Visual Modeler

Administration Guide

Release 9.1.04

Visual Modeler

Administration Guide

Release 9.1.04

Note
Before using this information and the product it supports, read the information in "Notices" on page 143.

Contents

Chapter 1. Visual Modeler Tasks Checklist

This topic describes a checklist of tasks that you must perform to ensure that the Visual Modeler is functioning appropriately.

- Install the IBM® Sterling Selling and Fulfillment Foundation and perform the necessary configurations. For more information, refer to the *Sterling Selling and Fulfillment Foundation: Installation Guide*.
- Create products and define their details using the IBM Sterling Business Center application. Alternatively, you can load the products using the data load functionality. For more information about creating products using the Sterling Business Center application, refer to the *Business Center: Item Administration Guide*.
- Create storefronts using the Applications Manager and perform storefront administration tasks. For more information about creating an organization, refer to the *Sterling Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- Configure the models by performing the following tasks:
  - Install the Visual Modeler application and integrate it with the Sterling Selling and Fulfillment Foundation to enable them to exchange information. For more information about integrating the Visual Modeler with Sterling Selling and Fulfillment Foundation, refer to the *Visual Modeler: Implementation Guide*.
  - Configure the Sterling Selling and Fulfillment Foundation appropriately to enable the Visual Modeler and the Sterling Selling and Fulfillment Foundation to exchange information. For more information about integrating the Visual Modeler with Sterling Selling and Fulfillment Foundation, refer to the *Visual Modeler: Implementation Guide*.
  - Note that you must extract the `sic_properties.zip` file located in the `<INSTALL-DIR>/repository/external` folder and copy the .properties files to the same location where the properties file are located as configured in the Applications Manager.
  - In the Visual Modeler application, create a storefront with the same Skin ID as the Organization Code of the catalog organization defined in the Applications Manager.
  - In the Visual Modeler application, create configuration models. While creating the models, associate the products to the option items appropriately.
  - Compile and test the models.
- Create configurable products using the Sterling Business Center application and associate models with those products. For more information, refer to the *Business Center: Item Administration Guide*.
- Generate the search index. For more information about generating the search index, refer to the *Business Center: Item Administration Guide*.
- Install IBM Sterling Web and configure it appropriately to enable a user to create a cart and place an order. For more information, refer to the *Sterling Selling and Fulfillment Suite: Applications Installation Guide* and the *Sterling Web: Implementation Guide*.
- Access the Web channel application and verify that orders can be placed and processed by performing the following tasks:
  - Browse the catalog to find a configurable product.
  - Configure the product and add the configuration to a cart.

- Place an order for the cart.
- Check the order and shipment status.

Chapter 2. Configuring the Visual Modeler

The properties of the Visual Modeler are defined in a set of configuration files and in the Knowledgebase. When the servlet container is started, the Visual Modeler loads and accesses the configuration files in order to determine Visual Modeler settings.

System properties for each enterprise (tenant and storefronts) are managed in the CMGT_SYS_PROPERTIES database table: for each storefront default values are populated when the storefront is first created. When a system property is changed through the Web UI, then the new value is stored in the table.

After implementation, you can modify the settings using the System Administration page and the Business Rules Manager.

Site System Administration

Site system administration is performed by site system administrators. Site system administrators access the Visual Modeler through the site administration URL, which is of the form:

`http://server:port/Sterling/en/US/enterpriseMgr/admin`

The default login ID is admin, password admin. Site system administrators manage properties that are common to all storefronts, such as logging. For more information, refer to Chapter 45, “Site System Administration,” on page 135.

Enterprise System Administration

Enterprise system administration manages enterprise-level (either tenant or storefront) settings in the Visual Modeler. For example, you can specify the email settings in the System Properties page.

You can only modify system configuration settings if you have the appropriate access role. In the reference implementation provided with the Visual Modeler, only users with the Program Management function can access the System Administration pages. These users access the Visual Modeler through an enterprise administration URL which is of the form:

`http://server:port/Sterling/en/US/enterpriseMgr/matrix`

Enterprise system administrators manage properties that are specific to their enterprise: changes that they make do not affect other enterprises. For more information, refer to Chapter 39, “Modify System Settings,” on page 121.

Business Rules

Business Rules define the behavior of the Visual Modeler. For example, this includes punchin and punchout specifications, the behavior of imports and exports, cluster configuration, and other product management specifications. These business rules are specified in the properties files of the Visual Modeler, and are managed through the business rule administration interface. For more information, refer to Chapter 42, “Manage Business Rules,” on page 127.

Job Scheduling

You can create cron jobs for different activities in the Visual Modeler. There are storefront-level and enterprise-level cron jobs. Each storefront manages its own set of cron jobs. Only enterprise administrators can manage the enterprise level cron jobs.

You can schedule system-level or application-level cron jobs to run according to a specific frequency between a certain date and time range.

Note: When a job is run using the Job Scheduler, its execution status is recorded. On occasion, a job may execute successfully but the status is recorded as "Timed Out".

Chapter 3. Managing Users

Create a New Enterprise User

About this task

To create a new enterprise user:

Procedure

1. Click **System Users** in the System Administration panel on the Visual Modeler enterprise home page.
2. The User List page is displayed. Click **Create User**.

   Note: The **Create User** button is displayed only if you have access privileges to create a new user.
3. The User Detail page is displayed. Enter the details of the new user. Note the following:
   - Username: This username must be unique throughout the Visual Modeler.
   - Password: Use letters and numbers from the keyboard with no spaces or other punctuation.
   - User Functions: Select the **Program Management** check box. Select any other function that this user will perform by selecting the appropriate check boxes. The list of functions is determined at implementation time.

     Note: Do not select the ERPAdministrator user type for standard users. Users of this user type cannot log in through the Web user interface.
   - Preferred Locale: Select the preferred locale which will apply when the user logs in. The drop-down list displays the names for the supported locales.
4. Click **Save**.

   The User Detail page is displayed with new tabs.
5. You can update information on the **Info** tab and click **Save**.

Modify an Enterprise User Profile

About this task

You can change user profile information for another enterprise user or partner user if you have the right level of entitlement access to the user.

To modify an enterprise user profile:

Procedure

1. Click **System Users** in the System Administration panel on the Visual Modeler enterprise home page.
2. The User List page is displayed. Click the link corresponding to the user whose profile you wish to modify. If you cannot see the user whose details you want to update, then you can search for the user.
3. The User Detail page is displayed. Modify the user details as appropriate.
4. Click **Save**.

Delete an Enterprise User

About this task

To delete an enterprise user:

Procedure

1. Click **System Users** in the System Administration panel on the Visual Modeler enterprise home page.
2. The User List page is displayed. Click **Delete** corresponding to the user you wish to delete.

   The user is deleted from the Visual Modeler. However, note that the username that belonged to this user is still present in the system. No new user can re-use this username.

Search for an Enterprise User

About this task

To search for an enterprise user:

Procedure

1. Click **System Users** in the System Administration panel on the Visual Modeler enterprise home page.
2. Select Username, First Name, or Last Name, from the drop-down list, and enter the full or partial string for the search. You can use "*" as a wild card character.

   For example, if you select First Name and enter "An*", then you will find all enterprise users whose first name begins with "An" such as Andrew and Anne.
3. Click **Go**.
4. You can click **Advanced Search** to perform a more detailed search.

Search for Any User

About this task

You can view the user details of any user in the Visual Modeler. If you have the appropriate function, then you can also modify user details or delete them from the Visual Modeler.

You can use the Search for User by Name panel to perform a quick search for a user, or you can use the advanced search capabilities as follows:

Procedure

1. Click **Advanced Search** in the Search for User by Name panel.
2. Enter the search criteria and click **Go**.

   The search results page displays the users that match your search criteria.
3. Click the link corresponding to the user you are searching for.

Chapter 4. Managing the Enterprise Profile

Enterprise users are responsible for managing the profile of the storefront that is created to enable integration with the Sterling Selling and Fulfillment Foundation. The information is organized by tabs.

The following table describes the key fields displayed on the Profile Detail page:

Field	Description
Info Tab	
Profile Name	The display name for the profile. Profile names do not have to be unique. However, in several places in the user interface, profiles are listed by profile name. Distinguishing two profiles with the same name in any list of profile names can be confusing. Use a naming convention that ensures that profile names are effectively unique.
Detail Tab	
Organization ID	The unique identifier of the organization. This ID is used by the enterprise to uniquely identify each organization with which it does business.
External Partner ID	The Organization Code of the catalog organization that is created using the Applications Manager. This is required to ensure that the correct Organization Code is passed while making XAPI calls to the Sterling Selling and Fulfillment Foundation.
Commerce Tab	
Skin Url	Click the **Create New Skin** button to specify a skin URL for the profile. This should be a simple string and must be unique within the Visual Modeler. This string will be used in URLs used to access the storefront.

Chapter 5. Create a Storefront

About this task

The storefront that you create using the Visual Modeler is meant only for administration purposes. To ensure that the correct product information is used for defining the models in the Visual Modeler, you must create a storefront with the same Skin ID as the Organization Code of the catalog organization.

To create a storefront:

Procedure

1. Click **Go** in the Search for Organization by Name panel on the Visual Modeler home page.
2. The Profile List page is displayed. Click **Create Storefront**.

   The Organization Detail: New Profile page is displayed.
3. Enter basic information for the storefront administrator partner as you would do for any other partner. * denotes required fields.
4. In the Detail tab, in the **External Partner ID** field, enter the Organization Code of the catalog organization as defined in the Applications Manager. This is required to ensure that the correct Organization Code is passed while making XAPI calls to the Sterling Selling and Fulfillment Foundation.
5. Enter a skin URL for the new storefront. This should be a simple string and must be unique within the Visual Modeler. For example, you can use "anderel" or "storefront". This string will be used in URLs used to access the storefront. For example:

   ```
   http://server:port/Sterling/en/US/enterpriseMgr/anderel
   ```
6. Click **Save**.

   You must create at least one user to act as the first storefront administrator for the new storefront. Notify the organization for whom you have created the storefront and provide them with their new storefront URL and their storefront administrator userid.

Chapter 6. Visual Modeler Interface

This topic describes the general layout of the Visual Modeler interface.

Note: Models are compiled to XML files. Consequently, do not use the following characters when naming models and model entities such as groups, properties, and rules: "&", "/". "@", "!", and the quote characters " and '.

The Visual Modeler page consists of three frames:

- **Model Groups**: When you first access the Visual Modeler page, this frame displays the root model group, that is, the highest group in the model group hierarchy.
- You can expand a model group to display the model groups within it by clicking the plus (+) sign.
- **Models and Groups**: This displays the models and groups that are children of the model group selected in the Model Groups frame.
- **Content**: This displays information about the model group selected in the Model Groups frame. The information is collected into the following tabs:
  - **General Info**: Displays the children of the model group (where you can select, delete, and reorder children). You can also create new model groups, new models, and new groups. You can also upload models or model groups to the currently selected model group.
  - **Properties**: You can define new properties in this tab which you can then attach to any model, option class, or option item within the model group for which the property was defined. In the same way, you can also use the property in rules defined for any model or model group in the hierarchy below the model group for which it is defined.
  - **Rules**: You can define rules for the model group. These rules can be attached to any models, option classes, or option items in the hierarchy below the model group for which it is defined.
  - **Lists**: The list you define here can be used in any properties in the hierarchy below the model group for which it is defined.

In addition, the Visual Modeler page contains a *toolbar* across the top with access to the following tasks:

- **Edit**: This enables you to edit a model, option class group, or option item group highlighted in the Models and Groups frame.
- **Compile**: This enables you to compile a model, option class group, or option item group into an XML file. See Chapter 15, “Compiling a Model,” on page 47. Only compiled models can be associated with configurable products.
- **Test**: This enables you to test a model that you are creating or modifying. See Chapter 14, “Test a Model,” on page 45.
- **Copy**: This enables you to copy a selected entity (model group, model, option class group, and so on).
- **Import**: This enables you to import an entity into your library of entities. See Chapter 28, “Import a Model Group or Model,” on page 89.
- **Export**: This enables you to export an entity. See Chapter 29, “Export a Model Group or Model,” on page 91.

- **Report**: This enables you to produce a report on some entity in the model library. See Chapter 32, “Run A Report,” on page 97.
- **Search**: This enables you to search for entities based on selected search parameters. See Chapter 11, “Search for Entities,” on page 39.

When you build a model, you use the model detail page. The detail page contains the following frames:

- **Toolbar**: as described above.
- **Navigation**: Click the plus (+) sign to expand the model or group and display the elements of the model or group: the sub-models, option classes, option items, or groups.
- **Content**: This displays information about the model selected in the Navigation frame. By navigating to a particular node in the model, you can create and update information on that node. This information is collected into the following tabs:
  - **General Info**: Displays general information about the model or group, as well as the children. You can delete or re-order children in this frame. You can translate the model, assign a product ID to the model, create option classes and (for option item groups) option items, and attach groups. You can also download models from here.

    Note: Only the General Info tab appears for option class groups or option item groups.
  - **Display**: This tab enables you to define display properties at the model level. These properties include things like constant guiding text, as well as pre- and post-pick guiding text. Some display properties have default values which can be overridden by display values set at the option class or option item levels. Note that all the properties displayed on the **Display** tab can also be set by setting UI properties on the **Properties** tab. See “Working with Display Properties” on page 99 for more information about display properties and UI properties.
  - **Properties:** If the current node is a model, then this tab consists of two tabs: **Attach** and **Define**, otherwise you can use this tab to attach properties to the node. In the **Attach** tab, you can attach to the model properties to which the model has access. (The model has access either to properties defined specifically for the model itself or to properties defined for any model group above the model in the model group hierarchy.) In the **Define** tab, you can define new properties for use locally, in the model's structure.
  - **Rules**: If the current node is a model, then this tab consists of two tabs: **Attach** and **Define**; otherwise you can use this tab to attach rules to the node. In the **Attach** tab, you can attach to the model rules to which the model has access. (The model has access either to rules defined specifically for the model or rules defined for any model group above the model in the model group hierarchy.) In the **Define** tab, you can define new rules for use locally, in the model's structure.
  - **Lists**: If the current node is a model, then the lists you define here can be used locally, in any properties you define for the model.
  - **Tables**: If the current node is a model, then you create constraint tables here.
  - **Tabs**: If the current node is a model, then you can create a tab-based configuration for your customers here. See “Create a Tabbed User Interface” on page 51.

– **Worksheets**: If the current node is a model, then this tab enables you to manage properties using worksheets. These provide you with a quick way to view and manage related properties and option items. See Chapter 19, “Using Worksheets,” on page 59.

Chapter 7. Access the Visual Modeler

About this task

To access the Visual Modeler:

Procedure

1. Click **Configuration Models in the Product Configuration Administration panel** on the Visual Modeler home page.

   The Visual Modeler page is displayed.
2. In the Models and Groups frame, click a model or a group.

   This displays the current structure (option classes, option items, and groups) for the selected model or group. Click the plus (+) sign to the left to expand the structure of the model.
3. Click **Edit** in the taskbar.

   This displays the detail page for the model, option class group, or option item group.
4. In the Navigation frame, click on the plus (+) sign to expand the model or group.
5. Click an option class.

   This displays the following tabs in the Content frame:
   - **General Info**: This tab provides general information about the selected option class. A list box displays the children belonging to this option class. You can also assign a product ID here, define a ratio for the class (the number by which the option item quantity will be multiplied to get the necessary option item quantity). You can create nested option classes and option items as well as attach groups.
   - **Display**: This tab enables you to set display property values specific to the selected option class.
   - **Properties**: You can associate with the option class properties to which the option class has access. (The option class has access either to properties defined specifically for the model to which the option class belongs or to properties defined for any model group above the option class in the model group hierarchy.)
   - **Rules**: You can attach rules defined for the model, as well as for the model group to which it belongs (or to any ancestor model group).
6. In the Navigation frame, click the plus (+) to expand the option class.

   This displays the children of the option class: these may be option items or option classes.
7. Click an option item.

   This displays the following tabs in the Content frame:
   - **General Info**: This tab provides general information about the selected option item: name and description, effectivity dates, and a field for assigning a product ID.
   - **Display**: You can set display property values specific to the selected option item.
   - **Properties**: You can associate with the option items properties to which the option item has access. (The option item has access either to properties

defined specifically for the model to which the option item belongs or to properties defined for any model group above the option item in the model group hierarchy.)

- **Rules**: You can attach any accessible rules to the option item. (The option item has access to any rules defined at any level above it in the model group hierarchy.)

Chapter 8. Working with Models and Model Groups

Create a Model Group

About this task

To create a model group:

Procedure

1. Navigate to and select the model group under which you wish to create the new model group. See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to display the model group.
2. Click **New Model Group**.

   This displays the New Model Group tab.
3. Enter a name and description for the new model group.

   Note: On Windows platforms, there is a 256 character limit for a fully-qualified pathname (this includes the pathname and the filename). Therefore, in Visual Modeler, take care not use long names for either model groups or models, particularly if you are using non-ASCII characters. When you compile a model, Visual Modeler recreates the model group structure as directories in the file system and, in the process, expands any non-ASCII characters.
4. Click **Save** or **Save and Edit** to save the new model group.

   The new model group appears in the Model Groups frame. If you clicked **Save and Edit**, then the Visual Modeler page appears, ready for you to edit the new model group. See "Modify a Model Group."

Modify a Model Group

About this task

To modify a model group:

Procedure

1. Navigate to and display the model group you want to modify.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to display the model group.

   This displays the General Info tab where you can modify the name and description of the group. You can also do one or more of the following:
   - Delete model groups, models, or groups that are children of the selected model group (see "Delete the Children of a Model Group" on page 18).

