

## Chapter 7: Edit Errors

Computerized edits are automated processes that test the validity of data fields. The following sets of edits are integrated into SEER\*DMS: SEER Edits, SEER Extended Edits, edits defined by and for the local registry, and system edits that enforce database integrity.

- The SEER Edits cover fields submitted to SEER and represent the edits implemented in the SEER\*Edits software.
- The SEER Extended edits were developed by the NCI to extend edits beyond those fields sent to the SEER Program. The extended edits are available to all SEER\*DMS registries and validate fields that are not required to be transmitted to SEER.
- Registry edits are defined and maintained by registry staff. These would include edits to validate registry-specific fields, edits to implement validation not covered by SEER edits, and registry-specific versions of SEER edits (the registry may decide to vary the logic of a SEER edit by creating a registry-defined edit and disabling the SEER edit).
- System edits are edits to enforce database constraints. The SEER\*DMS system edits enforce database integrity of fields common to all SEER\*DMS registries. The SEER\*DMS Registry system edits enforce database integrity of registry-specific fields.

SEER, SEER Extended, and all system edits are maintained and deployed by the SEER\*DMS development team. Registry staff are responsible for defining and maintaining registry-specific edits that are not system edits. The SEER\*DMS Edits Manager provides an interface for adding, modifying, and deleting registry edits.

SEER\*DMS data screens indicate the edits that failed and the fields associated with those edits. It is recommended that, whenever possible, changes be made to the patient set and not to the record. Therefore, only errors that must be resolved prior to screening should be resolved in record data fields (see *Chapter 8: Resolving Record Errors*). All other errors should be resolved when the data are incorporated into a patient set. Patient set edit errors may be resolved in ad hoc editing or when performing a worklist task. Patient Set tasks are described in detail in *Chapter 12: Consolidating Data*, *Chapter 13: Visual Editing*, and *Chapter 14: Resolving Patient Set Errors*.

A patient set exits the workflow when all edits are cleared. However, a Resolve Patient Set Errors task is not created if an edit error is introduced because of data modifications made via ad hoc editing or Mass Change imports.

Edits are executed each time a user opens, validates, or saves a record or patient set in SEER\*DMS. Edits are also executed whenever an automated process updates a patient set (this includes updates via Mass Change imports). The *Patient Set Edits System Task* can be used to execute the edits on all patient sets or a cohort of patient sets. The primary purpose of this task is to ensure that new or modified edits have been applied prior to generating an extract or report that is dependent on submissibility or edit frequencies.

In this chapter, you'll learn about

- Viewing Edit Documentation
  - Viewing Edit Documentation while Editing Data
  - Using the Edits Help Page
- Using the Edits Manager
- Understanding Edits in SEER\*DMS
- Defining and Maintaining Registry Edits
- System Task to Execute Edits in Patient Sets
- Reports Related to Edits
- Color Codes Used in SEER\*DMS Editors

## Viewing Edit Documentation

SEER\*DMS provides four ways to view documentation related to edits:

- As you are editing or viewing a record or patient set, you may use the Edits page of the SEER\*DMS editor to view edit documentation for each failed edit (see the *Viewing Edit Documentation while Editing Data* section of this chapter for instructions).
- All system users have access to the Help menu which includes an Edits section. You may use the Edits Help page to search for an edit by edit ID, group, severity level, and/or search text (see the *Using the Edits Help Page* section of this chapter for instructions).
- If you have the *system\_administration* permission you may use the Edits Manager to view documentation and source code for all edits. You may also use the Edits Manager to modify registry-defined edits. (See the *Using the Edits Manager* and *Defining and Maintaining Registry Edits* sections of this chapter for instructions.)
- If you have the *system\_administration* permission, you can search the XML configuration files in which the SEER, SEER Extended, and system edits are defined. Refer to the *System Administration Page* section of Chapter 27 for instructions to access the system files. The edit system files are named according to the edit group. Once you select the XML file, use the Firefox search tools to find a particular edit.

## Viewing Edit Documentation while Editing Data

All users with permission to view or edit record or patient set data may view documentation for edits that fail during that editing session.

