Release Management v1

Managing Releases in ServiceNow

PDF generated using the open source mwlib toolkit. See http://code.pediapress.com/ for more information.
PDF generated at: Wed, 29 Nov 2017 20:53:13 PST

Release v1

Release Overview

Overview

The Release Management application provides Release Tables to store information about the planned release, and the tasks that will be required to execute the release. It also provides Phase Automation to power the workflow around the release process.

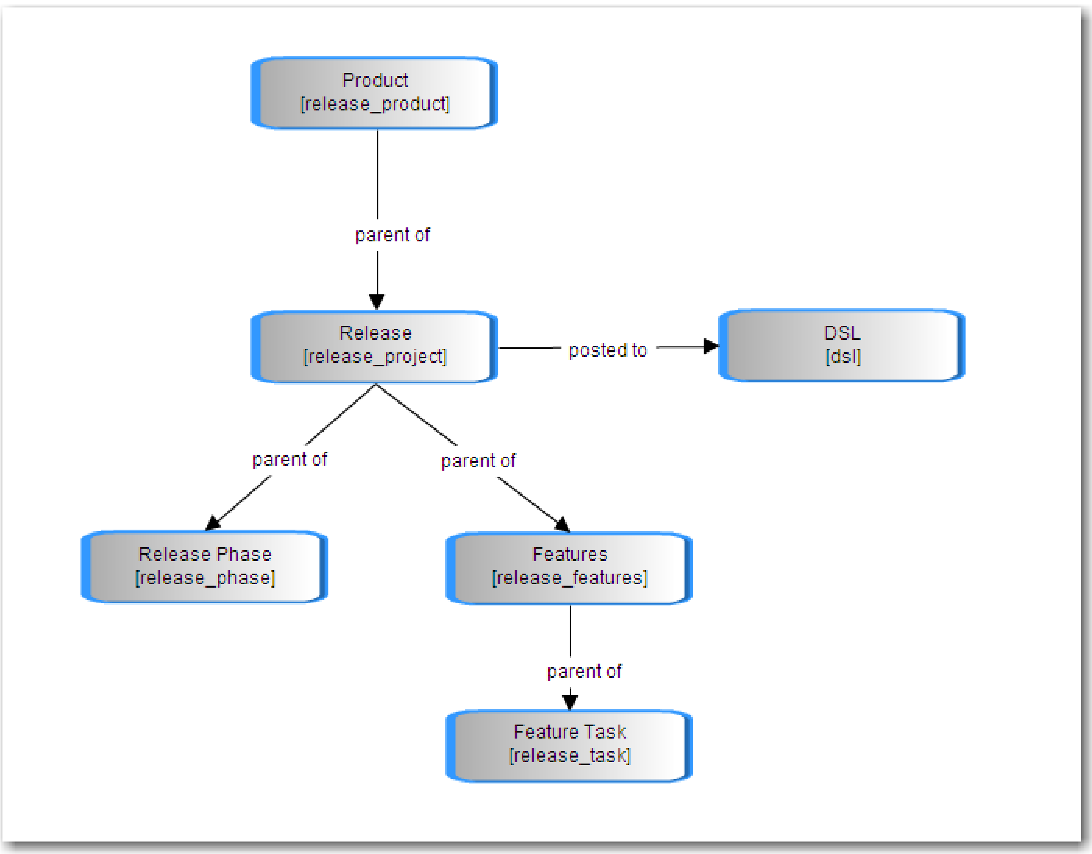
For information on how to manage the release process using the tools outlined here, see Managing a Release Process.

Release Tables

The following tables represent the concepts that power Release Management:

- **Product [release_product]** - represent whole products whose releases are being managed.
- **Release [release_project]** - represent individual versions of the product.
- **Release Phase [release_phase]** - represent the different stages of work required to complete a release.
- **Feature [release_feature]** - represent each feature within the release.
- **Feature Task [release_task]** - represent tasks to create individual features.
- **DSL [dsl]** - represent digital media that store the product at time of release. Although it is not a CI, it can be related to a CI with a reference field.

The tables are inter-related as follows:



Phase Automation

Business Rules on the Release tables automate the process of Release phases.

Phase Generation

The business rule **SNC Release Insert Phases** generates the phases for a release based on the type of the Release, if the **Phases Wanted** field on the Release form is checked. Once the Release is saved, the business rule will create standard Phases. The form will need to be reloaded before they will be visible.

Phase Advancement

Phase workflow is powered by business rules.

