is true, a discrepancy note should be created. As such, ensure the first line of the DiscrepancyNoteAction tag is as follows: ***IfExpressionEvaluates="true"*** *Note that "true" must be in lower case.*

The <Run> tag allows you to specify when you want the Rule to be executed. For each option specify true or false (must be in lower case).

For a DiscrepancyNoteAction, you must specify the message that the data entry person will see if the Rule is triggered. Type that message between the opening and closing <Message> tags.

For our example, the RuleRef (and the completed Rule) would be as follows:



```
<RuleImport>
    <RuleAssignment>
        <Target Context="OC_RULES_V1">SE_VISIT1.F_DRUGADMIN.IG_DRUGADMINISTRATION.I_ADMIN_DT</Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="I_ADMIN_DT_R1">
            <DiscrepancyNoteAction IfExpressionEvaluates="true">
                <Run AdministrativeDataEntry="true" InitialDataEntry="true" DoubleDataEntry="true" ImportDataEntry="true" Batch="true" />
                <Message>Drug Administration Date is less than Date of Informed Consent. Please check both dates and either make a correction or create a discrepancy note.</Message>
            </DiscrepancyNoteAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="I_ADMIN_DT_R1" Name="Drug Administration Date must be greater than Informed Consent Date">
        <Description>If drug administration date is less than date of informed consent a discrepancy note must be logged.</Description>
        <Expression>I_ADMIN_DT lt SE_REGISTRATION.F_CONSENT.IG_INFORMEDCONSENT.I_CONSENT_DT</Expression>
    </RuleDef>
</RuleImport>
```

Functional approval by Laura Keita. Signed on 2017-08-10 3:42PM

Approved for publication by Ben Baumann. Signed on 2017-08-10 4:07PM

Not valid unless obtained from the OpenClinica document management system on the day of use.

4.4.1 Rule Assignment and Rule Definition

RuleAssignment

A Rule Assignment executes a Rule Expression for a specific target. It contains the following entities.

- **Target**: The item that will be affected by the Rule or for which the RuleAction is being performed.

- **RuleRef**: The RuleOID that will be executed and specifications for when the Rule should be run.

  Within a RuleAssignment you may have one or more RuleRefs.

  A RuleRef also specifies the Action(s) you want the Rule to perform.
  Out of the box actions by OpenClinica currently include

o DiscrepancyNoteAction

o EmailAction

o NotificationAction

o InsertAction

o ShowAction

o HideAction

o EventAction

- **Run**: This specifies when the Rule should be executed. Specify "true" or "false" (must be in lower case) for each possible status as follows:

  If "false" for any of the following options, the Rule will not execute for the specified setting.

  - **AdministrativeDataEntry**
    If "true" the Rule will run if a user saves a change to the data after the form has been marked "complete."

  - **InitialDataEntry**
    If "true" the Rule will run when the user clicks Save after initial data entry.

  - **DoubleDataEntry**
    If "true" the Rule will run when the user clicks Save after double data entry.

  - **ImportData**
    If "true" the Rule will run when data is imported into the form.

  - **Batch**
    If "true" the Rule will run if a user runs the Rule in batch mode (via Tasks>Monitor and Manage Data>Rules and then click the Run icon OR via Tasks>Monitor and Manage Data>CRFs>View the CRF and click the link Run All Rules for This CRF).

RuleDef

Rule Definitions in OpenClinica are independent entities. The same Rule definitions could be used in multiple assignments. Rule Definition elements are themselves made of multiple elements.

- **RuleOID**
  A user-defined, unique OID for the Rule. This must be entered in all upper-case alphanumeric characters. Underscores are also allowed.

- **Name**
  A brief, user-provided description of what the Rule does. This is only visible on the Manage Rules screen (Tasks>Monitor and Manage Data>Rules).

- **Description**
  A detailed, user-provided description of what the Rule does. This is only visible on the Manage Rules screen if the user clicks the Show More link above the table. (Tasks>Monitor and Manage Data>Rules, and then click Show More).

- **Expression**
  The condition that triggers the Rule action. Expressions can include logical operators, mathematical operators, and can be performed using dates. For more information see Expression Element and Rule Expressions.

4.4.2 Expression Element

The expression element appears in RuleDef Element. It is an expression that evaluates to a result when it is run. For example the expression below might evaluate to true or false depending on the value of ITEM_OID.

```
<Expression> ITEM_OID eq 50</Expression>
```

OpenClinica defines a specific language for creating these expressions. This language supports Logical, Arithmetic, and Conditional operators. We will list those in tables below. Alongside operators OpenClinica supports the user of Integers, Strings and OpenClinica variables in order to do the evaluation.

The expression above is using ITEM_OID, which is an OpenClinica variable and 50, which is a number, with the operator being an equal. OpenClinica out of the box supports the following operators

The Equality and Relational Operators

eq	Equal to	Variables used with this operator could be of any type
ne	Not Equal to	Variables used with this operator could be of any type
ct	Contains	Variables used with this operator could be of any type
gt	Greater Than	Variables used with this operator should be of a number type
gte	Greater Than or Equal to	Variables used with this operator should be of a number type
lt	Less Than	Variables used with this operator should be of a number type
lte	Less Than or Equal to	Variables used with this operator should be of a number type

Some example expressions using the above operators

- ITEM_OID eq YELLOW
- ITEM_OID lt (-15)
- ITEM_OID gt 5

- 5 gte ITEM_OID
- YELLOW ne GREEN
- ITEM_OID ct "2014"

The Conditional Operators

and	and	Variables used with this operator should be of a boolean type
or	or	Variables used with this operator should be of a boolean type

Some example expression using the above operators

- ITEM_OID eq YELLOW and ITEM_OID_2 gt 5
- (ITEM_OID eq YELLOW) and (ITEM_OID_2 gt 5)
- (YELLOW neq GREEN) or (ITEM_OID_2 gt 5)

The Arithmetic Operators

+	Addition	Variables used with this operator should be of a number type
-	Subtraction	Variables used with this operator should be of a number type
*	Multiplication	Variables used with this operator should be of a number type
/	Division	Variables used with this operator should be of a number type

Some example expression using the above operators

- ITEM_OID + 35 eq 50
- ITEM_OID 10 eq 34 AND ITEM_OID_2 gt 45
- ((ITEM_OID 10) eq 34) AND (ITEM_OID_2 gt 45)

Statements can be used in these expressions. Now OpenClinica utilizes what is known as contextual statements, meaning as in the examples above you can put

ITEM_OID and OpenClinica will automatically complete the statement so to make it legal. Lets use an example

```
<Target Context="OC_RULES_V1">SED_OID[ALL].CRF_OID.GROUP_OID.ITEM_OID</Target>
<Expression> ITEM_OID eq 50</Expression>
```

What will actually happen here when this expression is being evaluated is that ITEM_OID will be substituted by

```
SED_OID[ordinal].CRF_OID.GROUP_OID.ITEM_OID
```

In order to support cross form edit checks we actually include a complete statement. One side not regarding that is we cannot use ALL in the expressions.

Dates within Rules

Rules supports the following operators to be used with dates

The Equality and Relational Operators

eq	Equal to	ITEM_OID eq 2008-12-12
ne	Not Equal to	ITEM_OID ne 2008-12-12
gt	Greater Than	ITEM_OID gt 2008-12-12
gte	Greater Than or Equal to	ITEM_OID gte 2008-12-12
lt	Less Than	ITEM_OID lt 2008-12-12
lte	Less Than or Equal to	ITEM_OID lte 2008-12-12

The format of the date included in an expression should be **yyyy-MM-dd .**

Example: January 01, 2010 should be written as : 2010-01-01

You can also use _CURRENT_DATE to compare values against the current server date. For example, with a NotificationAction, to notify Participants of a Form that must be completed on the next day, you could use an expression such as:

<Expression>**SE_OID.STARTDATE eq (_CURRENT_DATE +1) and SE_OID.STATUS ne "complete"**</Expression>

This would send a notification one day in advance of the expected form completion date as long as the form was not already completed by the Participant.

4.4.3 Target Element

We will further explain how the Target element is setup.

Rules are assigned to a particular data item in a CRF, via the Target element. The Target must be a valid Item OID, and can be a full or relative OID path. The full OID path would include Event Definition, optional ordinal, CRF, CRF Version, Group, optional ordinal, and Item. A relative path can also be specified by providing only part of the path, normally just the Item OID. Each RuleAssignment may have multiple RuleRefs.

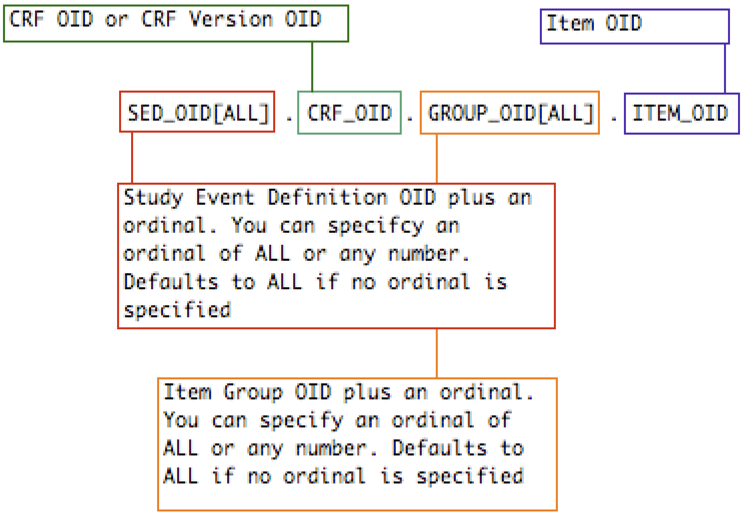
```
<Target
Context="OC_RULES_V1">SED_OID[ALL].CRF_OID.GROUP_OID[ALL].ITEM_OID</Target>
```

As mentioned above the Target element will specify which item the rules will be evaluated against. Although your expression may be evaluating multiple variables, you must specify one variable on which the error message will appear and the action will occur for. Every one of the OIDs must be in upper case. To specify the exact item we need to provide the following information.

- SED_OID[ALL] We need to tell OpenClinica which Study Event definition the Item were after is going to be. We can also specify an ordinal if this happens to be a repeating event, this could be set to ALL or an integer. If you set it to ALL then all event repeats will evaluated. If you specify a specific integer then only that specific event will be evaluated.
- CRF_OID We specified which Event Definition this item is in, now we need to specify which CRF or CRF Version this item is in. Specifying a CRF Version OID will allow the rules to be evaluated against only that specific version. Whereas specifying a CRF OID will allow evaluating against all CRF versions.

- GROUP_OID[ALL] We specified the Study Event Definition and the CRF where the item is, now we need to specify the Group this item is in. This is important because a group might be a repeating or a non-repeating one. If it is a repeating Group OpenClinica allows you to specify if you want the rule to be evaluated against all of the repeats for the variable or a specific ro one of the repeats (a specific row in the repeating group-table) by setting the ordinal to ALL or an integer.
- ITEM_OID Finally we specify which item this rule will be evaluated against. If you have specified a DiscrepancyNoteAction or EmailAction, this is the also the variable that will be specified in the Discrepancy Note or the Email.

As you see we identify the entities by using their OIDs (unique identifiers). You can deduce from the example above that OpenClinica defines its own syntax to create these statements. The separator used throughout all these statements is the .


CRF OID or CRF Version OID
Item OID
SED_OID[ALL] . CRF_OID . GROUP_OID[ALL] . ITEM_OID
Study Event Definition OID plus an ordinal. You can specifcy an ordinal of ALL or any number. Defaults to ALL if no ordinal is specified
Item Group OID plus an ordinal. You can specify an ordinal of ALL or any number. Defaults to ALL if no ordinal is specified

4.4.4 Action Types

An ActionType is specified in the RuleAssignment and determines what should happen when the expression executes. ActionTypes include DiscrepancyNoteAction, EmailAction, NotificationAction, HideAction, InsertAction, ShowAction, and EventAction. A parameter must always be provided that defines when the action will be triggered to execute - either when the expression evaluates to true or to false.

DiscrepancyNoteAction

The DiscrepancyNoteAction is used to automatically log discrepancy notes on the target item.

When triggered, this action provides a message, specified in the rule definition, to the data entry user. The user can choose to update the value(s) he has provided that caused the rule to fire, or he may proceed with the values unchanged by clicking save on the information again. In this case the DiscrepancyNoteAction will log a discrepancy note on the Target with a status of New and a type of Failed Validation Check. The color of the discrepancy note flag will turn to red.

In the Description field of the discrepancy note, the text will start with the RuleDef OID followed by a colon. After the colon the Message defined by the rule creator will display.

DiscrepancyNoteAction is executed after CRF validations are executed and as the data is being saved to the database.

If a DiscrepancyNoteAction has already been executed on a given target, and the value has not changed, a second discrepancy note will not be logged on the item again.

EmailAction

The EmailAction is used to send email to an email address or list of addresses.

When triggered, this action provides a message, specified in the rule definition, to the data entry user. The user can choose to update the value(s) he has provided that caused the rule to fire, or he may proceed with the values unchanged by clicking save on the information again. In this case the EmailAction will send an email to the list of addresses specified in the rule definition. These addresses are specified when the Rule is created. No record of the email action is saved in OpenClinica.

EmailAction is executed after CRF validations are executed and as the data is being saved to the database.

NotificationAction

The NotificationAction is used to send an SMS and/or email notification to a Study Participant (when used in conjunction with OpenClinica Participate), or to send an email notification to any specified email address.

When triggered, this action provides a notification to the specified recipients. To send a notification to Study Participants, address the message to **${participant}**. OpenClinica replaces ${participant} with the communication means for the Participant as specified when the Participant was connected to OpenClinica. The means of communication for Participants can be SMS, email, or both based on the connection information provided. (For more information, see Connect Participants.

The elements of a NotificationAction include the following:

To: ${participant} and/or a specific email address to whom you're addressing the notification. Separate multiple addresses with a comma.

Subject: The subject of the notification (similar to the Subject line of an email)

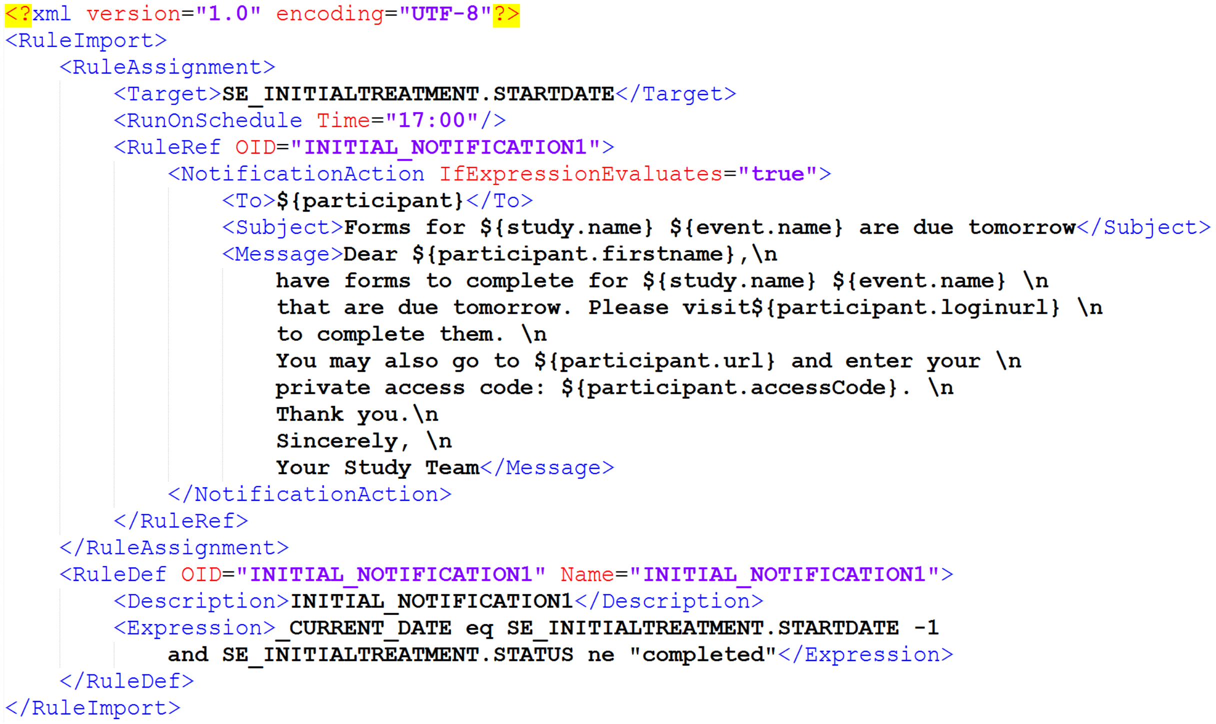
Message: The content of the notification.

Parameters can be passed to the "To" and "Message" elements as follows:

Parameter	**Description**	**To**	**Message**
${participant.firstname}	Participant first name		**x**
${participant.loginurl}	Participant URL with automatic login		**x**
${participate.url}	Participant URL without automatic login (requires an access code)		**x**
${study.name}	The name of the Study as defined in OpenClinica.		**x**

${participant.accessCode}	The single-use code the Participant must use to access OpenClinica Participate.		x
${event.name}	The name of the Event as specified in the Build Study task in OpenClinica.		x
${participant}	The Participant contact information as provided when the Participant was connected to the Study. This may have been a mobile phone number (for SMS notification), email address, or both. OpenClinica automatically sends the notification to any and all means that were provided for the Participant.	x	

For example, the following Rule will send a notification to any Participant who has an Event scheduled for the following day:



```
<?xml version="1.0" encoding="UTF-8"?>
<RuleImport>
    <RuleAssignment>
        <Target>SE_INITIALTREATMENT.STARTDATE</Target>
        <RunOnSchedule Time="17:00"/>
        <RuleRef OID="INITIAL_NOTIFICATION1">
            <NotificationAction IfExpressionEvaluates="true">
                <To>${participant}</To>
                <Subject>Forms for ${study.name} ${event.name} are due tomorrow</Subject>
                <Message>Dear ${participant.firstname},\n
                    have forms to complete for ${study.name} ${event.name} \n
                    that are due tomorrow. Please visit${participant.loginurl} \n
                    to complete them. \n
                    You may also go to ${participant.url} and enter your \n
                    private access code: ${participant.accessCode}. \n
                    Thank you.\n
                    Sincerely, \n
                    Your Study Team</Message>
            </NotificationAction>
        </RuleRef>
    </RuleAssignment>
    <RuleDef OID="INITIAL_NOTIFICATION1" Name="INITIAL_NOTIFICATION1">
        <Description>INITIAL_NOTIFICATION1</Description>
        <Expression>_CURRENT_DATE eq SE_INITIALTREATMENT.STARTDATE -1
            and SE_INITIALTREATMENT.STATUS ne "completed"</Expression>
    </RuleDef>
</RuleImport>
```

Notice that the Rule Expression uses _CURRENT_DATE and compares it to SE_INITIALTREATMENT.STARTDATE to determine that the Event is scheduled for the next day. (In other words, if TODAY is equal to the day BEFORE the visit and the status is not yet "completed", send a notification. If you want to send a reminder one day AFTER the visit, you would use _CURRENT_DATE eq SE_INITIALTREATMENT_STARTDATE +1, meaning that the rule will trigger if _CURRENT_DATE is one day AFTER the visit was scheduled and the status is not yet "completed".)

To force a line break, use n at the end of a line.

ShowAction

The ShowAction is used to show previously hidden CRF items or groups.

Note: A ShowAction Rule must be written in combination with a HideAction Rule. A Rule with a ShowAction will not work properly if there is no HideAction associated with it. Both actions must be created under the same Rule OID.

Within this action type is a DestinationProperty. This property is an OID of either an item or a group. When action is triggered, the items or groups specified will be shown.

To specify more than one item, or more than one group, there must be multiple DestinationProperty OIDs. This can not be a comma separated list of OIDs.

When triggered, and when the CRF items to be shown are in the same section of the CRF, this action provides a message, specified in the rule definition, to the data entry user. The message is used to alert the data entry person of new fields that are shown on their current section. Also, the items that are currently shown will be highlighted in yellow.

If the items are in a different section of the CRF, or another CRF entirely, the user will not see the message nor will they see the items highlighted in yellow to indicate they have been shown.

The Show action is not supported across event definitions.

ShowAction is executed after CRF Validations, and DiscrepancyNote, Email, Insert and Hide actions have been executed. Data has already been saved into the database before the new hidden items are shown.

Note: If the Rule XML ifExpressionEvaluates parameter equals blank (i.e. ifExpressionEvaluates = ""), once imported, the value will default to FALSE.

HideAction

The HideAction is used to hide previously shown CRF items or groups.

Note: A HideAction Rule must be written in combination with a ShowAction Rule. A Rule with a HideAction will not work properly if there is no ShowAction associated with it. Both actions must be created under the same Rule OID.

Within this action type is a DestinationProperty. This property is an OID of either an item or a group. When the action is triggered, the items or groups specified will be hidden. To specify more than one item, or more than one group, there must be multiple DestinationProperty OIDs. This can not be a comma separated list of OIDs.

Items that contain a value in the database can not be set to hidden and will always be shown.

The Hide action is not supported across event definitions.

HideAction is executed after CRF Validations, and DiscrepancyNote, Email and Insert actions have been executed. Data has already been saved into the database before the items are hidden.

