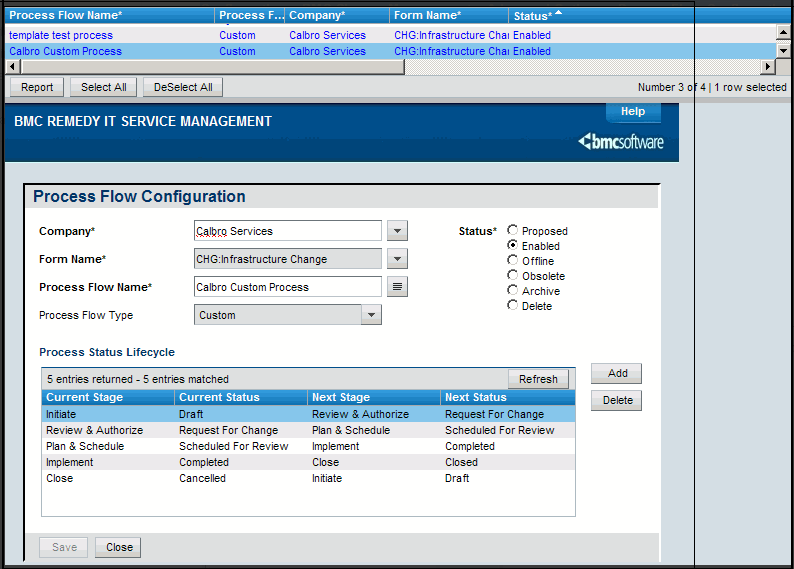
**Quick start to approvals in BMC Change Management**

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Action** | **Menu item** | **Comments** |
| [1](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step2) | Review the custom change management process defined for BMC Change Management, if any. | **Custom Configuration** >**Change Management** >**Advanced Options** >**Process Flow Configuration** | For more information, see [Configuring custom process flows](https://docs.bmc.com/docs/display/change81/Configuring+custom+process+flows). |
| [2](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step3) | Review the approval process configuration records that are installed by default. | **Custom Configuration** >**Foundation** > **Advanced Options** > **Approval Process Configuration** | For more information, see   * [Setting up approval process configuration records for your company](https://docs.bmc.com/docs/display/change81/Setting+up+approval+process+configuration+records+for+your+company) * [BMC Change Management approval state transitions](https://docs.bmc.com/docs/display/change81/BMC+Change+Management+approval+state+transitions) For detailed information about defining additional approval processes and rules in the BMC Remedy Approval Server, see [Setting up the approval process](https://docs.bmc.com/docs/display/ars8000/Setting+up+the+approval+process) in the BMC Remedy AR System documentation. |
| [3](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step4) | Review the states in the change request that drive the approval process. |  | For more information, see [Approval states on the Status Flow tab](https://docs.bmc.com/docs/display/change81/Approval+states+on+the+Status+Flow+tab). |
| [4](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step5) | Map approvers to the global approval process configuration records. | **Custom Configuration** >**Change Management** >**Approval** > **Approval Mappings** | For more information, see   * [Approval mappings](https://docs.bmc.com/docs/display/change81/Approval+mappings) * [BMC Change Management approval mappings](https://docs.bmc.com/docs/display/change81/BMC+Change+Management+approval+mappings) * [Release Management approval mappings](https://docs.bmc.com/docs/display/change81/Release+Management+approval+mappings) |
| [5](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step6) | Create a change or release request, and move it through its various approval phases. |  | For more information, see [Working with change requests](https://docs.bmc.com/docs/display/change81/Working+with+change+requests) or [Using Release Management](https://docs.bmc.com/docs/display/change81/Using+Release+Management). |
| [6](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step7) | Approve the change or release request. |  | For more information, see   * [Approvals - Approving change requests](https://docs.bmc.com/docs/display/change81/Approvals+-+Approving+change+requests) * [Managing release management approvals](https://docs.bmc.com/docs/display/change81/Managing+release+management+approvals) |
| [7](https://docs.bmc.com/docs/display/public/change81/Configuring+approvals+-+Quick+start#Configuringapprovals-Quickstart-step8) | Review the status of the change or release request after it is approved. |  |  |