The business rule **SNC Release phase events**:

- Activates phases when they are approved.
- If the phase is approved or marked **Closed Complete**, activates the next phase.
- Updates the Release's state according to the state of the phase, and according to which state is active.

The business rule **SNC Release phase**:

- Sets the Phase's state based on approvals.
- Updates the end-date based on completion.

The business rule **SNC Release phase - new**:

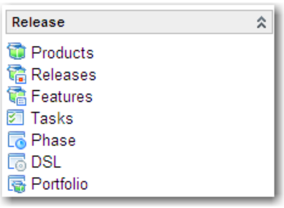
- Documents an added phase in the Release's journal field.

The business rule **SNC Release Delete Phases**:

- Deletes phases when a Release is deleted.

Release Application and Modules

The following modules are available in the **Release** Application:



- **Products** - The **Product [release_product]** list.
- **Releases** - The **Release [release_project]** list.
- **Features** - The **Feature [release_feature]** list.
- **Tasks** - The **Feature Task [release_task]** list.
- **Phase** - The **Release Phase [release_phase]** list.
- **DSL** - The **DSL [dsl]** list.
- **Portfolio** - The **Business Service [cmdb_ci_service]** list, with a Release-specific view.

Defining a Product with Release Management

Note: *This article applies to Fuji. For more current information, see Define a Profuct with Release Management* [1] *at* http://docs. servicenow.com The ServiceNow Wiki is no longer being updated. Please refer to http://docs.servicenow.com for the latest product documentation.

Note: *This documentation applies to Release v1. For Release v2, please refer to Define a Product with Release v2* [2] *at* http://docs. servicenow.com The ServiceNow Wiki is no longer being updated. Please refer to http://docs.servicenow.com for the latest product documentation.

Overview

Before a release process can be managed, a **Product** needs to be defined. A Product record stores information about the product for reference purposes, as well as grouping together all of the releases and features for a single product.

Creating a Product

Before a release process can be managed, define the product whose release will be managed.

To define a product, navigate to **Release > Product**

Field	Input Value
Number	Unique identifier for the Product. Generated using Number Maintenance.
Name	The name of the Product.
Manager	A reference to the product's manager.
Business Service	The Business Service in the CMDB that corresponds to the product.
Plan Approval	A group to approve any **Plan** phases for the release.
Build Approval	A group to approve any **Build** phases for the release.
Acceptance Approval	A group to approve any **Accept** phases for the release.
Release Approval	A group to approve any **Release** phases for the release.
Deploy Approval	A group to approve any **Deploy** phases for the release.
Blackout Approval	A group to approve any **Blackout** phases for the release.
Short Description	A short description of the product.
Description	A more verbose description of the product.
Notes	A journal field of notes for the product.

Relating to Business Services

The reference field **Business Service** can be used to link the product with a corresponding Business Service in the CMDB. Business Services keep information about how they relate to other Configuration Items, and can track any incidents, problems, or changes related to it. Specifying a Business Service for the Product connects information from the release process to other processes in the instance.

References

[1] https://docs.servicenow.com/bundle/jakarta-it-business-management/page/product/release-management/reference/r_DefineProductReleaseMgmt.html
[2] https://docs.servicenow.com/bundle/jakarta-it-business-management/page/product/release-management/reference/r_DefiningAProductWithReleaseV2.html