InsertAction

The InsertAction is used to insert data into CRF items.

Within this action type is a DestinationProperty. This property is an OID of an item. When the action is triggered, the specified item(s) will have a value inserted. To specify more than one item, there must be multiple DestinationProperty OIDs. This can not be a comma separated list of OIDs.

Within the DestinationProperty OID is the option to provide either a Value or a ValueExpression.

- **Value** is a static value you want inserted into the DestinationProperty. To insert a date, the format must be yyyy-mm-dd (e.g., 2014-12-24).
- **ValueExpression** allows you to calculate a new value or copy a value from one field to another. To use another items value as part of the ValueExpression you must provide a valid Item OID. ValueExpression supports all data types, including date and partial date (pdate).

The Insert action is not supported across event definitions.

EventAction

The EventAction is used to schedule a future event or events. This is also known as scheduling or calendaring, and allows you to schedule one or multiple events based on the current event.

Within this action, the Target is the trigger that fires the rule. There is an EventDestination, which is the STARTDATE of the future event that the rule will schedule, and there is a ValueExpression that calculates the STARTDATE for the future event.

To define an EventAction, you must identify the Target, define the EventDestination, and ValueExpression, and specify the conditions for when to run the Rule (RunOnStatus). Following are three different techniques for implementing an EventAction:

- Schedule all subsequent events based on the date of the first event (**View a sample**)
  - There is ONE Target (trigger) in this option - the Rule is triggered only for that Target
  - If an Event STARTDATE is changed, this Rule does not update future Events

- Schedule all subsequent events based on the date of the current event (**View a sample**)
  - There are MULTIPLE Targets (triggers) in this option - the Rule is triggered for any of the Targets
  - If an Event STARTDATEis changed, this Rule dynamically updates all subsequent events

- Schedule the next event (and only the next event) based on the status of the current event (**View a sample**)
  - This option does not display the entire visit schedule. Initially it only displays the first event. Once the first Event is started or completed, it displays the next event - and only the next event. Once the next Event data entry is started or completed, it displays the next event, and so on.

4.4.5 Testing Expressions

To ensure the logic you provided in a rule is correct, there is a Test Rule feature. This feature simply allows you to confirm that the rule is actually testing the feature you wanted to test. To fully test your rules, it is best practice to enter test data for a test subject.

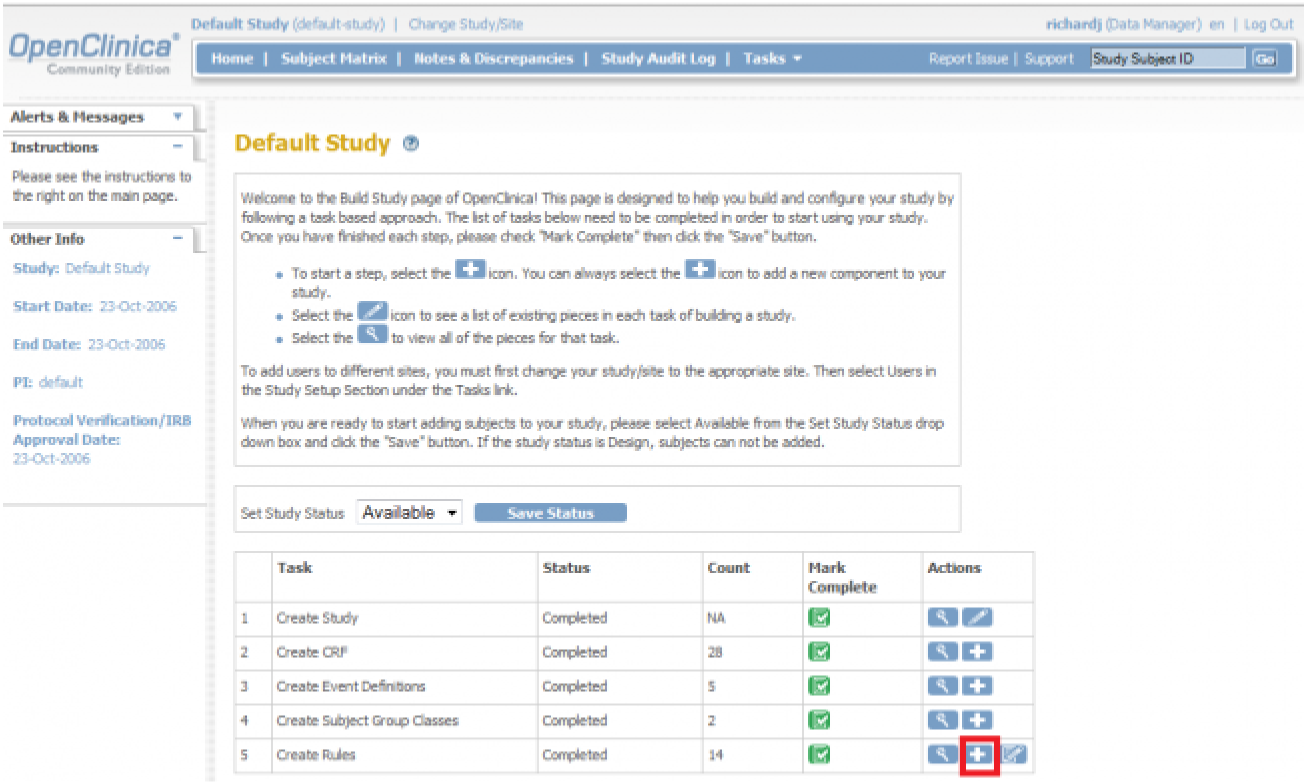
Ensure the rule is uploaded to OpenClinica:

1. Complete writing the rule

2. Log into OpenClinica

3. Select **Tasks > Build Study**

4. On the Create Rules row of the Build Study page, click +



OpenClinica Community Edition
Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go
Alerts & Messages
Instructions

Please see the instructions to the right on the main page.

Study: Default Study

Start Date: 23-Oct-2006

End Date: 23-Oct-2006

PI: default

Protocol Verification/IRB Approval Date: 23-Oct-2006

Default Study

Welcome to the Build Study page of OpenClinica! This page is designed to help you build and configure your study by following a task based approach. The list of tasks below need to be completed in order to start using your study. Once you have finished each step, please check "Mark Complete" then click the "Save" button.

- To start a step, select the + icon. You can always select the + icon to add a new component to your study.
- Select the ✎ icon to see a list of existing pieces in each task of building a study.
- Select the 🔍 to view all of the pieces for that task.

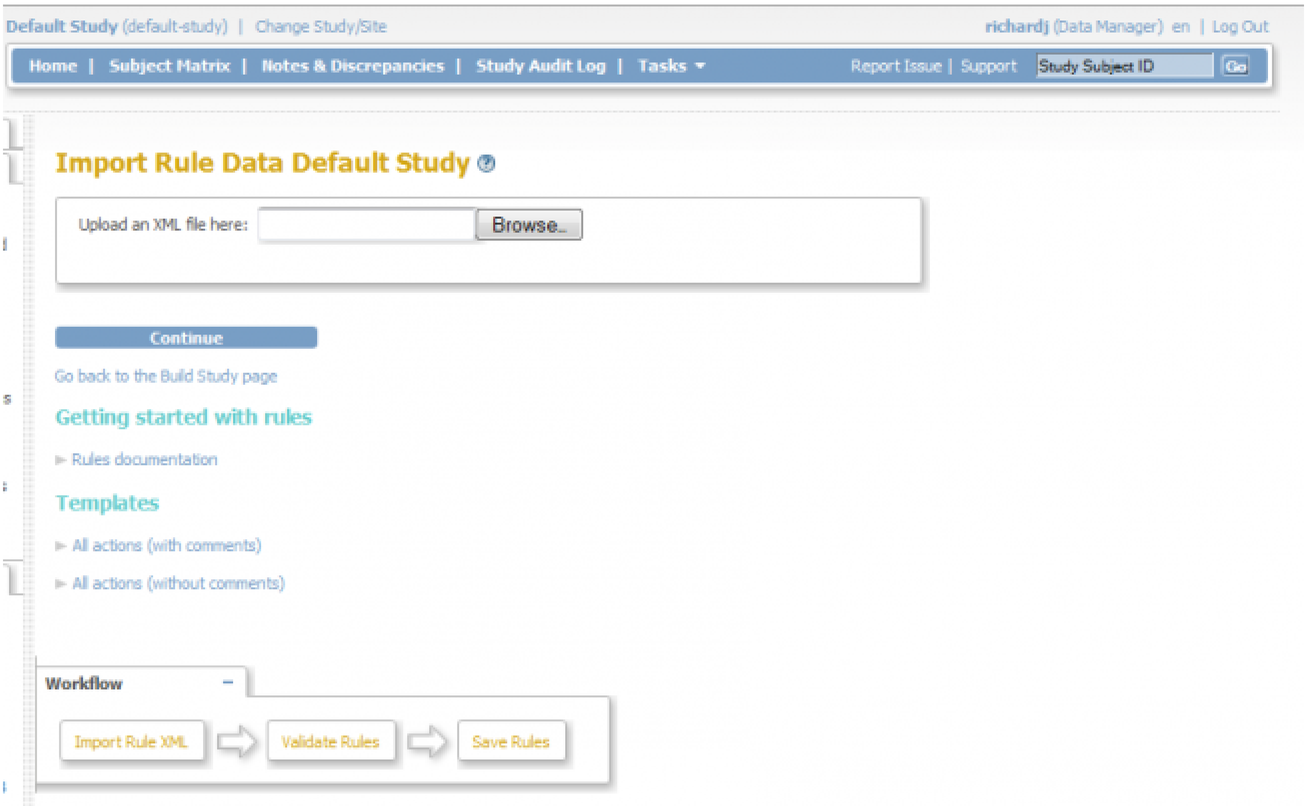
To add users to different sites, you must first change your study/site to the appropriate site. Then select Users in the Study Setup Section under the Tasks link.

When you are ready to start adding subjects to your study, please select Available from the Set Study Status drop down box and click the "Save" button. If the study status is Design, subjects can not be added.

Set Study Status: Available

	Task	Status	Count	Mark Complete	Actions
1	Create Study	Completed	NA	☑	🔍 ✎
2	Create CRF	Completed	28	☑	🔍 +
3	Create Event Definitions	Completed	5	☑	🔍 +
4	Create Subject Group Classes	Completed	2	☑	🔍 +
5	Create Rules	Completed	14	☑	🔍 + ✎

The following window displays:



Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go

Import Rule Data Default Study

Upload an XML file here:

Browse...
Continue
Go back to the Build Study page

Getting started with rules

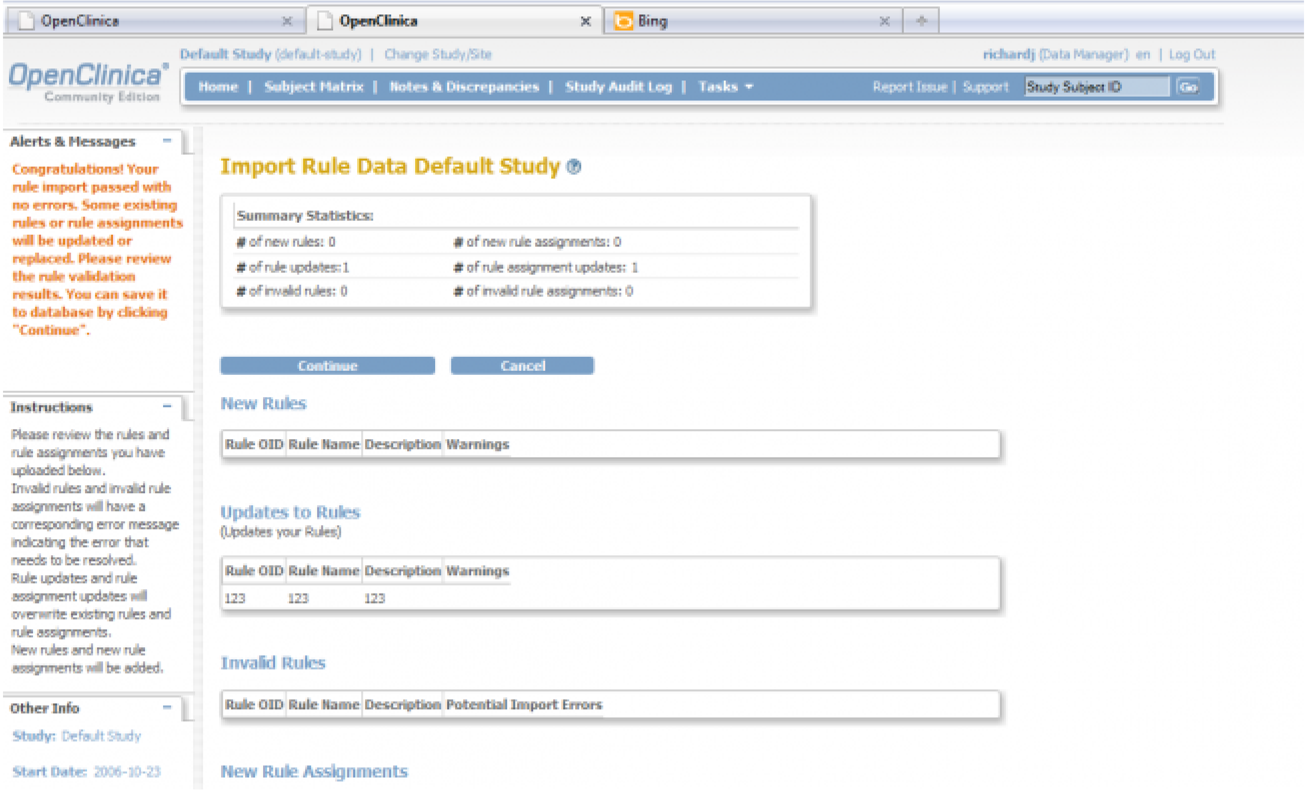
Templates

All actions (with comments)
All actions (without comments)
Workflow
Import Rule XML
Validate Rules
Save Rules

5. Click **Browse**, then locate and **Open** the XML file containing the rule to be uploaded. You may only upload one file at a time.

6. Click **Continue**, this will prompt the system to check the validity of the rule

7. If the rule is valid, a successful confirmation message is displayed. To save the rule in OpenClinica, you must select "Continue." Otherwise, click "Cancel."


OpenClinica
OpenClinica
Bing
OpenClinica® Community Edition
Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go
Continue
Cancel

Alerts & Messages

Congratulations! Your rule import passed with no errors. Some existing rules or rule assignments will be updated or replaced. Please review the rule validation results. You can save it to database by clicking "Continue".

Instructions

Please review the rules and rule assignments you have uploaded below.
Invalid rules and invalid rule assignments will have a corresponding error message indicating the error that needs to be resolved.
Rule updates and rule assignment updates will overwrite existing rules and rule assignments.
New rules and new rule assignments will be added.

Other Info

Study: Default Study

Start Date: 2006-10-23

Import Rule Data Default Study

Summary Statistics:

# of new rules: 0	# of new rule assignments: 0
# of rule updates:1	# of rule assignment updates: 1
# of invalid rules: 0	# of invalid rule assignments: 0

New Rules

Rule OID	Rule Name	Description	Warnings

Updates to Rules
(Updates your Rules)

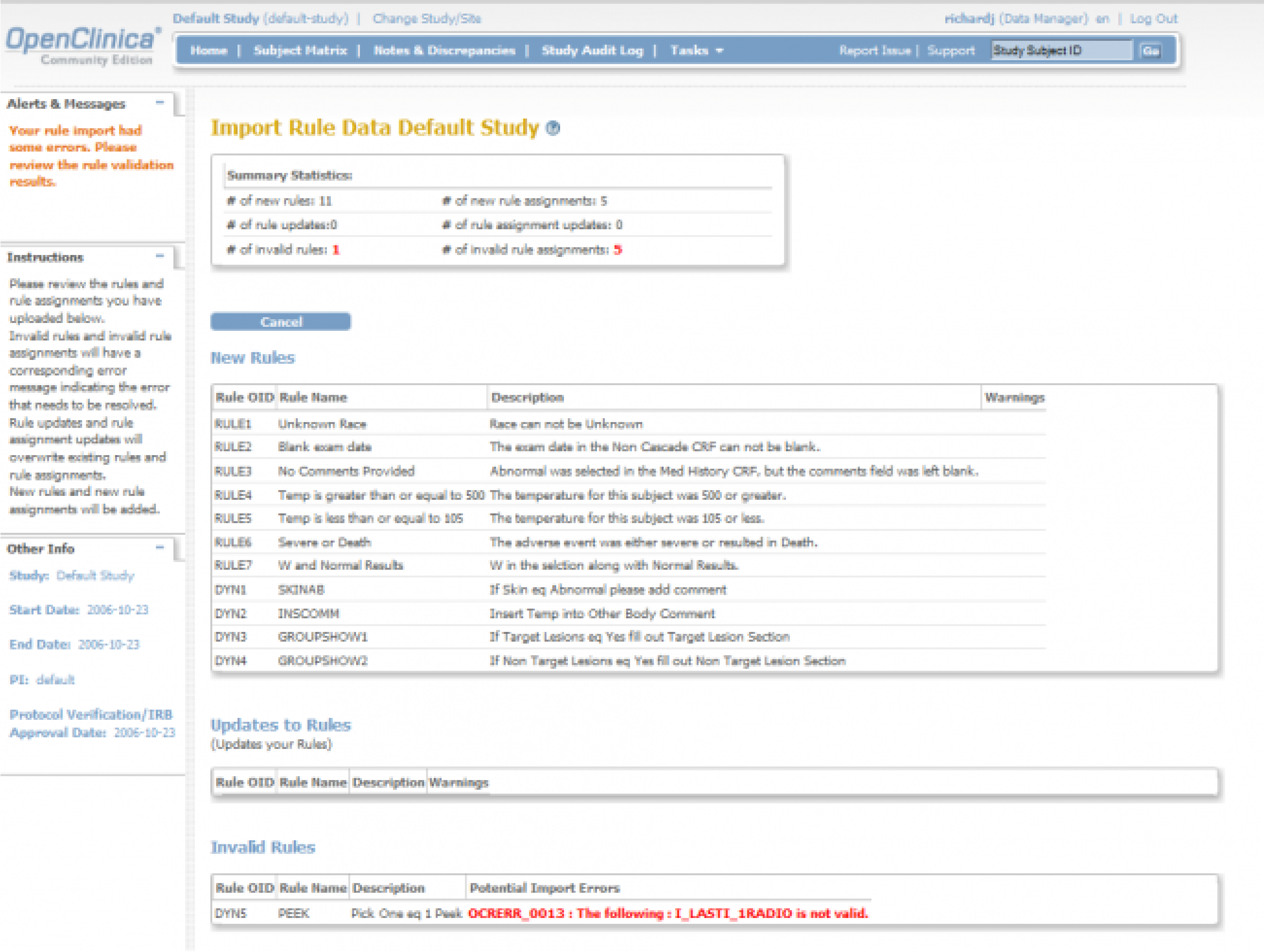
Rule OID	Rule Name	Description	Warnings
123	123	123	

Invalid Rules

Rule OID	Rule Name	Description	Potential Import Errors

New Rule Assignments

8. If the rule in invalid, the system displays an error message containing the source of errors. Note that for invalid rules there is no "Continue" button; "Cancel" is the only option at the bottom of the page.



OpenClinica®
Community Edition
Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go
Alerts & Messages

Your rule import had some errors. Please review the rule validation results.

Please review the rules and rule assignments you have uploaded below.
Invalid rules and invalid rule assignments will have a corresponding error message indicating the error that needs to be resolved.
Rule updates and rule assignment updates will overwrite existing rules and rule assignments.
New rules and new rule assignments will be added.