*To view information related to an edit that failed for a particular patient set or record:*

1. Open the record or patient set in the SEER\*DMS editor.
2. Click **Edits** in the left navigation section of the editor.
3. Click the Information Icon  shown next to the Edit ID.
4. The edit's ID, message, severity level, group, and list of entities will be displayed in a popup window. These components are described in the *Understanding Edits in SEER\*DMS* section of this chapter.

## Using the Edits Help Page

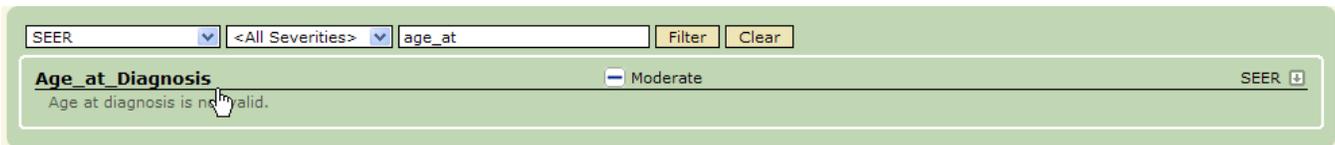
Requires system permission: *none* (all users have access to the help pages)

*To search the Edits Help page for documentation related to an edit:*

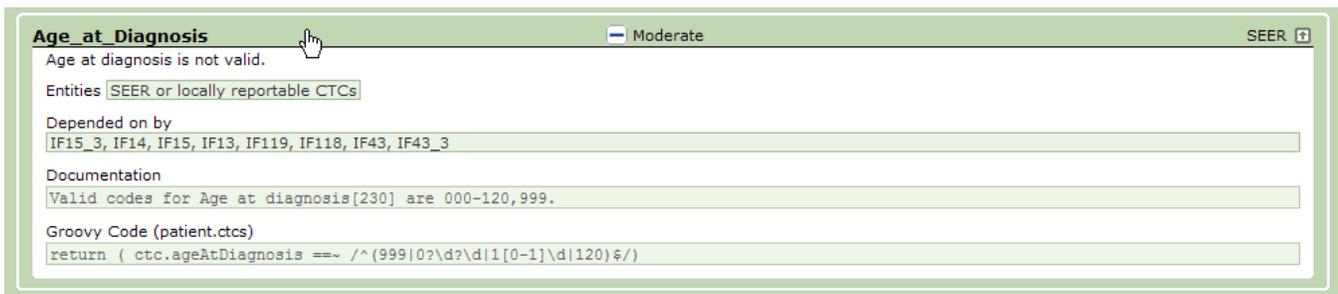
1. Select **Help > Edits**.
2. By default, all edits will be listed. You may use the filter to search by Edit Group, Severity Level, and/or text used in the message, ID, or source code.
  - a. Use the first drop-down menu to search by **Edit Group**.

- i. **All Groups** – This is the default setting.
  - ii. **<Your Registry's Name>** - Edits in this group are defined and maintained by registry staff.
  - iii. **SEER** – These are the edits implemented in the SEER\*Edits software package to validate fields submitted to SEER.
  - iv. **SEER\*DMS** – SEER\*DMS edits enforce database integrity of fields common to all SEER\*DMS registries.
  - v. **SEER\*DMS Registry** – SEER\*DMS Registry edits enforce database integrity of registry-specific fields (fields in the *patient\_registry*, *ctc\_registry*, *record\_registry*, and other *\_registry* tables in the database).
  - vi. **SEER-Extended** - Edits developed by NCI and IMS for the SEER\*DMS registries to validate fields that are not required in data submitted to SEER.
- b. Use the second drop-down menu to search by **Severity Level**. Each edit is assigned a severity level of low, moderate, high, or critical. Please refer to the *Understanding Edits in SEER\*DMS* section of this chapter for a detailed explanation of edit severity levels.
  - c. Enter text in the *keyword* box to search for text in the edit message, ID, or Groovy source code.
  - d. Click **Filter** to apply the search criteria. You may use **Clear** to reset the filter to the default values (All Edit Groups, All Severity Levels, and no search text).

The following example shows the result of using the filter to find Edit Group = SEER, Severity Level = All, and search text = *age\_at*.