     Note: Click **Save All Changes** to save your changes before you leave the **General Info** tab.
   - Create a model group as a child of this group. See "Create a Model Group."
   - Create a model as a child of this group.
   - Create either an option class group or an option item group.
2. Click the **Properties** tab to create or modify properties for this model group.

   See Chapter 18, "Properties in Visual Modeler," on page 53.

Note: Click **Save All Changes** to save your changes before you leave the **Properties** tab.

3. Click the **Rules** tab to create or modify rules for this model group.

   See "Define a Rule" on page 71 or "Modify a Rule" on page 71

4. Click the **Lists** tab to create or modify lists for this model group.

   See "Define a List" on page 69 or "Modify a List" on page 69

Delete the Children of a Model Group

About this task

To delete one or more children in a group (a model group, a model, an option class group, or an option item group), use the following procedure:

Procedure

1. Navigate to and select the parent model group that contains the child you want to delete.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to display the model group.

2. In the list box, select one or more model groups (MG), models (M), option class groups (OCG) or option item groups (OIG) to be deleted.
   - You cannot delete a model group if the group has children. You must delete the children first.
   - You cannot delete a model if it is attached as a sub-model elsewhere in the model group hierarchy.
   - You cannot delete an option class group if it is attached to another model or option class group.
   - You cannot delete an option item group if it is attached to another model, option class group, or option item group.
3. Click **Delete**.
4. Click **Save All Changes**.

   The model group hierarchy will no longer display the deleted items.

5. Enter the Destination Model Group.
   a. Click **Browse**.

      This displays a Hierarchy Browser or Hierarchical Entity Chooser.

   b. Browse the model group hierarchy until you find the destination model group.
   c. Select the destination model group.
   d. Click **Done**.

      The model group appears in the Destination Model Group field.

6. As desired, modify the Destination Name field.

   The name defaults to the name of the model group being copied.

7. Click **Copy** in the Copy window.

   The model group is copied to the destination model group.

Copy a Model Group

About this task

You can copy a model group and its components into another model group.

To copy a model group:

Procedure

1. Navigate to and select the model group you wish to copy.
   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to display the model group.
2. In the taskbar, click **Copy**.
   This displays the Copy window.
3. Enter the Destination Model Group.
   a. Click **Browse**.
      This displays a Hierarchy Browser or Hierarchical Entity Chooser.
   b. Browse the model group hierarchy until you find the destination model group.
   c. Select the destination model group.
   d. Click **Done**.
      The model group appears in the Destination Model Group field.
4. As desired, modify the Destination Name field.
   The name defaults to the name of the model group being copied.
5. Click **Copy** in the Copy window.
   The model group is copied to the destination model group.

Create a Model

About this task

To create a model:

Procedure

1. Navigate to and display the model group under which you wish to create a model.
   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to display the model group.
2. In the **General Info** tab, click **New Model**.
   This displays the New Model tab.
3. Enter a name and description for the new model.
   If you plan to associate the model with a product ID, consider skipping this step. If the name and description match the name and description of the product ID, you can auto-fill these fields when you assign the product ID in Step 5.

   Note: On Windows platforms, there is a 256 character limit for a fully-qualified pathname (this includes the pathname and the filename. Therefore, in Visual Modeler, take care not use long names for either model groups or models, particularly if you are using non-ASCII characters. When you translate a model,

Visual Modeler recreates the model group structure as directories in a file system and, in the process, expands any non-ASCII characters.

4. Select the Start Date and End Date for the model.

   These are the dates within which the model is available for configuration. If the current date is outside these dates, the model is not available for configuration for any product with which it is associated.

5. If applicable, assign a product ID.

   See "Associate a Product with a Model, Option Class, or Option Item" on page 21.

6. Click **Save** or **Save and Edit** to save the new model.

   If you click **Save**, the **New Model** tab remains and the new model appears in the Models and Groups frame. You can create another model in this group.

   If you click **Save and Edit**, the Model Detail page appears with the new model in the Navigation frame. You can now add properties, rules, lists, and constraint tables for this model. You can also associate the model with a product. See "Modify an Existing Model."

Modify an Existing Model

About this task

To modify a model:

Procedure

1. In the model group hierarchy, navigate and display the Model Detail page for the model you want to modify.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to display the Model Detail page.

2. In the **General Info** tab, you can do one or more of the following:
   - Modify the name, description, and/or the start and end dates.
   - Delete one or more of the option classes or groups associated with the model. See "Delete the Children of a Model" on page 21.
   - Arrange the order of the children in the list.
   - Assign a product to the model, or change the current product assignment.

     See "Associate a Product with a Model, Option Class, or Option Item" on page 21.

     Note: Click **Save All Changes** to save your changes before you leave the **General Info** tab.
   - Create one or more option classes. See "Create an Option Class" on page 25.
   - Attach an option class group. See "Attach a Model, Option Class Group, or Option Item Group to an Option Class" on page 35.
   - Modify display properties. See "Working with Display Properties" on page 99.
3. Click the **Properties** tab to define properties for or to attach properties to this model.

   See Chapter 18, "Properties in Visual Modeler," on page 53.

4. Click the **Rules** tab to define rules for or attach rules to this model.

   See "Define a Rule" on page 71 or "Attach a Rule" on page 72.

5. Click the **Lists** tab to create lists for this model.

See “Define a List” on page 69.

6. Click the **Tables** tab to create or modify constraint tables.

   See “Create a Constraint Table” on page 85 or “Modify a Constraint Table” on page 85.

Delete a Model

About this task

You delete a model by finding the model group that is its parent, then deleting the model from that group. You cannot delete a model if it is attached as a sub-model elsewhere in the model group hierarchy.

See “Delete the Children of a Model Group” on page 18 for the procedure.

Note: When you delete a model, you must also delete the corresponding compiled model XML files from the location where the models are stored. This location is configured using the Applications Manager.

Delete the Children of a Model

About this task

Use this procedure to delete one or more option classes or groups that are children of a model:

Procedure

1. Navigate to and display the Model Detail page for the model with the elements you want to delete.

   See Chapter 7, “Access the Visual Modeler,” on page 15 for information on how to display the model.

   The **General Info** tab contains a list box showing the option classes (OC), option class groups (OCG) or option item groups (OIG) that are children to the model.

2. In the list box, select one or more objects to be deleted.
3. Click **Delete**.

   Note: Attached sub-models and groups are not deleted by this action. Only the attachment to those models and groups is removed. See “Delete a Group” on page 38.

4. Click **Save All Changes**.

   The model hierarchy no longer displays the deleted children.

Associate a Product with a Model, Option Class, or Option Item

About this task

You can reference a model, option class, or option item to a product ID in the product catalog. If the product ID has been assigned to one or more price lists, then this enables you to associate a price with the entity. In addition, if the item associated with a product is selected as part of a configuration, then when the user adds the configured product to their cart, the item is displayed with associated product ID and product information.

To associate a product with a Model, Option Class, or Option Item:

Procedure

1. In the model group hierarchy, find the entity that you want to associate with a product ID.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the **General Info** tab for the model, option class, or option item, click **Browse** to search for the product ID in the product catalog.
   - The product ID must exist in the product catalog. You create the product using the IBM Sterling Business Center application.
   - See Chapter 16, "Search the Product Catalog for a Product ID," on page 49 for help in browsing for a product ID. When you select the product ID, the product ID is displayed in the Assigned Product ID field and its product name and description are auto-filled into those fields.
   - You can manually enter the product ID in the Assigned Product ID field, but the Product Name and Product Description fields are not auto-filled until you save the information.
   - You can use the product name as the name of the new model. If the Name field is blank, then the field will be auto-filled with the product name. If the field has an entry already, then you will be prompted to use the product name.
   - If you are modifying a model, then you can click **Product Detail** to view the details of the assigned product.
3. Click **Save All Changes**.

Copy a Model

About this task

You can copy a model and its components into a model group.

To copy a model:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the model you want to copy. The model name is displayed in the Models and Groups frame.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the model you want to copy.

   This displays the current structure of the model.
3. Click **Copy** in the taskbar.

   This displays the Copy window.
4. Enter the Destination Model Group.
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the hierarchy until you find the destination model group.
   c. Select the destination model group.
   d. Click **Done**.

The model group appears in the Destination Model Group field.

5. As desired, modify the Destination Name field.

   The name defaults to the name of the model being copied.

6. Click **Copy** in the Copy window.

   The model is copied to the destination model group.

Copy a Model Reference

About this task

You can re-use a model as part of another entity without having to recreate the model. You do this by attaching the model to the entity. The attachment then becomes a model reference. You can copy this model reference; that is, instead of copying the actual model, you can copy the reference to a model that is attached.

To copy a model reference:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the entity with the model reference you want to copy. The entity name is displayed in the Models and Groups frame.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. In the Models and Groups frame, click on the entity that contains the reference you want to copy.

   This displays the current structure of the entity.

3. Click **Edit** in the taskbar.

   This displays the General Info tab.

4. In the Navigation frame, find and select the model reference you want to copy.
5. Click **Copy** in the taskbar.

   This displays the Copy window for copying model references.

6. Enter the Destination Option Class.
   a. Click **Browse**.

      This displays a Hierarchy Browser.

   b. Browse the model group hierarchy until you find the destination option class.
   c. Select the destination option class.
   d. Click **Done**.

      The option class appears in the Destination Option Class field.

7. As desired, modify the Destination Name field.

   The name defaults to the name of the model reference being copied.

8. Click **Copy** in the Copy window.

   The model reference is copied to the destination option class.

Embed a Model

About this task

You can embed a model within an option class.

To embed a model:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the model structure you want to embed. The model name is displayed in the Models and Groups frame.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the model whose structure you want to embed.

   This displays the current structure of the model.
3. Click **Edit** in the taskbar.

   This displays the General Info tab.
4. Click **Copy** in the taskbar.

   This displays the Copy window for embedding models.
5. Enter the Destination Option Class by typing or by browsing.

   To browse for the option class:

   a. Click **Browse**.

      This displays a Hierarchy Browser.

   b. Browse the model group hierarchy until you find the destination option class.

   c. Select the destination option class.

   d. Click **Done**.

      The option class appears in the Destination Option Class field.
6. As desired, modify the Destination Name field.

   The name defaults to the name of the model being embedded.
7. Click **Copy** in the Copy window.

Chapter 9. Working with Option Classes and Option Items

Option classes and option items comprise configurable parts or services of a model. You can think of option classes as representing questions or components that need to be configured, while option items represent answers or choices of components. Sometimes the answer to a question can give rise to further questions. In these cases it is useful to nest option classes within other option classes to help guide a user to the configuration that best meets their needs.

Create an Option Class

About this task

To create an option class:

Procedure

1. In the model group hierarchy, navigate to and display the model, option class group, or option class in which you want to create the option class.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

   You can create an option class within another option class, within a model, or within an option class group.
2. To create option classes as children of the model or option class group:
   a. Click **New Option Class**.

      This displays the New Option Class tab.
   b. Proceed to Step 4.
3. To create nested option classes:
   a. In the Navigation frame, navigate to and select the option class where you want to nest the new class.
   b. Click **New Option Class**.

      This displays the New Option Class tab.
   c. Proceed to Step 4.
4. Enter a name and description for the new option class.

   If you plan to associate the option class with a product ID, then you might consider skipping this step. If the name and description match the name and description of the product ID, then you can auto-fill these fields when you assign the product ID in Step 6.
5. Define the effectivity dates by modifying the start and end dates.

   You can click the calendar icon to select the dates from a calendar.
6. If applicable, assign a product ID.

   See "Associate a Product with a Model, Option Class, or Option Item" on page 21.
7. Click **Save** to save the new option class and remain at the **New Option Class** tab (to create additional option classes); click **Save and Edit** to save the new option class and display the option class tabs for editing.

   The new option class appears in the Navigation frame. The new option class is selected, ready to be modified.

Modify an Option Class

About this task

To modify an option class:

Procedure

1. In the model group hierarchy, navigate to and display the model, option class group, or option class that contains the option class.

   See Chapter 7, “Access the Visual Modeler,” on page 15 for information on how to navigate the model group hierarchy.
2. In the Navigation frame, find and click on the option class that you want to modify.

   This displays the General Info tab for the option class.
3. Modify Name, Description, and Start and End Dates as applicable.
4. Enter a ratio in the **Ratio** field, if applicable.

   The ratio field determines the quantity of option items that are added to a customer's order. The quantity of any child item selected is multiplied by this ratio to compute the “extended” quantity of the child item. For example, a bicycle model may have a wheel option class defined with a ration of “2”. When a user selects a particular wheel item from this option class, then two wheels will be added to the configured product.

   You can enter the **Ratio** as either a whole number or a decimal.
5. As applicable, modify the order of the children or delete the children.

   See “Delete the Children of an Option Class” on page 29.
6. If applicable, assign a product ID or modify the current assignment.

   See “Associate a Product with a Model, Option Class, or Option Item” on page 21.
7. Before you click the other tabs, click **Save All Changes**.
8. Click the **Display** tab to modify the display properties for this option class.

   See “Working with Display Properties” on page 99.
9. Click the **Properties** tab to attach properties to this option class.

   See “Attach a Property” on page 54.
10. Click the **Rules** tab to attach rules to this option class.

    See “Attach a Rule” on page 72.

    When you have completed modifying the option class, click **Save All Changes**.

    You can also create option items for this option class. See “Add Option Items to an Option Class.”

Add Option Items to an Option Class

About this task

To add option items to an option class:

Procedure

1. In the model group hierarchy, navigate to the option class to which you want to add the option items.

See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. In the **General Info** tab, click **New Option Item** to display the **New Option Item** tab.
3. Enter a name and description for the new option item.

   If you plan to associate the option item with a product ID, then you might consider skipping this step. If the name and description match the name and description of the product ID, then you can auto-fill these fields when you assign the product ID in Step 5.
4. Define the effectivity dates by modifying the start and end dates.
5. If applicable, assign a product Id.

   See "Associate a Product with a Model, Option Class, or Option Item" on page 21.
6. Click **Save** or **Save and Edit**.

   The new option item appears in the model hierarchy in the Navigation frame.

Copy an Option Class

About this task

You can copy an option class and its components into a model, an option class group, or another option class.

To copy an option class:

Procedure

1. Navigate to and select the parent model group for the model or option class group that contains the option class.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the model or option class group that contains the option class.

   This displays the current structure of the model or option class group.
3. Click **Edit** in the taskbar.

   This displays the General Info tab for the model or option class group.
4. In the Navigation frame, find and click on the option class that you want to copy.

   This displays the General Info tab for the option class.
5. Click **Copy** in the taskbar.

   This displays the Copy window for option classes.
6. Enter the destination model, option class group, or option class as follows:
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the destination model, option class group, or option class and select it.
   c. Click **Done**.

      The model, option class group, or option class appears in the Destination Model/OCG/Option Class field.
7. Enter the Destination name.

The name defaults to the name of the option class being copied.

8. Click **Copy** in the Copy window.

   The option class is copied to the destination model, option class group, or option class.

Modify an Option Item

About this task

To modify an option item:

Procedure

1. Find the option item that you want to modify.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

   When you click the option item, the **General Info** tab is displayed.
2. If applicable, modify the name, description, start/end dates.
3. If applicable, assign a product Id.

   See "Associate a Product with a Model, Option Class, or Option Item" on page 21

   Note: Before you click the other tabs, click **Save All Changes**.
4. Click the **Display** tab to modify the display properties for this option item.

   See "Working with Display Properties" on page 99.
5. Click the **Properties** tab to attach properties to this option item.

   See "Attach a Property" on page 54.
6. Click the **Rules** tab to attach rules to this option item.

   See "Attach a Rule" on page 72.

Copy an Option Item

About this task

You can copy an option item into an option item group or an option class.

To copy an option item:

Procedure

1. Find the option item that you want to copy.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

   When you click the option item in the Navigation frame, the **General Info** tab is displayed.
2. Click **Copy** in the taskbar.

   This displays the Copy window for option items.
3. Enter the destination option item group or option class.
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the option item group or option class and select it.

c. Click **Done**.

   The option item group or option class appears in the Destination: Option Item Group/Option Class field.

4. Enter the Destination name.

   The name defaults to the name of the option item being copied.

5. Click **Copy** in the Copy window.

   The option item is copied to the destination option item group or option class.

Deleting an Option Class

You can delete an option class by deleting the option class as a child of the parent to which it belongs. An option class can exist as a child of one of the following:

- A model. See "Delete the Children of a Model" on page 21.
- An option class. See "Delete the Children of an Option Class."
- An option class group. See "Delete the Children of a Group" on page 38.

Deleting the option class automatically deletes any option items, nested option classes, or attachments to groups.

Note: Nested groups are not deleted when you delete an option class, but only the attachment to those groups is deleted.

Delete the Children of an Option Class

About this task

You can delete option items and nested option classes, as well as any attachments to groups.

To delete the children of an option class:

Procedure

1. Navigate to and display the detail page for the model or option class group that contains the option class.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. In the Navigation frame, navigate to and select the option class.

   This displays the General Info tab which contains a list box showing the children of the option class.

3. Click the item to be deleted: option item (OI), option class (OC), model, option class group (OCG), or option item group (OIG).

   Note: Nested groups are not deleted. However, the attachment to those groups is removed.

4. Click the **Delete** button.

5. Click **Save All Changes**.

   The items are no longer displayed in the Navigation frame.

Chapter 10. Working with Groups

Create a Group

About this task

To create a group:

Procedure

1. In the Model Groups frame, navigate to and select the model group for which you are creating the option class group or option item group.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

   This displays the General Info tab for the group. Make sure you are creating the group within the appropriate model group. The group will be available for attachment to any items below this model group in the model group hierarchy.
2. Click **New Option Group**.