Study: Default Study

Start Date: 2006-10-23

End Date: 2006-10-23

PI: default

Protocol Verification/IRB Approval Date: 2006-10-23

Import Rule Data Default Study

Summary Statistics:	
# of new rules: 11	# of new rule assignments: 5
# of rule updates:0	# of rule assignment updates: 0
# of invalid rules: 1	# of invalid rule assignments: 5

New Rules

Rule OID	Rule Name	Description	Warnings
RULE1	Unknown Race	Race can not be Unknown	
RULE2	Blank exam date	The exam date in the Non Cascade CRF can not be blank.	
RULE3	No Comments Provided	Abnormal was selected in the Med History CRF, but the comments field was left blank.	
RULE4	Temp is greater than or equal to 500	The temperature for this subject was 500 or greater.	
RULE5	Temp is less than or equal to 105	The temperature for this subject was 105 or less.	
RULE6	Severe or Death	The adverse event was either severe or resulted in Death.	
RULE7	W and Normal Results	W in the selction along with Normal Results.	
DYN1	SKINAB	If Skin eq Abnormal please add comment	
DYN2	INSCOMM	Insert Temp into Other Body Comment	
DYN3	GROUPSHOW1	If Target Lesions eq Yes fill out Target Lesion Section	
DYN4	GROUPSHOW2	If Non Target Lesions eq Yes fill out Non Target Lesion Section	

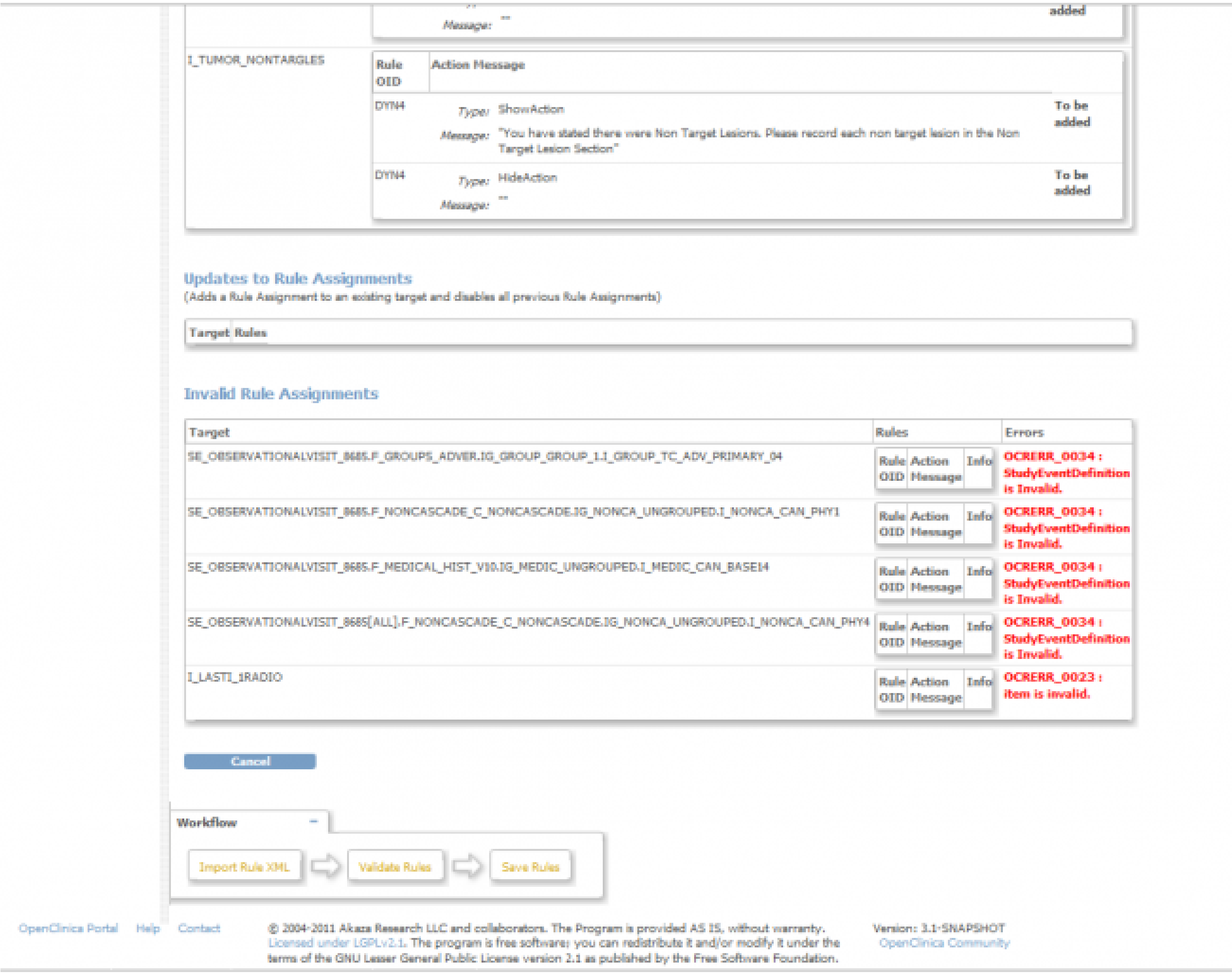
Updates to Rules

(Updates your Rules)

Rule OID	Rule Name	Description	Warnings

Invalid Rules

Rule OID	Rule Name	Description	Potential Import Errors
DYN5	PEEK	Pick One eq 1 Peek	OCRERR_0013 : The following : I_LASTI_1RADIO is not valid.



	Rule OID	Action Message	
		Message: ""	added
I_TUMOR_NONTARGLES	DYN4	*Type:* ShowAction *Message:* "You have stated there were Non Target Lesions. Please record each non target lesion in the Non Target Lesion Section"	To be added
	DYN4	*Type:* HideAction *Message:* ""	To be added

Updates to Rule Assignments

(Adds a Rule Assignment to an existing target and disables all previous Rule Assignments)

Target	Rules

Invalid Rule Assignments

Target	Rules	Errors
SE_OBSERVATIONALVISIT_8685.F_GROUPS_ADVER.IG_GROUP_GROUP_1.I_GROUP_TC_ADV_PRIMARY_04	Rule OID · Action Message · Info	OCRERR_0034 : StudyEventDefinition is Invalid.
SE_OBSERVATIONALVISIT_8685.F_NONCASCADE_C_NONCASCADE.IG_NONCA_UNGROUPED.I_NONCA_CAN_PHY1	Rule OID · Action Message · Info	OCRERR_0034 : StudyEventDefinition is Invalid.
SE_OBSERVATIONALVISIT_8685.F_MEDICAL_HIST_V10.IG_MEDIC_UNGROUPED.I_MEDIC_CAN_BASE14	Rule OID · Action Message · Info	OCRERR_0034 : StudyEventDefinition is Invalid.
SE_OBSERVATIONALVISIT_8685[ALL].F_NONCASCADE_C_NONCASCADE.IG_NONCA_UNGROUPED.I_NONCA_CAN_PHY4	Rule OID · Action Message · Info	OCRERR_0034 : StudyEventDefinition is Invalid.
I_LASTI_1RADIO	Rule OID · Action Message · Info	OCRERR_0023 : item is invalid.

© 2004-2011 Akaza Research LLC and collaborators. The Program is provided AS IS, without warranty. Licensed under LGPLv2.1. The program is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 2.1 as published by the Free Software Foundation.

9. Repeat step 1-7 as necessary to upload a new file/ update an invalid rule.

Testing Rules Expression in OpenClinica

In order to make creating/managing Rules easier, OpenClinica provides you with a tool to execute rules in a test environment. With the Test Rule page you will be able to modify, edit and run the rules for the desired results. You will be presented with a real time response on the validity of your test rules before uploading them into OpenClinica.

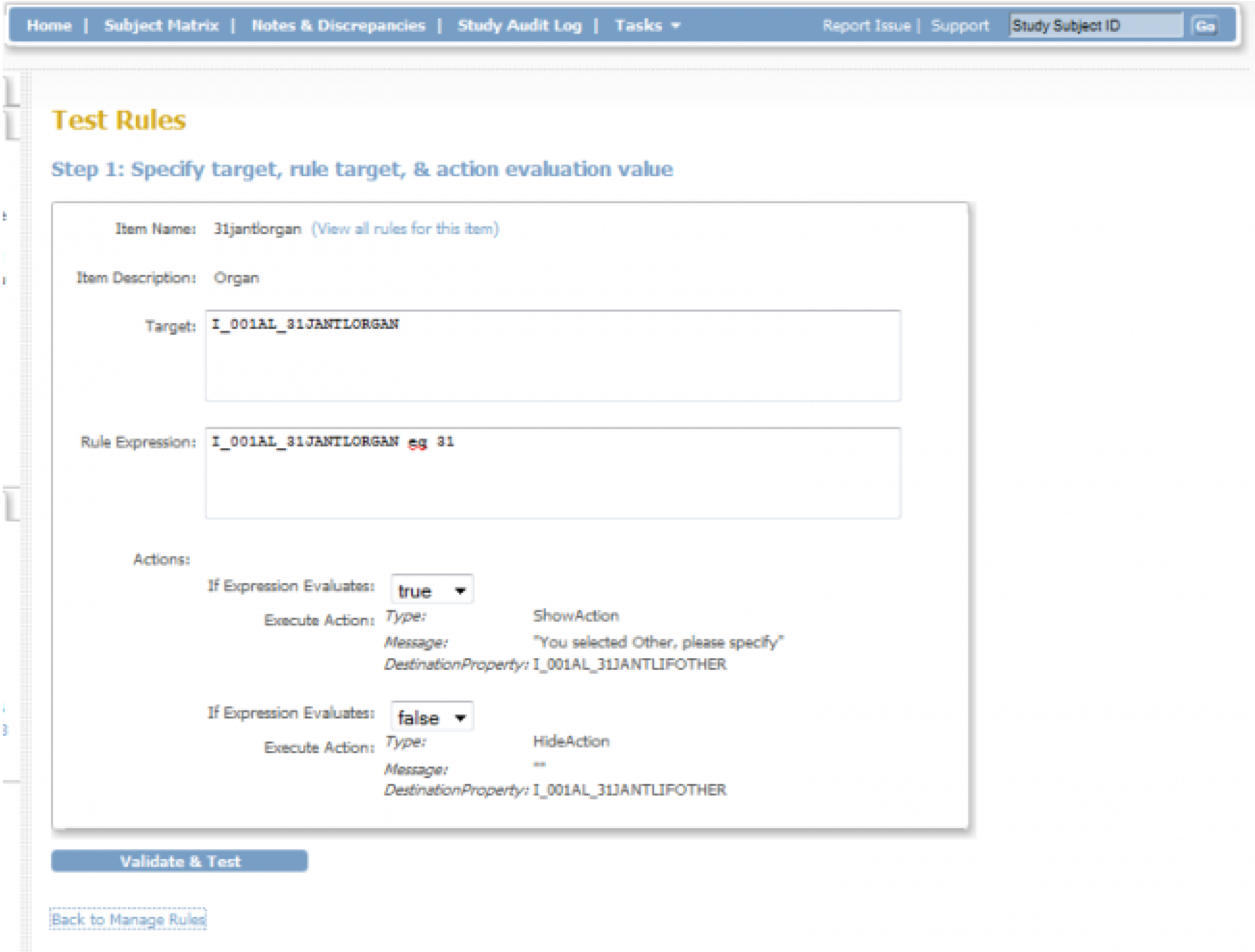
Once the rules get uploaded successfully to OpenClinica, you can verify the validity of the rule by:

1. On the Build Study page, click the View 🔍 icon to access the Manage Rules for <<Study Name>> page. Click here for more information regarding the Manage Rules page.

2. Locate the rule that you wish to validate and test

3. Under the "Actions" column of that rule, click the "Test" ⇨ icon

4. You will be prompted to the Test Rules page (see below). The Test Rules page allows you to enter specific information about the rule you want to test.



Test Rules

Step 1: Specify target, rule target, & action evaluation value

Item Name: 31jantlorgan (View all rules for this item)

Item Description: Organ

Target: I_001AL_31JANTLORGAN

Rule Expression: I_001AL_31JANTLORGAN eq 31

Actions:

If Expression Evaluates: true

Execute Action: *Type:* ShowAction
Message: "You selected Other, please specify"
DestinationProperty: I_001AL_31JANTLIFOTHER

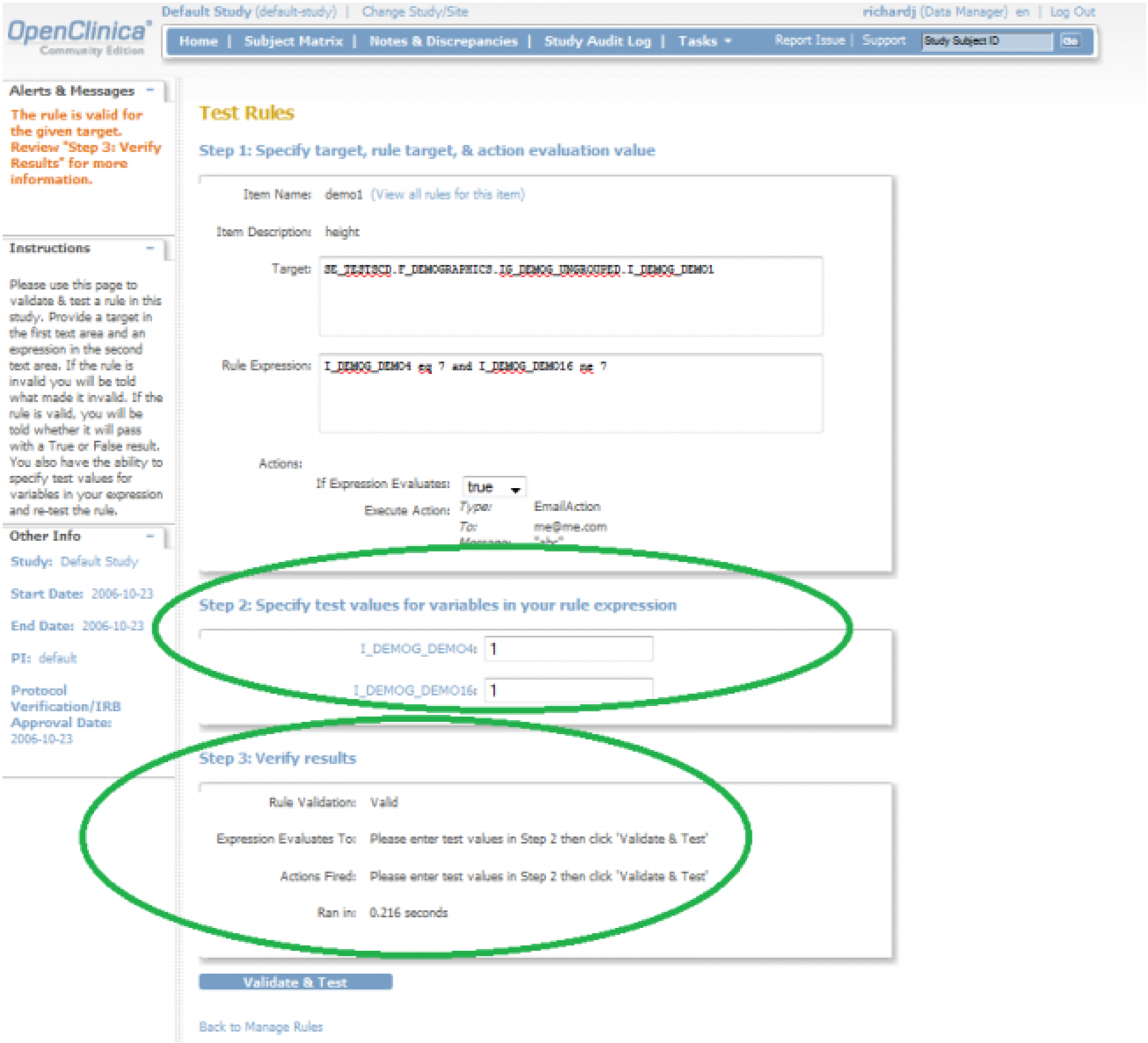
If Expression Evaluates: false

Execute Action: *Type:* HideAction
Message: ""
DestinationProperty: I_001AL_31JANTLIFOTHER

- Enter Target Name: This would be the same statement element as the Target element in RuleAssignment in the XML file
- Enter the Rule Expression: This would be the same element as the expression element in the RuleDef element in the XML file.
- Select the evaluation action from the drop-down true or false

5. Click Validate and Test

6. **Note**: If the rule is valid, you will be told whether it will pass with a True or False result. You also have the ability to specify test values for variables in your expression and re-test the rule. If the rule is valid, then a successful confirmation message is displayed. Also, addditional sections named "Step 2" and "Step 3" are displayed (see below). If a rule is invalid, OpenClinica will provide you with feedback in the form of an error message with information on why your rule was considered invalid.


OpenClinica Community Edition
Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go
Alerts & Messages
Instructions
Other Info
Validate & Test
Back to Manage Rules

The rule is valid for the given target. Review "Step 3: Verify Results" for more information.

Please use this page to validate & test a rule in this study. Provide a target in the first text area and an expression in the second text area. If the rule is invalid you will be told what made it invalid. If the rule is valid, you will be told whether it will pass with a True or False result. You also have the ability to specify test values for variables in your expression and re-test the rule.

Study: Default Study

Start Date: 2006-10-23

End Date: 2006-10-23

PI: default

Protocol Verification/IRB Approval Date: 2006-10-23

Test Rules

Step 1: Specify target, rule target, & action evaluation value

Item Name: demo1 (View all rules for this item)

Item Description: height

Target: SE_TESTSCD.F_DEMOGRAPHICS.IG_DEMOG_UNGROUPED.I_DEMOG_DEMO1

Rule Expression: I_DEMOG_DEMO4 eq 7 and I_DEMOG_DEMO16 ne 7

Actions:

If Expression Evaluates: true

Execute Action: Type: EmailAction

To: me@me.com

Message: "abc"

Step 2: Specify test values for variables in your rule expression

I_DEMOG_DEMO4: 1

I_DEMOG_DEMO16: 1

Step 3: Verify results

Rule Validation: Valid

Expression Evaluates To: Please enter test values in Step 2 then click 'Validate & Test'

Actions Fired: Please enter test values in Step 2 then click 'Validate & Test'

Ran in: 0.216 seconds

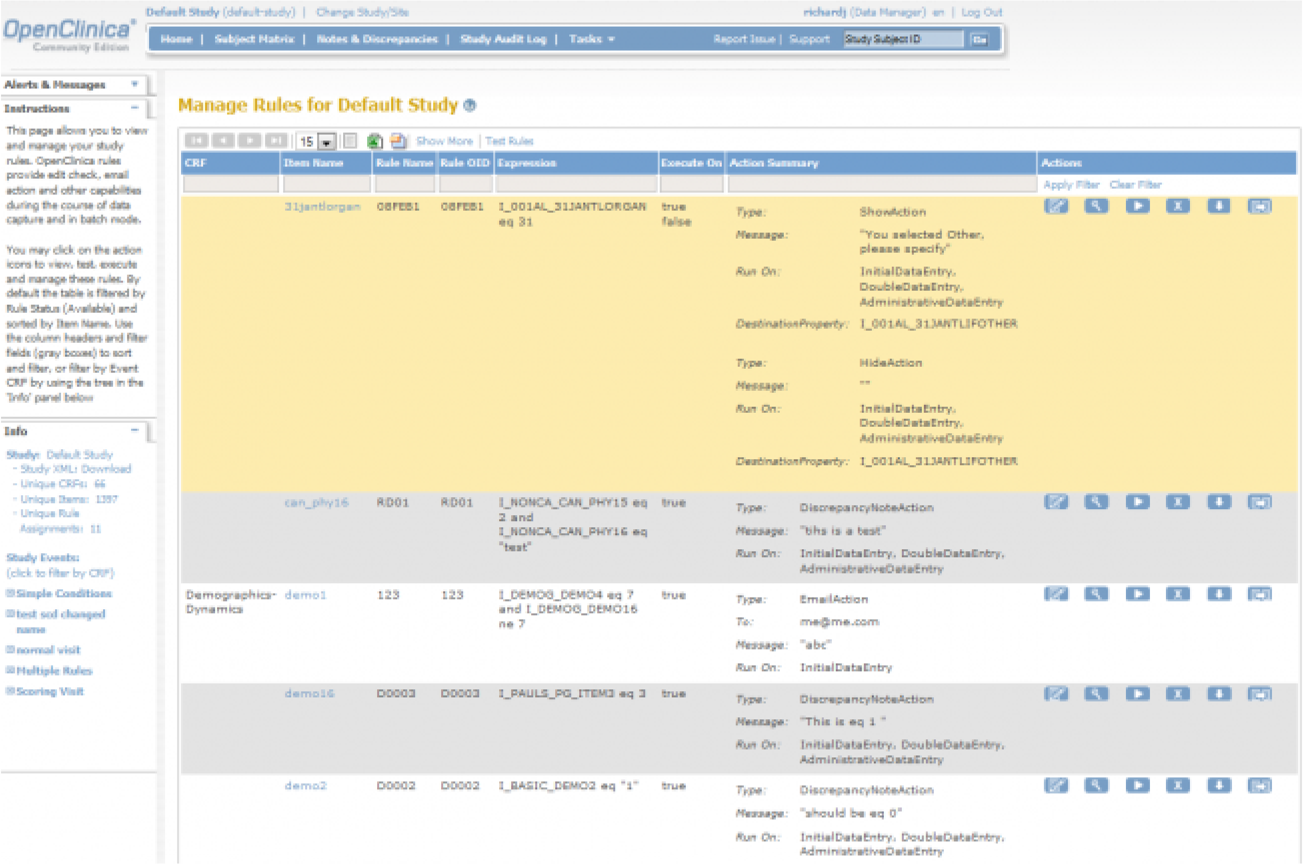
7. Now, you can specify the test values for variables in the rule expression at your discreton and verify if the rule functions as expected by clicking "Validate and Save". Repeat this step as necessary.

4.5 Manage Rules

You can access all the rules that you have created through the Manage Rules page and make changes any time. You can get to the Manage Rules page by selecting Tasks from the navigation bar, and select Rules menu under the Monitor and Manage Data group. This will prompt the Manage Rules on the study level; that is, it will display all of the available rules pertaining to a study. However, you can also access the Manage Rules page on the item level; that is, accessing each rule individually.

4.5.1 Manage Rules (Study Level)

The Manage Rules page allows you to view and manage the Rules for your Study. The list of Rules is filtered by default to display only Available (active) Rules.


OpenClinica
Community Edition
Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go
Alerts & Messages
Instructions
Info
15
Show More | Test Rules
Apply Filter Clear Filter

Manage Rules for Default Study

This page allows you to view and manage your study rules. OpenClinica rules provide edit check, email action and other capabilities during the course of data capture and in batch mode.