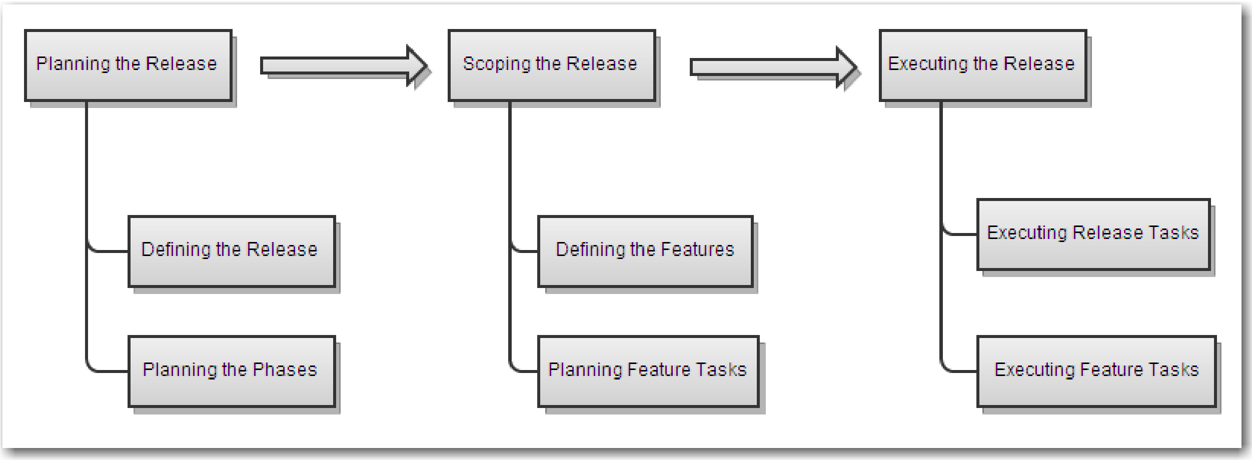
Managing a Release Process

Note: *This article applies to Fuji and earlier releases. For more current information, see Release Management Application* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**'

Overview

Once a product is defined, a release can be planned and executed. The steps in managing a release process are:

1. **Planning the release**: defining the Release record, and the release phases.
2. **Scoping the release**: defining features and feature tasks.
3. **Executing the Release**: performing the steps required to complete the release.



Planning the Release

Planning the release process involves:

1. Defining the Release
2. Planning Phases

Defining the Release

To create a release, navigate to **Release > Product**, select the product whose release will be created, and click the **Release** related list's **New** button. Populate the following fields:

Field	Input Value
Number	Unique identifier for the Release. Generated using Number Maintenance.
Name	The name of the Release.
Due Date	The date planned for the release to be distributed.
Manager	A reference to the product's manager.
Workflow State	A choice list of the state of the release. Note that phases will be automatically generated once the **State** reaches **Approved.**
Phases Wanted	If checked, phases will be dynamically generated.
Planned start date	The date planned for beginning work on the release.
Planned end date	The date planned for completing work on the release.
Product	A reference to the release's product.
Priority	A choice list of the release's priority.
Requested By	A reference to the user who requested the release.
Type	A choice list of release types. Note that this field will be used to automatically generate the Release Phases.
Risk	A choice list of risk.
Active	Determines whether or not the release is currently being used.
Actual start date	The date that work actually began on the release.
Actual end date	The date that work actually ended on the release.
Short Description	A short description of the release.
Description	A more verbose description of the release.

Planning Phases

If **Phases Wanted** is checked, the phases will be dynamically populated, and can be filled in with extra information. If **Phases Wanted** is unchecked, the phases will need to be created from scratch.

The following fields define the phase:

<table>
<tr><th>Field</th><th>Input Value</th></tr>
<tr><td>Number</td><td>Unique identifier for the Phase. Generated using Number Maintenance.</td></tr>
<tr><td>Name</td><td>The name of the Phase. Select from the following choices:
Plan
Build
Accept
Release
Deploy
Back Out
The selected name is used to generate approvals based on the different approval fields on the Product form.</td></tr>
<tr><td>Assigned To</td><td>A reference to the person responsible for the phase.</td></tr>
<tr><td>Active</td><td>Determines whether or not the phase is currently being performed.</td></tr>
<tr><td>Expected start</td><td>The date planned for beginning work on the phase.</td></tr>
<tr><td>Order</td><td>The order of the phase relative to other phases. Phases are evaluated from lowest to highest.</td></tr>
<tr><td>Short Description</td><td>A short description of the phase. If the phase is dynamically generated, it will automatically populate with a short description in the following format:
<code>[PHASE NAME] phase for project [RELEASE NUMBER] ([RELEASE SHORT DESCRIPTION])</code></td></tr>
<tr><td>Description</td><td>A more verbose description of the phase.</td></tr>
<tr><td>Work Notes</td><td>A journal field for storing notes about the phase's execution.</td></tr>
</table>

The related list of **Approvals** will be generated based on the Product's approval fields, or can be manually specified.

Scoping the Release

Once the release process is defined, the features and their tasks can be scoped. To add or create new features, use the **New** and **Edit** buttons on the **Feature** related list.

Populate the following fields on the **Feature** form:

Field	Input Value
Number	Unique identifier for the feature. Generated using Number Maintenance.
Name	The name of the feature.
Product	A reference to the parent product. If created from the release, it will be populated with the Release's product.
Release	A reference to the parent release. If created from the release, it will already be populated.
Priority	A choice list of the feature's priority.
Planned start date	The date planned for beginning work on the feature.
Planned end date	The date planned for completing work on the feature.
Product	A reference to the feature's product.
Priority	A choice list of the feature's priority.
Risk	A choice list of risk.
Percent Complete	A decimal field to store the percent of the feature that is complete. *Since Spring 2010 Stable 3, this field is a Percent Complete Field Type.*
Planned start date	The date planned for beginning work on the feature.
Planned end date	The date planned for completing work on the feature.