Click the edit's ID to expand the display of the edit information on your screen, as shown below. The edit's Message is listed first, followed by Entities, Dependencies, Documentation, and the Groovy source code. See the *Understanding Edits in SEER\*DMS* section of this chapter for a description of each component.



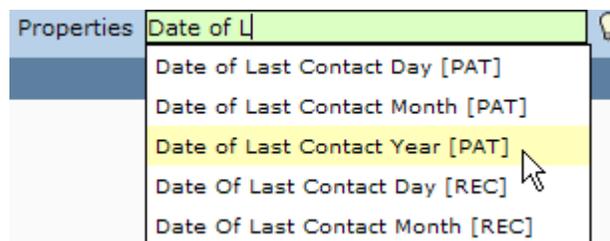
## Using the Edits Manager

Requires system permission: *system\_administration*

Use the Edits Manager to search for an edit or view a list of edits based on search criteria. If you are modifying or deleting an edit, follow the steps below to search for the edit and then refer to the *Defining and Maintaining Registry Edits* for further instructions.

To filter the list of edits in the Edits Manager:

1. Select **System > Edits**.
2. By default, all edits will be listed. You may use the filters to search by Edit ID, Edit Group, Severity Level, database Properties, Data Type, and text used in the message, ID, or source code.
3. The **Edit ID** is the edit's name or label (examples: IF104, SEER\_IR09, Age\_at\_Diagnosis). Use the **Edit ID** filter to search for an edit based on a complete or partial Edit ID.
4. Use the **Group** filter to limit the list to the edits in a particular group. The edit groups are listed below.
  - a. **<Your Registry's Name>** - Edits defined and maintained by registry staff.
  - b. **SEER** – These are the edits implemented in the SEER\*Edits software package to validate fields submitted to SEER.
  - c. **SEER\*DMS** – These are system edits that enforce database integrity of fields common to all SEER\*DMS registries.
  - d. **SEER\*DMS Registry** – These are system edits that enforce database integrity of fields specific to your registry.
  - e. **SEER-Extended** - Edits developed by NCI and IMS for the SEER\*DMS registries to validate fields that are not required in data submitted to SEER.
5. Use the **Severity** filter to search for edits based on severity level. Each edit is assigned a severity level of low, moderate, high, or critical. Please refer to the *Understanding Edits in SEER\*DMS* section of this chapter for a detailed explanation of edit severity levels.
6. Use the **Properties** filter to search for edits which reference a specific database field in the Groovy source code. You may click the Lookup icon  to search for a field; or enter text directly into the filter (SEER\*DMS will display a list of properties that match your text as you type). The text used in the Properties lookup is the Description shown on the Field Mappings help page.



7. Use the **Data Type** filter to search for edits based on the tables used in the edits' Entities.
8. Use the **Info** filter to search by text included in the Edit ID or Message.
9. If the filters are set appropriately, click **Apply**.
10. Click the ID of an edit to view or modify the edit's logic, message, or documentation. The specific components of an edit are described in the *Understanding Edits in SEER\*DMS* section of this chapter.
11. If you would like to restore the default filter settings, click **Reset**.

# Understanding Edits in SEER\*DMS

The components of SEER\*DMS edits are described below.

## Edit ID

A unique identifier assigned to each edit. You must specify an ID for a new registry edit (1-50 alphanumeric characters may be used).

*Guidelines for specifying an ID for a registry edit:*

- Use your registry's two-character abbreviation as a prefix for all registry-defined edits (AK, CN, CT, DT, HI, IA, LA, NM).
- If you are creating an edit designed to replace or supplement a SEER Edit, the ID should consist of your registry's prefix followed by the SEER Edit ID (e.g., DT\_IF29, IA\_IF02).