   This displays the New Option Class/Item Group tab.
3. Enter a name and description for the group.
4. Select the type of group (Option Class Group or Option Item Group).
5. Click **Save** or **Save and Edit**.

   The group appears in the hierarchy. You can now begin to build the group. The first step is to create one or more option classes. See "Create an Option Class" on page 25

Modify a Group

About this task

When you modify a group and then compile it, the modifications are reflected in any model to which the group is attached, once the model is recompiled.

To modify a group:

Procedure

1. In the model group hierarchy, navigate to and select the option class group or option item group that you want to modify.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

   This displays the General Info tab for the group.
2. Modify the name and description, reorder or delete the children.

   See "Delete the Children of a Group" on page 38 for information about deleting the children of a group.
3. (Option item groups only) If applicable, define start/end dates.
4. Click **Save All Changes**.

   You can also do the following:

   - Add option classes to an option class group. See "Create an Option Class" on page 25.

- Attach groups to the group. See "Attach a Group to a Model or Another Group" on page 34.

Copy an Option Class Group

About this task

You can copy an option class group to a model group.

To copy an option class group:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the group you want to copy. (See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.)
2. In the Models and Groups frame, click on the group you want to copy.
   The current structure of the group, if any, appears in the content frame.
3. Click **Copy** in the taskbar.
   This displays the Copy for option class groups window.
4. Enter the Destination Model Group.
   a. Click **Browse**.
      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the destination model group and select it.
   c. Click **Done**.
      The model group appears in the Destination Model Group field.
5. Enter the Destination name.
   The name defaults to the name of the option class group being copied.
6. Click **Copy** in the Copy window.
   The option class group is copied to the destination model group.

Embed an Option Class Group

About this task

You can embed an option class group within a model, another option class group, or an option class.

To embed an option class group:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the option class group you want to embed. The group name is displayed in the Models and Groups frame.
   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the group you want to embed.
   This displays the current structure of the group.
3. Click **Edit** in the taskbar.
   This displays the General Info tab.

4. Click **Copy** in the taskbar.

   This displays the Copy window for Embedding Option Class Groups.
5. Enter the destination model, option class group, or option class as follows:
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the destination model, option class group, or option class and select it.
   c. Click **Done**.

      The model, option class group, or option class appears in the Destination Model/Option Class Group/Option Class field.
6. Click **Copy** in the Copy window.

   The option class group is embedded in the destination model, option class group, or option class.

Copy an Option Item Group

About this task

You can copy an option item group to a model group.

To copy an option item group:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the option item group you want to copy.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the option item group you want to copy.

   The current structure of the group, if any, appears in the content frame.
3. Click **Copy** in the taskbar.

   This displays the copy window for option item groups.
4. Enter the Destination Model Group.
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the destination model group.
   c. Select the destination model group.
   d. Click **Done**.

      The model group appears in the Destination Model Group field.
5. Enter the Destination name.

   The name defaults to the name of the option item group being copied.
6. Click **Copy** in the Copy window.

   The option item group is copied to the destination model group.

Embed an Option Item Group

About this task

You can embed an option item group within another option item group or option class.

To embed an option item group:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the option item group you want to embed. The group name is displayed in the Models and Groups frame.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the group you want to copy.

   This displays the current structure of the group.
3. Click **Edit** in the taskbar.

   This displays the General Info tab.
4. Click **Copy** in the taskbar.

   This displays the Copy Window for embedding option item groups.
5. Enter the destination option item group or option class.
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the destination option item group or option class and select it.
   c. Click **Done**.

      The option item group or option class appears in the Destination OIG/Option Class field.
6. Click **Copy** in the Copy window.

   The option item group is embedded in the destination option item group or option class.

Attach a Group to a Model or Another Group

About this task

You can attach a model only to an option class (see "Attach a Model, Option Class Group, or Option Item Group to an Option Class" on page 35). You can attach an option class group to a model, an option class, or another option class group. You can attach an option item group to an option class or to another option item group.

To attach a group to a model or another group:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the model or group to which you want to attach the group.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the model or group to which you want to attach the option class group or option item group.

3. Click **Edit**.

   This displays the General Info tab for the model or group.
4. In the **General Info** tab, click **Attach**.

   This displays the Attach tab.
5. Enter a name and description for the attachment to the group or model.
6. Select the option class group or option item group to be attached.
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the option class group or option item group.
   c. Select the group.
   d. Click **Done**.

      The group appears in the selection field.
7. Click **Assign**.

   You can click **Return to General** to return to the **General Info** tab.

   The name you entered for the attached group or model appears in the model hierarchy in the Navigation frame.

Attach a Model, Option Class Group, or Option Item Group to an Option Class

About this task

To attach a model, an option class group, or an option item group to an option class:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the model with the option class.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the model or option class group that contains the option class.

   The current structure of the model or group, if any, appears in the content frame.
3. Click **Edit**.

   This displays the General Info tab for the model or group.
4. In the Navigation frame, navigate to and select the option class to which you want to attach the group.

   This displays the General Info tab for the option class.
5. In the **General Info** tab, click **Attach**.

   This displays the Attach tab.
6. Enter a name and description for the attached group or model.
7. Select the model, option class group, or option item group to be attached.
   a. Click **Browse**.

      This displays a Hierarchy Browser.
   b. Browse the model group hierarchy until you find the model, option class group, or option item group.

c. Select the model or group.
d. Click **Done**.

   The model or group appears in the selection field.

8. Click **Assign**.

   You can click **Return to General** to return to the **General Info** tab.

   The name you entered for the attached model or group appears in the model hierarchy in the Navigation frame.

View the Structure of an Attached Group

About this task

Once a group is attached, you can view the group's structure.

To view the structure of an attached group:

Procedure

1. Navigate to the level in the hierarchy (model, option class or option item) where the group is attached.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. Click **Show Detail**.

   This displays a read-only view of the group's structure.

Copy an Option Class Group Attachment

About this task

You can copy a reference to an option class group; that is, rather than copy the group itself, you copy the reference to the group. You can copy the reference into either a model, an option class group or into an option class.

To copy an option class group attachment:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the entity with the option class group attachment you want to copy. (See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.)
2. In the Models and Groups frame, click on the entity that contains the attachment you want to copy.

   The current structure of the model appears in the content frame.

3. Click **Edit** in the taskbar.

   This displays the model in the Navigation frame and the **General Info** tab for the group.

4. In the Navigation frame, navigate the model until you find the attached group you want to copy.
5. Click the attached group.
6. Click **Copy** in the taskbar.

   This displays the Copy window for option class group attachments.

7. Enter the destination model, option class group, or option class as follows:

a. Click **Browse**.

   This displays a Hierarchy Browser.

b. Browse the model group hierarchy until you find the destination model, option class group, or option class and select it.

c. Click **Done**.

   The model, option class group, or option class appears in the Destination Model/Option Class Group/Option Class field.

8. Enter the Destination name.

   The name defaults to the name of the option class group being copied.

9. Click **Copy** in the Copy window.

   The attachment is copied to the destination model, option class group, or option class.

Copy an Option Item Group Attachment

About this task

You can copy a reference to an option item group; that is, rather than copy the group itself, you copy the reference to the group. You can copy the reference into either an option item group or into an option class.

To copy an option item group attachment:

Procedure

1. In the Model Groups frame, navigate to and select the model group that contains the entity with the option item group attachment you want to copy. (See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.)
2. In the Models and Groups frame, click on the entity that contains the attachment you want to copy.

   The current structure of the entity appears in the content frame.

3. Click **Edit** in the taskbar.

   This displays the entity in the Navigation frame and the **General Info** tab for the group.

4. In the Navigation frame, navigate the entity until you find the attached group you want to copy.
5. Click the attached group.
6. Click **Copy** in the taskbar.

   This displays the Copy window.

7. Enter the destination option item group or option class.

   a. Click **Browse**.

      This displays a Hierarchy Browser.

   b. Browse the model group hierarchy until you find the option item group or option class and select it.

   c. Click **Done**.

      The option item group or option class appears in the Destination Option Item Group/Option Class field.

8. Click **Copy** in the Copy window.

   The attachment is copied to the destination option item group or option class.

Delete a Group

About this task

You can delete a group by finding the model group that is the parent of the group you want to delete, and then deleting the group from that model group. See "Delete the Children of a Model Group" on page 18 for the procedure.

To delete a group:

Procedure

1. Navigate to and select the model group that is the parent of the group you want to delete.
2. In the list box, select one or more model groups (MG), option class groups (OCG), or option item groups (OIG) to be deleted.
   - You cannot delete a model group if the group has children. You must delete the children first.
   - You cannot delete an option class group if it is attached to another model or option class group.
   - You cannot delete an option item group if it is attached to another model, option class group, or option item group.
3. Click **Delete**.
4. Click **Save All Changes**.

   The model group hierarchy will no longer display the deleted items.

Delete the Children of a Group

About this task

To delete one or more option classes or groups that are children of a group:

Procedure

1. Navigate to and select the parent model group that contains the group with the children you want to delete.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. In the Models and Groups frame, click on the group.

   This displays the current structure of the group.
3. Click **Edit**.

   This displays the General Info tab that contains a list box showing the children belonging to the group. This can include option classes (OC) and option class groups (OCG).
4. In the list box, select one or more elements to be deleted.
5. Click **Delete**

   Note: Groups are not deleted by this action. Only the attachment to those groups is removed. See "Delete a Group."
6. Click **Save All Changes**.

   The model hierarchy no longer displays the deleted elements.

Chapter 11. Search for Entities

About this task

You can search for entities that contain properties and property values that you specify as parameters. You can search across the entire hierarchy, or you can limit your search to model groups, models, option classes, option items, and rules, or you can limit your search even further to the currently selected model or group.

To search for entities:

Procedure

1. Access the Visual Modeler page.

   See Chapter 7, “Access the Visual Modeler,” on page 15 for information on how to navigate the model group hierarchy.
2. If you want to search within a specific model or group, then navigate to and select the model or group.
3. Click **Search** in the toolbar.

   This displays the Search window.
4. Select the scope for the search from the Search drop-down list.

   You can search all entities, you can limit the search to model groups, models, option classes, option items, or rules. If you are searching within a specific model or group, then you can limit your search to **Current Model** or **Current Group**.
5. Enter either a property name or a property value or both.

   Click **Browse** to display a browser window to select a property from a list of all the properties in the Visual Modeler.

   Use the drop-down list to select AND or OR. Select AND to produce search results that include both the Property Name and Property Value parameters you select. Select OR to produce search results that include either parameter.

   When you enter a property value, the search results will include property values that contain the property value you enter. For example, if you enter "75", then the search results will include any properties with the value "75" as well as property values such as "7550-1" or "MX-75-1".
6. Click **Search**.

   The search results will display below the parameters. By default, the result is sorted in ascending order by property name. You can click on one of the following columns to sort:

   - Property Name
   - Value
   - Location

     When you click the column title the first time, the column is sorted in ascending order.

Chapter 12. Include a Sub-Model in a Model

About this task

You can include one model in another model so that a sub-component of the parent model can be modeled and configured separately.

Suppose that you have a model A, and you want to use Model B as an option item in Model A, so that end-users can configure the Model B component as part of a session to configure model A.

To include a sub-model in a model:

Procedure

1. Create Model B as a model in its own right, and compile it. Make a note of the location of this model in the model group and model hierarchy. For example: Matrix/Computers/Workstations/Configurable Monitors/Matrix Monitor.
2. Navigate to Model A and to the location in the Model hierarchy at which you want to include Model B as an option item.
3. Create the option item and enter a name, description, and effectivity dates for it. Click **Save**.
4. Click the **Properties** tab.
5. Select CONFIG: SUBMODEL NAME in the Unattached Properties drop⌂down list.
6. In the Value field, enter the fully qualified name to Model B. For example, Matrix/Computers/Workstations/Configurable_0020Monitors/Matrix_0020Monitor. Note the use of escape characters to encode special characters such as spaces. See Chapter 13, “Special Characters Encoding,” on page 43 for more information.
7. Click **Attach**.
8. Click **Save All Changes**.
9. A separate property called CONFIG: SUBMODEL RETURN controls whether end-users return to the main model after configuring the child model.
   a. If you want to have end-users return to the main model when they have finished configuring Model B, then set the value of CONFIG: SUBMODEL RETURN to "true".
   b. If you want to have end-users return to directly to the calling application when they have finished configuring Model B, then set the value of CONFIG: SUBMODEL RETURN to "false".
10. Click **Attach**.
11. Click **Save All Changes**.
12. Click **Compile** to re-compile Model A.
13. To test the model, click **Test**.

Chapter 13. Special Characters Encoding

You must encode any special characters in model group and model names when you provide model group path names and model names.

The following table lists some common special character encodings:

Character	Encoding
" "(blank)	_0020
"-"	_002D
"/"	_002F
"!"	_0021
"@"	_0040
"#"	_0023
"$"	_0024

Chapter 14. Test a Model

About this task

You can test the model at any point while you are creating it.

The test model feature performs the following steps:

1. Compiles the model into an XML file.
2. Launches the browser.
3. Displays the model as a HTML page.

To test a model:

Procedure

1. Navigate to the model that you want to test.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click **Compile**. After you click Compile, a pop-up window, prompting whether to compile related models (sub-assemblies, if any) or not is displayed. If you want the system to compile related models, select the Compile Related models check box. If you want only the current model to be compiled, do not select the check box. Note that compiled sub-assemblies are required for configuring the model during runtime.

   A dialog box reports that compilation is successful.

   Note: If you are compiling a model that has subassemblies for the first time, then you must compile the subassemblies first. The main model does not function without the subassemblies being compiled first.
3. Click **Test Model**.

   This displays a configuration window as the end-user will see it, based on the current model.

   Note: If you click **Compile and Test**, then both actions are taken.

Chapter 15. Compiling a Model

Before a model can be associated with a configurable product and a customer can use the model you have created to configure a product, you must compile the model into XML format and store the model in a location accessible by IBM Sterling Configurator. Only compiled models can be associated with configurable products. After you create the model, you click a button to compile the model into an XML file. USD is not considered the default currency when you test a model. Currency and Organization Code are the mandatory parameters required for pricing items. The value of the Currency is fetched based on the preferences set by the user in the Applications Manager. For more information about Organization Code and defining currency definitions, refer to the *Sterling Selling and Fulfillment Foundation: Application Platform Configuration Guide*. The Organization Code set in the Visual Modeler for the current storefront is used.

Note: To compile all the models for the locales configured in Visual Modeler, select the **Compile All Locales** check box.

Compile a Model

About this task

To compile a model:

Procedure

1. Navigate to the model that you want to compile.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click **Compile**. After you click Compile, a pop-up window, prompting whether to compile related models (sub-assemblies, if any) or not is displayed. If you want the system to compile related models, select the Compile Related models check box. If you want only the current model to be compiled, do not select the check box. Note that compiled sub-assemblies are required for configuring the model during runtime.

   Note: If you are compiling a model that has subassemblies for the first time, then you must compile the subassemblies first. The main model does not function without the subassemblies being compiled first.

Results

The model is compiled into an XML file. This XML file is placed in the following location in `debs_home/Sterling/WEB-INF/data/config/`. This directory contains several directories, one for each locale. The model resides within the directory representing your preferred locale in either the folder representing the root model group folder or in one of the folders representing model groups within the root model group. They are stored in the shared location of a clustered deployment of the Visual Modeler.

Note: If your implementation of the Visual Modeler makes use of a staging and a production system, then bear in mind that the XML files may have to be moved over to the production environment or the model directories must be shared

between the systems.
In addition, the product records in the Knowledgebase for configurable products may have to be updated to point to the location of the XML files.

If your model group and model hierarchy include special characters (that is, non-alphanumeric characters), then these are encoded in the directory and files names that correspond to them. See Chapter 13, “Special Characters Encoding,” on page 43 for more information.

Compile All Models

About this task

Rather than compile models one by one, you can also compile all the models in a model group at once.

To compile all the models in a model group:

Procedure

1. Navigate to the model group whose models you want to compile. This can be the top-level model group.
2. Click **Compile All**.
3. In the Compile All Models window, click **Compile All Models**.
4. The Compile All Models Status window is displayed.
5. When it reports that all the models have been compiled, then click **Close**.

Chapter 16. Search the Product Catalog for a Product ID

About this task

To search the product catalog, perform the following steps:

Procedure

1. When you assign a product ID, you can click **Browse** to display the Hierarchical Entity Chooser.
2. You can use this window to navigate through the hierarchy until you find the product ID that you want to assign to the model object. You can click the **Search** tab to search through products unassigned to any product category.
3. Click **Done** when you find the product ID that you want to assign. The product ID appears in the Assigned Product ID field.

Chapter 17. Working with a Tabbed User Interface

Create a Tabbed User Interface

About this task

You can design your end-user interface so that, rather than being displayed in a single frame, the option classes appear within a series of tabs. You do this by first selecting the Tabbed Configurator JSP template at the model level, which sets the display property UI:JSP Filename (see "Working with Display Properties" on page 99). You then design the end-user interface using **Tabs** tab.

To create a tabbed user interface:

Procedure

1. Navigate to the model for which you want to create the tabbed interface.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Display** tab.

   The **Display** tab displays.
3. Select Tabbed Configurator from the JSP Template drop-down list.

   This automatically sets the UI: JSP FILENAME property to Configurator_Tabbed.jsp.
4. Click the **Tabs** tab.

   This displays the Tabs tab.
5. Enter a name for the tab in the Tab Name field.
6. Click **Add**.

   The content frame displays an area for editing the new tab.
7. Select the option classes or option class groups for the tab.
   a. Select an option class or option class group from the drop-down list.
   b. Click **Add**.
8. Repeat the last step for each option class or option class group you want in the tab

   Note: If you are creating a tabbed UI, then not all option classes must be accounted for in the tabs. Any option classes not included in a tab will not be displayed to the end-user.
9. Click **Move Up** or **Move Down** to arrange the order of the entities. To remove an entity, click the entity, then click **Remove**.
10. Click **Save All Changes**.