Type	A choice list of feature types. Choose from: • Feature • Bug fix • Improvement • Documentation • Major Enhancement
Requested By	A reference to the user who requested the feature.
Requested Date	The date that the feature was requested.
Due Date	The date that the feature is due.
RFC	A reference to a Change Request.
Workflow State	A choice list of the state of the feature.
Actual start date	The date that work actually began on the feature.
Actual end date	The date that work actually ended on the feature.
Short Description	A short description of the feature.
Documentation	An HTML Field for more verbose description of the feature.
Related List: Feature Task	
Number	Unique identifier for the feature task. Generated using Number Maintenance.
Assigned To	A reference to the user responsible for the Feature Task.
Due Date	The date that the feature task is due.
Type	A choice list of feature types. Choose from: • Feature • Bug fix • Improvement • Documentation • Major Enhancement
State	The state of the feature task.
Workflow State	A choice list of the state of the feature task.
Percent Complete	A decimal field to store the feature task's percent complete. *Since Spring 2010 Stable 3, this field is a Percent Complete Field Type.*
Feature	A reference to the parent feature. If created from the feature, it will already be populated.
Short Description	A short description of the feature task.
Description	A more verbose description of the feature task.

Executing the Release

Once the Release is planned and scoped, phases and feature tasks are ready to be executed. Typically, the feature tasks are performed during the **Build** phase.

As phases are completed, a business rule will update the Release and activate the next phase.

Completing the Release

Once the release is complete, all related problems will be closed and entries will be posted to the DSL.

References

[1] https://docs.servicenow.com/bundle/jakarta-it-business-management/page/product/release-management/concept/c_Release.html

Customizing the Release Process

Note: *This documentation applies to Release v1. For Release v2, please refer to Release Management Plugins.*

Overview

The Release Process is automated by a series of Business Rules which can be customized to tailor the behavior of the release process.

The method by which Phases are generated can be customized without editing business rules.

Impact

Customizing business rules prevents those business rules from being updated during future upgrades. Where possible, create a new business rule instead of customizing an existing one.

Release Business Rules

The release process is automated by a series of business rules.

Task Automation

The Release Task Automation business rules power the workflow of the process. Customizing these business rules can suit the Release Management process to the particular process of the organization.

These business rules help drive the Release tasks to completion:

- **SNC Release Delete Phase** - Deletes the phases associated with a given release.
- **SNC Release feature** - Ensures that features have the same product associated as its parent release.
- **SNC Release Insert Phases** - Generates the phases for a release. For more information, see Customizing Phase Generation.
- **SNC Release phase** - Changes state of release phase based on approvals.
- **SNC Release phase - new** - Updates the Release if a new phase is added.
- **SNC Release phase events** - Manages release phases by activating or deactivating phases.
- **SNC Release Project - New** - Sets the Release Manager based on the Product.
- **SNC Release task events** - Completes percent complete for the Feature based on the Feature Tasks.

For more information on how the Release Phase business rules work together, see Phase Automation.

Event Logging

Event Logging business rules are used to mark events in the Event Registry, which in turn can power email notification or record important information in the log. Customize these business rules to trigger email notifications or log events which are not marked out-of-box.

These business rules log events, which can be used for email notifications:

- **SNC Release feature events** - Logs events for changes to features.
- **SNC Release project events** - Logs events for changes to releases.
- **SNC Release phase events** - Logs events for changes to phases, in addition to automating phase activation.

Release Completion

Release Completion business rules update the release tables on completion of a release, including marking child tasks inactive, posting entries to the DSL, and recording features to the Product. Customize these business rules to determine what occurs after the Release completes.

These business rules are triggered when a Release is completed, to update the Product tables:

- **SNC Release Complete** - Closes all related problems (including those related to features and to the change request), and posts entries to the DSL.
- **SNC Release project - feature** - Makes features inactive if a release is marked inactive, and changes Product if Release's Product changes.
- **SNC Release project complete** - Marks release inactive upon completion.

Customizing Phase Generation

Phases are generated by the business rule **SNC Release Insert Phases** when the Release is inserted.

Customizing Phases

The phases are determined by a choice list on the Release field **Phases**, which is dependent on the **Type** field. For example, if the type is **Bug Fix**, the **Phases** choice list is Build, Accept, and Deploy. Those choices will be the names of the phases on insert.