You may click on the action icons to view, test, execute and manage these rules. By default the table is filtered by Rule Status (Available) and sorted by Item Name. Use the column headers and filter fields (gray boxes) to sort and filter, or filter by Event CRF by using the tree in the 'Info' panel below

Study: Default Study
- Study XML: Download
- Unique CRFs: 66
- Unique Items: 1397
- Unique Rule Assignments: 11

Study Events:
(click to filter by CRF)
⊞Simple Conditions
⊞test sed changed name
⊞normal visit
⊞Multiple Rules
⊞Scoring Visit

CRF	Item Name	Rule Name	Rule OID	Expression	Execute On	Action Summary	Actions
	31jantlorgan	08FEB1	08FEB1	I_001AL_31JANTLORGAN eq 31	true false	Type: ShowAction Message: "You selected Other, please specify" Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry DestinationProperty: I_001AL_31JANTLIFOTHER Type: HideAction Message: "" Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry DestinationProperty: I_001AL_31JANTLIFOTHER	
	can_phy16	RD01	RD01	I_NONCA_CAN_PHY15 eq 2 and I_NONCA_CAN_PHY16 eq "test"	true	Type: DiscrepancyNoteAction Message: "tihs is a test" Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry	
Demographics-Dynamics	demo1	123	123	I_DEMOG_DEMO4 eq 7 and I_DEMOG_DEMO16 ne 7	true	Type: EmailAction To: me@me.com Message: "abc" Run On: InitialDataEntry	
	demo16	D0003	D0003	I_PAULS_PG_ITEM3 eq 3	true	Type: DiscrepancyNoteAction Message: "This is eq 1 " Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry	
	demo2	D0002	D0002	I_BASIC_DEMO2 eq "1"	true	Type: DiscrepancyNoteAction Message: "should be eq 0" Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry	

View or Restore Removed Rule(s)

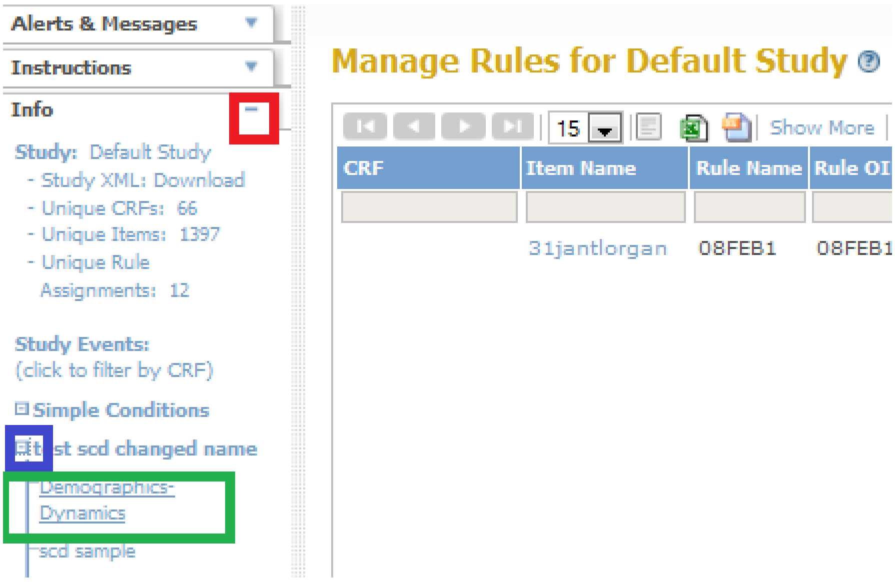
To view and/or restore Rules that have been removed, click the **Show More** link above the table, click in the **Rule Status** column, and click **Removed** from the dropdown list.

Manage Rules page (Study Level)

The column headers (once **Show More** link is selected) and the cells with detail information in the Manage Rules table:

Column	Description
Run On Schedule	True or False, indicating whether the Rule was created to run at a set time on a daily basis using the RunOnSchedule statement.
Run Time	If Run On Schedule is True, this is the time that the Rule is scheduled to run.
Target	The Target element defined in the RuleAssignment in the XML file.
Study Event	The Study Event affected by the Rule.
CRF	The CRF affected by the Rule.
Version	The version of the CRF affected by the Rule.
Group	The Group Label affected by the Rule.
Item Name	The Item Name affected by the Rule.
Rule Name	The user-defined brief description of the Rule.
Rule OID	The user-provided OID that uniquely identifies the Rule.
Rule Description	The user-provided statement describing the actions the Rule will perform.
Expression	The condition that must be met to trigger the Rule
CRF Validations	Yes or No to indicate whether validation exists for the item in a CRF Template.
Execute On	True or False to indicate whether the Rule should execute if the Expression condition is met (TRUE) or not met (FALSE.
Action Type	The action that the Rule will execute. For example, DiscrepancyNoteAction, EmailAction, NotificationAction, InsertAction, etc.
Action Summary	A summary of the Rule characteristics. This includes the action type prompted if the underlying condition is met, when it is run, and Rule properties.
Actions	Role-based actions that can be applied to the Rule. These depend on User Role. See Overview section for more details. **View** Displays the details of this rule **Run** Runs the rule as specified **Remove** Removes a Rule assignment so it will not be executed unless it is restored **Restore** Restores a Rule assignment that had been removed (must click Show More to display this action) **Download** Downloads the XML code for the Rule **Test** Opens the Test Rules page and allows you to do basic testing of the Rule

Use the column headers and filter fields (grey boxes) to sort and filter, or filter by Event CRF by using the tree in the 'Info' panel which is located on the left hand pane (see figure below). By default the table is filtered by Rule Status (Available) and sorted by Item Name.



Info

Study: Default Study
- Study XML: Download
- Unique CRFs: 66
- Unique Items: 1397
- Unique Rule Assignments: 12

Study Events:
(click to filter by CRF)

Simple Conditions
test scd changed name
Demographics-Dynamics
scd sample

Manage Rules for Default Study

CRF	Item Name	Rule Name	Rule OI
	31jantlorgan	08FEB1	08FEB1

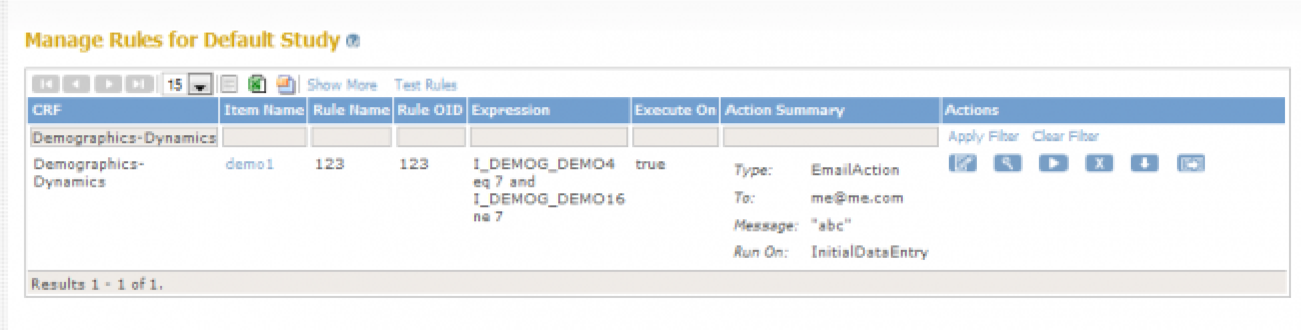
Filtering Rules Using the Info Pane

1. Locate the Info pane. Verify if it is expanded. If it is not, click on the maximize icon on the right (red box).

2. Click on the expand icon to the left of Study Event you are interested in (blue box)

3. A list of CRFs corresponding to the Study Event is populated in a tree hierarchy. Select the CRF (green box)

4. The Rules are filtered based on the CRF and Study Event you selected (see figure below)



Manage Rules for Default Study

CRF	Item Name	Rule Name	Rule OID	Expression	Execute On	Action Summary	Actions
Demographics-Dynamics							Apply Filter Clear Filter
Demographics-Dynamics	demo1	123	123	I_DEMOG_DEMO4 eq 7 and I_DEMOG_DEMO16 ne 7	true	Type: EmailAction To: me@me.com Message: "abc" Run On: InitialDataEntry	

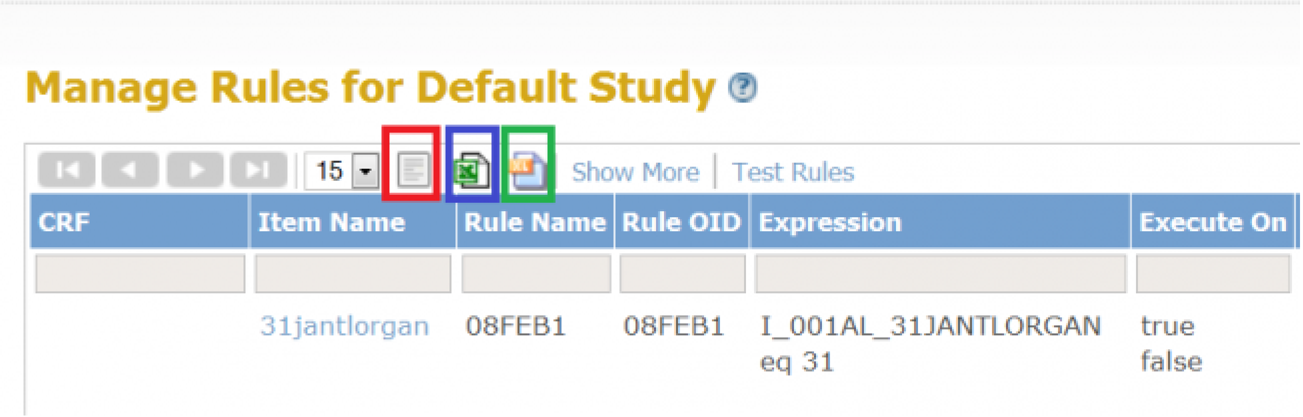
Results 1 - 1 of 1.

Filter Result

The green arrows at the top of the table allow you to navigate through the pages. If you have fewer than 15 Rules in your study the arrows will be greyed out and only one page will be displayed. With the drop-down menu next to the arrows you can choose how many subjects should be displayed on one page (15, 25, or 50). You will also see a Test Rules link which, when you select it, brings you to the Test Rules page. You may click on the action icons to view, test, execute and manage these rules.

4.5.2 Export Rules

On the Manage Rules on a study level page, you also have an option of downloading/exporting rules in CSV, Excel (for data export) and XML format (for rule file). All of the download/ export icons are located on the Manage Rules in <<Study Name>> header in between the drop down menu of how many subjects to be displayed and the Show More/ Hide link. Refer to the screenshot below.



Manage Rules for Default Study

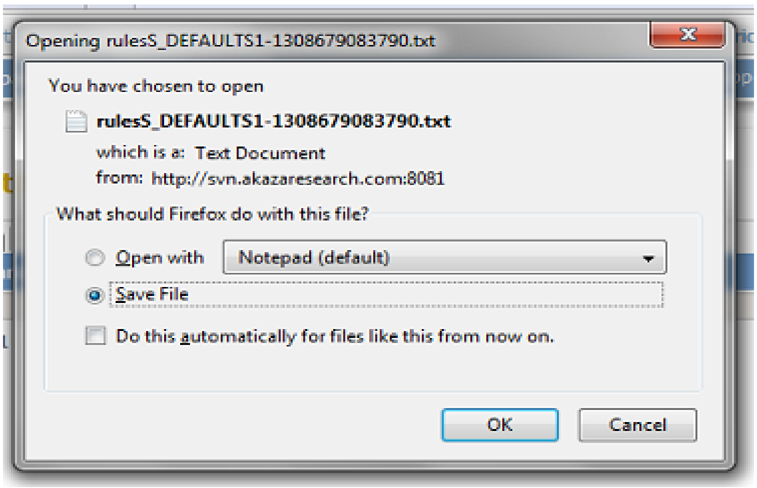
CRF	Item Name	Rule Name	Rule OID	Expression	Execute On
	31jantlorgan	08FEB1	08FEB1	I_001AL_31JANTLORGAN eq 31	true false

Download icons for rules

Left (red box): Exporting Rules in CSV

Middle (blue box): Exporting Rules in Excel format

Right (green box): Downloading the Rules file (in XML)



Opening rulesS_DEFAULTS1-1308679083790.txt

You have chosen to open

rulesS_DEFAULTS1-1308679083790.txt

which is a: Text Document

from: http://svn.akazaresearch.com:8081

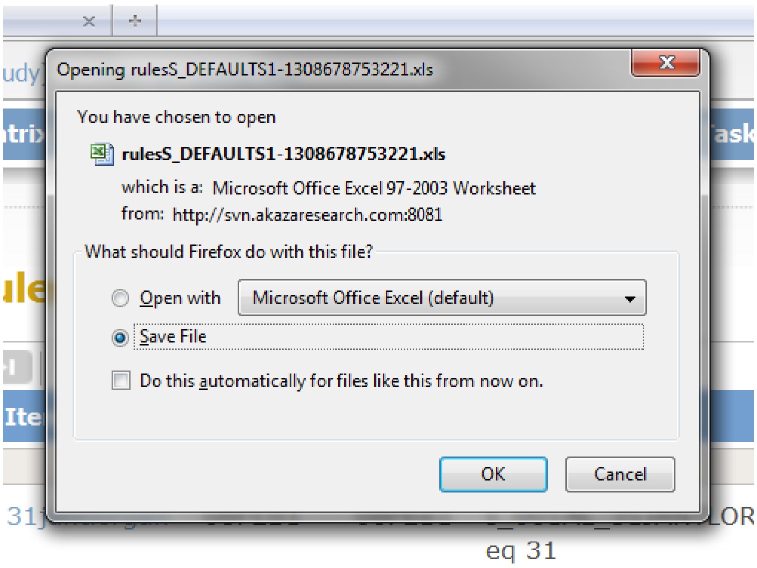
What should Firefox do with this file?

○ Open with Notepad (default)

● Save File

□ Do this automatically for files like this from now on.

Download Rule in CSV format. Note that the rule gets saved as a txt file but it could be opened with Excel.



Opening rulesS_DEFAULTS1-1308678753221.xls

You have chosen to open

rulesS_DEFAULTS1-1308678753221.xls

which is a: Microsoft Office Excel 97-2003 Worksheet

from: http://svn.akazaresearch.com:8081

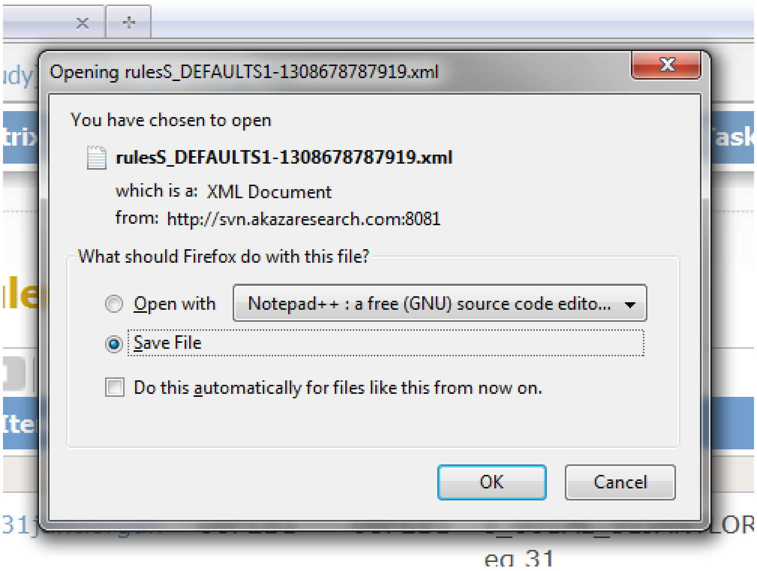
What should Firefox do with this file?

○ Open with Microsoft Office Excel (default)

● Save File

□ Do this automatically for files like this from now on.

Download Rule in XLS format.



Opening rulesS_DEFAULTS1-1308678787919.xml

You have chosen to open

rulesS_DEFAULTS1-1308678787919.xml

which is a: XML Document

from: http://svn.akazaresearch.com:8081

What should Firefox do with this file?

○ Open with Notepad++ : a free (GNU) source code edito...

● Save File

□ Do this automatically for files like this from now on.

Download Rule in XML Format.

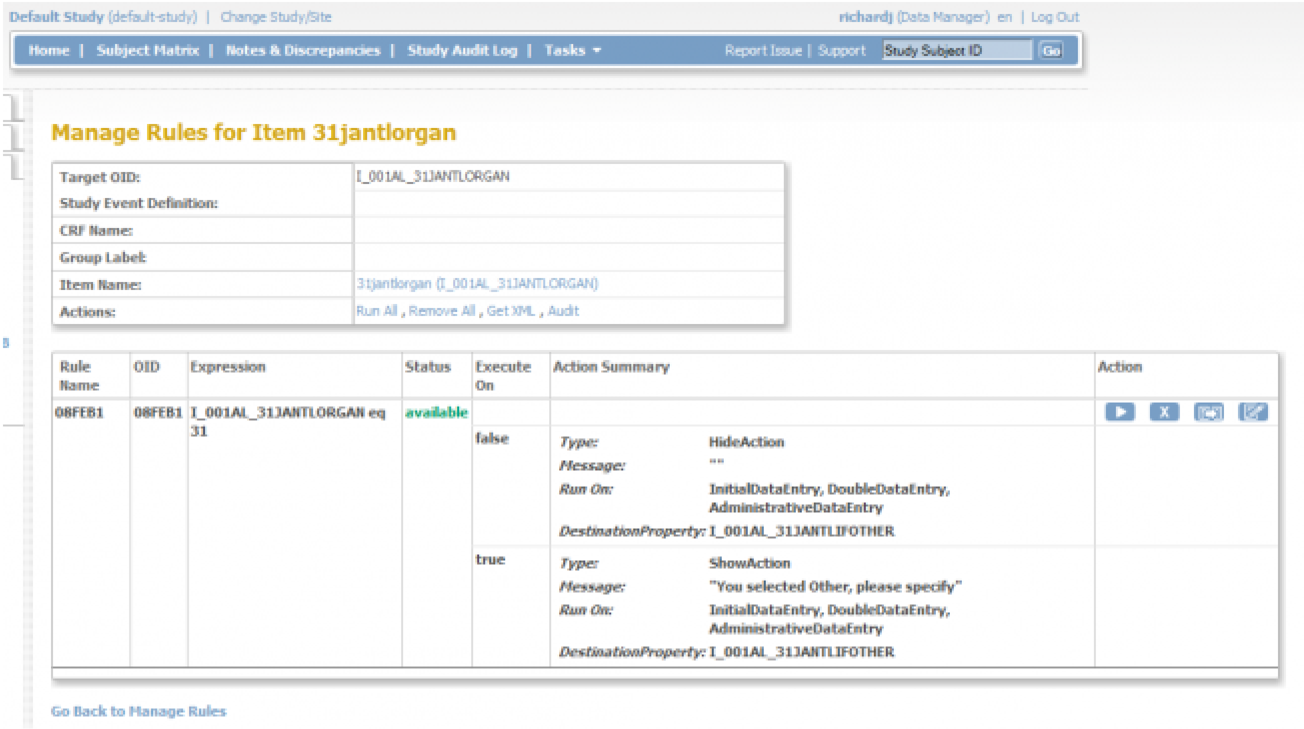
4.5.3 Manage Rules (Item Level)

To view detail information about the available rules for a particular Item select the view icon under the actions column. You will be presented with the Manage Rule for Item [Name] page. On this page you will have specific data about the Item that the rule is targeted for, along with the ability to perform actions on the individual rule by using the Run, Remove or Test icon under the actions column in the Rule table or perform actions on all available rules associated with that particular item.

Manage Rule for Item Page:

1. Target OIDs All the Object indentifiers that are needed to create the statement in which the rule will evaluate
2. Study Event Definition - The Study Event Definition which is affected by the Rule
3. CRF Name -   Indicates detailed information about the name of the CRF
4. Group Label - This is the Group which is affected by the Rule
5. Item Name - This is a link to the Item Metadata page and also provides information on the Item Metadata CRF Version Level Attributes
6. Actions:

- Run All - Allows you to execute all available rules associated with that particular item
- Remove All - Allows you to remove all available rules associated with that particular item.
- Get XML- This enables you to download in an XML format all available rules associated with that particular item.