Modify a Tab

About this task

To modify a tab:

Procedure

1. Navigate to the model with the tabbed interface.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Tabs** tab.

   This displays the Tabs tab.
3. Find the tab element you want to modify.
4. To rearrange the order of the entities within the tab:
   a. Find and select the entity you want to move.
   b. Click **Move Up** or **Move Down**.
5. To remove an entity:
   a. Find and select the entity you want to remove.
   b. Click **Remove**.
6. To rearrange the location of the tab within the list of tabs, click the up or down arrows in the far right.
7. Click **Save All Changes**.

Delete a Tab

About this task

To delete a tab:

Procedure

1. Navigate to the model with the tabbed interface.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Tabs** tab.

   This displays the Tabs tab.
3. Find the tab element you want to delete.
4. On the far right, click the **Delete** icon (X) for that tab.
5. Click **Save All Changes**.

Chapter 18. Properties in Visual Modeler

A property is an attribute of a model, option class, or option item. It is used as a basic building block for rule creation.

The Visual Modeler provides a set of built-in properties which are understood by the Sterling Configurator engine. These control the behavior of the engine and the presentation of the model to the end-user. These properties are summarized in Chapter 21, “Visual Modeler Properties,” on page 65. You can also define properties and they are available for use in any part of the model group and model hierarchy beneath the point at which they are defined. These defined properties are used to describe the product so that the Sterling Configurator engine can ensure that the user-configured model is valid.

You can also use properties as variables and write rules that reason on the properties' values using functions such as value and expand.

Define a Property

About this task

To define a property:

Procedure

1. Navigate to and select the location in the model group hierarchy where you want to create the property.

   See Chapter 7, “Access the Visual Modeler,” on page 15 for information on how to navigate the model group hierarchy.

   This is important since where you create the property determines what objects in the hierarchy can use the property.
2. When you reach the appropriate level, click the **Properties** tab.
3. If you are working in a model, then within the **Properties** tab, click the **Define** tab.
4. Enter a name for the property

   Note: Do not begin a property name with "UI:" or "CONFIG". Do not include a period (.) in a property name.
5. Select a property type from the drop-down list.
   - **Number**: use this for any property whose value is determined by a number. For example, the weight of an item could be expressed as a real number of grams (including decimals).
   - **String**: use this for any property that is expressed as a word or phrase. For example, you can use a string-valued property to indicate the color of an option item.

     If you select this type, then the Localize field is enabled. If you check this box, then you can enter values for this property in any of the supported locales. In other words, if you enter the original value in English, then you can change the system locale to German and then modify the property's

value in German. The German value will appear for those users whose locale is German; the English value will appear for those users whose locale is English.

- **List**: use this for any property where the value of the property must be selected from a list. For example, the availability of an item might be limited to specifying one or more days of the week. You can capture this in the form of a property by defining a list called "Weekdays" whose values are Sunday, Monday, and so on, concluding with Saturday.

6. If applicable, define a default value that this property takes. You can override this value when you apply the property to an item or class.

   If you selected "List" as the property type, then the Value field displays a drop-down selection of the current lists available. Select a list. See “Define a List” on page 69 for information about creating lists.
7. Click **Add**.

   The new property appears in the boxes below the fields.
8. Click **Save All Changes** to save the new property.

Attach a Property

About this task

You define a property at the model group or model level (see “Define a Property” on page 53). You attach a property to a model, an option class, or an option item.

To attach a property:

Procedure

1. Click **Configuration Models in the Product Configuration Administration panel** on the Visual Modeler home page.
2. In the Navigation frame, navigate to the object to which you want to attach the property.
3. Click the **Properties** tab.

   This displays two sets of fields: one called Unattached Properties for selecting properties and defining values for them, and one called Attached Properties that shows the properties that are currently attached.

   Note: The **Properties** tab for a model contains two tabs: **Attach** and **Define**. You use the **Define** tab to define properties. See “Define a Property” on page 53.
4. Select a property from the Unattached Properties drop-down list.

   The property will display any default value defined for it.
5. Enter a value for the property. You can set the value of a property simply by entering its value in the text field, or you can use a property editor window to set a value. See “Edit Property Values using Property Editor Window” on page 55.
6. Click **Attach**.

   The newly-attached property appears among the Attached Properties.
7. Click **Save All Changes**

   Note: You must perform this last step. Otherwise the property will not be attached.

Edit Property Values using Property Editor Window

About this task

The Numeric Property Editor window and the String Property Editor window are used to edit property values.

To edit property values using Property Editor window:

Procedure

1. You can invoke the property window editor simply by clicking the **Edit** button next to any property.

   When you do so, a Property Editor window is displayed
2. You can use this window to specify a fixed value of a property or to specify a formula that is used to calculate a value at runtime. If the first character of the text area is "=", then the editor window assumes that you want to create a formula, and the expansion fields are activated to help you define the formula.
3. The syntax of a formula depends on whether you are editing a numeric or a string property:

   If you are working on a numeric property, then when you specify a formula, use the drop-down lists as follows:

   a. Function: select one of the defined functions.

   b. Property: specify the property whose values should be used to calculate the function.

   c. Location: specify where the named property (or properties) should be located. You can select an option item or select one of the following values for the location:

      - unspecified: select this to use the named property anywhere it is defined in the model. First, the current position is checked to see if the property is defined at that location, if not, then the standard algorithm is followed to see if the property is defined anywhere else in the model.
      - relative: select this to use the named property at the current location.

      If you are working on a string property, then when you specify a formula, use the drop-down lists as follows:

   d. Choose among gather, match, and expand:

      - gather: use this in assigning actions to a string property. It finds all occurrences of the specified property in the property pool and creates a string with the semicolon separating the values of these occurrences.
      - match: use this in writing rule fragments. It provides a mechanism to compare a string to the value of a property.
      - expand: use the expand function as described in “Working with Display Properties” on page 99.

Modify or Remove an Attached Property

About this task

You can only modify the value of an attached property at the local level to which it is attached. To modify the name or default value, see “Modify or Delete a Property Definition” on page 56.

To modify or remove an attached property:

Procedure

1. Click **Configuration Models in the Product Configuration Administration panel** on the Visual Modeler home page.
2. In the Model Groups frame, navigate to the element to which the property is attached.

   If the property is attached to a model:

   a. In the Model Groups frame, click the model group that contains the model.
   b. In the Models and Groups frame, click the model to which the property is attached.
   c. Click **Edit** in the toolbar.

      If the property is attached to an option class or option item:

   d. In the Model Groups frame, click the model group that contains either the model or group with the option class or option item.
   e. In the Models and Groups frame, click the model or group.
   f. Click **Edit** in the toolbar.
   g. In the Navigation frame, find and click the option class or option item.
3. Click the **Properties** tab.

   This tab displays two sets of fields: one called Unattached Properties for selecting properties and defining values for them, and one called Attached Properties that shows the properties that are currently attached.

   Note: If the property is attached to a model, you will see two tabs within the Properties tab: Attach and Define. The Attach tab is automatically displayed.
4. Find the property you want to modify or remove.
5. Modify or remove the attached property:
   - If necessary, change the value of a property.

     Note: This only changes the value locally, at the level it is attached. To change the default value of the property, see "Modify or Delete a Property Definition."
   - To remove an attached property, click **Remove**.
6. Click **Save All Changes**.

Modify or Delete a Property Definition

About this task

To modify or delete a property definition:

Procedure

1. Click **Configuration Models in the Product Configuration Administration** panel on the Visual Modeler home page.
2. Navigate to and select the location in the model group hierarchy where the property is defined.
   - At the root model group level:

     The Visual Modeler page automatically displays the root model group when you access Visual Modeler. If the root model group is not selected, then click on the root model group.
   - At the model group level, navigate to and click the model group in the Model Groups frame.

- At the model level, navigate to and click the model group that contains the model. Then, in the Models and Groups frame, click the model. Now click **Edit Model** in the toolbar.

  In all of these cases, this displays the General Info tab for the group or model.

3. Click the **Properties** tab.

   At the model group level, this displays the properties defined at that level.

   At the model level, this displays two tabs: **Attach** and **Define**. If the property is attached anywhere in the model group hierarchy, you will not be able to modify the property type. If the property is attached anywhere in the model group hierarchy, you will not be able to delete the property definition.
4. If you want to modify an unattached property, click the **Define** tab. Within the **Define** tab, find the property you want to modify or delete.
5. If you want to modify an attached property, click the **Attach** tab. Within the **Attach** tab, find the property you want to modify or remove.
6. Modify or delete the property definition (property type or value).
7. Click **Save All Changes**.

   Name changes and value changes will be propagated to anywhere the property is attached. The value change is the default value for the property. It will not override any values set for the attached property.

Chapter 19. Using Worksheets

Worksheets provide quick access to a group of properties, enabling you to easily maintain all of a model's properties in one place. A worksheet is a table that assigns property values to option items:

- Rows represent option items
- Columns represent properties

Each worksheet belongs to a model and can be used to set the values of properties of that model. You can still set the values for properties as described in "Attach a Property" on page 54.

For example, suppose that a model of a computer has an option class for hard drives. Each hard drive option item has a number of properties such as capacity, RPM, latency, and buffer cache. You can create a worksheet to maintain the hard drive properties, similar to the following table:

Option Item	Capacity	RPM	Latency	Buffer Cache
WD Protege	160	5400	5.00	2
WD Caviar	250	7200	4.20	2
WD Caviar SE	250	7200	4.20	8
WD Essential	250	7200	4.20	2

Create a Worksheet

About this task

To create a worksheet:

Procedure

1. Navigate to the model for which you want to create a worksheet.
2. Click the **Worksheets** tab.
3. Click **New**.
4. In the New Worksheet window, enter a name for the worksheet, and click **Create**.
5. Add the option items whose properties you want to set using this worksheet. You do this by clicking **Add Row**, and then navigating to each option item in turn using the entity picker window.
6. Add the properties to the worksheet by clicking **Add Column** and in the Add Column dialog box, select each property from the drop-down list of properties defined for this model. You can create a new property by clicking **New Property** in the Add Column window, and then entering the new property details in the Define New Property window.
7. After you have added the rows and columns for your worksheet, you can enter values for each option item and property.
8. Click **Save All Changes**.

Modify a Worksheet

About this task

You can modify a worksheet at any time. Changes to property values are effective immediately, and will be compiled with the other model details when you next compile the model.

To modify a worksheet:

Procedure

1. Navigate to the model to which the worksheet belongs.
2. Click the **Worksheets** tab.
3. Select the name of the worksheet form the drop-down list.
4. Click **Select**.
5. In the worksheet, you can do the following:
   - Change the name of the worksheet: click the worksheet name and enter a new name for the worksheet.
   - Add a new row: click **Add Row** and select option items as required.
   - Move a row: click the link to the row, and select its new position from the drop-down list of rows.
   - Remove a row: click the link to the row, and click **Delete**.
   - Add a new column: click **Add Column**, and select the property from the drop-down list.
   - Move a column: click the column name, and select its new position form the drop-down list of columns.
   - Delete a column: click the column name and click **Delete**.

Export a Worksheet

About this task

There are times when it is more convenient to manage the values of properties when you have the worksheet in the form of a spreadsheet that you maintain on your local machine. You can export a worksheet in the form of a comma-separated values (CSV) file, and then open this file in your preferred spreadsheet program to manage the values. You can then import the modified spreadsheet to update the values in the worksheet. See “Import a Worksheet” on page 61 for details on importing a worksheet.

To export a worksheet:

Procedure

1. Navigate to the model to which the worksheet belongs.
2. Click the **Worksheets** tab.
3. Select the name of the worksheet from the drop-down list.
4. Click **Select**.
5. Click **Export**.
6. In the File Download window, click **Save**.
7. In the Save As window, navigate to the directory on your local machine to which you want to save the file, and then click **Save**.

The file is saved to your local machine.

Import a Worksheet

About this task

When you have finished editing a spreadsheet for a worksheet, save it as a comma-separated values (CSV) file. Follow these steps to import the worksheet into the Visual Modeler.

To import a worksheet:

Procedure

1. Navigate to the model to which the worksheet belongs.
2. Click the **Worksheets** tab.
3. Click **Import**.
4. In the Worksheet Import window, click the **Browse** button.
5. In the Choose File window, navigate to and select the spreadsheet that you want to import.
6. Click **Open**.
7. In the Worksheet Import window, click **Import Now**.

   The spreadsheet is imported into the Visual Modeler.

Chapter 20. Properties as Variables

You can evaluate the value of a property in defining rules and properties using this syntax: ${*function(...)*}. This enables you to define a property as a function of another property. This can be useful in defining display properties and in defining mathematical formulae for rules. For example, you can use ${expand(property[,default[,format]])} to display properties of models.

For example, suppose that you have a numerical property called "Monitor Size"defined on a series of monitors that expresses the screen size in inches and suppose that you want to present this information in a table in the form "17.00 inches". You can define a property called Display Monitor Size by "${expand("Monitor Size","n/a",0.00)} inches". Now use this new property in the display of the model and users will see the size expressed as "17.00 inches" if the underlying Monitor Size property has the value "17". Note that if the Monitor Size property is not defined, then "n/a inches" is displayed.

Chapter 21. Visual Modeler Properties

The following table summarizes the properties that are built in to the Visual Modeler. Note that UI properties are covered in “Visual Modeler UI Properties” on page 99.

Property	Type	Comments
CONFIG: FIRST FIRE	numeric	1 if this is the first time firing rules, 0 otherwise.
CONFIG: POOL SIZE	numeric	Number of copies of a model to keep in the model pool.
CONFIG: REPEAT FIRING	string	yes" or "true" turns on looping in the rule engine, causing rules to fire as long as the current state keeps changing. Since rules are removed from the rule list whenever they fire, this is not an infinite loop.
CONFIG: SUBMODEL NAME	string	The encoded name of another model. Encoding replaces potentially unsafe file system characters with _XXXX where XXXX is the hex representation of their Unicode character code. For example, a space is represented by "_0020". See Chapter 13, “Special Characters Encoding,” on page 43 for more information.
CONFIG: SUBMODEL RETURN	string	yes" or "true" implies that when we punch into a submodel specified by the previous property we will be returning with that models BOM as a child of this model.
_cacheKey	string	Used on a model node to contain the key used to store the model in the model cache.
_description	string	The description of an item.
_errorCount	numeric	Number of errors encountered during rule firing.
_fileSize	string	String representation of a Long value, size of the XML file for a model.
_lastModified	string	Last modified date for a model as a string (number of seconds since some important date).
_modelTabs	list	List of tab names for the model.
_name	string	The name of an option item, option class, or model.
_parent.<item names>	varies	Properties inherited by a submodel from the parent.
_pickItems	list	Internally used to keep track of picked items.
_pickmap.<itemKey>	string	Mapping of an item to an option class.
_picks	list	Internally used to keep track of picked items.
_quantity	integer	Quantity selected, if >0 the item is picked.
_sequence	numeric	Rule firing sequence, if 0 this is the first time through the loop, 1 is the second, and so on.
_tabMembers<#>	list	Where <#> is a tab number (0...N), these properties contains the names of the root level option classes that are part of the tab whose index is <#>.

Chapter 22. Lists in Visual Modeler

In many cases, the values a property may take can be expressed as a number or as a string of characters. In some cases however, a property has to take one of a certain number of pre-specified values such as the days of the week, or one of a set of manufacturer-specified formats such as SM, M, L, or XL.

In these situations, the best approach to take is to define a property, of List type. Then you can write rules to test whether the value of the first property is in the list that is the value of the List property.

Thus, if you have a property called ShirtSize and you want to restrict the choices that a user can select to SM, M, L, or XL, then the steps are:

1. Create a list called ShirtSizeList. Enter values for the list: in this case SM, M, L, and X.
2. Create a property called AvailableShirtSizes whose type is List and assign it the value ShirtSizeList.
3. Create the ShirtSize property and assign it to option items as appropriate.
4. Create a rule that specifies that the value of the ShirtSize property must be in the list of the AvailableShirtSizes property.

Chapter 23. Lists

Define a List

About this task

To define a list:

Procedure

1. Navigate to the model group or model for which you want to define the list.
   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Lists** tab.
   This displays any lists already defined.
3. Click **New**.
   This displays the New List tab.
4. Enter a name and description for the list.
5. Define the values for the list.
   a. Enter a value in the New Value field.
   b. Click **Add Item**.
6. Repeat the last step for each value you want to add.
7. Click **Save** to save the values and remain at the **New List** tab.
   When you click **Save and Return**, you save the values and return to the **Lists** tab. The new list appears among the defined lists.

Modify a List

About this task

To modify a list:

Procedure

1. Navigate to the model which contains the list you want to modify.
   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Lists** tab.
   This displays any lists already defined.
3. Click the name of the list you want to modify.
   This displays the Edit List tab.
4. Modify the name or description.
5. Delete values from the list.
   a. Select one or more values in the list.
   b. Click **Delete**.
6. Add values to the list.
   a. Type a value in the New Value field.
   b. Click **Add Item**.

7. Modify values in the list.

   There is no way to modify a value in a single step. You must delete the old value and add the new one.
8. Click **Save** to save the values and remain at the **Edit List** tab.

   When you click **Save and Return**, you save your changes and return to the **Lists** tab.

Delete a List

About this task

To delete a list:

Procedure

1. Navigate to the model which contains the list you want to delete.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Lists** tab.

   This displays any lists already defined.
3. Among the defined lists, find the list you want to delete.
4. Click **Delete** on the same line as the list you want to delete.

   The list disappears from among the defined lists.