## Group

In SEER\*DMS, edits are organized into five groups:

1. SEER Edits validate fields submitted to SEER. These edits were developed and are maintained by the developers of the SEER\*Edits software.
2. SEER Extended Edits are defined and maintained by SEER for the SEER\*DMS registries. These edits validate fields that are supported by SEER\*DMS, but not required to be transmitted to SEER.
3. Registry Edits are defined and maintained by registry staff. These would include edits to validate registry-specific fields, edits to implement validation not covered by SEER edits, and registry-specific versions of SEER edits (the registry may decide to vary the logic of a SEER edit by creating a registry-defined edit and disabling the SEER edit).
4. SEER\*DMS edits enforce database integrity of fields common to all SEER\*DMS registries
5. SEER\*DMS Registry edits enforce database integrity of registry-specific fields. The SEER\*DMS development team creates these edits, if necessary, when registry-specific fields are added to the database.

## Entities

When defining an edit, you must specify the database entities to which the edit will be applied. The available entities include complete database tables, for example, some edits may be run against all patient sets (entity = patients). Other entities are subsets of database structures (for example, edits may be run against "CTCs with sequence = 1" or "CTCs needing review").

Registry staff may not add or remove entities to the list. The list of available entities is defined by the SEER\*DMS development team when the system is configured for the registry.

*Guidelines for specifying Entities for registry edits:*

- Record Edits: Select an entity related to Records ("Records" may be the only appropriate entity or your registry may have entities that define subsets of records).
- Patient Set Edits: Based on the fields used in the edit's logic, select an entity related to the database table at the lowest level of the patient set hierarchy.
  - Identify the database table used for each field used in the logic of the edit.
  - Refer to the Simplified DB Model of the Patient Set provided in the Database Documentation section of the SEER\*DMS Web Portal.

- Determine the lowest level table used in the edit's logic.
- Select an entity related to that table.
- For example, an edit comparing age at diagnosis and age at birth uses the CTC and Patient tables. The CTC table is at a lower level in the patient set hierarchy than the patient table. Therefore, you should select an entity related to the CTC table.
- If your edit requires an entity that is not listed, submit a change request via the SEER\*DMS Technical Support Squish project.

## Message

The edit message is the text that is displayed in the patient set or record editor when the edit fails. The Edit ID and Message are shown when you hover over a field involved in the failed edit. The edit messages are also displayed in all edit reports. The maximum length for an edit message is 500 characters.

*Guidelines for defining Messages for registry edits:*

- The edit message should reflect the logic of the edit and should be written in language that is easily understood by data editors.
- It is recommended that the message text includes "review required" if the edit is associated with a review flag.

## Severity

Each edit is assigned a severity level of low, moderate, high, or critical. The severity level is used to determine whether the data can be saved, whether a manual task is required, and to prioritize the edits for the user performing editing tasks.

Record edits are assigned a high severity level if the data item is required in order to screen the record for reportability or special studies. Therefore, a record edit with a high severity level will trigger a Resolve Record Errors task. If a record only has fields with low or moderate edit errors, the fields are flagged but the record moves forward in the workflow and is screened for reportability. If a record has one or more fields with high or critical edit errors, the record is sent to a manual Resolve Record Errors task for review prior to reportability screening.

Edits with a critical error level must be resolved in order to save the record or patient set. The critical error level is typically reserved for edits that check values to determine if they violate database constraints. For example, a critical edit error would occur if an "A" were entered in a field that requires a numeric value. *Note: It is technically possible to configure the edits to use the critical error level for other purposes. However, it is recommended that you only assign the critical error level to edits that check for database constraints.*

*Guidelines for setting the severity level of a registry edit:*

- You should rarely define an edit as critical. Typically, the SEER\*DMS development team will define a critical edit, if necessary, when a registry-specific data field is added to the database. Remember that your editing staff will not be able to save their changes if the changes generate a critical error.
- Record Edits:
  - Set the severity level to High if you want SEER\*DMS to create a Resolve Record Errors task when a record fails this edit.
  - Set the severity level to High if the edit is tied to the record reviewed flag for the purpose of initiating follow-back. A manual Resolve Record Errors task will provide a user with the opportunity to create the follow-back, set the reviewed flag, and save the record. The record would then move forward in the workflow.