Default Study (default-study) | Change Study/Site
richardj (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go

Manage Rules for Item 31jantlorgan

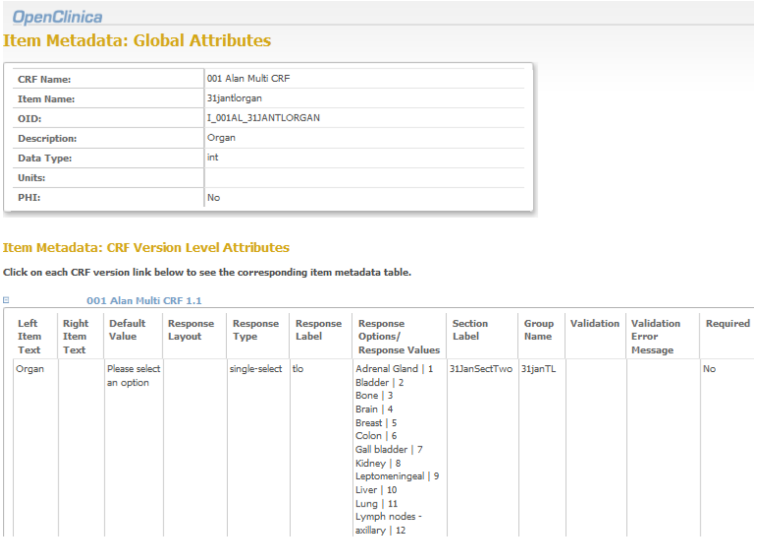
Target OID:	I_001AL_31JANTLORGAN
Study Event Definition:	
CRF Name:	
Group Label:	
Item Name:	31jantlorgan (I_001AL_31JANTLORGAN)
Actions:	Run All , Remove All , Get XML , Audit

Rule Name	OID	Expression	Status	Execute On	Action Summary	Action
08FEB1	08FEB1	I_001AL_31JANTLORGAN eq 31	available			▶ X
				false	*Type:* HideAction *Message:* "" *Run On:* InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry *DestinationProperty:* I_001AL_31JANTLIFOTHER	
				true	*Type:* ShowAction *Message:* "You selected Other, please specify" *Run On:* InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry *DestinationProperty:* I_001AL_31JANTLIFOTHER	

Go Back to Manage Rules

Manage Rules page (Item Level)

When selecting the *Item Name* link on the Manage Rule for Item page the Item Metadata page is displayed giving you information on Item Metadata Global Attributes and Item Metadata: CRF Version Level Attributes.



Item Metadata: Global Attributes

CRF Name:	001 Alan Multi CRF
Item Name:	31jantlorgan
OID:	I_001AL_31JANTLORGAN
Description:	Organ
Data Type:	int
Units:	
PHI:	No

Item Metadata: CRF Version Level Attributes

Click on each CRF version link below to see the corresponding item metadata table.

001 Alan Multi CRF 1.1

Left Item Text	Right Item Text	Default Value	Response Layout	Response Type	Response Label	Response Options/ Response Values	Section Label	Group Name	Validation	Validation Error Message	Required
Organ		Please select an option		single-select	tlo	Adrenal Gland | 1 Bladder | 2 Bone | 3 Brain | 4 Breast | 5 Colon | 6 Gall bladder | 7 Kidney | 8 Leptomeningeal | 9 Liver | 10 Lung | 11 Lymph nodes - axillary | 12	31JanSectTwo	31janTL			No

Item Metadata

4.6 Using Rule Designer

In OpenClinica 3.3, you can build and test rules using OpenClinica and XML code or by using the OpenClinica Rule Designer interface. OpenClinica allows you to write the XML code for your rules, which requires familiarity with the rules template and XML. The Rule Designer provides an interface that allows you to drag-and-drop OpenClinica objects to build your rule, and OpenClinica writes your XML code for you behind the scenes - with this option, XML knowledge is not required!

In this section, we will learn more about the Rule Designer and its capabilities.

Functional approval by Laura Keita. Signed on 2014-06-04 11:00AM

Approved for publication by Ben Baumann. Signed on 2014-06-12 3:51PM

Not valid unless obtained from the OpenClinica document management system on the day of use.

4.6.1 Accessing Rule Designer

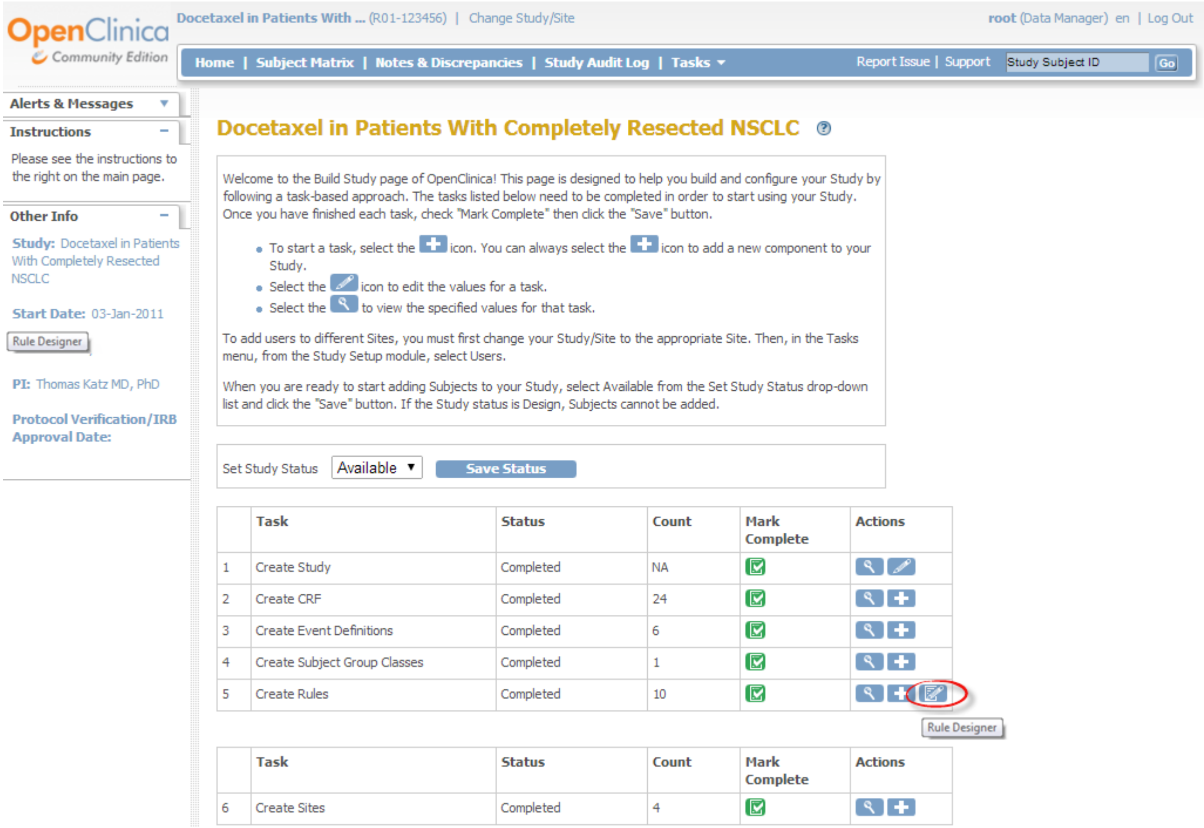
Accessing Rule Designer

You can access Rule Designer from the Build Study page or from the Manage Rules page. Each option is outlined below:

1. To access Rule Designer from the Build Study page:

- Log into OpenClinica.
- On the **Tasks** menu, select **Build Study**.
- In the **Create Rules** row **Actions** column, click 📝 (the Rule Designer icon - see below).

Accessing Rule Designer from the Build Study page



OpenClinica
Community Edition
Docetaxel in Patients With ... (R01-123456) | Change Study/Site
root (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks ▾
Report Issue | Support
Study Subject ID
Go
Alerts & Messages
Instructions

Please see the instructions to the right on the main page.

Study: Docetaxel in Patients With Completely Resected NSCLC

Start Date: 03-Jan-2011

PI: Thomas Katz MD, PhD

Protocol Verification/IRB Approval Date:

Docetaxel in Patients With Completely Resected NSCLC

Welcome to the Build Study page of OpenClinica! This page is designed to help you build and configure your Study by following a task-based approach. The tasks listed below need to be completed in order to start using your Study. Once you have finished each task, check "Mark Complete" then click the "Save" button.

- To start a task, select the ✚ icon. You can always select the ✚ icon to add a new component to your Study.
- Select the ✎ icon to edit the values for a task.
- Select the 🔍 to view the specified values for that task.

To add users to different Sites, you must first change your Study/Site to the appropriate Site. Then, in the Tasks menu, from the Study Setup module, select Users.

When you are ready to start adding Subjects to your Study, select Available from the Set Study Status drop-down list and click the "Save" button. If the Study status is Design, Subjects cannot be added.

Set Study Status: Available ▾ — Save Status

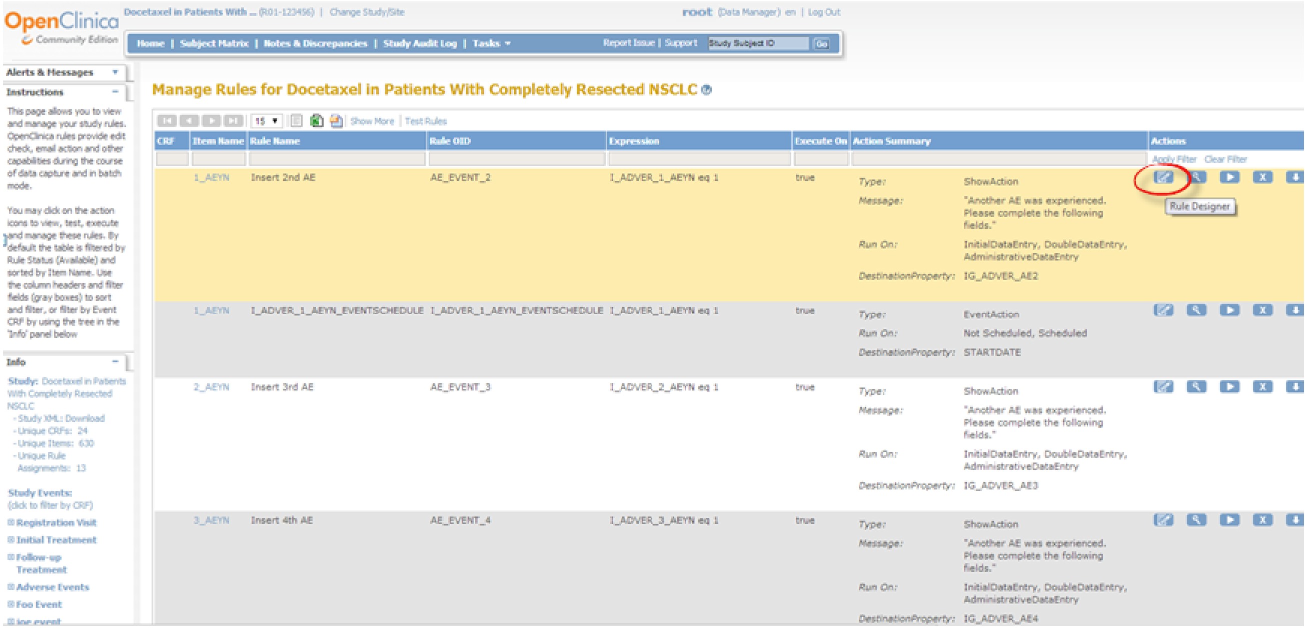
	Task	Status	Count	Mark Complete	Actions
1	Create Study	Completed	NA	☑	🔍 ✎
2	Create CRF	Completed	24	☑	🔍 ✚
3	Create Event Definitions	Completed	6	☑	🔍 ✚
4	Create Subject Group Classes	Completed	1	☑	🔍 ✚
5	Create Rules	Completed	10	☑	🔍 ✚ 📝

	Task	Status	Count	Mark Complete	Actions
6	Create Sites	Completed	4	☑	🔍 ✚

2. To access Rule Designer from the Manage Rules page:

- Log into OpenClinica.
- On the **Tasks** menu, select **Build Study** and in the **Create Rules** row, click 🔍 (the View icon)
  OR
  On the **Tasks** menu, in the **Monitor and Manage Data** section, select **Rules**.
- On the **Manage Rules** page, click 📝. (See below).

Accessing Rule Designer from the Manage Rules page


OpenClinica
Community Edition
Docetaxel in Patients With ... (R01-123456) | Change Study/Site
root (Data Manager) en | Log Out
Home | Subject Matrix | Notes & Discrepancies | Study Audit Log | Tasks
Report Issue | Support
Study Subject ID
Go
Alerts & Messages
Instructions
Info
15
Show More | Test Rules
Apply Filter Clear Filter
Rule Designer

Manage Rules for Docetaxel in Patients With Completely Resected NSCLC

This page allows you to view and manage your study rules. OpenClinica rules provide edit check, email action and other capabilities during the course of data capture and in batch mode.

You may click on the action icons to view, test, execute and manage these rules. By default the table is filtered by Rule Status (Available) and sorted by Item Name. Use the column headers and filter fields (gray boxes) to sort and filter, or filter by Event CRF by using the tree in the 'Info' panel below

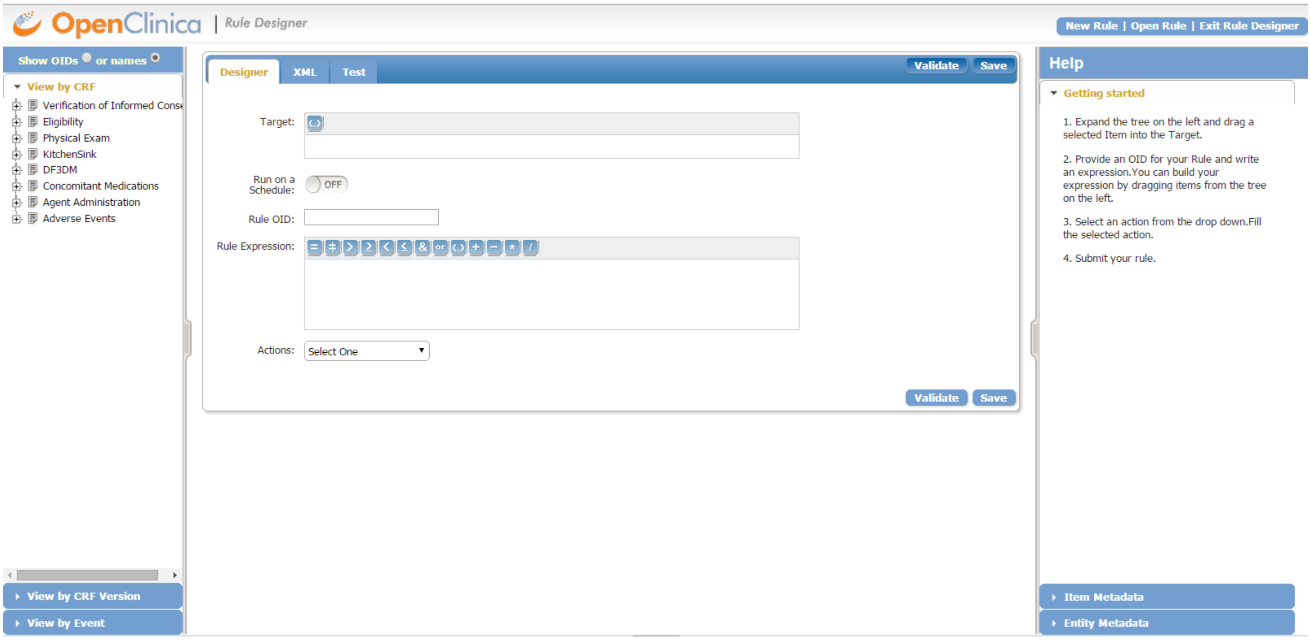
Study: Docetaxel in Patients With Completely Resected NSCLC
- Study XML: Download
- Unique CRFs: 24
- Unique Items: 630
- Unique Rule Assignments: 13

Study Events:
(click to filter by CRF)
⊞ Registration Visit
⊞ Initial Treatment
⊞ Follow-up Treatment
⊞ Adverse Events
⊞ Foo Event
⊞ joe event

CRF	Item Name	Rule Name	Rule OID	Expression	Execute On	Action Summary	Actions
	1_AEYN	Insert 2nd AE	AE_EVENT_2	I_ADVER_1_AEYN eq 1	true	Type: ShowAction Message: "Another AE was experienced. Please complete the following fields." Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry DestinationProperty: IG_ADVER_AE2	
	1_AEYN	I_ADVER_1_AEYN_EVENTSCHEDULE	I_ADVER_1_AEYN_EVENTSCHEDULE	I_ADVER_1_AEYN eq 1	true	Type: EventAction Run On: Not Scheduled, Scheduled DestinationProperty: STARTDATE	
	2_AEYN	Insert 3rd AE	AE_EVENT_3	I_ADVER_2_AEYN eq 1	true	Type: ShowAction Message: "Another AE was experienced. Please complete the following fields." Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry DestinationProperty: IG_ADVER_AE3	
	3_AEYN	Insert 4th AE	AE_EVENT_4	I_ADVER_3_AEYN eq 1	true	Type: ShowAction Message: "Another AE was experienced. Please complete the following fields." Run On: InitialDataEntry, DoubleDataEntry, AdministrativeDataEntry DestinationProperty: IG_ADVER_AE4	

The Rule Designer interface displays.

The Rule Designer interface



Target:

Run on a Schedule: OFF

Rule OID:

Rule Expression:

Actions: Select One

Getting started

1. Expand the tree on the left and drag a selected Item into the Target.

2. Provide an OID for your Rule and write an expression.You can build your expression by dragging items from the tree on the left.

3. Select an action from the drop down.Fill the selected action.

4. Submit your rule.

4.6.2 Building Rules in Rule Designer

The Rule Designer interface is divided into three main panes:

- The left hand pane, which is further divided into three subtabs: View by CRF, View by CRF Version, and View by Event. Every time each of these subtabs is selected, a list of CRFs/ Events will be listed. Each of these CRFs/ Events has an adjacent node which can be clicked to expand into a hierarchy/ minimize to display or hide any associated Item OID. You will then be able to locate, select, and finally drag and drop any of the desired OIDs to the applicable entries on the main (center) tab.

- The center pane, which is divided into three main tabs: Designer, XML and Test tab. The Designer tab contains entries that you can use to drag and drop OIDs to the desired fields (i.e. Target, Rule OID, Rule Expression). Note that if you hover to any of the entry fields on the Designer tab, there will be an associated pop-up help menu that will get displayed. For example, if your cursor is placed on the Target field, there will be a back pop-up message displayed to provide an explanation of a Target field. The XML tab displays the associated XML lines generated as you are updating the fields on te Designer tab. The Test tab can be accessed once all of the information entered on the Designer tab is valid.
- The right hand pane, which is further divided into two subtabs: The Help subtab and the Item Metadata subtab. The help tab contains direction how you can build your rule on the Designer tab.The Item Metadata subtab contains the metadata associated to selected OID/ record. The Item Metadata gets updated everytime a new OID/ record is selected.

On the top right of the page, there are 3 additional links that you may access at any point: New Rule, Open Rule and Exit Rule Designer.

Building a New Rule in Rule Designer

1. In order to create a new rule, you will initially need to access the Designer tab in Rule Designer.

2. Specify the target field. The Target field is the variable which you want the error message to appear on. It is also associated with any discrepancy note/ email action.

3. If you want the Rule to run at a specified time each day, turn on the option to Run On Schedule. These were specifically designed for Participate Notification Action Rules so that Participants can be notified of upcoming forms that should be completed. Though any Rule can be set to run on a schedule, it is not recommended to set a large number of Rules to run on schedule since this could slow server processing. Rules are run automatically whenever data is saved, and can also be run in batch via Manage Rules or Manage CRFs.

4. Specify the Rule_OID field. This is a unique identifier of any new rule that you are creating and this is how your rule will be referred in the database. It must be 32 characters or shorter, be in uppercase, cannot include any spaces/special characters (underscore permitted)

5. Specify the Rule Expression. This is the condition that will trigger certain actions. You may use the Arithmetic, Logical or Relational Operator in order to build the rule expression. For more information regarding using thee operators, refer to section 6.3: Creating Rules. You can also use the drag and drop CRF Item form the left hand pane in order to build your rule.

6. Specify what actions will be triggered as a result of implementing the rule. The available actions are in the "Actions" drop down menu:

- Discrepancy Note. This will trigger a discrepancy note on a certain CRF Item. Refer to Notes and Discrepancies section for more information.
- Email action. This will prompt the system to generate an automated email to individual(s). Both the email action and discrepancy notes form edit checks.
- Notification Action. This sends an SMS and/or email notification to Patient Participants, or an email to any specified email address. Variables can be used in the SMS/email Subject and/or in the body of the notification. Click on the button for the variable you want to include. For more information on the variables that can be included, see Notification Action.
- Show action. This will trigger a certain CRF Item be displayed as a result of meeting a certain criteria.

- Hide action. A CRF Item will be hidden if the rule condition is met. Note that both Show and Hide Action are used to implement dynamic logics.
- Insert action. This will prompt the system to insert data from one CRF item into another based on previously established rules.
- Event action. This allows you to pre-schedule an Event or Events based on the Startdate or Status of a specified Event.

7. Note that each rule may trigger one or more actions. To add more actions, simply access the drop down menu for the second time and select applicable actions. The second action will be lined below the first action. If you want to remove action, simply click on the remove X icon next to each action.

8. Specify the data entry stages where the rule can be run. In OpenClinica, there are 4 possible values: Admin Data Entry, Initial Data Entry, Double Data Entry, Batch. Select all of the checkboxes if you want them to run at all stages.

9. Click Validate. This will verify if the form contains valid rule. If the rule is valid, you will get a successful confirmation message "Form has no errors. This is a Valid Rule"

Building a New Rule in Rule Designer

1. In order to create a new rule, you will initially need to access the Designer tab in Rule Designer.

2. Specify the target field. The Target field is the variable which you want the error message to appear on. It is also associated with any discrepancy note/ email action.

3. If you want the Rule to run at a specified time each day, turn on the option to Run On Schedule. These were specifically designed for Participate Notification Action Rules so that Participants can be notified of upcoming forms that should be completed. Though any Rule can be set to run on a schedule, it is not recommended to set a large number of Rules to run on schedule since this could slow server processing. Rules are run automatically whenever data is saved, and can also be run in batch via Manage Rules or Manage CRFs.