   Note: This last step is important! If you click **Delete**, but do not click **Save All Changes**, then the list will not be deleted.
5. Click **Save All Changes**.

Chapter 24. Rules

Define a Rule

About this task

To define a rule:

Procedure

1. Navigate to the detail page for the model group or model where you want to create the rule.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Rules** tab.
   a. If you are defining the rule at the model level, then the **Rules** tab displays two tabs: **Attach** and **Define**. Click the **Define** tab.
   b. The model group level contains a single tab for defining the rule.
3. Click **New**.

   This displays the New Rule tab.
4. Select a classification for the rule and specify a priority.

   You can create your own rule classifications: see "Create a Rule Classification" on page 72. Rule priorities are used to determine the order in which rules are fired: lowest numbers fire first. You should use values between 0 and 100: 50 is the default value.
5. Enter a name and description for the rule. Also, select whether the rule is triggered when the rule's conditions are met (success) or not met (failure).
6. Define the fragments of the rule.

   See Chapter 25, "Working With Rule Fragments," on page 77.
7. Define the rule actions.

   You can define messages to be displayed, a rule expansion formula, or you can assign properties and values. See Chapter 26, "Working with Rule Actions," on page 81.

   Note: No syntax checking is performed on rules. The configurator engine will fail to load a model if there is a syntax error in any of the assigned rules.
8. Click **Save**.

Modify a Rule

About this task

To modify a rule:

Procedure

1. Navigate to the model group or model where the rule was created.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Rules** tab.

The **Rules** tab for the Model level displays two tabs: **Attach** and **Define**. To modify the rule, click the **Define** tab. The model group level contains a single tab for defining the rule.

The **Rules** tab displays a table with the currently defined rules.

3. Find the rule you want to modify, then click the **Edit** icon.

   The **Edit Rule** tab displays.
4. Modify the Name and Description as necessary.
5. Add or modify Comments as necessary.
6. Modify whether the rule is triggered when the rule's conditions are met (success) or not met (failure, as necessary).
7. Modify the rule fragments in the Fragments table.

   See Chapter 25, "Working With Rule Fragments," on page 77.
8. Add or modify actions in the Actions area.

   You can define messages to be displayed, a rule expansion formula, or assign properties and values. See Chapter 26, "Working with Rule Actions," on page 81.

   Repeat these steps for each rule you want to modify. You can click **Where Used** at the bottom of the tab to view the entities to which the rule is attached. See "View Rule Attachments" on page 73.

Create a Rule Classification

About this task

To create a rule classification:

Procedure

1. Navigate to the rule creation page: see "Define a Rule" on page 71.
2. Click **...** next to the Classification drop-down list.
3. In the Rule Classification Window, enter a name for the classification, and click **Add Item**.
4. Click **Save and Return**.

Attach a Rule

About this task

To attach a rule:

Procedure

1. Navigate to the level in the model hierarchy (model, option class or option item) where you want to attach the rule.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Rules** tab.

   At the model level, the **Rules** tab contains two tabs: **Attach** and **Define**. At the option class and option item levels, the **Rules** tab looks like the **Attach** tab.

   The **Attach** tab displays a drop-down list of the unattached rules, as well as a table showing the rules that are currently attached.
3. Select a rule from the drop-down list in the Unattached Rules table.

4. Click **Attach**.

   The rule is appended to the end of the current rules in the Attached Rules table.
5. Define the start and end dates for the rule.
6. If you want this rule to be a checkpoint, check the box the Stop Firing column.

   When checked, this rule acts as a checkpoint: if any errors have occurred up to this point in the rule firing, then processing will stop at this point and the errors will be displayed. If no errors have occurred, then rule firing will continue until all the rules are fired or the next checkpoint is hit.
7. Determine the sequence.

   The rules will fire within the element to which they are attached in the order they appear in the list. You can modify the order using the up or down arrows to the right of the rule.
8. Click **Save All Changes**.

View the Details of an Attached Rule

About this task

Once you have attached a rule, you can view the details of the attached rule by clicking the rule's name in the **Attach** tab.

To view the details of an attached rule:

Procedure

1. Navigate to the level in the hierarchy (model, option class, or option item) where the rule is attached.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Rules** tab.

   At the model level, the **Rules** tab contains two tabs: **Attach** and **Define**. At the option class and option item levels, the **Rules** tab looks like the **Attach** tab.

   The **Attach** tab displays a drop-down list of the unattached rules, as well as a table showing the rules that are currently attached.
3. Find the rule among the list of attached rules in the lower part of the frame.
4. Click the name of the rule.

   This displays the Rule Detail Viewer.

View Rule Attachments

About this task

You can use this procedure to see where a rule is attached.

To view rule attachments:

Procedure

1. Navigate to the model group or model where the rule was created.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Rules** tab.

The **Rules** tab for the Model level displays two tabs: **Attach** and **Define**. To modify the rule, click the **Define** tab. The model group level contains a single tab for defining the rule.

The **Rules** tab displays a table with the currently defined rules.

3. Find the rule you want to modify, then click the **Edit** icon.

   The **Edit Rule** tab displays.

4. Click **Where Used**.

   The Rule Usage window displays.

Unattach a Rule

About this task

To unattach a rule:

Procedure

1. Navigate to the level in the model hierarchy (model, option class or option item) where the rule is attached.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. Click the **Rules** tab.

   At the model level, the **Rules** tab contains two tabs: **Attach** and **Define**. At the option class and option item levels, the **Rules** tab only contains attachments.

   The Rules tab displays a drop-down list of the unattached rules, as well as a table showing the rules that are currently attached.

3. Find the rule in the Attached Rules table.
4. Click the **Delete** symbol (**X**) at the end of the rule's row in the table.

   The rule returns to the Unattached Rules table.

5. Click **Save All Changes**.

Delete a Rule

About this task

To delete a rule:

You can delete rules when they are no longer required.

Note: You cannot delete a rule if the rule is currently attached to any node in the model hierarchy.

Procedure

1. Navigate to the model group or model where you created the rule.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. Click the **Rules** tab.

   At the model level, the **Rules** tab contains two tabs: **Attach** and **Define**. At the model group level, you can only define. If the rule you want to delete was created at the model level, then click the **Define** tab.

3. Find the rule you want to delete.
4. Click the **Delete** icon next to the rule you want to delete.

5. Click **Save All Changes**.

Move or Copy a Rule

About this task

It is sometimes necessary to re-organize your model hierarchy and in doing so, you may need to move the rule definitions too. You can move or copy a rule: *moving* means that the rule definition is deleted from its previous location whereas *copying* means that you create a copy of the rule without deleting its original definition.

To move or copy a rule:

Procedure

1. Navigate to the rule definition that you wish to move or copy.
2. Click **Copy Rule**. The Rule Copy or Move window displays.
3. Click **Browse** to open the entity picker window.
4. Navigate to the model group or group to which you want to move or copy the rule definition, select it and click **Done**.
5. If you want to, you can then change the name of the rule definition.
6. Click **Move** or **Copy** as appropriate.

   An error message is displayed if a rule with the same name already exists in the target location. If a property referenced in the rule does not exist in the new location, then it is created at the same time as the rule is.
7. Click **Close**.

Specify the Rule Firing Sequence

About this task

Each time a model is validated, the rules are fired to determine whether each rule succeeds or fails. You can control the order in which rules fire by setting a priority for each rule.

To specify the rule firing sequence:

Procedure

1. Navigate to the model.
2. Click the Rules tab.
3. Click the Firing Sequence sub-tab. The Firing Sequence tab displays.
4. Enter a priority for each rule: this should be an integer between 0 and 100.

   The higher the value the lower the priority: that is rules with lower priority value will fire before rules with a higher priority value. The default value is 50.

Review Rule Firing

About this task

When you are testing a model, you can review the model firing behavior.

To review the rule firing behavior:

Procedure

1. Navigate to the model whose rule firing you want to review.
2. Click **Test**.
3. In the Product Configurator window, click **Show Trace Log**.
4. The secondary window displays a trace of the results of the rule firing. You can review this to determine if rules are firing as you expect.

Controlling Rule Firing

When Sterling Configurator validates a model and the current set of picks, it tests each rule in turn to evaluate them for success or failure, and performs expansion and assignment actions as appropriate. There are different ways in which rule firing can behave:

- Each rule is tested only once when a model is validated. This is referred to as single-pass rule firing.
- You can configure a model so that if any rules are fired, the fired rules are removed from the rules list and the remaining rules are tested again. This process continues until no more rules are fired. This is referred to as multiple-pass rule firing.

  The property CONFIG: REPEAT FIRING controls this behavior. By default, only single-pass firing is performed.

Force Multiple-Pass Rule Testing

About this task

You can specify whether rules are fired once or multiple times.

To control rule firing:

Procedure

1. Navigate to the model whose rule firing you want to control.
2. Click **Properties**.
3. Select CONFIG: REPEAT FIRING from the **Unattached Properties** drop⌂down list.
4. Set its value to “true” and click **Attach**.
5. Click **Save All Changes**.

   You can verify that the rules fire only once by following the steps described in “Review Rule Firing” on page 75. In the summary section of the trace log, you should see that there was one firing phase .

Chapter 25. Working With Rule Fragments

As you create rules, you must create rule fragments that perform the rule logic. This topic describes the procedures to define or modify the fragments of a rule when you are creating (“Define a Rule” on page 71) or modifying a rule (“Modify a Rule” on page 71).

Click the **Display Rule Fragment** icon, to toggle the visibility of sections of the Fragments area and enable working with them. For example, click the arrow to display the foreach section.

The following table describes the buttons that display in the Fragments area of the New Rule tab.

Button	Name
	Display Rule Fragment
	New Operator button
	New Fragment button
	Delete button
	Edit button

Click the **New Operator** icon to create a nested level for creating fragments. The new level displays a new set of **New Fragment** and **New Operator** links. You use this **New Fragment** link to create the fragments at this nested level. If you click **New Operator** at this level, then you will create another nested level below this one with another set of **New Fragment** and **New Operator** links for that level.

Click the **New Fragment** icon to create a fragment at the currently displayed level (in this case, the top level) in the rule structure. Click **New Fragment** again to create a second fragment at the currently displayed level. In other words, the rule would be:

FragmentA AND FragmentB

Click the **Delete** button to delete the fragment.

Click the **Edit** button to modify the fragment.

Foreach

A rule condition can have a property called the foreach property. Use the foreach property to loop through the property pool, identify all instances of a specified property, and act on a set of found values. You associate the foreach property with a property defined in the model. Each occurrence of the foreach property is bound to an "as" property.

For example, to increase the price of any node found selected in the model that has the SKU MXDS-7500 and for which the rackMountable property is true, you could use foreach in the following way:

```
foreach sku as tempSku
      IF value(tempSku) == literal(MXDS-7500)
      AND propval(itemType) == literal("rackMountable")
      THEN UI: PRICE = value(UI: PRICE) * 1.1
```

Example: To Create a Simple Level of Fragments

About this task

In this example, you are attempting to create a rule consisting of two fragments, joined by a single operator, with no nested levels: `FragmentA AND FragmentB`. When you access the tab, the **New Rule** tab appears.

Procedure

1. Select the boolean operator you want for these fragments.
2. Click the **New Fragment** icon

   This displays the New Fragment tab.
3. Define the fragment.
   a. Check the **Not** check box if you want to define the fragment as a negative: "NOT (sum(PropertyA <= 250))".
   b. Select the first function from the Function1 drop-down list.
   c. Select a property from the Property1 drop-down list.
   d. Select the operator.
   e. Select the second function from the Function2 drop-down list.
   f. In the Property2 field, select a property from the drop-down list or enter a literal value in the field (if you selected "literal" as the function).
   g. Select the value for the If Not Specified drop-down list.
4. Click **Save and Return**.

   This re⌂displays the New Rule tab with the new fragment.
5. Click the **New Fragment** icon to create the next fragment in the rule.
6. Repeat Step 3 to define the second fragment.
7. Click **Save and Return**.

   This re⌂displays the New Rule tab with the fragment you created.

   You can click **Save** to save the rule and continue defining the rule. Click **Save and Return** to return to the list of rules in the **Define** tab.

Example: To Create Nested Fragments

About this task

In this example, the modeler is creating the following rule with nested fragments:

```
(FragmentA AND FragmentB) OR (FragmentC AND FragmentD)
```

Procedure

1. Click **New Operator** icon.

   A nested level appears in the **New Rule** tab. This level has its own drop-down boolean operators, as well as its own **New Fragment**, **New Operator**, and **Delete Operator** icons.

2. Create two fragments as described in “Example: To Create a Simple Level of Fragments” on page 78.

   Use the nested drop-down list to select the boolean operator for these fragments. The default is AND.

   Use the nested **New Fragment** icon to create the fragments at this nested level.

   When the two fragments are completed, the **New Rule** tab appears.

   You can nest as many fragments as you want by clicking the nested **New Operator** icon. Each time, a new nested operator will appear with a new set of nested icons. You use these nested icons to create the fragments for the nested level.
3. Using the top-level list, select the boolean operator (in this example, OR) that will join the two nested level.
4. Click the **New Operator** icon at the top level.

   A new nested level appears in the fragments tab. This level has its own drop-down boolean operators, as well as its own **New Fragment**, **New Operator**, and **Delete Operator** icons.
5. Create two fragments as described in “Example: To Create a Simple Level of Fragments” on page 78.

   Use the nested drop-down list of boolean operators and the nested **New Fragment** icon for these fragments.

   When the two fragments are completed, the **New Rule** tab appears.

Modify a Fragment

About this task

To modify a fragment:

Procedure

1. Click the fragment you want to modify. This displays the Edit Fragment tab .
2. Modify one or more elements of the fragment.
   a. Check the **Not** check box if you want to define the fragment as a negative: “NOT (sum(PropertyA <= 250))”.
   b. Select the first function from the Function1 drop-down list.
   c. Select a property from the Property1 drop-down list.
   d. Select the operator.
   e. Select the second function from the Function2 drop-down list.
   f. In the Property2 field, select a property from the drop-down list or enter a literal value in the field (if you selected “literal” as the function).
   g. Select the value for the If Not Specified drop-down list.
3. Click **Save And Return**.

Delete a Fragment

About this task

Find the fragment you want to delete in the Fragments table, then click the **Delete** icon in the Actions column on the same line as the fragment.

Chapter 26. Working with Rule Actions

You can include a rule action when you are creating (“Define a Rule” on page 71) or modifying a rule (“Modify a Rule” on page 71). You define rule actions in the lower part of the **New Rule** or **Edit Rule** tab. Rule actions comprise three types of actions:

- **Message Actions**: A message that is displayed when the rule is triggered.
- **Formula and Expansion Actions**: Defines an expansion action based on a rule expansion formula.
- **Assignment Actions**: Assigns the value calculated by the rule formula to one or more properties when the rule is triggered.

Create a Message Action

About this task

When you are creating (“Define a Rule” on page 71) or modifying a rule (“Modify a Rule” on page 71), you perform this task in the Message Actions area of the **New Rule** or **Edit Rule** tab.

To create a message action:

Procedure

1. Select the type of message action from the drop-down list: Error, Warning, Suggestion.
2. Type the message.
3. Click **Add Item**.
4. Repeat these steps to enter additional messages.
5. Click **Save** to save the message action and continue editing. Click **Save and Return** to save the message and return to the **Define** tab.

Create an Expansion Action

About this task

When you are creating (“Define a Rule” on page 71) or modifying a rule (“Modify a Rule” on page 71), you perform this task in the Expansion Actions area of the **New Rule** or **Edit Rule** tab.

To create an expansion action:

Procedure

1. Enter a formula.

   The results of formula will be used to perform the expansion.
2. Enter a minimum and a maximum amount of the formula result.

   The minimum amount is the minimum value the rule formula result must be greater than. This value can be negative or greater than or equal to zero. The value must be less than the maximum value (Max). The maximum amount is the maximum value the rule formula result must match. This value must be greater than the minimum value (Min)

Note: Min and Max work slightly differently: for a fragment to evaluate to true, the rule formula must evaluate to greater than the Min value, but less than or equal to the Max value.

3. Enter the quantity of the expansion items (must be greater than zero). You can use the supported functions to calculate the quantity and so you can specify the quantity as a function of a property. For example:

   2*value(Memory Cards)

4. Enter the item that will be expanded.

   You must provide the full path to the expansion item within the current model. For example, the rule adds an option item called either 64MB, 128MB, or 256MB, located in option class AutoMemory in the current model.

   When a rule is used in multiple models, this fully qualified path could be difficult to specify since the current model name will very likely not be "MXWS-7650" for all the models where the rule is attached. To facilitate the use of expansion rules across multiple models, you can usual special symbols as follows:

   - You can begin the path with a period (.), which means “from the attachment point of the rule”. In other words, if you attach a rule to a model, then ".Memory.64MB" means "an option item called 64MB in an option class called Memory in the current model".
   - You can begin the path with an asterisk (*), which means from the root of the model group hierarchy.
   - If the name of a path component includes a quote character (' or "), then you must escape the quote character or wrap the whole expression in quotes. For example, to get the gauge property from the Tubing.3" pipe.threading option item, you can use

     ```
     x = value(Tubing.3\"pipe.threading.gauge)
     or
           x = value('Tubing.3"pipe.threading.gauge')
     To retrieve Board.8'plank.thickness, use
           x = value(Board.8\'plank.thickness)
     or
           x = value("Board.8'plank.thickness")
     ```

5. Repeat these steps to enter additional items.
6. Click **Save All Changes**.

   The result of an expansion action picks a quantity selected on an option item. If the option item quantity is a drop-down list, ensure that the possible calculated values are consistent with the pickable values: otherwise, the drop-down list will not be able to display the calculated value.