To customize a Release Type's phases:

1. Navigate to the Release form.
2. Add the **Phases** field to the form by configuring the form.
3. Right click the label and select **Configure Choices** (**Personalize Choices** in versions prior to Fuji).
4. Add or remove choices as desired and save.

Customizing Phase Short Descriptions

The generated phase's Short Description is determined within the business rule. To customize it, find this line of code:

```
var desc = grPhase.name.getDisplayValue() + ' phase for project ' + current.number + ' (' + current.short_description + ')';
```

This results in the default short description in the format:

```
[PHASE NAME] phase for project [RELEASE NUMBER] ([RELEASE SHORT DESCRIPTION])
```

For example, to get this format:

```
[RELEASE NUMBER]: [PHASE NAME]
```

Change the line to:

```
var desc = current.number + ': ' + grPhase.name.getDisplayValue();
```

Moving from Release v1 to Release v2

	Functionality described here requires the **Release Management v2** plugin.

Overview

Release Management Plugins enhance release functionality by implementing a new data structure and new user interfaces to make managing a release more flexible and more powerful.

Upgrading from Release v1 to Release v2

To upgrade from Release v1 to Release v2, simply activate the appropriate plugin.

Activating the plug in will create a new application and new tables. The old applications and tables will not be deleted; the **Release** application will be deactivated, but is accessible through **System Definitions > Applications**.

The existing Release data will not be converted into Release v2 data, but will be accessible through the old application for historical purposes.

Comparing Release v1 to Release v2

Once the new plugin is activated, make sure to review the new data structure and user interfaces to understand how Release v2 differs from the original Release Management application.

New Data Structure

New Tables

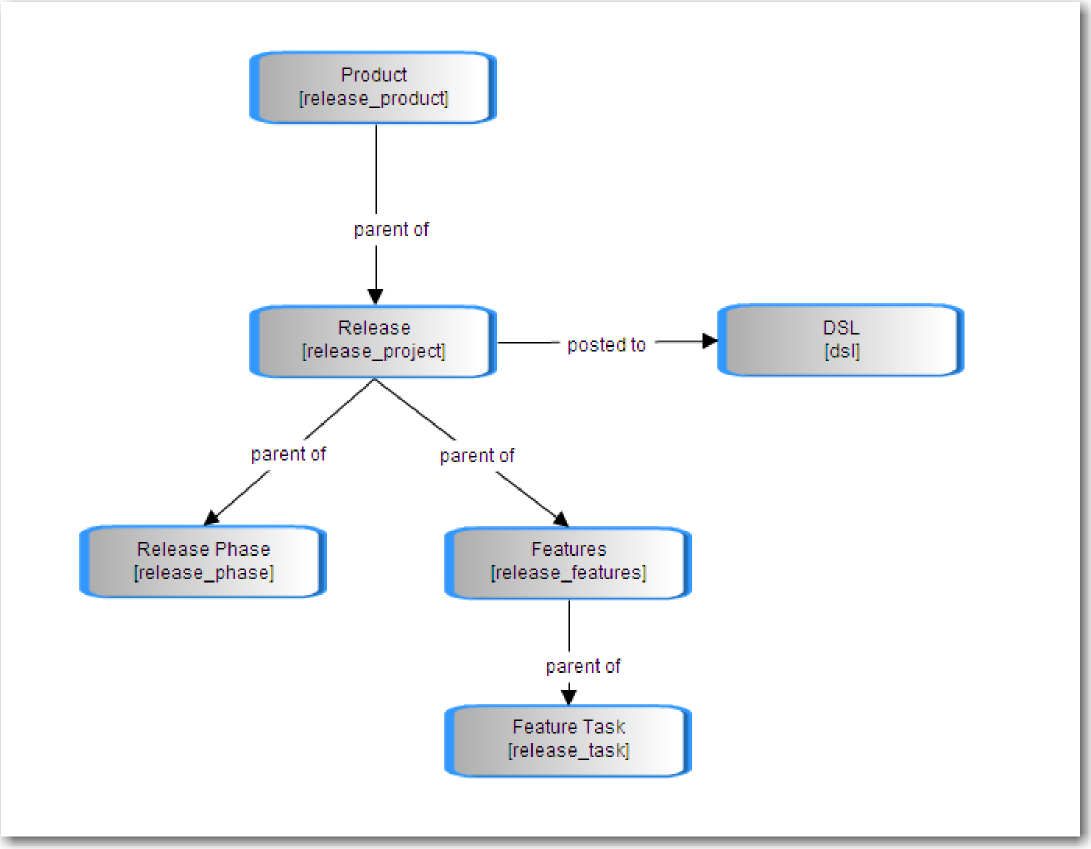
Each of the different Release Management Plugins implements a different set of tables (see Release Management Plugins for specifics). These tables differ from the old Release tables because they are now extensions of the

Planned Task table, meaning they can now use all of the functionality specific to the **Task** table (such as Approval and Assignment Rules), as well as the standard time-related functionality provided by Planned Task.