4. Specify the Rule_OID field. This is a unique identifier of any new rule that you are creating and this is how your rule will be referred in the database. It must be 32 characters or shorter, be in uppercase, cannot include any spaces/special characters (underscore permitted)

5. Specify the Rule Expression. This is the condition that will trigger certain actions. You may use the Arithmetic, Logical or Relational Operator in order to build the rule expression. For more information regarding using thee operators, refer to section 6.3: Creating Rules. You can also use the drag and drop CRF Item form the left hand pane in order to build your rule.

6. Specify what actions will be triggered as a result of implementing the rule. The available actions are in the "Actions" drop down menu:

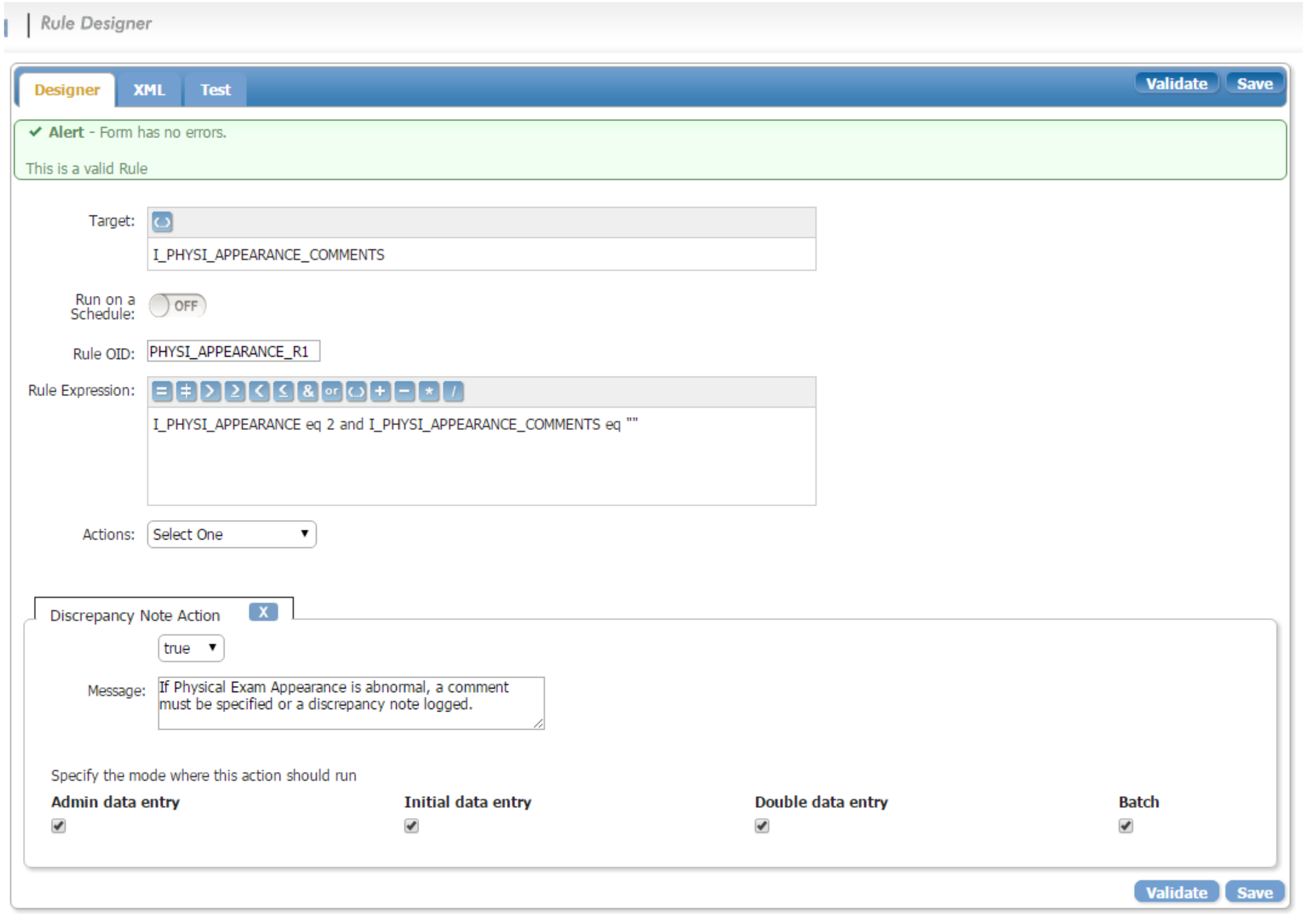
- Discrepancy Note. This will trigger a discrepancy note on a certain CRF Item. Refer to Notes and Discrepancies section for more information.
- Email action. This will prompt the system to generate an automated email to individual(s). Both the email action and discrepancy notes form edit checks.
- Notification Action. This sends an SMS and/or email notification to study Participants who are users of Participate, or an email to any specified email address.

- Show action. This will trigger a certain CRF Item be displayed as a result of meeting a certain criteria.
- Hide action. A CRF Item will be hidden if the rule condition is met. Note that both Show and Hide Action are used to implement dynamic logics.
- Insert action. This will prompt the system to insert data from one CRF item into another based on previously established rules.
- Event action. This allows you to pre-schedule an Event or Events based on the Startdate or Status of a specified Event.

7. Note that each rule may trigger one or more actions. To add more actions, simply access the drop down menu for the second time and select applicable actions. The second action will be lined below the first action. If you want to remove action, simply click on the remove X icon next to each action.

8. Specify the data entry stages where the rule can be run. In OpenClinica, there are 4 possible values: Admin Data Entry, Initial Data Entry, Double Data Entry, Batch. Select all of the checkboxes if you want them to run at all stages.

9. Click Validate. This verifies that the form contains valid rule. If the rule is valid, the following confirmation message displays: "Form has no errors. This is a Valid Rule."



✓ Alert - Form has no errors.

This is a valid Rule

Target: I_PHYSI_APPEARANCE_COMMENTS

Run on a Schedule: OFF

Rule OID: PHYSI_APPEARANCE_R1

Rule Expression: I_PHYSI_APPEARANCE eq 2 and I_PHYSI_APPEARANCE_COMMENTS eq ""

Actions: Select One

Discrepancy Note Action X

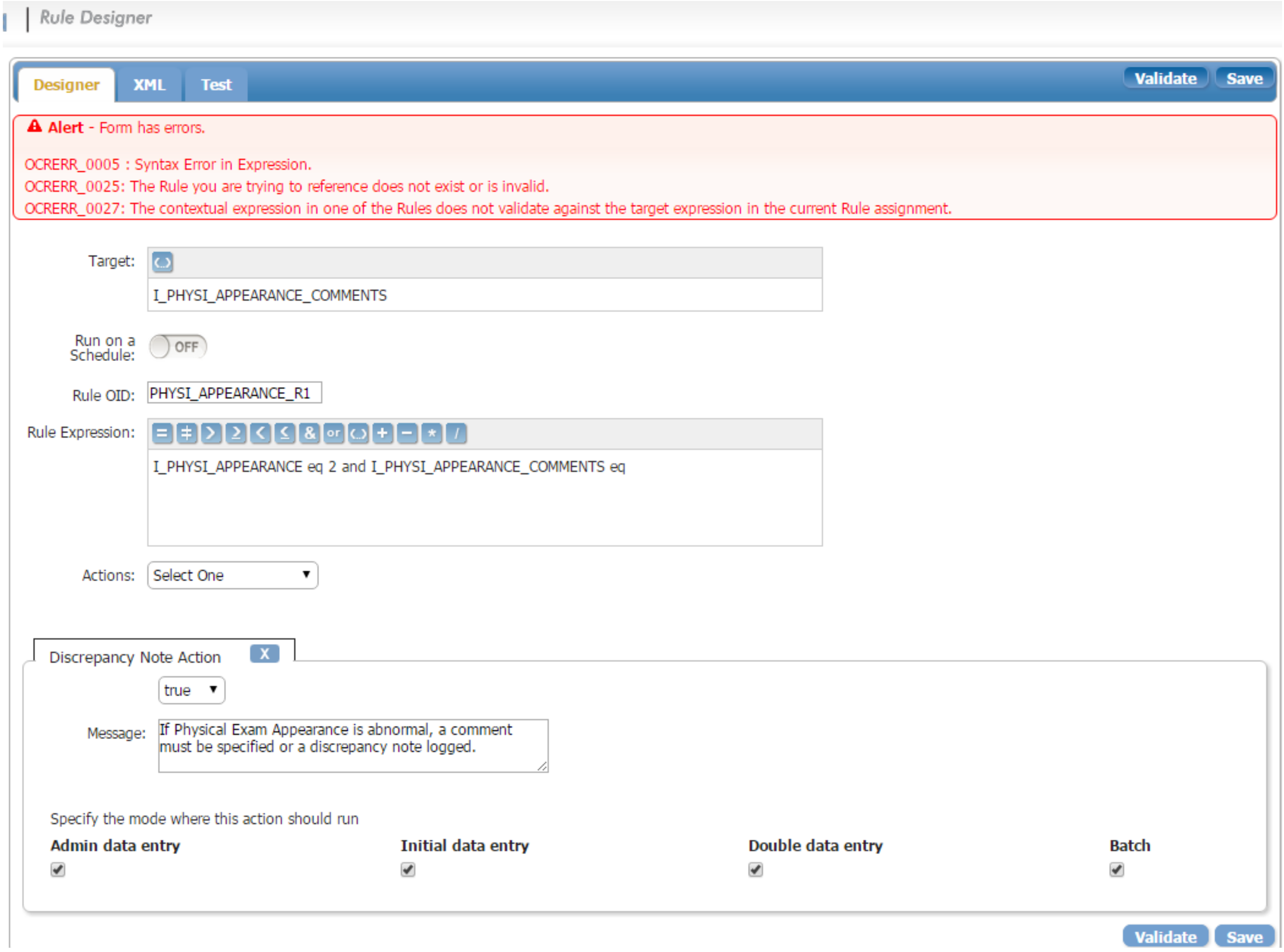
true

Message: If Physical Exam Appearance is abnormal, a comment must be specified or a discrepancy note logged.

Specify the mode where this action should run

Admin data entry	Initial data entry	Double data entry	Batch
☑	☑	☑	☑

If some of the rule components are invalid, an error message displays on the top of the page detailing the sources of the errors. Refer to the sample of screenshot below.



Alert - Form has errors.

OCRERR_0005 : Syntax Error in Expression.
OCRERR_0025: The Rule you are trying to reference does not exist or is invalid.
OCRERR_0027: The contextual expression in one of the Rules does not validate against the target expression in the current Rule assignment.

Target: I_PHYSI_APPEARANCE_COMMENTS

Run on a Schedule: OFF

Rule OID: PHYSI_APPEARANCE_R1

Rule Expression: I_PHYSI_APPEARANCE eq 2 and I_PHYSI_APPEARANCE_COMMENTS eq

Actions: Select One

Discrepancy Note Action

true

Message: If Physical Exam Appearance is abnormal, a comment must be specified or a discrepancy note logged.

Specify the mode where this action should run

Admin data entry	Initial data entry	Double data entry	Batch
☑	☑	☑	☑

10. You also have the option of saving the rule to OpenClinica. To achieve this purpose, you select "Save". You will get a confirmation message "Submitted successfully to your instance".

4.6.3 Testing Rules in Rule Designer

Testing a New Rule in Rule Designer

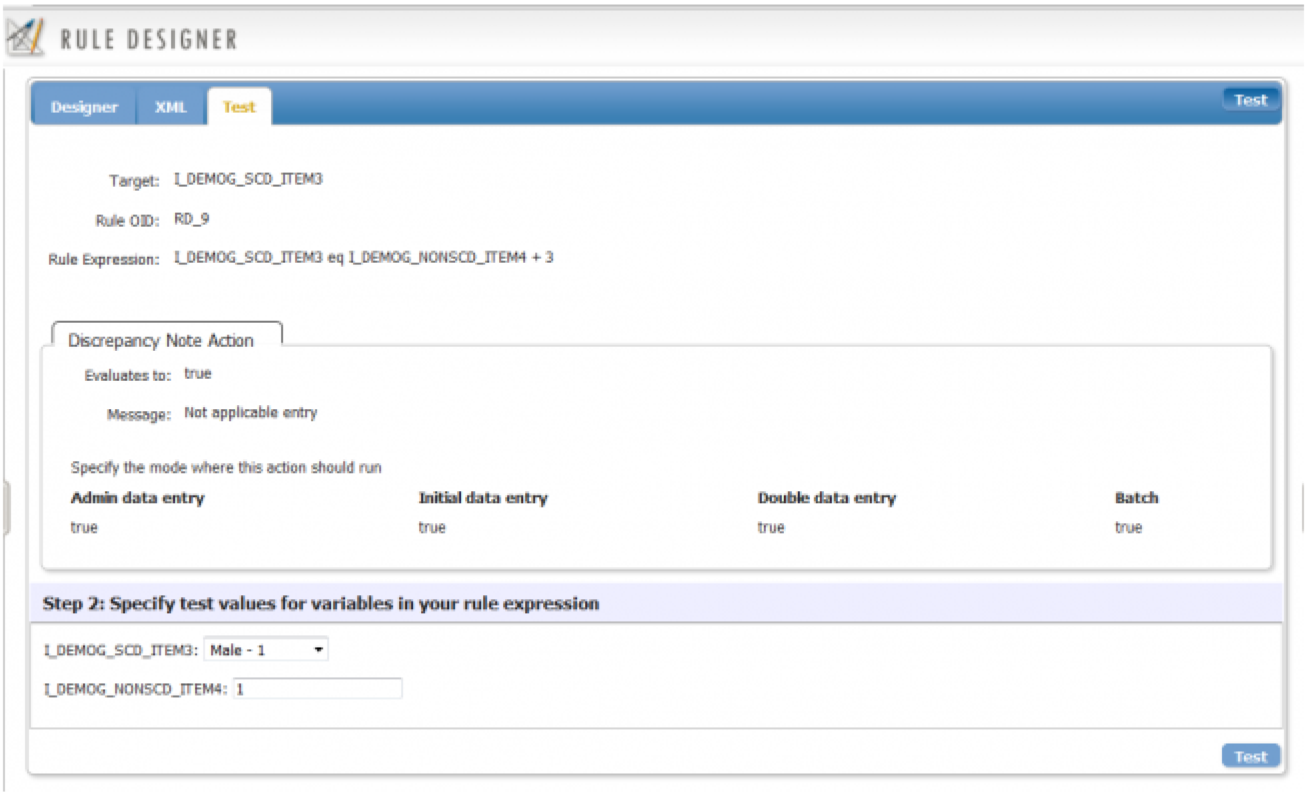
OpenClinica Rule Designer offers an easy interface to test the rules so that you will be able to instantaneously observe the simulation of the rule that you just created on CRF Items. The Test tab serves as a "mock-up" of rules if it were to be deployed on actual CRF Items. Based on the outcome of the test, you can determine if further changes to the rule are necessary.

1. In order to test a rule, you must ensure that your rule is valid. Refer to Building a New Rule in Rule Designer section for more information. Once complete, access the Test tab on the center pane.

2. Initially, you will notice that there will be only 2 tabs displayed on the Test tab. The upper tab contains the information pertaining to the action you selected on the Designer tab. The bottom tab contains data fields for the impacted CRF items. You can modify the input variables on this tab as desired.

3. Modify the input parameters of the impacted CRF Items. Once complete, click "Test"

4. Once run, there will be an additional tab displayed on the bottom of the second tab named "Verify Results". This tab contains the outcome of the test if the rule were to be deployed as is in OpenClinica. You can evaluate the result and determine if you need to commit further edits on the rule. You can also modify the input variables from step 2 and rerun the test again as many times as possible.



Target: I_DEMOG_SCD_ITEM3

Rule OID: RD_9

Rule Expression: I_DEMOG_SCD_ITEM3 eq I_DEMOG_NONSCD_ITEM4 + 3

Discrepancy Note Action

Evaluates to: true

Message: Not applicable entry

Specify the mode where this action should run

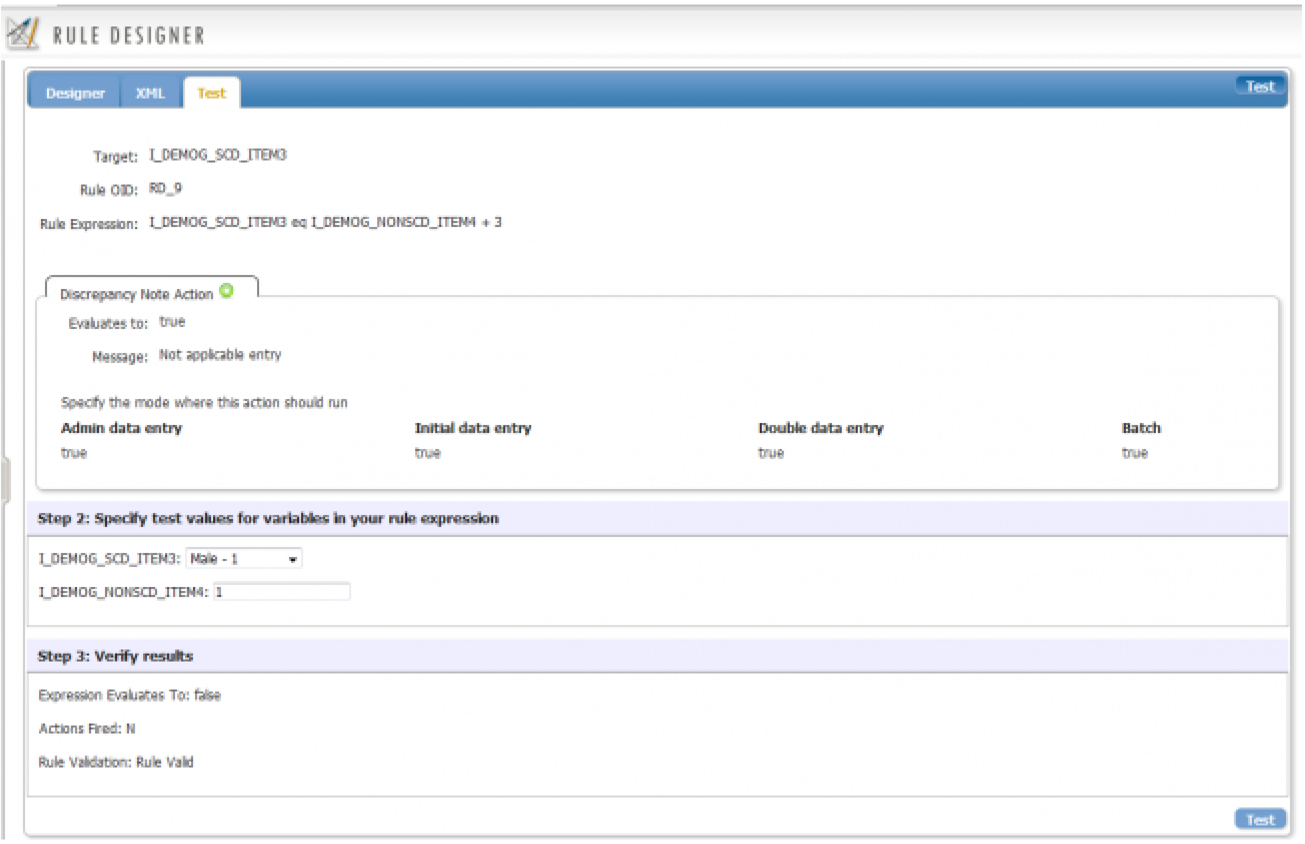
Admin data entry	Initial data entry	Double data entry	Batch
true	true	true	true

Step 2: Specify test values for variables in your rule expression

I_DEMOG_SCD_ITEM3: Male - 1

I_DEMOG_NONSCD_ITEM4: 1

The Test tab before the rule is executed. Note that there are only 2 tabs available.



Target: I_DEMOG_SCD_ITEM3

Rule OID: RD_9

Rule Expression: I_DEMOG_SCD_ITEM3 eq I_DEMOG_NONSCD_ITEM4 + 3

Discrepancy Note Action

Evaluates to: true

Message: Not applicable entry

Specify the mode where this action should run

Admin data entry	Initial data entry	Double data entry	Batch
true	true	true	true

Step 2: Specify test values for variables in your rule expression

I_DEMOG_SCD_ITEM3: Male - 1

I_DEMOG_NONSCD_ITEM4: 1

Step 3: Verify results

Expression Evaluates To: false

Actions Fired: N

Rule Validation: Rule Valid

The Test tab is displayed after the rule is executed (tested) for the first time. Note that "Verify Results" tab is now displayed.

4.7 Rule Expressions

Previously, we gave some basic examples on how RuleExpressions should be written. This section provides more detailed information and examples on how to design expressions and use rule expression operators with each data type. All of the information below is based on 3.1.

Below is a list of the operators available for use in Rule Expressions. Parentheses () can be used to designate which parts of the expression should be evaluated first. 1 + 1 * 32 equals 33. Multiplication occurrs before addition. However, if you add parentheses around the addition you would have (1 + 1) * 32. In this example, the total is 64. Parentheses will work the same way with OpenClinica Rule Expressions.