Create an Assignment Action

About this task

When you are creating (“Define a Rule” on page 71) or modifying a rule (“Modify a Rule” on page 71), you perform this task in the Assignment Actions area at the bottom of the **New Rule** or **Edit Rule** tab.

To create an assignment action:

Procedure

1. Select a property from the drop-down list. The table below summarizes some of the special properties that can be assigned.

2. Enter a value for the property. You can use the supported functions to calculate the value and so you can specify the value as a function of a property. For example:

   ```
   2*value(Memory Cards)
   ```

   When you are assigning a value to a property whose type is String, you must use the following syntax to refer to properties:

   ```
   ${function(arg1, arg2, ..., arg N}
   ```

   For example, ${expand("Color", "Black", 0)}. See "Example Uses of Expand" for other examples of the usage of the expand function.
3. Type the entity to which you want to assign the property and its value.

   If you leave this field blank, the assignment defaults to the entity to which the rule is attached.
4. Click **Add Item**.
5. Repeat these steps to add additional items.
6. Click **Save All Changes**.

   The following table summarizes some of the available properties for assignment. These properties may change in each release, so check with your representative for further information if required.

Property	Action
_constraintMessage	String: a message on an item because it is constrained
_constraintType	Integer: type of constraint; 0 is suggest, 1 is warn, and 2 is error
_description	String: an items description
_amEntitled	Integer: 0 false, 1 true
_isConstrained	Integer: 0 false, 1 true
_isSelected	Integer: 0 false, 1 true
_isViewable	Integer: 0 false, 1 true
_itemKey	Integer: database key of the item
_pickOverride	Integer: 0 false, 1 true; pick was overridden by a rule
_quantity	Integer: quantity; 0 quantities are not in the rule pool
_ratio	Numeric: ratio of this item to its children, computed if nested within another parent
_rawRatio	Numeric: raw ratio used in previous computation
_rulePick	Integer: 0 false, 1 true
_tabLevel	Integer: depth of this item

Example Uses of Expand

The syntax of the expand function is:
${expand(property[,defaultValue[,format[,lookup]]])}.

For example, suppose that you want to display the name of the model as the name of the associated product together with the product description. At the model level,

set the value of the UI: DISPLAY NAME property to: ${expand("UI: PRODUCT NAME")} or ${expand("UI: PRODUCT DESCRIPTION","Description not available")}.

Doing this ensures that if the product name or description changes and you recompile the model, the name or description displays with the new version when users next configure the product.

Here are some further examples of the expand function:

- String-valued property:
  - ${expand("color")}
  - ${expand("color", "Black")}
- Numeric-valued property:
  - ${expand("weight")}
  - ${expand("weight", 0.0)}
  - ${expand("weight", 0.0, #.00)}

Chapter 27. Working with Constraints

Create a Constraint Table

About this task

Constraint tables enable you to limit a customer's choice of one or more option items based on the customer's choice of another option item. For example, the choice of an exterior color for a car might limit the choice of interior colors. You create an option constraint by creating a constraint table. You define constraint tables at the model level.

To create a constraint table:

Procedure

1. Navigate to the model where you want to create the constraint table.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Tables** tab.

   This displays two tabs: **General Info** and **Records**. The **General Info** tab displays general information about the table displayed in the Table Name field.
3. Click **New**.

   This displays the Create New Constraint Table tab.
4. Enter a Table Name, a Description, and a date range (Start Date/End Date) for the table. (You can click the **Calendar** icon to select dates from the calendar.)
5. Enter a message.

   This message appears when the end-user chooses a selection which is incompatible with a constraint defined in the table.

   a. Select the message type: error, warning, or suggestion.

   b. Enter the message in the Message field.
6. Click **Save Changes**.

   This re-displays the Tables tab with the new table in the Table Name field. The next step is to create the option constraints that are a part of the table. You do this in the **Records** tab. See "Define Option Constraints" on page 86.

Modify a Constraint Table

About this task

To modify a constraint table:

Procedure

1. Navigate to the model that contains the table you want to modify.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Tables** tab.
3. Select the table from the Table Names drop-down list.
4. Modify the table. You can:

- Define option constraints (see “Define Option Constraints”).
- Modify option constraints (see “Modify an Option Constraint” on page 87).
- Delete option constraints (see “Delete Option Constraints” on page 87).
- Modify the name, description, or effectivity dates in the **General Info** tab.
- Modify the error/warning/suggestion message in the **General Info** tab.

Define Option Constraints

About this task

After you create a table and the option classes that will provide the constraints, you define the constraints. Each row in the table represents a constraint.

To define an option constraint:

Procedure

1. Navigate to the model that contains the table for which you want to define the constraint.

   See Chapter 7, “Access the Visual Modeler,” on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Tables** tab.

   This displays two tabs: **General Info** and **Records**. The **General Info** tab displays general information about the table displayed in the Table Name field. The **Records** tab is where you will define the constraints.
3. Select the table from the Table Names drop-down list.
4. Click the **Records** tab.

   This displays the currently defined option constraints.
5. Add columns to the constraint table.
   a. Select an option class from the Table Column name drop-down list.

      The drop-down list includes all the option classes belonging to the model including any option classes nested within option classes as well as option classes that are part of option class groups attached to the model. The drop-down list will display the path to the option class relative to the model.

      For example, there are two slections in the drop-down list called **Monitor** and **Software**. Notice that the Navigation frame shows two option classes by these names directly below the model.

      The drop-down list has another selection, **Software.Application**. Notice that the model has an option class called **Software** directly below the model, with a nested option class called **Application**. Notice how the drop-down list indicates the path relative to the model, **Software.Application**.

      The drop-down list also includes a selection, **MX-7500 Service.Warranty**. This corresponds to the option class group, **MX-7500 Service**, directly below the model. **Warranty** is an option class within the group.
   b. Click **Add**.

      The column name is added to the table.
   c. Repeat the last two steps for every column you want to add.
6. Define an option constraint.
   a. Click **New Constraint** to add a new row to the table.
   b. Click **Edit**.

This displays the option classes as table columns, along with their option items.

The option items that display include any option items belonging to an any option item group attached to the option class.

c. Define compatibility ("Selected Values are all"). That is, will the selections you make in one column be valid or invalid with the selections in the other column(s)?

d. Select one or more option items in each column.

e. Click **Save**.

A new row appears in the table.

7. Repeat the last step for each constraint you want to define.

Modify an Option Constraint

About this task

To modify an option constraint:

Procedure

1. Navigate to the model that contains the table with the constraint you want to modify.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. Select the table from the Table Names drop-down list.
3. Click the **Records** tab.

   This displays the currently defined option constraints.

4. Find the constraint row you want to modify and click **Edit**.

   This displays the constraint information.

5. Modify the constraint information.

   a. Modify compatibility.

      Will the selections you make in one column be valid/invalid with the selections in the other column(s)?

   b. Modify the option items in each column.

   c. Click **Save**.

      The row is changed based on your modifications.

Delete Option Constraints

About this task

To delete an option constraint:

Procedure

1. Navigate to the model that contains the table with the constraint you want to delete.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.

2. Click the **Tables** tab.
3. Select the table from the Table Names drop-down list.
4. Click the **Records** tab.

This displays the currently defined option constraints.

5. Find the constraint row you want to delete.
6. Click **Delete** (**X**).

   The constraint row is deleted.

Delete a Constraint Table

About this task

To delete a constraint table:

Procedure

1. Navigate to the model that contains the table with the constraint you want to delete.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Tables** tab.
3. Select the table from the Table Names drop-down list.
4. Click the **Delete** button.

   The constraint table is deleted.

Chapter 28. Import a Model Group or Model

About this task

You can import model groups and models in the form of XML files. You can either import the entity relative to its original root model group, or you can designate a location into which to import. The model will appear in the Navigation frames, enabling you as modeler to add to or modify the imported model.

To import a model group or model:

Procedure

1. Access the Visual Model page.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. If you want to import to a selected point, then navigate to the model group within which you want to import the file.
3. Click **Import** in the toolbar.

   This displays the Visual Modeler Import window.
4. Click **Browse** to find the XML file you want to import.

   When you select the file, the file will be displayed in the field along with the complete path to the file.
5. Select the import option.
   - **Import with its original structure relative to its root model group**

     When you make this selection, the Import process will ignore any Destination Model Group indicated at the top of the window.
   - **Import into the selected destination model group**
6. Click **Import Now**.

   The imported model group or model and its structure will be imported based on the import option you selected.

Chapter 29. Export a Model Group or Model

About this task

To export a model group or model:

Procedure

1. Navigate to the model group or the model that you want to export.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click **Export** in the toolbar.

   You can either open the XML file at its current location, choosing a desired text processing tool, or you can save the file to a desired location.

Chapter 30. Dynamic Instantiation

Dynamic instantiation provides a way to allow users to configure products on the fly while avoiding the need to create option items for each possible product configuration in your model. For example, consider a server rack. The user can decide on the number of slots they need and create dynamic instantiation controls for each type of component, such as servers and storage arrays that can fit into a slot, AC or DC power, and so on. As the modeler, you create the rack model, then create option classes for each of the rack's configurable features (such as servers and storage arrays) and set them as dynamic instantiation control classes. An end-user buying computer racks navigates to the rack product on your site and clicks the **Configure** button next to the servers and storage array choices. This causes a new option item to be added to the model for that configurable feature. The user can then configure each option item by clicking the **Configure** button that appears next to each added item. When the entire rack and all the configurable features have been added and configured, the user clicks the **Add** button located in the button bar at the top of the Configurator page to add the rack to their cart.

Chapter 31. Test Dynamic Instantiation

About this task

To test how the dynamic instantiation process works, perform the following steps:

Procedure

1. On the Model Group Navigation page, click **New Model**.

   The New Model page displays.
2. Enter a name for the model, then click **Save and Edit**.

   The Model Navigation page for the new model displays.
3. Click **New Option Class**.

   The New Option Class page displays.
4. Enter a name for the new option class, then click **Save and Edit**.
5. Click the **Display** tab, then choose Dynamic Instantiation from the **UI Control** drop-down list.

   Set other Display properties as appropriate, then click **Save All Changes**.
6. Click the **Properties** tab, then set the following properties from the **Unattached Properties** drop-down list:
   a. CONFIG: SUBMODEL NAME

      Enter the name of an existing submodel for the property value, then click **Attach**.
   b. CONFIG: SUBMODEL RETURN

      Enter the name of an existing submodel to which the end-user should return after clicking the **Add** button, then click **Attach**.
7. Click **Save All Changes**.
8. Return to the new model's root node, then click the Compile and Test icon to test your dynamic instantiation option class.

Chapter 32. Run A Report

About this task

You can run a report on a model that you specify. You can select the types of information you want in the report:

- Rule definitions
- List definitions
- Property definitions
- Display Settings
- Attached Properties
- Attached Rules
- Expand Groups

To run a report:

Procedure

1. Access the Visual Modeler page.
   See Chapter 7, "Access the Visual Modeler," on page 15.
2. Click **Report** in the toolbar.
   This displays the Report Entry window.
3. Enter the model you want to report on.
   You can click **Browse** to find and select the model in the model hierarchy.
4. Select the locale in which you want to run the report.
5. Select a date to report.
   This produces a report for the models for whom the selected date falls within the range of their effectivity dates. The report does not display any models (or entities within the model) for whom the selected date falls outside their effectivity dates.
6. Select the information you want to include in the report.
7. Click **Run Report**.
   A report is displayed based on the parameters you entered.

Chapter 33. UI Properties

A property is an attribute of a model, option class, or option item. UI properties are used to determine the look-and-feel of a product as it is configured. You can use UI properties to control how option classes are displayed, how to display properties of option items, as well as basic guiding text and pictures.

The Visual Modeler provides a set of built-in UI properties which are understood by the Sterling Configurator engine. These control the behavior of the engine and the presentation of the model to the end-user. These properties are summarized in Chapter 34, “Define Display Property Values,” on page 107.

Working with Display Properties

The Visual Modeler provides certain *display properties* that come pre-defined with the Visual Modeler. These display properties enhance the customer experience by enabling you to provide values that define various aspects of the model or its elements. They can all be specified using the **Display** tab of a model, option class, or option item, or as UI properties in the **Properties** tab. For example, you can define a "Pre-Pick Guiding Text" for an option class either by defining it on the **Display** tab or by specifying the value of the UI: PRE-PICK GUIDING TEXT property in the **Properties** tab.

Display properties also allow you to create fields and options that end-users may use to enter their own values rather than values specified by you. See Chapter 37, “Define User-Entered Values,” on page 117. Note that every property displayed on the **Display** tab corresponds to a UI property. This means that display properties can also be set using the **Properties** tab provided that you know which UI property matches the display property. See Chapter 34, “Define Display Property Values,” on page 107 for more details.

Visual Modeler UI Properties

The following table summarizes the UI properties that are built in to the Visual Modeler.

Property	Type	Comments
UI: ADDITIONAL DESCRIPTION	string	You can use this property to add additional descriptive text to an option class. use this property in conjunction with the UI: DISPLAY RESULTS property.
UI: ALIGNMENT	string	Horizontal" or "Vertical" controls layout of radio buttons and check box controls.

Property	Type	Comments
UI: AUTOMATIC POST	string	yes" or "true" turns on automatic posting for an option class. After a customer makes a pick of an option item, then you usually want the server to re-display the page so that rules can be fired and any changes to the available option classes displayed. However, if you do not want picks in an option class to cause a re⌂display, then set this property to "no" or "false". This is equivalent to selecting On User Request from the Submit to Server Display property drop-down list. The option class is displayed with **Update** button: after making a pick in this option class, a user can click the **Update** button to request a re-display of the page from the server.
UI: CLASS DISPLAY NAME	string	Use this property at the model level to determine what is displayed as the displayed name of option classes. By default, this property takes the value ${expand("_description")} which means that the value of the option class's **Description** field is displayed. For example, if you want to display option class names instead of descriptions, then set this property to ${expand("_name")}. You can overwrite this value at a single class by using the UI: DISPLAY NAME property.
UI: COLUMN ALIGNMENT	string	Used in the tabular display control to specify the alignment of the values in the column. The tabular display control uses the ";" character to separate entries from each other, so the format of this column is something like: "left;left;center;right".
UI: COLUMN HEADINGS	string	Used in the tabular display control to specify the titles of columns. Each title is separated from each other with the ";" character. For example: "Speed;Pins;Manufacturer". See Chapter 35, “Display Properties in a Tabular Form,” on page 113 for an example of using this property.
UI: COLUMN PROPERTIES	string	A semi-colon-separated list of property names used in the tabular display of properties. For example: "SPEED;NOPINS;SUPPLIER", where SPEED, NOPINS, and SUPPLIER are properties defined on option items in an option class. See Chapter 35, “Display Properties in a Tabular Form,” on page 113 for an example of using this property.
UI: COLUMN SPAN	numeric	Controls how many columns an option class requires for its display in the customer-facing display of the model. This is the same as entering a number for the **Number of Columns** field on the **Display** tab.

Property	Type	Comments
UI: CONFIG CELL HTML CLASS	string	Sets the CSS class attribute in the HTML. Use this property to control the look-and-feel of cells. Note that the Visual Modeler uses the **internal.css** CSS file when you test models.
UI: CONSTANT GUIDING TEXT	string	Defines the guiding text that will always be shown for an option class. This is the same as entering text for the **Constant Guiding Text** field in the **Display** tab.
UI: CONTROL	string	The name of the JSP fragment used to render an option class. Do not use UI: JSP FILENAME at the option class level.
UI: DEFAULT SELECTION	string	true" or "yes" on an item makes the item a default selection within its parent option class.
UI: DISPLAY ADDITIONAL INFO	string	Use this property to provide a description specific to a particular instance of a sub-model. If you attach this property to the root node of a submodel and pass it as an output property to the parent model, the parent model displays the description next to the item in the parent model. This allows you to give feedback to the end-user about how the sub-model is configured. This is particularly useful for dynamic instantiation, where there can be multiple instances of a sub-model, each configured differently, and you want to provide an appropriate description for each instance of the submodel.
UI: DISPLAY NAME	string	Use this property to determine what is displayed as the displayed name of the option class. By default, this property takes the value ${expand("_description")} which means that the value of the option class's **Description** field is displayed.
UI: DISPLAY RESULTS	string	This property is deprecated. A property that is displayed along with the description of items. This special property also allows the usage of text expansion macros. Currently we support: ${expand(propname[,defaultValue[,pictureString]])} but the name of this "function", expand in this case, is accessed via the object manager. An example usage is to set a description string in the UI: ADDITIONAL DESCRIPTION property, and then set the value of this property to ${expand("UI: ADDITIONAL DESCRIPTION")}.
UI: HELP URL	string	A URL that is used to turn an option class description into a hyperlink, typically used to provide additional information about what that option class is for, but could also be a datasheet or any other hyperlink. Clicking on the hyperlink will bring up the page in a new window. This is the same as entering text for the **Help URL** field in the **Display** tab.