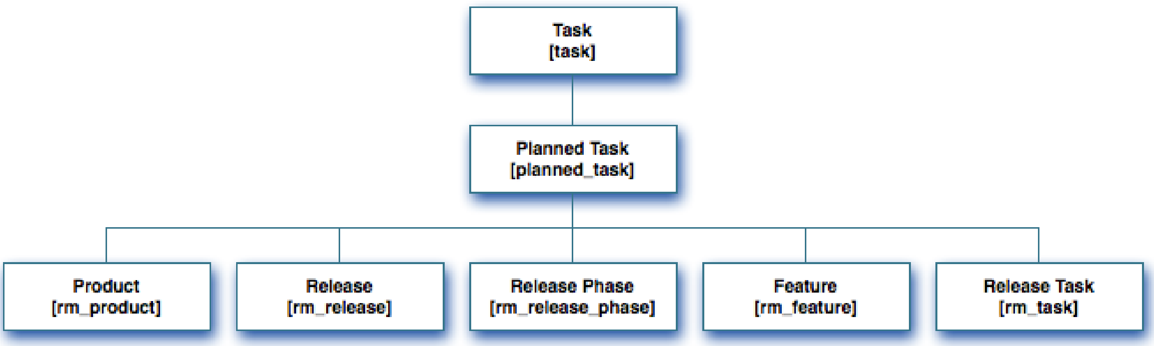
New Table Relationships

The change in the data structure is not simply new tables. The relationships between the tables are implemented differently, to allow more flexibility.

The old tables were separate tables related by a parent-child relationship:

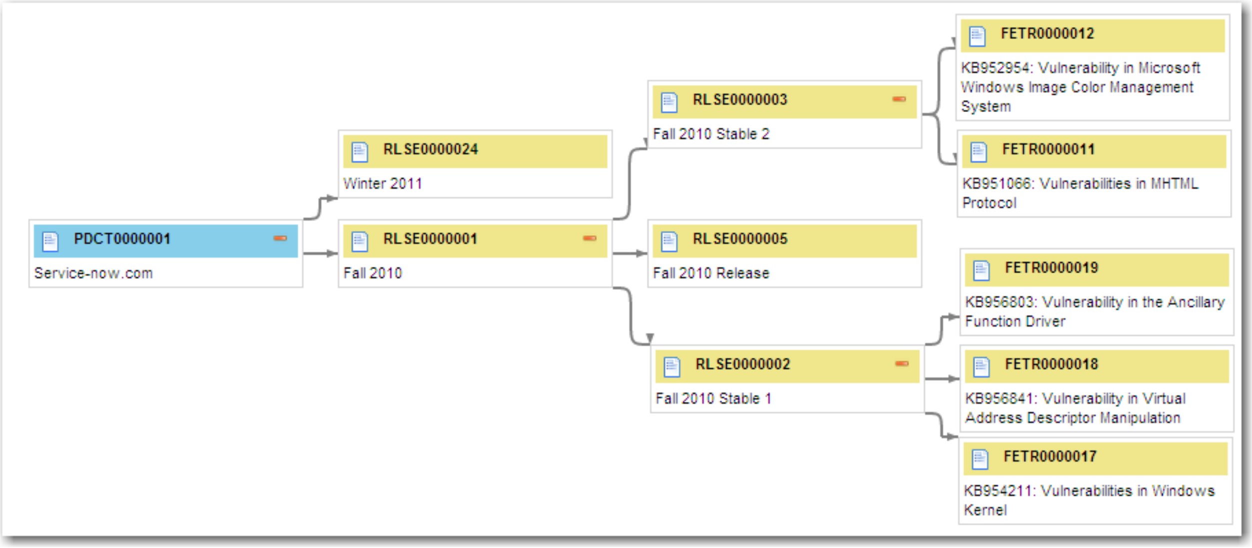


The new tables are now all extensions of the **Planned Task** table:



Individual records can be associated with themselves in parent-child relationships using the **Parent** reference field available to all of them. However, because the **Parent** reference field can point to any task, records on the release tables can be associated with any records anywhere on the release table. Therefore, it is possible to have Parent and Child releases and Parent and Child features.