To configure BMC Change Management approvals

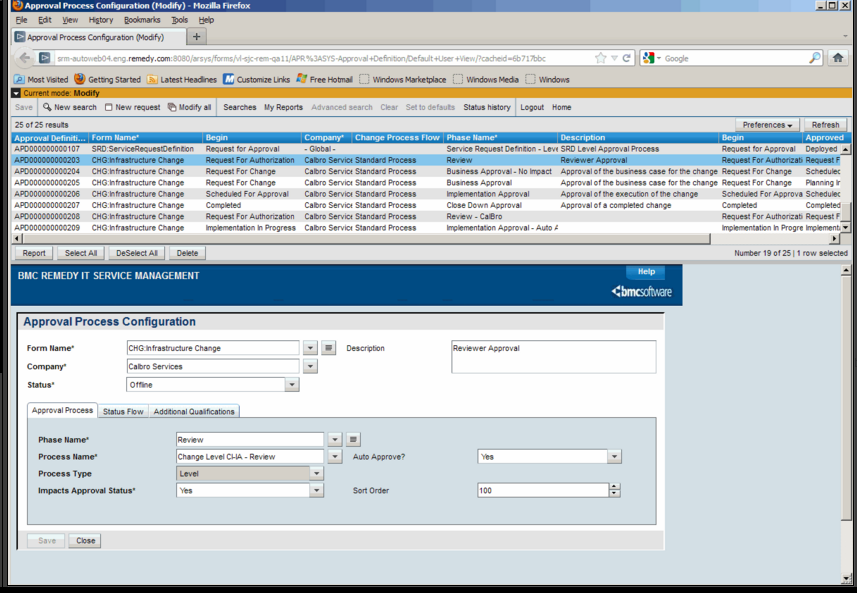
1. Review the change management process flow. In the Application Administration Console, select **Custom Configuration** > **BMC Change Management** > **Advanced Options** > **Process Flow Configuration**.   
     
   **Custom Process Flow**

Click the following figure to expand it.



2.Review the approval process configuration records that are installed by default. In the Application Administration Console, select **Custom Configuration > Foundation > Advanced Options > Approval Process Configuration**.   
  
**Reviewing the approval process configuration records**

Click the following figure to expand it.

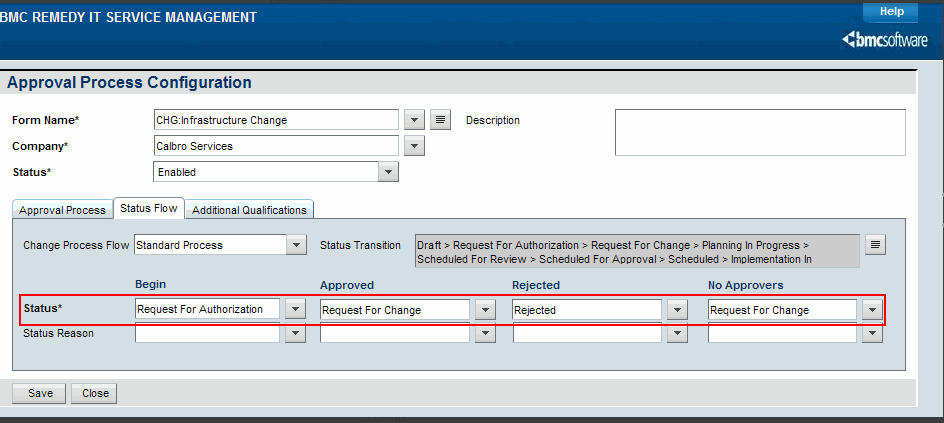


**Note**

BMC does not recommend that you modify the **Global** company record in this form. However, you can create company specific records for the approval process flow.

3 Click the **Status Flow** tab in the approval process configuration records to review the change states that drive the approval process.   
  
**Reviewing the status flow of the approval process configuration record**

Click the following figure to expand it.

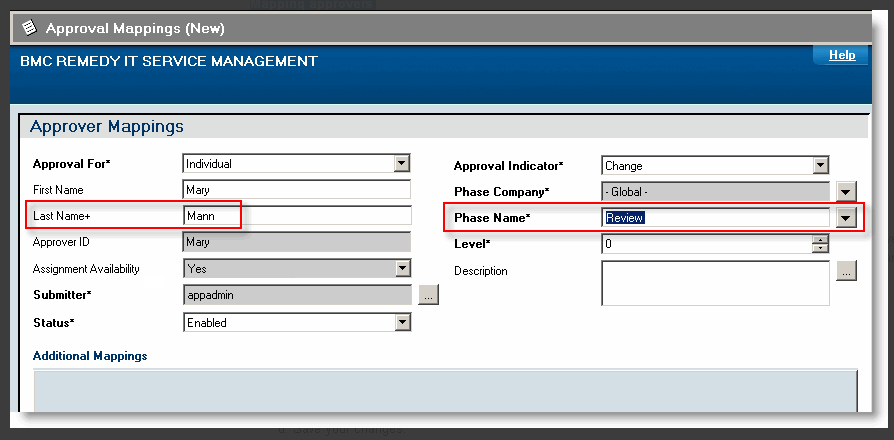


The Change Process Flow drop-down displays the configured change process flows. It displays the standard process and custom processes created. The contents in the **Begin**, **Approved**, **Rejected**, and **No approvers** fields are displayed based on the Change process selected. This is a mandatory field.   
  