- If you want the edit to flag fields as having a problem but do not want a Resolve Record Errors task created, set the severity level to either low or moderate. Registry policy dictates when to use low versus moderate.
- Patient Set Edits:
  - Registry policy dictates whether to assign a severity of low, moderate, or high to a patient set edit.
  - For SEER and SEER\*DMS edits, the low severity level typically indicates that a review flag will over-ride the edit.
  - The severity levels of failing edits do not affect the workflow routing of patient sets. If a patient set is saved with edit failures in a Visual Edit Patient Set or Consolidate task it will move to a Resolve Patient Set Errors task, regardless of the severity level of the failing edits.

## Dependencies

SEER\*DMS uses “dependencies” to conditionally execute an edit based on the success of other edits. Typically, a dependency is added to an edit to ensure that a valid value is provided for a required field. For example, several SEER edits require age at diagnosis. These edits are not executed if the “*age\_at\_diagnosis*” edit fails.

There are two sets of dependencies for each edit. An edit is not executed if any of the edits listed in its “Depends on” section fail. Similarly, if the edit fails then none of the edits in the “Depended on by” section will be applied.

To view examples, review the dependencies for the following SEER edits: Age\_at\_diagnosis (Age at diagnosis is not valid), IF118 (Age, Primary Site, Histology conflict), IF13 (Age, Birth Date and Date of Diagnosis conflict).

Dependencies can only be defined between edits within the same group. For example, a registry edit can only depend on registry edits and cannot depend on SEER edits.

## Groovy Code

SEER\*DMS edits are implemented in Groovy, a scripting language for the Java platform. The Internet has several Groovy references including the Groovy home page at <http://groovy.codehaus.org>. You do not need to be fluent in Groovy to add or modify edits. SEER\*DMS edits use a small subset of the Groovy syntax. A working knowledge of regular expressions and Groovy logic statements are all that is needed to add and modify edits in SEER\*DMS. To define a new edit, it is recommended that you copy-and-paste the definition from an existing edit and use that code as a template.

*Guidelines for writing the Groovy code for a registry edit:*

- An edit error is triggered if the code returns FALSE for the record or patient set. The edit passes if the code returns TRUE.
- Use the Groovy code of a similar edit as a template.
- If you copy-and-paste the Groovy code of a SEER, SEER\*DMS, or SEER Extended edits, verify that an appropriate entity is available. If your edit requires an entity that is not listed, submit a change request via the SEER\*DMS Technical Support Squish project.
- The Available Contexts are displayed at the bottom of the page. Your Groovy code may include references to any of the Available Contexts. Refer to the SEER\*DMS Web portal for tutorials on using contexts in registry edits.

## Contexts

A context is a Java naming system. In SEER\*DMS, contexts are used to define arrays, hash tables, and functions used by the edits. For example, there are a large number of contexts defined for the SEER\*Edits. Primarily, these represent data tables required by the SEER Edits logic. Registry staff may not add or modify contexts in the current version of SEER\*DMS.

## Documentation

An edit's documentation is descriptive text that is more comprehensive than the edit message. If available, the edit documentation is displayed in the edit popup shown in the patient set and record editor. Edit documentation is also displayed in the detailed edit report (RPT-064D).

Documentation from the SEER\*Edits software is displayed in SEER\*DMS for the SEER Edits. Registry staff are responsible for writing and maintaining documentation for the registry edits.

## Defining and Maintaining Registry Edits

Requires system permission: *system\_administration*

The instructions below provide an overview of the steps required to add, delete, or modify an edit. Refer to the *Understanding Edits in SEER\*DMS* section of this chapter for specific guidelines when selecting values for each component of the edit.

*To define a new registry edit:*

1. Select **System > Edits**.
2. Click **Add Edit**.
3. Set all components shown on the **General** tab (specific guidelines are provided in *Understanding Edits in SEER\*DMS*).
  - a. Enter the **Edit ID**.
  - b. Select one item listed in the **Entities** drop-down list.
  - c. Click the **Severity** level that you wish to assign to this edit. Highlighting is used to identify the current selection. In the example below, the Severity level is Moderate.