Rule expression operators, in order of precedence:

( )
* /
+ -
gt gte lt lte eq ne ct
and or

At each level of precedence the expression is evaluated from left to right.

In the examples below, we will be building the expressions with the default structure

- SED_OID = The OID depicting a particular study event definition. If the rule is referencing

more than one event definition, _X, where X is a number, will be appended to the OID
- CRF_OID = The OID depicting a particular CRF. This will cover all versions of this CRF. If the rule is referencing more than one CRF, _X, where X is a number, will be appended to the OID
- CRF_VERSION_OID = The OID depicting a particular CRF Version. If the rule is referencing more than one CRF Version, _X, where X is a number, will be appended to the OID
- GROUP_OID = The OID depicting a particular Group. If the rule is referencing more than one Group, _X, where X is a number, will be appended to the OID
- ITEM_OID = The OID depicting a particular Item. If the rule is referencing more than one Item, _X, where X is a number, will be appended to the OID

4.7.1 Equal or Not Equal To

The following examples will demonstrate how to use both the equal (eq) and the not equal () operators with the following data types:

- INT
- REAL
- DATE
- ST
- FILE

Example 1

The first example will demonstrate how to specify a temperature is not equal to a particular negative value. The data type is INT so no decimal places are allowed.

```
<RuleDef OID="RULE1" Name="This is an example">
   <Description>The temperature is not equal to -10 degrees F</Description>
   <Expression>ITEM_OID ne (-10)</Expression>
</RuleDef>
```

Please be aware of the use of parentheses around the negative value. This is how you must write your RuleExpression if you want to evaluate negative values enterd as data in the CRF. If the DataType was REAL, decimal places would be allowed.

Also, if the field was a ST, quotes ("") must surround the string value. If these are not present, the RuleExpression will not evaluate correctly.

Example 2

The second example is similar, but it is stating that a value for one item is equal to the value for one item, or not equal to a value from a third item. The second item is part of the same CRF Version as the first item, but the third item is in a different event CRF. If you compare two fields together, they must be of the same DataType. If the third item is comparing an explicit value, it does not have to be the same DataType

```
  <RuleDef OID="RULE2" Name="This is an example">
    <Description>Temperature is different and the color of the sun is yellow</Description>
    <Expression>SED_OID.CRF_OID.GROUP_OID._ITEM_OID_3 eq "yellow" or
CRF_VERSION_OID.GROUP_OID.ITEM_OID_2 ne ITEM_OID_1</Expression>
  </RuleDef>
```

As you can see, the path will continue to grow to the left if you have to include the CRF or CRF Version, or the Study Event Definition. Each piece of the path is separated with a period (.) and everything must be capitalized.

Example 3

The following example uses DATEs with the equal and not equal operators. The same principle applies where the system can only compare items of the same DataType.

```
  <RuleDef OID="RULE3" Name="This is an example">
    <Description>Visit Date is the same as today's date</Description>
    <Expression>ITEM_OID_1 eq _CURRENT_DATE</Expression>
  </RuleDef>
```

_CURRENT_DATE is a system level property that will pull the "today's date" from the server OpenClinica is installed on.

```
  <RuleDef OID="RULE4" Name="This is an example">
    <Description>The date is not December 31, 2012r</Description>
    <Expression>ITEM_OID_1 ne 2012-12-31</Expression>
  </RuleDef>
```

Notice the date must be in the ISO 8601 format otherwise the system will reject the RuleExpression.

Example 4

When an item is defined as a FILE DataType, you can write rules to make sure the field is either blank, or not blank. You will not be able to write any rules comparing files or saying files are equal to each other. An example of a correct RuleExpression would be:

```
  <RuleDef OID="RULE5" Name="This is an example">
    <Description>no file is provided</Description>
    <Expression>ITEM_OID_1 eq ""</Expression>
  </RuleDef>
```

"" next to each other in a RuleExpression signifies a blank field for all DataTypes.

As we go through the rest of the operators, we will also demonstrate how to combine multiple operators into a RuleExpression which should provide even greater clarity on how to use Rules.

4.7.2 Greater Than or Equal To

The following examples will demonstrate how to use both the greater than (gt) and the greather than or equal to (gte) operators with the following data types:

- INT
- REAL
- DATE

The 'gt' and 'gte' operators can not be used with ST or FILE.

Example 1

The first example will demonstrate how to specify a temperature is greater than a particular positive value. The data type is INT so no decimal places are allowed.

```
<RuleDef OID="RULE1" Name="This is an example">
   <Description>The temperature for this subject was greater than 105 degrees</Description>
   <Expression>ITEM_OID gt 105</Expression>
</RuleDef>
```

If the DataType was REAL, decimal places would be allowed.

Example 2

The second example is similar, but it is stating that a value for one item is greater than or equal to the value for another item. The second item is part of the same Group as the first item. Both items must be of the same DataType in order for this comparison to work. If one item is an INT and the other is DATE, the RuleExpression will be invalid.

```
<RuleDef OID="RULE2" Name="This is an example">
   <Description>Temperature is different</Description>
   <Expression>ITEM_OID_1 gte ITEM_OID_2</Expression>
</RuleDef>
```

If the second item was not part of the same group, but contained in the same CRF, the Expression would have to be written as follows:

```
<RuleDef OID="RULE3" Name="This is an example">
   <Description>Temperature is different</Description>
   <Expression>ITEM_OID_1 gte GROUP_OID.ITEM_OID_2</Expression>
</RuleDef>
```

The path will continue to grow to the left if you have to include the CRF or CRF Version, or the Study Event Definition. Each piece of the path is separated with a period (.) and everything must be capitalized.

Example 3

The following example uses DATEs with the greater than or the greater than and equal to operators. The same principle applies where the system can only compare items of the same DataType.

```
<RuleDef OID="RULE4" Name="This is an example">
   <Description>Visit Date is greater than today's date</Description>
   <Expression>ITEM_OID_1 gt _CURRENT_DATE</Expression>
</RuleDef>
```

_CURRENT_DATE is a system level property that will pull the "today's date" from the server

OpenClinica is installed on.

```
<RuleDef OID="RULE5" Name="This is an example">
   <Description>The date is 2013 or later</Description>
   <Expression>ITEM_OID_1 gt 2012-12-31</Expression>
</RuleDef>
```

Notice the date must be in the ISO 8601 format otherwise the system will reject the RuleExpression.

As we go through the rest of the operators, we will also demonstrate how to combine multiple operators into a RuleExpression which should provide even greater clarity on how to use Rules.

4.7.3 Less Than or Equal To

The following examples will demonstrate how to use both the less than (lt) and the less than or equal to (lte) operators with the following data types:

- INT
- REAL
- DATE

The 'lt' and 'lte' operators can not be used with ST or FILE.

Example 1

The first example will demonstrate how to specify a temperature is less than a particular positive value. The data type is INT so no decimal places are allowed.

```
<RuleDef OID="RULE1" Name="This is an example">
   <Description>The temperature for this subject was less than 105 degrees</Description>
   <Expression>ITEM_OID lt 105</Expression>
</RuleDef>
```

If the DataType was REAL, decimal places would be allowed.

Example 2

The second example is similar, but it is stating that a value for one item is less than or equal to the value for another item. The second item is part of the same Group as the first item. Both items must be of the same DataType in order for this comparison to work. If one item is an INT and the other is DATE, the RuleExpression will be invalid.

```
<RuleDef OID="RULE2" Name="This is an example">
   <Description>Temperature is different</Description>
   <Expression>ITEM_OID_1 lte ITEM_OID_2</Expression>
</RuleDef>
```

If the second item was not part of the same group, but contained in the same CRF, the Expression would have to be written as follows:

```
<RuleDef OID="RULE3" Name="This is an example">
   <Description>Temperature is different</Description>
```

```
    <Expression>ITEM_OID_1 lte GROUP_OID.ITEM_OID_2</Expression>
</RuleDef>
```

The path will continue to grow to the left if you have to include the CRF or CRF Version, or the Study Event Definition. Each piece of the path is separated with a period (.) and everything must be capitalized.

Example 3

The following example uses DATEs with the less than or the less than and equal to operators. The same principle applies where the system can only compare items of the same DataType.

```
<RuleDef OID="RULE4" Name="This is an example">
    <Description>Visit Date is less than today's date</Description>
    <Expression>ITEM_OID_1 lt _CURRENT_DATE</Expression>
</RuleDef>
```

_CURRENT_DATE is a system level property that will pull the "today's date" from the server OpenClinica is installed on.

```
<RuleDef OID="RULE5" Name="This is an example">
    <Description>The date is 2013 or earlier</Description>
    <Expression>ITEM_OID_1 lt 2012-12-31</Expression>
</RuleDef>
```

Notice the date must be in the ISO 8601 format otherwise the system will reject the RuleExpression.

As we go through the rest of the operators, we will also demonstrate how to combine multiple operators into a RuleExpression which should provide even greater clarity on how to use Rules.

4.7.4 Arithmetic Operators

OpenClinica Rules support the following arithmetic operators:

- + (Addition)
- - (Subtraction)
- * (Multiplication)
- / (Division)

All four of the operators above will work with INT and REAL DataTypes. For DATEs, an INT can be added (+) or subtracted (-) from the DATE to increase or decrease it by a certain number of days. Subtraction (-) can also be used to get the difference between two DATEs, with the result being an INT representing the number of days between the two DATEs. The difference between dates is always computed as an absolute number. For example, 2011-11-19 - 2011-11-20 evaluates to 1, not -1. ST fields can not be used with any of the arithmetic operators.

Example 1

```
<RuleDef OID="RULE1" Name="This is an example">
    <Description>The average temperature is greater than 98.6</Description>
    <Expression>((ITEM_OID_1 + ITEM_OID_2) / 2) gt 98.6</Expression>
</RuleDef>
```

The above example is taking 2 fields that are REAL DataTypes, ITEM_OID_1 and ITEM_OID_2, adding them together to get a total, and then dividing by 2. If the average is greater than 98.6, this Rule will fire.

Example 2

Addition and subtraction with dates can be very powerful, but there are a few things to keep in mind. When working with dates and these arithmetic operators, the values derived will be in days. For example, you can not have a rule that checks for a visit that is greater than 1 year. You would have to write this rule saying the difference between the dates is greater than 365 days.

```
<RuleDef OID="RULE2" Name="This is an example">
   <Description>The surgery date is greater than 1 year from this visit date</Description>
   <Expression>ITEM_OID_1 - ITEM_OID_2 gte 365</Expression>
</RuleDef>
```

In the above RuleExpression, we are subtracting one date from another, and if the value is 365 or greater, the Rule should evaluate to true.

Example 3

OpenClinica computes addition/subtraction expressions from left to right. When combining dates and integers in an expression this has some interesting consequences. A date that has an integer added to or subtracted from it is still treated as a date. When two dates are compared, it uses the absolute value of the difference (ie, the result is always a positive number or 0). Thus the two expressions below are not always equivalent:

A) 0 - 1 + DATE2 - DATE1 ne 0
B) 0 + DATE2 - DATE1 - 1 ne 0

Assume DATE2 = 31-Dec-2000 and DATE1 = 01-Jan-2001

In case A, -1 is added to DATE2 and the result is a date that is one day before DATE2. OpenClinica then takes that date and subtracts DATE1 from it, and since for date math we use absolute values, computes the absolute value of the difference between the two dates.

A)   0 - 1 + DATE2   -   DATE1 ne 0
(-1) + 31-Dec-2000   -   DATE1 ne 0
30-Dec-2000   -   01-Jan-2001 ne 0
2 ne 0

In case B, OpenClinica first computes the absolute value of the difference between DATE2 and DATE1, then subtracts 1.

B)   0 + DATE2 - DATE1 - 1 ne 0
31-Dec-2000   -   01-Jan-2001   - 1 ne 0
1 - 1 ne 0
0 ne 0

4.7.5 6.7.5 The Contains Operator

The following examples will demonsrate how to use the contains (ct) operator with the following data types:

- ST
- INT
- REAL
- DATE
- FILE

Example 1:

The first example will demonstrate how, when using checkbox or multi-select items with a data type of ST, to check if a certain option is selected. For example, if you have a checkbox item with three options (Option 1, Option 2, and Unknown), you may want to make sure that 'Option 1' and 'Option 2' are not selected if 'Unknown' is selected:

```
   <RuleDef OID="RULE1" Name="This is an example using ST data type">
      <Description>'Unknown' and at least one other option selected</Description>
      <Expression>(ITEM_OID_1 ct "1" or ITEM_OID_1 ct "2") and ITEM_OID_1 ct
"99"</Expression>
   </RuleDef>
```

Similarly, the contains operator can be used with data types INT and REAL. Please note that all values must be enclosed in quotation marks in the expression when using the ct operator (i.e. "1" and "2").

Example 2:

This example will demonstration how to use the contains operator to check for a certain year when using an item of data type DATE. For example you may want to write a Rule that will verify that the dat of informed consent occurred during the year 2014:

```
   <RuleDef OID="RULE2" Name="This is an example using DATE data type">
      <Description>The year is 2014</Description>
      <Expression>ITEM_OID_1 ct "2014"</Expression>
   </RuleDef>
```

When set to evaluate to true, the Rule will fire if the date occurs in the year 2014. Alternatively, you could set the Rule to evaluate to false so that the Rule fires if the date does not occur in the year 2014.

Example 3:

When an item is defined as a FILE data type, a Rule can be written to make sure the filename is a certain extension type. For example, the contains operator can be used to check for a PDF file:

```
<RuleDef OID="RULE3" Name="This is an example using FILE data type">
      <Description>The extension must be .pdf</Description>
      <Expression>ITEM_OID_1 ct ".pdf"</Expression>
   </RuleDef>
```

If written to evaluate to false, the Rule will fire and an error message will appear if a file type other than .PDF is uploaded.

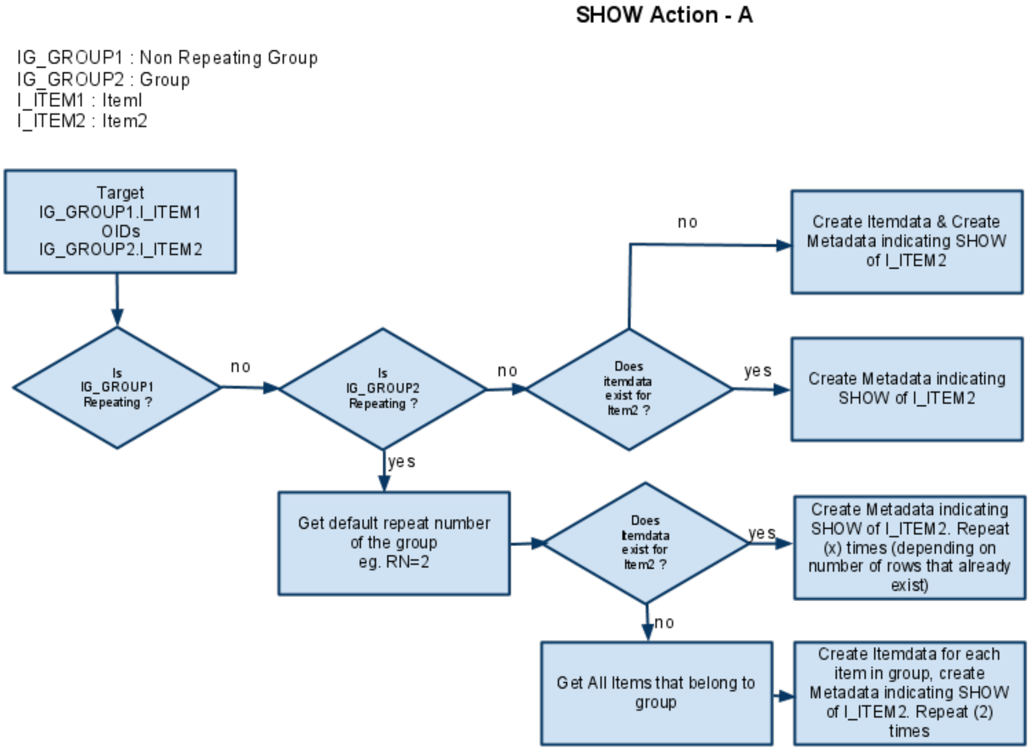
4.8 Rule Action Models

A set of Rule Action Models for ShowAction and InsertAction are displayed below. The behavior of these particular actions is complex and depends on a large number of factors. The diagrams below provide a map of the logic followed by the application when executing these actions, and illustrates the different ways groups and items are treated depending on their characteristics (such as whether a referenced item is part of a repeating or non-repeating group).

Approved for publication by Ben Baumann. Signed on 2014-03-24 8:43AM

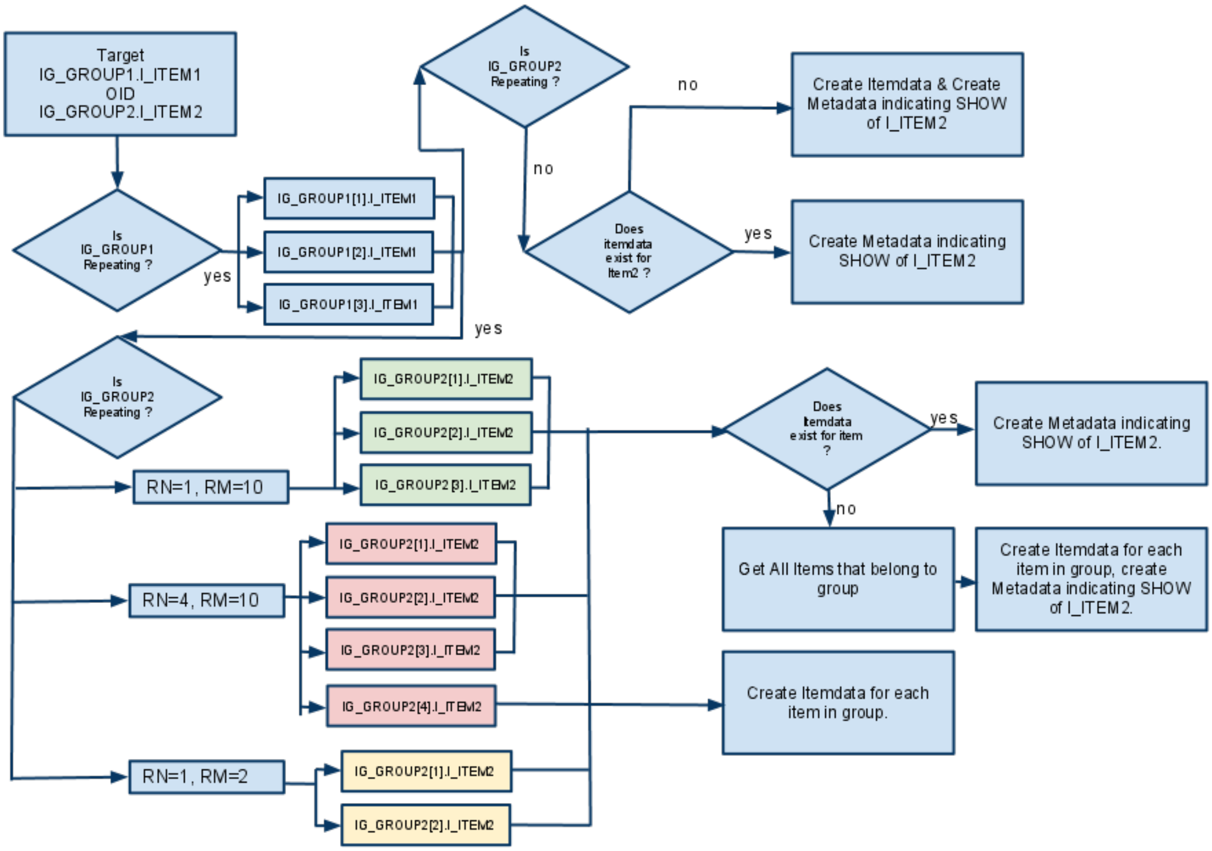
Not valid unless obtained from the OpenClinica document management system on the day of use.