**Approval and change request states**

| **Approval state field** | **Change request status** | **Approval Server action to the change request** |
| --- | --- | --- |
| **Begin** | Request For Authorization | When the Status of the change request is Request For Authorization, the approval process starts. |
| **Approved** | Request For Change | If the change request is approved, its Status becomes Request For Change. |
| **Rejected** | Rejected | If the change request is rejected, its Status changes to Rejected. |
| **No Approvers** | Request for Change | If there are no approvers defined for the change (that is, if no approvers are mapped to this phase), its Status changes to Request For Change. |

4 Map your approvers to the approval process configuration records. In the Application Administration Console, select**Custom Configuration** **> Change Management** **> Approval** **> Approval Mappings**.   
  
**Mapping approvers**

Click the following figure to expand it.



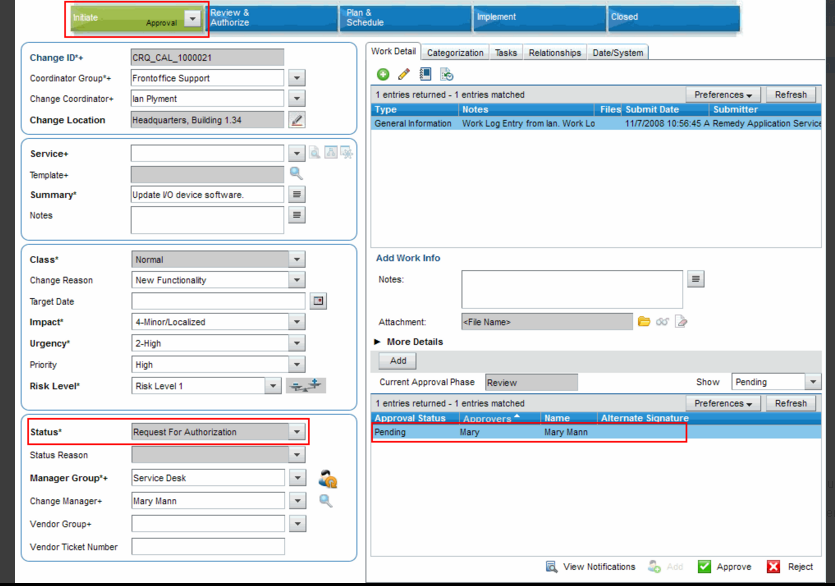
1. **Note**

Enter only the minimum values (shown in the red areas in the figure) that are needed to create the approver mappings. To enter values in the Additional Mappings area, *carefully* read [Approval mappings](https://docs.bmc.com/docs/display/change81/Approval+mappings) to understand the purposes of these fields and how they are used.

* 1. Enter Mann in the **Last Name** field and then press **Enter**.  
     Mary Mann is part of the Calbro Service sample data.
  2. Select **Global > Review** as the Phase Name.
  3. Ignore the remaining fields on the Approver Mappings form.
  4. Save your changes.
     + If you correctly mapped the approvers to the approval process configuration record, you will see the approver in the **Approvers** tab when the change request reaches the Request For Authorization status, as shown in the figure in step 6.
     + If you correctly mapped the approvers to the approval process configuration record, you will see the approver in the **Approvers** tab when the change request reaches the Request For Authorization status.

5 Create a change request, and then use the Process Flow Status wizard to work the change request through its approval phases.   
  
**Viewing approval information in a change request**

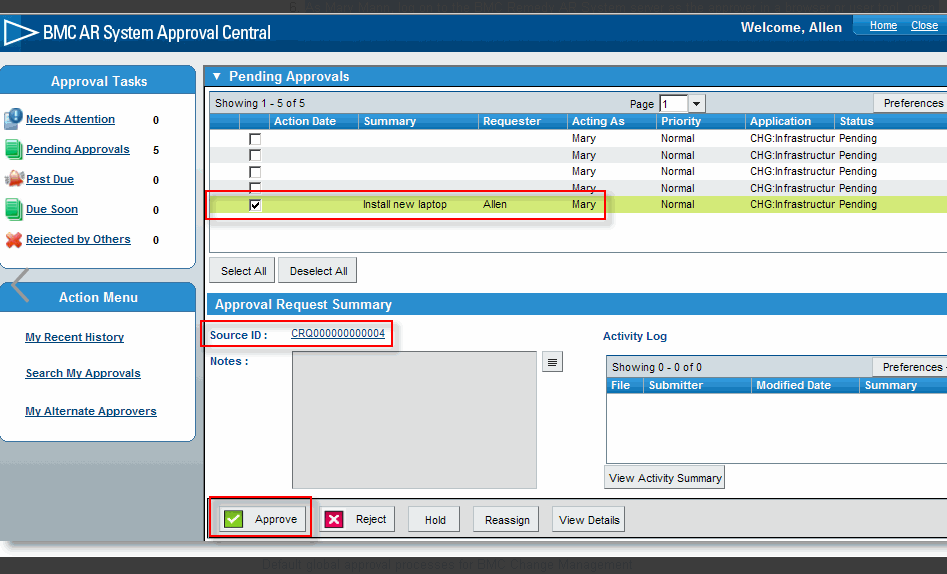
Click the following figure to expand it.