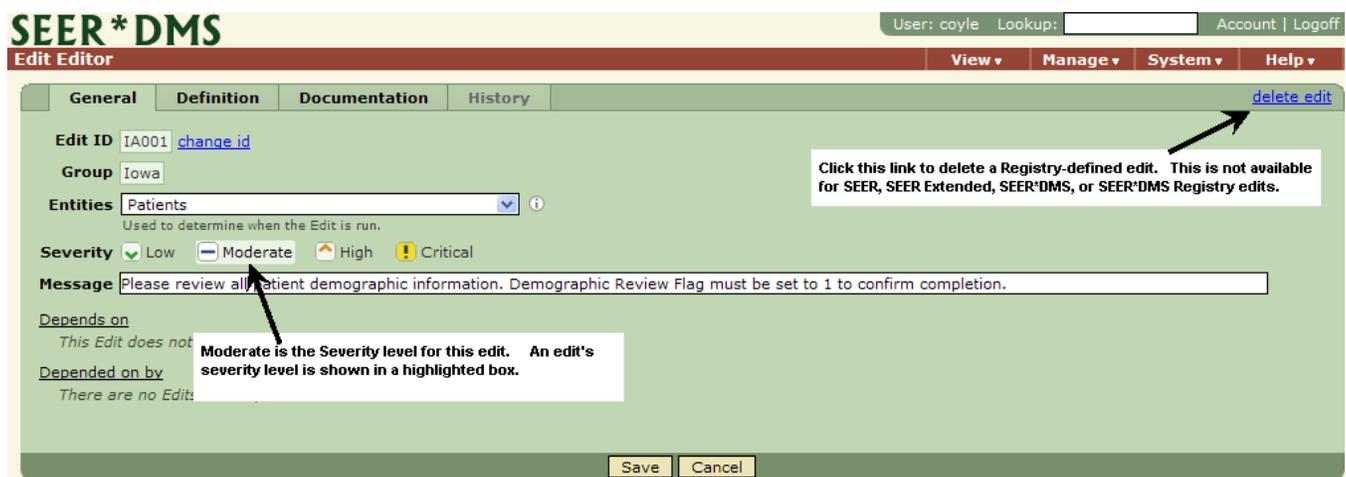
- d. Enter the **Message** text.
  - e. An edit is not executed if any edit listed in the **Depends on** section fails. To add an edit to the Depends On section:
    - i. Click **add dependency**
    - ii. Use the filter to search for the edit. Dependencies can only be defined between edits within the same group. Therefore, only registry edits will be listed.
    - iii. Click the **Edit ID**.
4. On the **Definition** tab, enter **Groovy** source code to define the logic of the edit. The Definition tab also includes a list of the Available Contexts. These Contexts are defined by the SEER\*DMS development team, as needed. Your Groovy code may include references to any of the Available Contexts. Contexts are organized by the edit groups. A Context is

directly available to edits within the Context's group. However, edits in other groups may reference Contexts in other groups. Refer to the SEER\*DMS Web portal for tutorials on using SEER edit contexts in registry edits.

5. On the **Documentation** tab, enter detailed documentation related to the new edit.

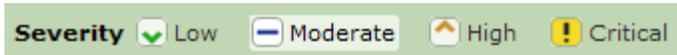
To delete a registry edit:

1. Select **System > Edits**.
2. Use the filters to find the edit as described in the *Using the Edit Manager* section of this chapter. To search for registry-defined edits, set the Group filter to your registry's name. (SEER, SEER Extended, SEER\*DMS, and SEER\*DMS Registry edits cannot be deleted.)
3. Click the edit's **ID**.
4. Review the list of edits that depend on this edit (listed under the *Depended on By* heading). Determine whether this edit must be retained in order for those edits to function properly; or if the listed edits require modifications due to this edit being deleted.
5. If you determined that the edit can be deleted, **delete edit**. (This link is shown in the upper right corner of the screen for Registry Edits. It is not displayed for SEER, SEER Extended, SEER\*DMS, or SEER\*DMS Registry edits).