4.8.1 ShowAction Models


SHOW Action - A
IG_GROUP1 : Non Repeating Group
IG_GROUP2 : Group
I_ITEM1 : ItemI
I_ITEM2 : Item2
Target IG_GROUP1.I_ITEM1 OIDs IG_GROUP2.I_ITEM2
Is IG_GROUP1 Repeating ?
no
Is IG_GROUP2 Repeating ?
no
Does itemdata exist for Item2 ?
no
Create Itemdata & Create Metadata indicating SHOW of I_ITEM2
yes
Create Metadata indicating SHOW of I_ITEM2
yes
Get default repeat number of the group eg. RN=2
Does Itemdata exist for Item2 ?
yes
Create Metadata indicating SHOW of I_ITEM2. Repeat (x) times (depending on number of rows that already exist)
no
Get All Items that belong to group
Create Itemdata for each item in group, create Metadata indicating SHOW of I_ITEM2. Repeat (2) times

SHOW Action - B

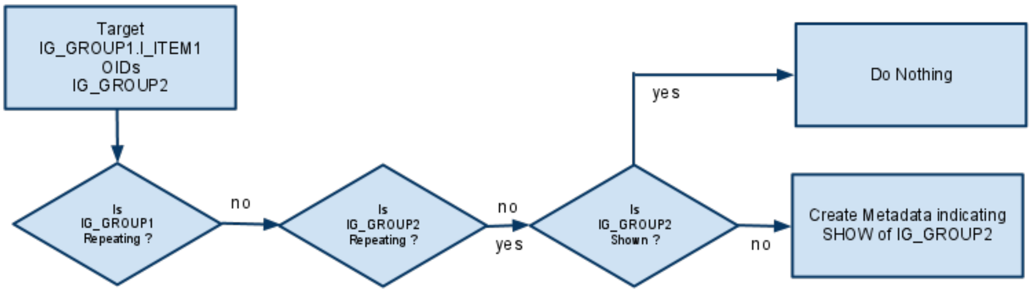
IG_GROUP1[ALL]: Repeating Group
IG_GROUP2 : Group
I_ITEM1 : ItemI
I_ITEM2 : Item2

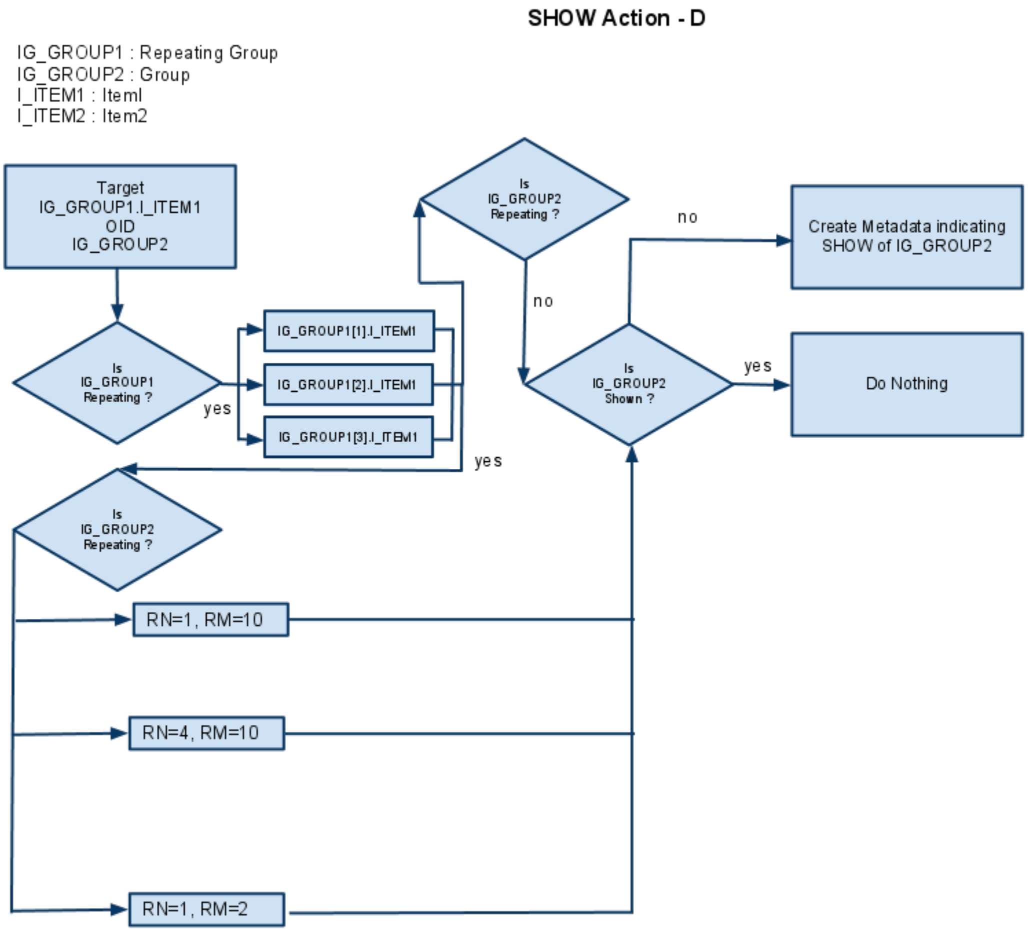


Target IG_GROUP1.I_ITEM1 OID IG_GROUP2.I_ITEM2
Is IG_GROUP1 Repeating ?
yes
IG_GROUP1[1].I_ITEM1
IG_GROUP1[2].I_ITEM1
IG_GROUP1[3].I_ITEM1
yes
Is IG_GROUP2 Repeating ?
no
no
Does itemdata exist for Item2 ?
Create Itemdata & Create Metadata indicating SHOW of I_ITEM2
yes
Create Metadata indicating SHOW of I_ITEM2
Is IG_GROUP2 Repeating ?
RN=1, RM=10
IG_GROUP2[1].I_ITEM2
IG_GROUP2[2].I_ITEM2
IG_GROUP2[3].I_ITEM2
Does Itemdata exist for item ?
yes
Create Metadata indicating SHOW of I_ITEM2.
no
Get All Items that belong to group
Create Itemdata for each item in group, create Metadata indicating SHOW of I_ITEM2.
RN=4, RM=10
IG_GROUP2[1].I_ITEM2
IG_GROUP2[2].I_ITEM2
IG_GROUP2[3].I_ITEM2
IG_GROUP2[4].I_ITEM2
Create Itemdata for each item in group.
RN=1, RM=2
IG_GROUP2[1].I_ITEM2
IG_GROUP2[2].I_ITEM2

SHOW Action - C

IG_GROUP1 : Non Repeating Group
IG_GROUP2 : Group
I_ITEM1 : Iteml
I_ITEM2 : Item2

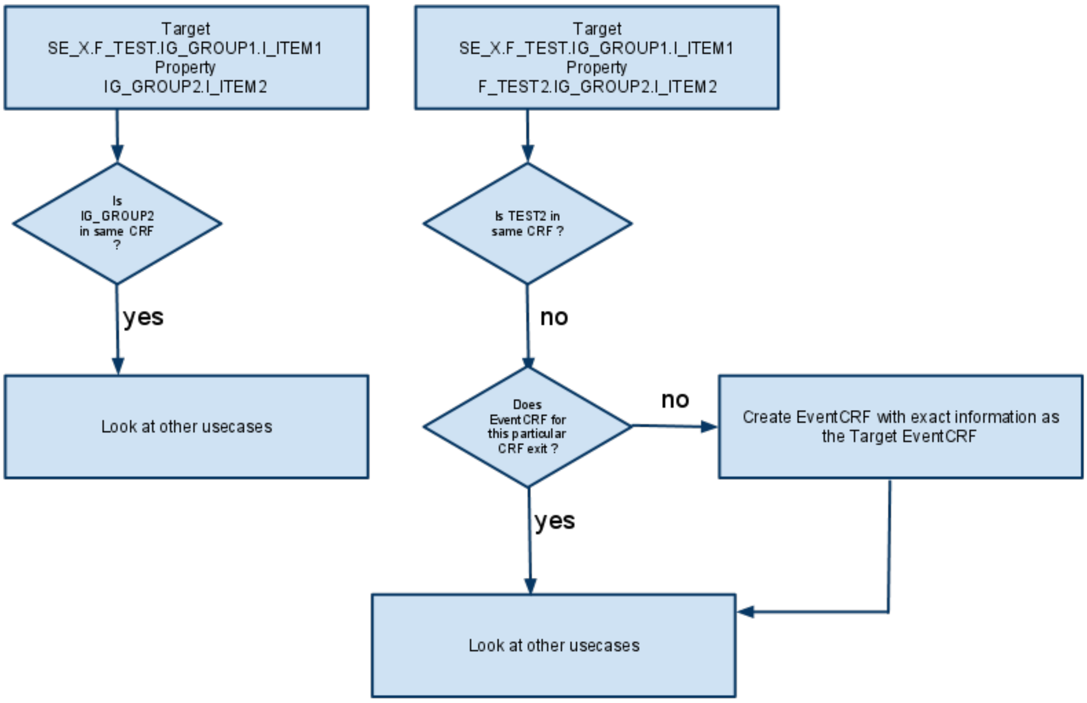




4.8.2 InsertAction Models

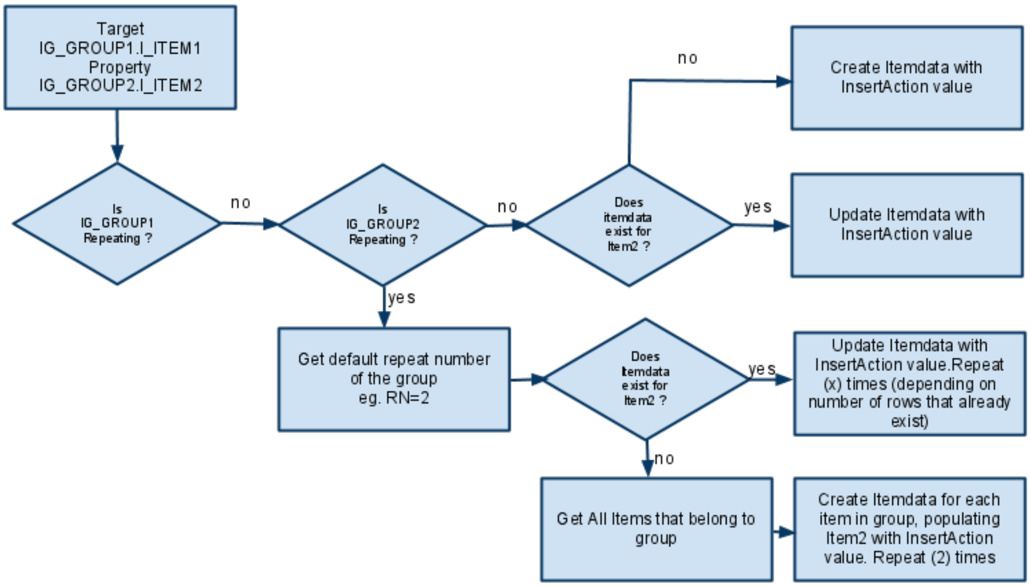
InsertAction - A1

IG_GROUP1	Non Repeating Group
IG_GROUP2	Group
I_ITEM1	Iteml
I_ITEM2	Item2
RN	Repeat number
RM	Repeat max


Target
SE_X.F_TEST.IG_GROUP1.I_ITEM1
Property
IG_GROUP2.I_ITEM2
Is IG_GROUP2 in same CRF ?
yes
Look at other usecases
Target
SE_X.F_TEST.IG_GROUP1.I_ITEM1
Property
F_TEST2.IG_GROUP2.I_ITEM2
Is TEST2 in same CRF ?
no
Does EventCRF for this particular CRF exit ?
no
Create EventCRF with exact information as the Target EventCRF
yes
Look at other usecases

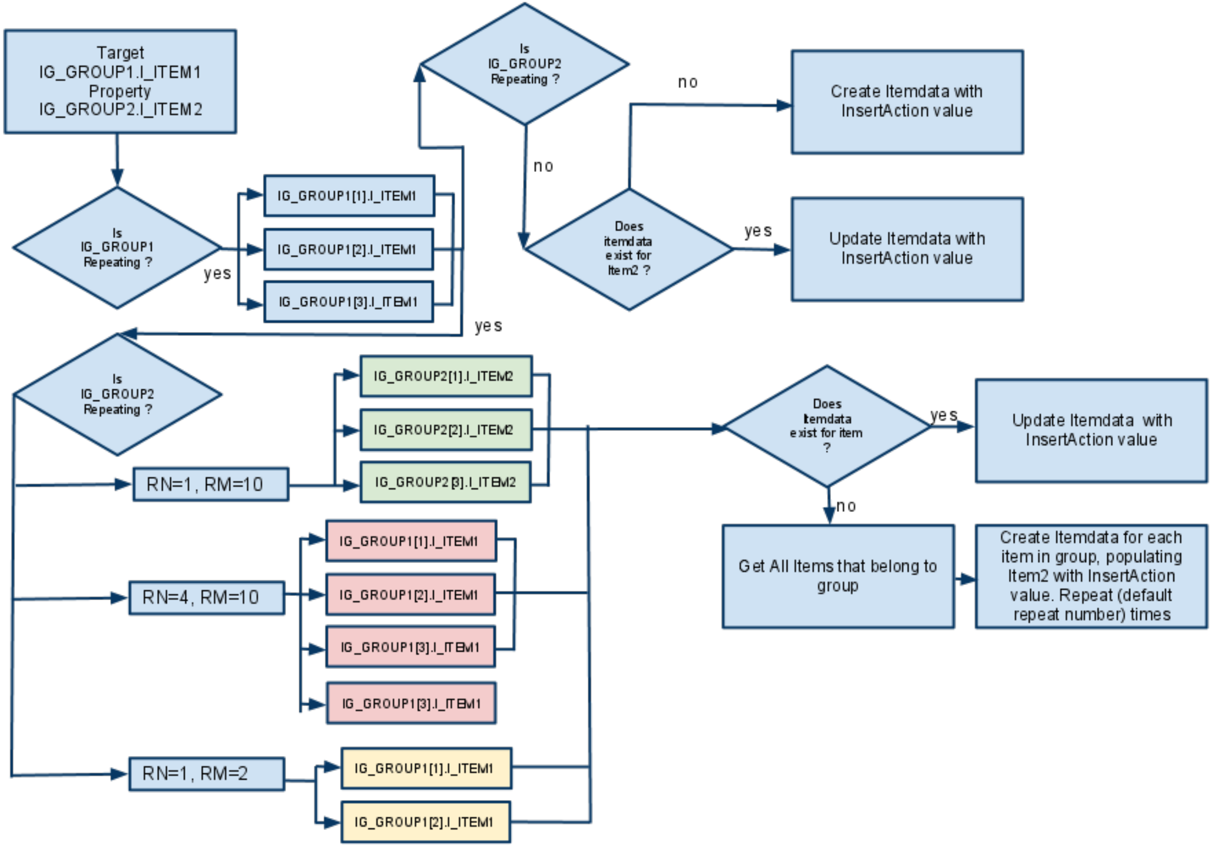
InsertAction - A2

IG_GROUP1 : Non Repeating Group
IG_GROUP2 : Group
I_ITEM1 : ItemI
I_ITEM2 : Item2
RN : Repeat number
RM : Repeat max


Target IG_GROUP1.I_ITEM1 Property IG_GROUP2.I_ITEM2
Is IG_GROUP1 Repeating ?
no
Is IG_GROUP2 Repeating ?
no
Does itemdata exist for Item2 ?
no
Create Itemdata with InsertAction value
yes
Update Itemdata with InsertAction value
yes
Get default repeat number of the group eg. RN=2
Does Itemdata exist for Item2 ?
yes
Update Itemdata with InsertAction value.Repeat (x) times (depending on number of rows that already exist)
no
Get All Items that belong to group
Create Itemdata for each item in group, populating Item2 with InsertAction value. Repeat (2) times

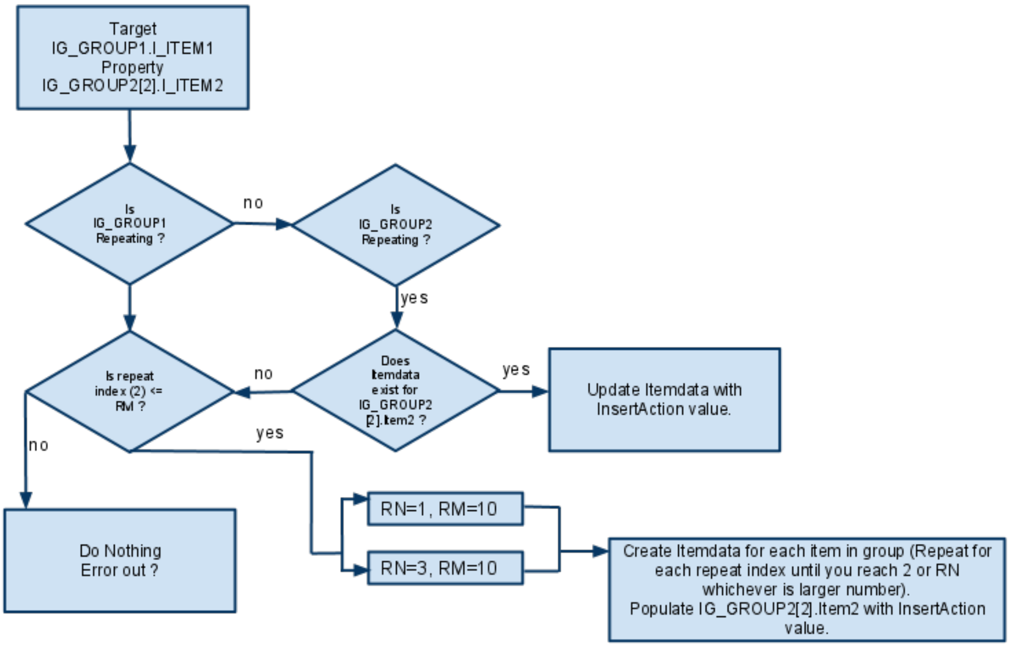
InsertAction - B

IG_GROUP1 : Repeating Group
IG_GROUP2 : Group
I_ITEM1 : ItemI
I_ITEM2 : Item2
RN : Repeat number
RM : Repeat max



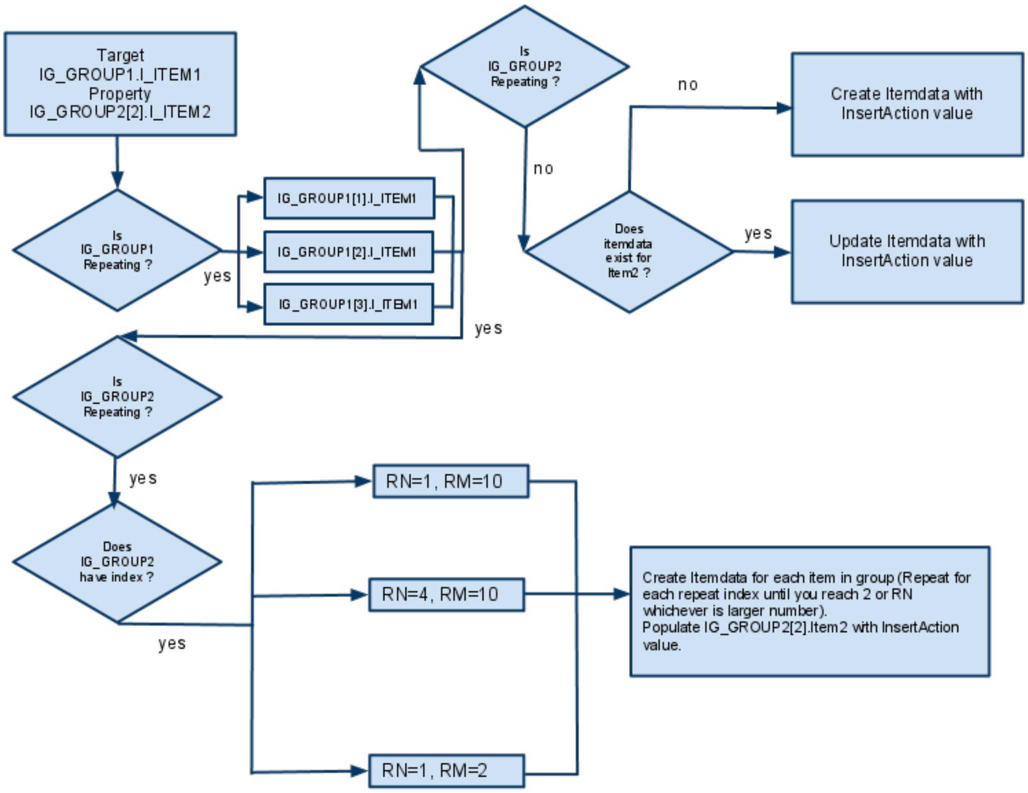
InsertAction - C

IG_GROUP1 : Non Repeating Group
IG_GROUP2 : Repeating Group
I_ITEM1 : Iteml
I_ITEM2 : Item2
RN : Repeat Number
RM : Repeat Max


Target IG_GROUP1.I_ITEM1 Property IG_GROUP2[2].I_ITEM2
Is IG_GROUP1 Repeating ?
no
Is IG_GROUP2 Repeating ?
yes
Does Itemdata exist for IG_GROUP2 [2].Item2 ?
yes
Update Itemdata with InsertAction value.
no
Is repeat index (2) <= RM ?
no
yes
Do Nothing Error out ?
RN=1, RM=10
RN=3, RM=10
Create Itemdata for each item in group (Repeat for each repeat index until you reach 2 or RN whichever is larger number). Populate IG_GROUP2[2].Item2 with InsertAction value.

InserAction - D

IG_GROUP1 : Repeating Group
IG_GROUP2 : Group
I_ITEM1 : Iteml
I_ITEM2 : Item2
RN : Repeat number
RM : Repeat max



Target IG_GROUP1.I_ITEM1 Property IG_GROUP2[2].I_ITEM2
Is IG_GROUP1 Repeating ?
yes
IG_GROUP1[1].I_ITEM1
IG_GROUP1[2].I_ITEM1
IG_GROUP1[3].I_ITEM1
yes
Is IG_GROUP2 Repeating ?
no
Does itemdata exist for Item2 ?
no
Create Itemdata with InsertAction value
yes
Update Itemdata with InsertAction value
Is IG_GROUP2 Repeating ?
yes
Does IG_GROUP2 have index ?
yes
RN=1, RM=10
RN=4, RM=10
RN=1, RM=2
Create Itemdata for each item in group (Repeat for each repeat index until you reach 2 or RN whichever is larger number).
Populate IG_GROUP2[2].Item2 with InsertAction value.