Property	Type	Comments
UI: ICON GRAPHIC	string	Used with an option class to display a picture along with the description of the option class. This is the same as entering text for the **Image** field in the **Display** tab: see "Associating Images to Option Items" on page 115 for information on how values in this field are resolved to URLs.
UI: IGNORE IN QUOTE	string	If this is set to "yes" or "true", whatever item this property is attached to will be filtered out of the summary page and not added to the shopping cart. This is the same as selecting the **Ignore in Quote** check box in the **Display** tab. Typically, this field is used to ensure that only selected option items are displayed in shopping carts and to suppress option classes in the list of items in a shopping cart.
UI: JSP FILENAME	string	The name of the JSP page that will render the model: **Configurator_Tabbed.jsp** or **configurator.jsp**. This property is added to support easier customization and eventually to allow different presentations per model. Using built-in customization elements of Sterling Configurator, it is possible to dynamically change pages as well.
UI: LEAD TIME	numeric	Attached to items in the model. It is used to build a maximum lead time for the entire model by finding the largest lead time of all items currently selected.
UI: NUMBER OF COLUMNS	numeric	Number of columns to divide the end-user configurator presentation. This property is defined at the model level to manage how many columns are used to display the option classes for a model. This property in conjunction with UI: COLUMN SPAN, UI: ROW SPAN, AND UI: SKIP COLUMNS controls how option classes are arranged on the page. This property is the same as setting the Number of Columns property in the **Display** tab.
UI: OPTION CLASS REQUIRED	string	yes" or "true" causes Sterling Configurator to require that a selection be made for an option class. For radio buttons this causes the None selection to be removed.
UI: OPTION CLASS SELECT	string	This property is used to specify what UI control should be used when no specific UI: CONTROL value is specified. Its use is primarily to support importing models from external configuration systems or from earlier releases of the Visual Modeler. It takes "single" or "multiple" as values, and is only used in the absence of a UI: CONTROL property to determine if a radio button or checkbox control should be shown for an option class.
UI: OPTION CLASS TYPE	string	Obsolete: do not use.

Property	Type	Comments
UI: OPTION CLASS VIEW	string	POPUP", "POPUP-QTY", or "INVISIBLE". This controls the display behavior of an option class. If POPUP, a standard option class is shown; if POPUP-QTY is selected, then a quantity box will be shown for each selected item within that control. Finally, INVISIBLE is used to prevent the display of the control entirely. INVISIBLE is often used to hide option classes until other picks made by the customer requires the class to be displayed.
UI: POPUP-QTY ALLOWED VALUES	string	This controls what values are available for a selection in a popup drop-down list. Use this at the option class level, in conjunction with setting UI: OPTION CLASS VIEW to POPUP⌂QTY. A "," separated list of allowed values. Ranges can be specified with “-”, so 1-4,7-9 is the same as 1,2,3,4,7,8,9. If you leave this field blank, then a text field is displayed with the current value; otherwise a drop-down list with the allowed values is displayed.
UI: POST PICK GUIDING TEXT	string	A guiding text message displayed with an option class description if the user has made at least one pick from within the option class. This is the same as entering text for the **Pre-Pick Guiding Text** field in the **Display** tab. This property is not displayed until a customer makes a pick.
UI: PRE PICK GUIDING TEXT	string	A guiding text message displayed with an option class description if the user has not made a pick from within the option class. This is the same as entering text for the **Post-Pick Guiding Text** field in the **Display** tab. Once a pick has been made, then this property is no longer displayed.
UI: PREVENT SELECTION	string	yes" or "true" causes the Sterling Configurator to prevent the user from selecting items that would violate a constraint table rule. If the Constraint Selections display property is set to "Hide constrained items", then this property is set to “yes”.
UI: PRICE	numeric	The price for an item that will be used only if STATIC_PRICING or OVERRIDE_PRICING is set in the business rules. In the case of OVERRIDE_PRICING, this value will be used if a price cannot be found for the item in the price list.
UI: PRICING SKU	string	The SKU to use when looking up the item in the price list. Note that if you set a product ID value for this property, then it overrides the value of the Assigned Product ID in determining prices.

Property	Type	Comments
UI: PRICING STYLE	string	Usually, you use this property at the option class level. It controls how prices of option items are displayed to the end user as follows: NONE: Do not display prices as user configures product. ABSOLUTE: Display prices next to option items as absolute prices. DELTA: Display prices next to option items as their effect on the price of the whole configured product. This property is the same as setting Pricing Style in the **Display** tab.
UI: PRODUCT ID	string	If a product has been associated with a node of a model, then this property can be used to retrieve the product ID of the associated product. The value of this property is resolved at compile time, so if the product ID is changed, then you must re-compile the model for the change to take effect.
UI: PRODUCT NAME	string	If a product has been associated with a node of a model, then this property can be used to retrieve the product name of the associated product. The value of this property is resolved at compile time, so if the product name is changed, then you must re-compile the model for the change to take effect.
UI: PRODUCT DESCRIPTION	string	If a product has been associated with a node of a model, then this property can be used to retrieve the description of the associated product. The value of this property is resolved at compile time, so if the product description is changed, then you must re-compile the model for the change to take effect.
UI: QUANTITY AVAILABLE	numeric	Do not use in this release. Used in the quantity matrix, this can optionally be attached to the items for the matrix. If so it will set the quantity available of each item. If the control is set to show quantity available this property value will be displayed in a secondary row for each item.
UI: REQUIRED	string	Obsolete: do not use.
UI: ROW SPAN	numeric	Controls how many rows an option class requires for its display in the end-user presentation of the page. In conjunction with UI: NUMBER OF COLUMNS and UI: COLUMN SPAN, this property controls the layout of the page viewed by end-users. This is the same as entering a number for the **Number of Rows** field in the **Display** tab.
UI: SHOW ITEM IMAGES	string	yes" or "true" controls whether item images are shown.

Property	Type	Comments
UI: SKIP COLUMNS	numeric	Number of columns to skip after this class. It is used to add to the count variable that is tracking how many cells are being used to lay out the option classes. This is the same as entering a number for the **Number of Columns to Skip** field on the **Display** tab. If you have used the UI: COLUMN SPAN property or UI: ROW SPAN property for another option class, then use this property to account for table cells in the layout that the multiple span class uses.
UI: SUPPRESS NAME DISPLAY	string	yes" or "true" causes Sterling Configurator to not display the names of option classes.
UI: SUPPRESS NONE SELECTION	string	yes" or "true" suppresses the NONE selection value for radio buttons.
UI: SUPPRESS UEV NONE VALUE	string	yes" or "true" suppresses the NONE selection for UEV combination boxes. Use this in conjunction with UI: UEV ALLOWED VALUES property. For example, if you have specified that a user-entered value field can only take the values Red, Green, Blue, then if the value of this property is set to "yes", then None will not appear in the drop⌂down list of selectable values. If you set the value of this property to "no", or do not attach this property, then None will be a selectable value.
UI: UEV ALLOWED VALUES	string	Comma-separated list of values for a combination box UEV control. Suppose that you want to allow customers to enter only one color from a small list of colors. Then enter the list like this: Black,Blue,Green,Red,White When this property is set, then the user-entered value option item is displayed as a drop-down list of these values. None is also displayed as a selectable option, unless you set the UI: SUPPRESS UEV NONE VALUE property to "yes". This property is the same as setting values in the Allowed Values display property.

Property	Type	Comments
UI: UEV ASSIGNMENT PROPERTY	string	The name of a property where a UEV will store its value. This property should be of the correct type to contain the UEV. Note: numeric properties can be used to hold INTEGER UEVs as well as NUMERIC UEVs. If the value of this property is just a property name, then the property will be set on the current item. If the value contains a path to a property as well as the property name, then the property will be set on the item referenced by the path if it exists. Once a user makes their pick in the user-entered value field, then the assigned property can be used by rules or in the display of the model, just like any other property. This property is the same as setting a value in the Assign Value to Property display property.
UI: UEV INTEGER VALUE	integer	Filled in by the engine when an integer UEV has a value in it. This provides you with a way to reference the value of the field without assigning it to another property.
UI: UEV LIST VALUE	list	Filled in by the engine when a list UEV has a value in it (not currently used). This provides you with a way to reference the value of the field without assigning it to another property.
UI: UEV NUMERIC VALUE	numeric	Filled in by the engine when a numeric UEV has a value in it. This provides you with a way to reference the value of the field without assigning it to another property.
UI: UEV POSTFIX	string	A string of text displayed after the UEV entry field. This property is the same as setting a value in the Text After Entry Field display property.
UI: UEV PREFIX	string	A string of text displayed before a UEV entry field. This property is the same as setting a value in the Text Before Entry Field display property.
UI: UEV SELECTION	varies	Obsolete: do not use.
UI: UEV SPECIAL	string	Used by the user entered value control to enable a file list or notes control. This will be phased out and replaced by a new file attachment control and notes control in future releases: do not use.
UI: UEV STRING VALUE	string	Filled in by the engine when a string UEV has a value in it. This provides you with a way to reference the value of the field without assigning it to another property.
UI: UEV TYPE	string	string", "integer", or "numeric"; the type of UEV control.
Note: To add additional macros, define a new class that implements the IExpansionHandler interface, and put a reference to it into the object manager.		

Chapter 34. Define Display Property Values

About this task

To define display property values:

Procedure

1. Navigate to and display the detail page for the model, option class, or option item.

   See Chapter 7, "Access the Visual Modeler," on page 15 for information on how to navigate the model group hierarchy.
2. Click the **Display** tab.

   This displays the display properties appropriate to the level.
3. Edit the desired fields.

   See the following table for an alphabetical list of the properties, where they can be assigned, and what they mean. Because each display property corresponds to a UI property, the table also provides the name of the corresponding UI property, and further information about each UI property is provided in "Visual Modeler UI Properties" on page 99.
4. Click **Save All Changes**.

Field Name/Property Name	Where Used	Description
Automatic Post/UI: AUTOMATIC POST	Model Option Class	Depending on the value you choose, this property specifies how posting is done: • **none**: No update is performed when the customer selects an option item. • **update**: An incremental update occurs when the customer selects an option item. • **final (default)**: A final update occurs when the customer selects an option item.
Constant Guiding Text/UI: CONSTANT GUIDING TEXT	Model Option Class	Used to add extra text to the displayed HTML page. This text is "constant", that is, it appears all the time, even after a selection is made. For example, guiding text for a configurable camcorder may state "Only lithium batteries type XYZ are compatible with this model."

Field Name/Property Name	Where Used	Description
Control/UI: CONTROL	Option Class	Enables you to determine how the option items are displayed: • **Radio button**: Items appear as radio buttons. Customer can only select one. • **Checkbox**: Option items will appear with check boxes; multiple selection allowed. • **Drop down list**: Items appear in a drop-down list. • **Combobox**: Items appear in drop-down list, but end-users can also type in a selection. • **Multiple Selection listbox**: Items appear in a scrollable list from which the customer can make multiple selections. • **Display All Children**: When you have nested option classes, nested classes appear with their option items visible (as opposed to option items only appearing when nested option classes are "picked"). • **User EnteredValue**: Items appear as user⌂entered fields. • **Tabular Display**: Items appear as rows in a table.
Default Selection/UI: DEFAULT SELECTION	Model Option Class Option Item	This property specifies that, if the user does not choose an entity in the option class, this entity (embedded model, nested option class, or option item) is automatically selected. You can use this property in conjunction with the **Option Class Required** special property. You can only assign it to one option item in an option class.
Display Template	Model	Select the type of user interface from the drop-down list: Tabbed UI or Non-tabbed UI. See “Create a Tabbed User Interface” on page 51.
Help URL/UI: HELP URL	Model Option Class	Enables you to display a link (URL) to a page that has additional information about the model, option class, or option item.
Icon Graphic/UI: ICON GRAPHIC	Model Option Class	Provides the location (fully qualified path) of a GIF format file to be displayed next to this model, option class, or option item.
Ignore In Quote/UI: IGNORE IN QUOTE	Model Option Class Option Item	This special property is attached to option classes and option items that will not be transferred into the summary page when these option items are selected by the customer or through an expansion rule.

Field Name/Property Name	Where Used	Description
Lead Time/ UI: LEAD TIME	Model Option Class Option Item	Enables you to specify a lead time between when a customer orders a product that includes this item and when that product can be expected to ship.
Option Class Required/UI: OPTION CLASS REQUIRED	Option Class	Enables you to specify whether or not a customer must make a selection in that option class to complete the configuration. Customer must select one of the option items to complete the configuration.
Option Class View/UI: OPTION CLASS VIEW	Option Class	Determines (1) if the items in this option class are displayed, and (2) if the pop-up quantity is displayed next to the option item. • **Popup**: When the customer clicks the drop⌂down arrow, the line is expanded to display all items. • **Popup-qty**: Customer types in a number in the quantity field. The entered value influences the quantity of option items that are ordered for this option class. • **Invisible**: Option class and its items are not displayed to the customer. This is typically used for an automatic expansion when the customer does not need to know about the added option items that are part of the configuration. For example, if a customer orders a special wheel, then invisible option items may include nuts and bolts that are included with the special wheel.
Popup-Qty Values/ UI: POPUP-QTY ALLOWED VALUES	Option Class	Enables you to set quantity for that item. The quantities specified appear as possible selections in a quantity box next to the item.
Post-Pick Guiding Text/UI: POST PICK GUIDING TEXT	Model Option Class	Used to add extra text displayed on the HTML page after the customer has made a selection. For example, a model of a computer has an option class called "Operating System", and that option class has an option item called "Windows 2000". Post⌂pick guiding text for that option item might be "Windows 2000 requires a minimum of 256MB of RAM; make sure the amount of RAM you select is at least 256MB".

Field Name/Property Name	Where Used	Description
Pre-Pick Guiding Text/UI: PRE PICK GUIDING TEXT	Model Option Class	This special property is assigned at the option class level, and is used to add extra text that is displayed on the HTML page. The text disappears once a selection is made. For example, pre-pick guiding text for a CPU option class of a configurable computer may state "Choose a Processor". Once a processor is chosen, the text disappears.
Prevent Selection of Items Resulting in Constraint Errors/UI: PREVENT SELECTION	Option Class	Enables you to prevent a customer from selecting items in this class that are incompatible with items in another class (based on an option constraint table). If the Constraint Selections display property is set to "Hide constrained items", then this property is set to "yes". If Option Class Required is selected, then you cannot check this box.
Pricing Style/UI: PRICING STYLE	Model Option Class	This special property enables you to specify how option class items will display price information. There are three possible values: • **none**: If you assign this property with the value *none*, then the option class' items are displayed without any pricing information. • **delta**: If you assign this property with the value **delta**, then the option items display pricing information in relation to the total base price of the configurable product. When end-users first see the option class, they see the option items with prices as "Add $xxx.xx", meaning "selecting this item adds this amount to the current configuration price of the model." Once the end-user selects an option item, the other option items will show either "Add $xxx.xx" or "Subtract $xxx.xx", depending on how choosing those option items will affect the price. • **absolute**: If you assign this property with the value *absolute*, then the option items display pricing information as the total cost of that item. This kind of pricing information is not relative to any base price. It is simply the cost of that item. • Only the absolute property set at Option Class level is considered while displaying the price information. •

Field Name/Property Name	Where Used	Description
Price/UI: PRICE	Model Option Class Option Item	This special property enables you to assign a specific price to the item. This property is used to attach a price to a model if your model, option class, or option item is not associated with a product ID (see “Associate a Product with a Model, Option Class, or Option Item” on page 21). **Note:** Prices assigned to option items in this way are not preserved when the configured product is returned to a cart.
Return From Submodel/CONFIG: SUBMODEL RETURN	Option Item	Setting this property to "no" allows end users to transition from one model to the next. If a user returns to a model, all selections and derived properties are reset.
User Entered Value Type/UI: UEV TYPE	Option Item	This property is displayed only if you selected "User entered value" for the Control display property of the option class to which the item belongs. This property enables you to define the type: **string**, **integer**, or **numeric**.
User Entered Value Prefix/UI: UEV PREFIX	Option Item	This property is displayed only if you selected "User entered value" for the Control display property of the option class to which the item belongs. This enables you to provide a text string that precedes a user-entered value (For example, "$").
User Entered Value Postfix/UI: UEV POSTFIX	Option Item	This property is displayed only if you selected "User entered value" for the Control display property of the option class to which the item belongs. This enables you to provide a text string that follows any user-entered values (for example, "inches", "feet", and so on).
User Entered Value Allowed Values/UI: UEV ALLOWED VALUES	Option Item	This property is displayed only if you selected "User entered value" for the Control display property of the option class to which the item belongs. This property enables you to define a comma-delimited list of values for numbers (1-3, 5, 9, 10-12, and so on). For strings, you can enter the name of a list property.
Validate Submodel/CONFIG: VALIDATE SUBMODEL	Option Item	This setting ensures that the submodel is correctly configured in nested configuration scenarios. Use Validate Submodel in conjunction with the Submodel Return property. The default behavior is not to validate a submodel configuration after returning to a parent model. When you set this property to "yes" and the Return From Submodel property to "yes", the submodel configuration will be validated after the user returns to the parent model and is configuring the parent or sibling. Consider using this setting carefully as there can be performance issues.

Field Name/Property Name	Where Used	Description
Automatic Post/UI: AUTOMATIC POST	Model Option Class	Depending on the value you choose, this property specifies how posting is done: • **none**: No update is performed when the customer selects an option item. • **update**: An incremental update occurs when the customer selects an option item. • **final (default)**: A final update occurs when the customer selects an option item.
Constant Guiding Text/UI: CONSTANT GUIDING TEXT	Model Option Class	Used to add extra text to the displayed HTML page. This text is "constant", that is, it appears all the time, even after a selection is made. For example, guiding text for a configurable camcorder may state "Only lithium batteries type XYZ are compatible with this model."

Chapter 35. Display Properties in a Tabular Form

About this task

To help users choose between two or more option items in an option class, it is often helpful to display one or more properties for each option item in the form of a table. For example, see the Tabular Display of an Option Class.

You cannot use the tabular display for pickable option items. Use tabular displays with another option class that allows users to make a selection.

To display properties in a tabular form:

Procedure

1. Navigate to the option class whose option items you want to display in a tabular form.
2. Either:
   a. Click the **Display** tab.
   b. Select Tabular Display from the Control drop-down list.
   c. Click **Save All Changes**.

      Or:

   d. Click the **Properties** tab.
   e. Select UI: CONTROL from the Unattached Properties drop-down list and enter "controls/displayProps.jsp" as its value.
   f. Click **Attach**.
3. Select UI: COLUMN HEADINGS from the Unattached Properties drop-down list and enter a semi-colon delimited list of headings as its value.