To modify a registry edit:

1. Select **System > Edits**.
2. Use the filters to find the edit as described in the *Using the Edit Manager* section of this chapter. To search for registry-defined edits, set the Group filter to your registry's name. (SEER, SEER Extended, SEER\*DMS, and SEER\*DMS Registry edits cannot be modified.)
3. Click the edit's **ID**.
4. Set all components shown on the **General** tab (refer to *Understanding Edits in SEER\*DMS* for more specific guidelines).
  - a. To modify the Edit ID, click **change ID**.
  - b. Select one item listed in the **Entities** drop-down list.
  - c. Click the **Severity** level that you wish to assign to this edit. Highlighting is used to identify the current selection. In the example below, the Severity level is Moderate.



- d. If you will be modifying the edit logic, update the **Message** text to reflect the changes.
  - e. An edit is not executed if edits listed in the **Depends on** section fail.
    - i. To add an edit to the Depends On section, click **add dependency**. Use the filter to search for the edit. Dependencies can only be defined between edits within the same group. Therefore, only registry edits will be listed. Click the **Edit ID**.
    - ii. To delete an edit from the Depends on section, click the **remove** link associated with the edit. The remove link is shown in the right-most column of the table.
  - f. Edits listed in the **Depended on by** section will not execute if the edit being modified fails. This section is shown for your information, but cannot be changed on this screen. The edits listed in this section depend on the edit that you are currently modifying. You must consider how your modifications will impact these edits.
5. On the **Definition** tab, modify the **Groovy** source code as needed. The Definition tab also includes a list of the Available Contexts. These Contexts are defined by the SEER\*DMS development team, as needed. Your Groovy code may include references to any of the Available Contexts. Contexts are organized by the edit groups. A Context is directly available to edits within the Context's group. However, edits in other groups may reference Contexts in other groups. Refer to the SEER\*DMS Web portal for tutorials on using SEER edit contexts in registry edits.
  6. On the **Documentation** tab, modify the detailed documentation to reflect the changes that you made to the logic.
  7. Click **Save**.
  8. Enter a comment describing your changes. These comments will be displayed in the edit's History table that is displayed wherever edit information is shown.

## System Task to Execute Edits in Patient Sets

All edits are executed each time a patient set is opened, validated, or saved in the SEER\*DMS editor. The Patient Set Edits system task enables you to re-execute the edits on patient sets in the database. You may run the edits on all patient sets or on a cohort defined by year of diagnosis. Use the Patient Set Edits task to ensure that new or modified edits are evaluated.

A polisher is a system utility that derives, calculates, or assigns data field values. For example, polishers are used to derive collaborative stage variables; assign census tract based on address; and calculate the age at diagnosis based on date of birth and date of diagnosis. When a patient set is opened, saved, or validated, a polisher will be executed if the value of a related data item changes. Occasionally, you may need to execute a polisher on a large number of patient sets. The Patient Set Edits task enables you to run selected polishers as well as the edits.

Polishers are assigned to "classes". Classes are simply categories used in SEER\*DMS to define when the polishers are executed during various system processes. A polisher may belong to multiple classes. The Polishers help page lists the polishers and the classes to which they belong.

The Patient Set Edits task has an option to execute the Edit Polishers. Edit polishers are polishers in the "Pre-Edits" and "Post-Edits" classes (the polishers in these classes are listed on the screen when you open the Patient Set Edits system task). These are polishers that need to be run with the Patient Set Edits task during the initial migration of data to SEER\*DMS. The edit polishers rarely need to be run as a group after deployment.

A polisher may need to be executed with the Patient Set Edits task when a new version of SEER\*DMS is released that includes a change to that polisher. The Patient Set Edits task allows you to select one to three "extra" polishers. All of the standard polishers (polishers in the "Validation" class) can be executed with the Patient Set Edits task.

To re-execute the edits for some or all patient sets in the database:

1. Click **System > Tasks**.
2. Click the **Patient Set Edits** link.
3. To limit the edits to data by year of diagnosis, enter the **Start Year**. Patient sets with a diagnosis date during or after this year will be considered.
4. If you would like to define an end range, enter an **End Year**. If the end date is not specified, today's date will be used by default.
5. To include data with unknown year of diagnosis, set **Include Unknown Year** to Yes.
6. If you wish to execute all polishers in the Pre-Edit and Post-Edit classes, set **Edit Polishers** to Yes. It is recommended that this option be set to *No* unless there is a specific need related to the transitioning of data into SEER\*DMS.
7. You may execute one to three individual polishers with the Patient Set Edits using the **Extra Polisher** drop-down menus. These are polishers in the Validation class (as listed on the Polishers help page).
8. You may enter text related to this task in the **Comment** field. The comment for the last execution of the task is stored in the database (utility\_history table).
9. Click **Start**.