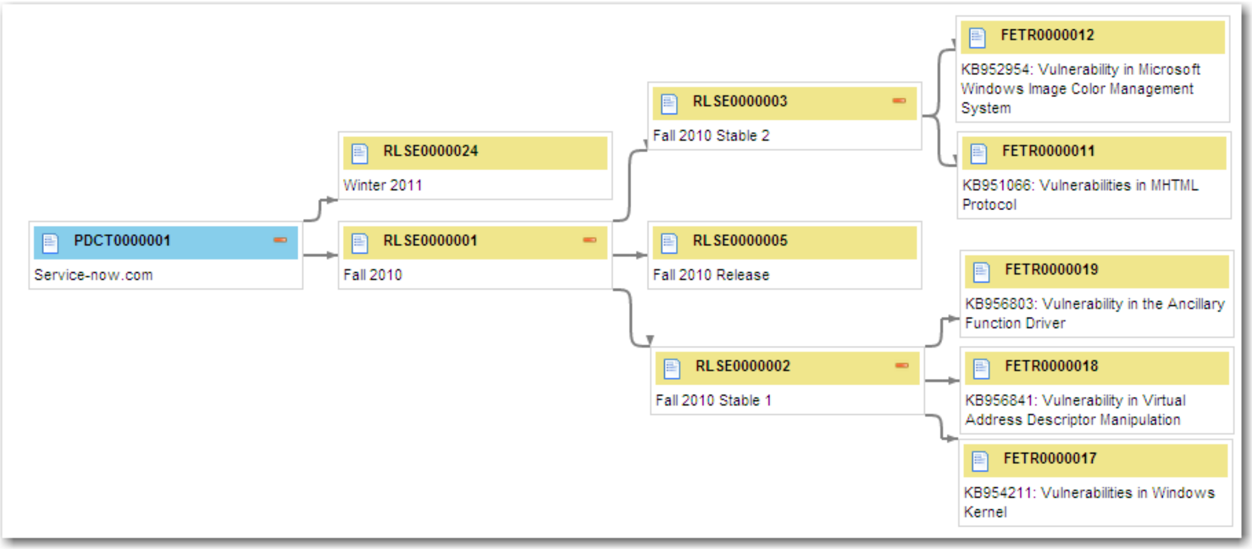
In this example (displayed using the hierarchy view), the Product has Releases is broken up into sub-releases, which have child features defined:



New User Interfaces

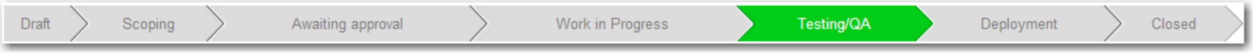
Release Management Hierarchies

The **Hierarchy** view of Products and Releases can be found from a related list on the form:



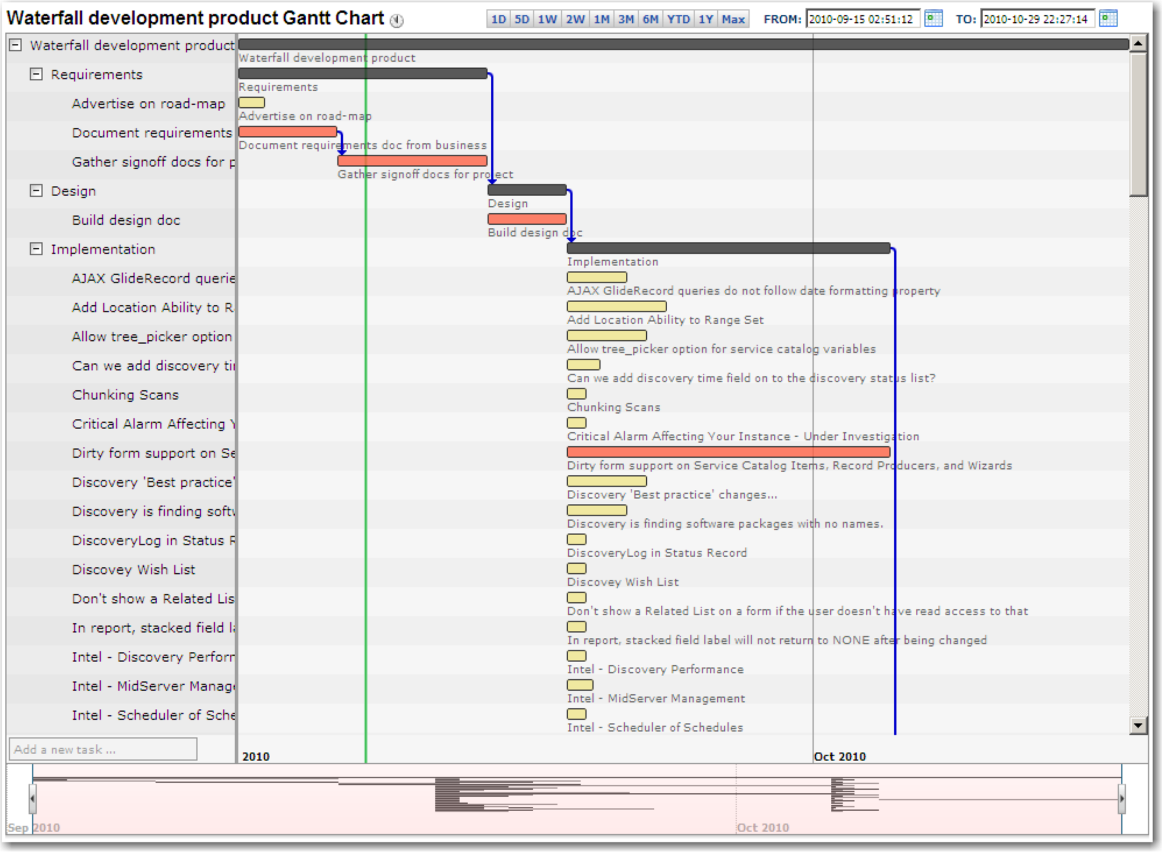
Release Management Process Flow

The **Release** and **Feature** forms include a process flow, which displays the current status of the record based on the **State** field. These are generated using the Process Flow Formatter Plugin:



Release Gantt Chart

Releases can also be managed using the Gantt Chart:



Article Sources and Contributors

Release Overview *Source*: http://wiki.servicenow.com/index.php?oldid=100930 *Contributors*: Davida.hughes, G.yedwab, Joseph.messerschmidt

Defining a Product with Release Management *Source*: http://wiki.servicenow.com/index.php?oldid=250489 *Contributors*: Davida.hughes, G.yedwab, Guy.yedwab, John.ramos, Joseph.messerschmidt

Managing a Release Process *Source*: http://wiki.servicenow.com/index.php?oldid=250698 *Contributors*: David.Bailey, Davida.hughes, G.yedwab, Guy.yedwab, John.ramos, Joseph.messerschmidt

Customizing the Release Process *Source*: http://wiki.servicenow.com/index.php?oldid=240468 *Contributors*: Davida.hughes, Emily.partridge, Fuji.publishing.user, G.yedwab, Guy.yedwab, Joseph.messerschmidt

Moving from Release v1 to Release v2 *Source*: http://wiki.servicenow.com/index.php?oldid=130260 *Contributors*: Davida.hughes, Emily.partridge, G.yedwab, Guy.yedwab, Joseph.messerschmidt

Image Sources, Licenses and Contributors

Image: Rm-relationships.png *Source*: http://wiki.servicenow.com/index.php?title=File:Rm-relationships.png *License*: unknown *Contributors*: G.yedwab

Image:Rm-application.png *Source*: http://wiki.servicenow.com/index.php?title=File:Rm-application.png *License*: unknown *Contributors*: G.yedwab

Image:Warning.gif *Source*: http://wiki.servicenow.com/index.php?title=File:Warning.gif *License*: unknown *Contributors*: CapaJC

Image:Release Process.png *Source*: http://wiki.servicenow.com/index.php?title=File:Release_Process.png *License*: unknown *Contributors*: G.yedwab

Image:Plugin.gif *Source*: http://wiki.servicenow.com/index.php?title=File:Plugin.gif *License*: unknown *Contributors*: CapaJC, Joseph.messerschmidt

Image:Rm2-concepts.png *Source*: http://wiki.servicenow.com/index.php?title=File:Rm2-concepts.png *License*: unknown *Contributors*: G.yedwab, Guy.yedwab

Image:Rm2-heirarchy.png *Source*: http://wiki.servicenow.com/index.php?title=File:Rm2-heirarchy.png *License*: unknown *Contributors*: G.yedwab

Image:Workflow Display.png *Source*: http://wiki.servicenow.com/index.php?title=File:Workflow_Display.png *License*: unknown *Contributors*: G.yedwab

Image:Rm2-gantt.png *Source*: http://wiki.servicenow.com/index.php?title=File:Rm2-gantt.png *License*: unknown *Contributors*: G.yedwab