   For example, "Size;Weight;Color".
4. Click **Attach**.
5. Select UI: COLUMN PROPERTIES from the Unattached Properties drop⌂down list and enter a semi-colon delimited list of the property names as its value.

   For example, "Monitor Size;Monitor Weight;Monitor Color".

   You can use property values as described in Chapter 20, “Properties as Variables,” on page 63 to help you display the values of properties exactly as you need.
6. Click **Attach**.

   Note that the number of columns in the table is inferred from the number of properties you define in the UI: COLUMN PROPERTIES property.
7. Click **Save All Changes**.
8. If you now click **Test**, then you can verify that the option class is now presented as a table with one row for each option item and one column for each property specified.

Chapter 36. Image properties

Associating Images to Models and Option Classes

About this task

Use the Icon Graphic field on the **Display** tab for models and option classes. This corresponds to the UI: ICON GRAPHIC property.

Associating Images to Option Items

You can attach images to option items and display them to end-users using the UI: ITEM IMAGE NAME property to specify an image for each option item. You must set the UI: SHOW ITEM IMAGES property to be “true” at the option class level.

The value of the UI: ITEM IMAGE NAME can be interpreted as a relative URL or as an absolute URL:

- If you enter "2of4stars.gif" or "../images/2of4stars.gif", then the image will be displayed by resolving the image location to:

  ```
  http://server:port/Sterling/en/US/images/2of4stars.gif
  ```

- You can use absolute URLs to point to different locations anywhere on the Web. This is particularly useful if you use a different Web server to serve up static content for your Web site. For example:

  ```
  http://imageserver:port/configurator/images/2of4stars.gif
  ```

Chapter 37. Define User-Entered Values

About this task

You can allow your customers to type in values for a configurable product's options. For example, you may want to let customers enter a color that is not one of the pre-defined colors in a model, or you may want to let them enter a product ID for a product that is not in your product catalog, but which you can fulfil by special order.

The User Entered Value properties, described in the table, enable customers to type in values. For example, suppose that you have a configurable product and you want to let the user specify their own choice of color.

To define user-entered values:

Procedure

1. Navigate to the model and click **Edit**.
2. Click **New Option Class**.
3. In the Name field, enter "Custom Color Class".
4. In the Description field, enter "Enter your preferred color".
5. Click **Save**.
6. Click the **Display** tab.
7. Set the Control display property to "User Entered Values".
8. Check the **Ignore in Quote** check box.
9. Click **Save All Changes**.
10. Click the **General Info** tab.
11. Click **New Option Item**.
12. In the Name field, enter "Custom Color Item".
13. In the Description field, enter "We will provide a color match before shipping.".
14. In the Navigation panel, navigate to the Custom Color Item option item.
15. Click the **Display** tab.
16. Select String, Integer, or Numeric from the **User Entered Value Type** drop-down list.
17. Click **Save All Changes**.
18. Click **Compile**.

    Note: You can use user-entered values in rules by referring to the appropriate UEV property: UEV: NUMERIC VALUE (for Integer or Numeric values) or UEV: STRING VALUE (for String values).

Chapter 38. UI Control Reset Behavior

Some UI controls allow the user to reset (clear) a selection and start over. A **Clear** button displays in the configuration UI by default to enable this reset behavior. The following table summarizes the default behavior of the **Clear** button in UI controls:

UI Control	Default View	Default Selection	Allowed User Action
Checkbox	Displays all values.	The model's default selections are checked. If there is no default value, nothing is checked.	Check or uncheck values. Clicking **Clear** checks the default value. If there is no default value, clicking **Clear** clears all values.
Radio Button	Displays all values.	The model's default selection is selected. If there is no default value, nothing is selected.	Check or uncheck values. Clicking **Clear** selects the default value. If there is no default value, clicking **Clear** clears all values.
List Box	Displays all values.	The model's default selection is selected. If there is no default value, nothing is selected.	Select any value in the list box. Clicking **Clear** selects the default value. If there is no default value, clicking **Clear** clears all values.
Multiple Selection List Box	Displays all values.	The model's default selections are selected. If there is no default value, nothing is selected.	Select or unselect any value. Clicking **Clear** selects the default value. If there is no default value, clicking **Clear** clears all values.
Display All Children	Displays all values.	No default selection.	User cannot take any action.
Drop-down List	Displays all the values in the drop-down.	The model's default selection is selected. If there is no default value, nothing is selected.	Select any value in the drop-down list. Clicking **Clear** selects the default value. If there is no default value, clicking **Clear** clears all values.
Dynamic Instantiation	Nothing displays.	No default selection.	User cannot take any action.
Tabular Display	Nothing displays.	No default selection.	User cannot take any action.
Single-Select Tabular Display	Displays all values and a **Reset** button.	The model's default selection is selected. If there is no default value, nothing is selected.	Select any value. Clicking **Clear** selects the default value. If there is no default value, clicking **Clear** clears all values. Clicking **Reset** clears all values.
Multi-Select Tabular Display	Displays all values.	The model's default selection is selected. If there is no default value, nothing is selected.	Select or unselect value(s). Clicking **Clear** selects the default value. If there is no default value, clicking **Clear** clears all values.

UI Control	Default View	Default Selection	Allowed User Action
Tabular Display with Quantity Box Selection	Displays all values with quantity boxes.	The model's default selection is selected. If there is no default value, nothing is selected.	The user cannot take any action.
User Entered Values	Displays all with text boxes.	No default selection.	Enter values. Clicking **Clear** clears all values.

Chapter 39. Modify System Settings

About this task

You perform the system administration tasks through the **System Administration** link on the Visual Modeler Administration page. This link is accessible only to authorized personnel.

You can modify system configuration settings only if you have the appropriate access function. In the reference implementation provided with the Visual Modeler, only users with the Program Management function (defined in the **Entitlements.xml** configuration file as EnterpriseProgramManagement) may access the System Administration pages.

To modify system settings:

Procedure

1. Click **System Services in the System Administration panel** on the Visual Modeler home page.

   The system configuration properties are organized into logically-related groups.
2. Access each group by clicking the corresponding link on the System Administration page.

   Each link takes you to a new page that displays the current values for each property.
3. Make the appropriate changes as necessary.

   See "Locale Settings" on page 123 and "Define Display Names" on page 123 for a description of each set of properties.
4. Click **Save All and return to List**.
5. By default, changes to the value of a system property take effect immediately, and are persisted to the file system. A server restart is not necessary, but if you do restart the server, the new value of the property remains in effect.

Chapter 40. Configuration Properties

Locale Settings

The locale names supported by your installation combine the ISO-639 language codes and ISO-3166 country or region codes. You can define display names that will appear for these locale names in the Visual Modeler. You can define a display name for each supported locale, that is, how each locale name will appear for each supported locale. For example, you can decide that, in the en_us locale, "en_us" will be displayed as "United States", while in the de_de locale (Germany), "en_us" will be displayed as "Vereinigte Staaten".

Note: If a display name is not defined for a locale name for the locale effective during a session, then the fields in which that locale name should appear will be blank.

Changes that you make to locale names become active when you restart the Visual Modeler.

Define Display Names

About this task

To define display names:

Procedure

1. Select an effective locale from the **Effective Locale** drop-down list.

   The locale names for the supported locales appear in the **Locale Name** column. The current display name, if any, for each locale appears in the next field in the language of the effective locale.
2. In the text field next to each locale name, type the display name you want to appear for each locale name.
3. Repeat the last two steps for each supported locale.
4. Click **Save** to save the changes and remain at the Locale Settings page; click **Save All and return to List** to save the changes and redisplay the System Administration page.

   If you click **Save**, then the **Effective Locale** field re-displays the default system locale, as defined in the Internationalization properties.

   Repeat these steps for each locale in the **Effective Locale** drop-down list.

   Note: If you change the effective locale without clicking **Save**, then any unsaved changes to Display Names will be lost.

Chapter 41. Job Scheduler Settings

The Visual Modeler supports the ability to schedule tasks that must be performed at regular intervals as cron jobs.

There are two types of cron jobs: system and application.

- System cron jobs run without session information and without an associated Visual Modeler user. Typically, they are used for low-level background tasks such as garbage collection. System cron jobs do not save their last execution time or execution status to the Knowledgebase because the same job may be run on several servers in a cluster.
- Application cron jobs are used when session information (such as a username or locale) is required to run the job or if audit information might be needed to determine how changes to data objects were made. Application cron jobs are initiated by posting an XML message to the Visual Modeler using the message URL for cron jobs: consequently, to enable application cron jobs, you must take care to set this URL correctly.

  For example, if the main URL used to access the Visual Modeler is:

  `http://server:port/Sterling/en/US/enterpriseMgr/matrix`

  then set the cron job message URL to:

  `http://server:port/Sterling/msg/matrix`

  Similarly, if the main URL used to access the Visual Modeler is:

  `http://server:port/store/en/US/enterpriseMgr/anderel`

  then set the cron job message URL to:

  `http://server:port/store/msg/anderel`

You can choose whether or not to allow either type of cron job to run on your implementation.

Note: Some cron jobs such as the search index builder must be run as an application cron jobs. To support advanced search, you must enable application cron jobs.

Application cron jobs are created specifying a username and password of a Visual Modeler user. You must ensure that the Password data field of the CronConfig data object is not set to store one-way encrypted values.

In a clustered installation of the Visual Modeler, if you want a job to run on all servers in the cluster, then make it a system cron job. If you want the job to be run on only one server in the cluster, then you must make it an application cron job.

Chapter 42. Manage Business Rules

About this task

During implementation, you can configure business rules by editing the property files provided with the Visual Modeler. After implementation, you can manage the business rules from the Business Rules Manager page. See Chapter 2, “Configuring the Visual Modeler,” on page 3 for an overview of business rules administration.

To manage the business rules in your Visual Modeler, you must be an enterprise user that has been assigned the appropriate function: typically, this is the Program Management function.

To manage business rules:

Procedure

1. Click **Business Rules in the System Administration panel** on the Visual Modeler home page.

   The Business Rules Manager page displays.

   The Business Rules link appears on the Visual Modeler home page only if you are assigned the appropriate function to perform this task.
2. Click a link to modify the desired set of business rules.

   Each business rule contains help text describing the rule.
3. Click **Save**.
4. Unless otherwise directed, the changes to the value of a business rule take effect immediately, and are persisted to the file system. This means that a server restart is not necessary, but if the server is restarted, the new value of the business rule continues to be used.

Chapter 43. Job Scheduling Tasks

Display a Scheduled Job

About this task

To display a scheduled job:

Procedure

1. Click **Job Scheduler in the System Administration panel** on the Visual Modeler home page.
2. Click the name of a cron job to display the details of the selected job.

Create a Job

About this task

Note: If you are running multiple instances of the Visual Modeler, then creating or modifying a cron job will affect any of these instances running off the same Knowledgebase instance.

To create a job:

Procedure

1. Click **Job Scheduler in the System Administration panel** on the Visual Modeler home page.
2. Click **Create New Job**.

   The Cron Job Configuration page displays.
3. Enter the information about the job

Field	Description
Job Name	The name of the cron job
Program	The java implementation class that executes the job
Command Line Arguments	The command line parameters that provide information about the job. For example, you can specify that a cron job should time out after 300 seconds (5 minutes) by setting the RequestTimeout parameter as follows: RequestTimeout=300
Cron Job Type	The type of the cron job: a system level cron job (such as cache cleaning) or an application level cron job (such as importing/exporting) If you select **Application**, then you must enter the username and password required for access to the particular data.
Frequency	How often will the job be run? Every three days? Every week? Every five minutes? and so on.

Field	Description
Start date and time/ End date and time	The effective start and end period between which dates and times the cron job will run. This, along with Frequency, determines when the job will run. For example, if you entered a frequency of three days, then the job will run every three days from the task start date and time until the task end date and time is reached. You can enter the same dates and times for both start date and time, in which case the job will be run only once, at a specific time.

4. Check the box next to **Active** to make the job available to be run.
5. Click **Save All Changes**.

Modify a Job

About this task

Note: If you are running multiple instances of the Visual Modeler, creating or modifying a cron job will affect any of these instances running off the same Knowledgebase instance as the cron job.

To modify a job:

Procedure

1. Click **Job Scheduler in the System Administration panel** on the Visual Modeler home page.
2. Click the name of a cron job to display the details of the selected job.

   The details are displayed on the Cron Job Configuration page.
3. Enter the information about the job.
4. Check the box next to **Active** to make the job available to be run.
5. Click **Save All Changes**.

Run a Cron Job Immediately

About this task

You may need to sometimes run a cron job immediately.

To run a cron job immediately:

Procedure

1. Click **Job Scheduler in the System Administration panel** on the Visual Modeler home page.
2. In the list of cron jobs, identify the job that you want to run immediately.
3. Click **Run Now**.

   The cron job will be immediately scheduled to run, but if jobs are ahead of it in the cron job queue, then it will not run until those jobs have completed.

Delete a Job

About this task

To delete a job:

Procedure

1. Click **Job Scheduler in the System Administration panel** on the Visual Modeler home page.
2. Check the box next to the job(s) you want to delete.
3. Click the Delete icon (**X**) in the Actions column.

View the History of a Cron Job

About this task

You may need to review how a cron job has run in the past.

To view the history of a cron job:

Procedure

1. Click **Job Scheduler in the System Administration panel** on the Visual Modeler home page.
2. In the list of cron jobs, identify the job whose history you want to view.
3. Click **Show History**.

Chapter 44. Visual Modeler Cron Jobs

If you install the Visual Modeler with the reference data, the installation includes the pre-defined cron jobs described in this section.

If you installed the Visual Modeler with minimal data, only the Cache Cleanup job is included. If you want to implement the other jobs, you must create them by following the steps in “Create a Job” on page 129. The following sections contain the information needed to create these jobs.

Note that all cron job timeout values are specified in seconds.

Specifying -1 as a timeout value means that the cron job never times out.

Cache Cleanup

This group of properties determine the frequency and class of the cron job used to clean the cache.

Maintain Configuration

This cron job deletes saved configurations of the specified configuration type that are older than the specified age. The default age for deletion is 10 days.

Cron Job Field	**Entry**
Program	com.comergent.apps.configurator.main.ConfigMaintainanceC
Command Line Arguments	ConfigType=Config&AgeInDays=10
Cron Job Type	Application

Chapter 45. Site System Administration

There is a distinction between *system administration* and *enterprise administration*:

- System administration is the responsibility of *system administrators*: they manage the basic system properties of the Visual Modeler and the system cron jobs.
- Enterprise administration is the responsibility of enterprise users: these users manage enterprise data. For example, enterprise users are responsible for administering models.

A system administrator can manage:

- System users: see "Create a System Administrator User."
- System profile: see "Manage the System Administrator Profile" on page 136.
- System properties: see "Update a System Property" on page 136.
- System cron jobs: see "Create a System Cron Job" on page 136.
- System status: see "View the System Status" on page 137.

All these tasks are performed from the system administration home page.

Access the System Administration Home Page

About this task

To access the System Administration Home page:

Procedure

1. Point your browser to the System Administration URL. By default, this is:
   `http://server:port/Sterling/en/US/enterpriseMgr/admin`
   Check your site documentation to identify this URL.
2. Log in as a system administrator. When the Visual Modeler is first installed, the default username/password combination is admin/admin. If other system administrator users have been created, then you can log in using one of these userids

   Note: You must change at least the password of the admin user to protect the system from unauthorized access. We suggest that you create a different system administrator user, and then delete the admin user.
3. From this page, you can perform the task described in the following sections.

Create a System Administrator User

About this task

To create a system administrator user:

Procedure

1. Log in as a system administrator.
2. Click **System Users**.
3. Click **Create User**.
4. Enter information for the new user as appropriate.

5. Click **Save**.

   The new user information is saved.
6. You can verify that the new user has been successfully created, by logging out and logging back in as the newly-created user.

Manage the System Administrator Profile

About this task

To manage the system administrator profile:

Procedure

1. Log in as a system administrator.
2. Click **View Your Organization Profile**.
3. Modify the profile details as appropriate.
4. Click **Save**.

Update a System Property

About this task

System administrators can manage the system-level properties for the Visual Modeler, including configuring logging settings, job scheduler categories, Configurator settings (number of models to cache, default template directory and page template), whether or not to use session-based caching, and so forth.

To update a system property:

Procedure

1. Log in as a system administrator.
2. Click **System Services**.
3. Click the link to the set of properties that you wish to update.
4. Modify the values of properties as required.
5. Click **Save All and Return to List**.

Create a System Cron Job

About this task

System administrators can manage system cron jobs.

To create a system cron job:

Procedure

1. Log in as a system administrator.
2. Click **Job Scheduler**.
3. Click **Create New Cron Job**.
4. Enter details for the new system cron job as appropriate.
5. Click **Save All Changes**.

View the System Status

About this task

To view the system status:

Procedure

1. Log in as a system administrator.
2. Click **System Status**.
3. Review the system status details as appropriate.

Index

Special characters

A

B

C

D

E

F

G

H

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

1623-14, Shimotsuruma, Yamato-shi

Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be

incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA 95141-1003

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2012. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2012.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Connect Control Center®, Connect:Direct®, Connect:Enterprise®, Gentran®, Gentran®:Basic®, Gentran:Control®, Gentran:Director®, Gentran:Plus®, Gentran:Realtime®, Gentran:Server®, Gentran:Viewpoint®, Sterling Commerce™, Sterling Information Broker®, and Sterling Integrator® are trademarks or registered trademarks of Sterling Commerce®, Inc., an IBM Company.

Other company, product, and service names may be trademarks or service marks of others.

Printed in USA