The edits will be re-evaluated for each patient set in the cohort. In order to avoid creating an inordinate number of worklist tasks, a Resolve Patient Set Errors task will *not* be created for each patient set with an edit error. If the logic of a new or modified edit is implemented incorrectly, it could erroneously create an edit error for a large number of patient sets. Therefore, you must use reports rather than tasks to identify the patient sets with errors and to evaluate the error levels in the patient set data. Two system reports, RPT-064A and RPT-064B, are available for identifying the edit errors that were triggered and the patient sets that are involved:

## Reports Related to Edits

Select **View > Reports** to access the system reports provided with SEER\*DMS. The reports listed below are the edit reports available in SEER\*DMS when this manual was printed.

| Report ID | Title  | Description   |
|-----------|--|---|
| RPT-064A  | Frequency of Edit Errors in Patient Set Data | Use this report to evaluate the error levels in the patient data. Set the parameters to generate frequencies of errors in all patient sets when verifying modifications that you made to edits or when you run the Patient Set Edits system task. |
| RPT-064B  | Patient Sets with Edit Errors                | A listing of Patient Sets with a failure related to a particular edit or a sample for all failing edits.  |
| RPT-064C  | SEER*DMS Edits                               | ID, Message, Severity, and Group of all edits available in SEER*DMS.  |
| RPT-064D  | Detailed Listing of SEER*DMS Edits           | Lists all components of edits, including the Groovy code, entities, history, and documentation.   |

## Color Codes Used in SEER\*DMS Editors

The SEER\*DMS record and patient set editors display data fields in different colors to highlight fields that contain errors, and fields that have been modified but not validated or saved. Although you can readily see the fields that require attention on data pages, it is recommended that you review the list of errors and related data fields on the Edit Errors page prior to making changes to a record or patient set. Refer to the chapter related to your specific task for further instructions.

If two or more edits are related to a field, the edit with the highest severity level will determine the color.

|   |   |
|---|---|
|  1981        | <b>White</b> indicates that the field does not have an error and has not been changed since the last save.  |
|  XXXX        | <b>Dark red with white text</b> indicates that the field triggered a <b>Critical Error</b> . The value can not be stored in the system due to database constraints. Critical Errors will only be seen in patient sets, since database constraints are not applicable to record data.  |
|  0000        | <b>Medium red with black text</b> indicates that the field triggered an error with a <b>High</b> severity level. In patient set data, the value may not be valid for this field or an inter-field edit may have detected a conflict with other fields. In record data fields, this indicates that an error was detected that must be resolved prior to screening. The record will not move past the Resolve Record Errors task until all errors with a severity level of high are resolved. |
|  9999        | <b>Light red</b> indicates that the field has an error with a <b>Low or Moderate</b> severity level. The value may not be valid for this field or an inter-field edit may have detected a conflict with other fields. Low and moderate errors should not be resolved when editing records, these editing tasks should be performed when consolidating the patient data.   |
|  1980      | <b>Orange</b> indicates that a field with an error was modified, but the change has not been validated against the SEER and local edits. Fields are validated when the record is saved or the Validate button is clicked. Once validated, this field will turn to yellow if the error was corrected or to a shade of red if the field still contains an error.  |
|  1990      | <b>Yellow</b> indicates that the field that did not have an error and was changed. The patient set or record has not been saved since the change was made. Once validated, this field will turn to a shade of red if there is an error.   |
|  1980      | <b>Blue with a black border</b> indicates that the field was changed by a Polisher.   |
|  0000      | <b>Blue with no border</b> indicates that the field is read-only and was changed by a Polisher. <i>Note: Some fields are modified when another field is changed.</i>  |
|  DOE, JOHN | Read-only fields that have not been changed are shown in a lighter shade (the color is determined by the color scheme that you are using).  |