* 1. Log on to BMC Change Management.
  2. Create a change request and assign it to a specific support group in your company.
  3. Use the Process Flow Status bar to move the change request to the Request For Authorization status.

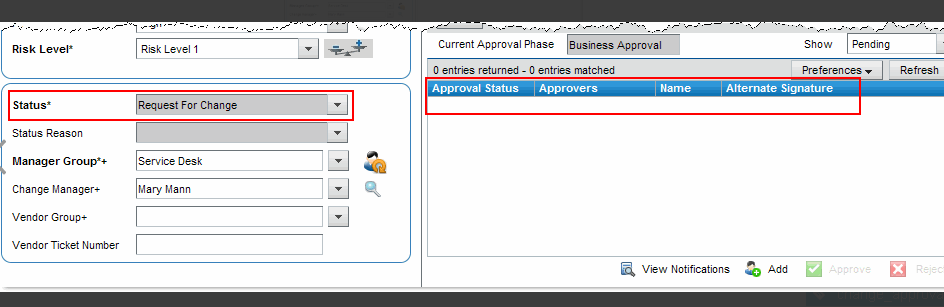
6 As Mary Mann, log on to the BMC Remedy AR System server as the approver in a browser or user tool, open Approval Central from the IT Home page, and then approve the change request.   
  
**Approving the change request in Approval Central**

Click the following figure to expand it.



7 Return to the change request, and then refresh it.   
  
**Flow of change request after approval**

Click the following figure to expand it.



# How approvals work with change requests

The change manager or change coordinator controls the overall progression of a change request, and can perform approvals at several points in the change lifecycle.

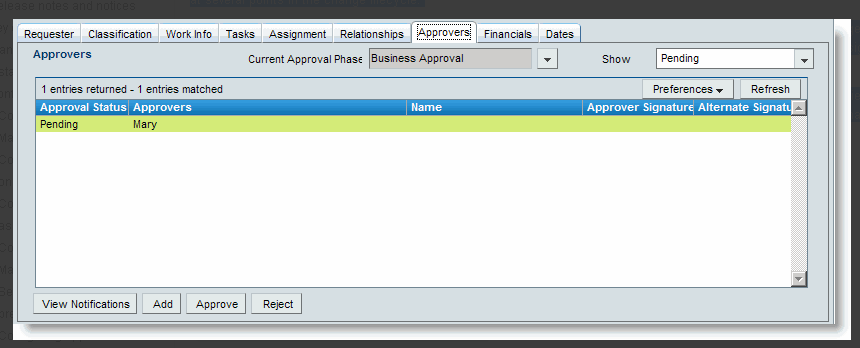
For change requests that have a status of Request for Change with no approvers, the change manager or the change coordinator can move the change forward to Planning In Progress, cancel it, or send it back to the requester by assigning it a status of Draft.

After a change is planned and a change request is built, the change request is set to a status of Scheduled for Review. The change manager or change coordinator can again review change plans, schedules, and so on, before moving the change to the Implementation Approval phase.

The following figure shows the **Approvers** tab on the Change form:

**Change request — Approvers tab**

Click the following figure to expand it.



The following options are available to control the functions that involve approvers:

| **Option** | **Description** |
| --- | --- |
| **Approve** | Allows defined approvers to approve the change. A change manager or change coordinator can approve the change on behalf of the selected approver, if they are defined as alternates.   This button is enabled only for users who are valid approvers for the change request. |
| **Reject** | Allows defined approvers to reject the change. A change manager or change coordinator can reject the change on behalf of the selected approver, if the change manager or change coordinator are defined as alternates.   This option is enabled only for users who are valid approvers for the change request. |
| **Add** | Allows you to assign a new individual or group to approve the change. |
| **View Notifications** | Opens the APR:Non-ApprovalNotifications form so that you can view notifications in either Pending or Notified status. The APR:Non-ApprovalNotifications form enables you to specify a person or group to be notified when a change reaches a particular approval phase, even though that person or group does not need to approve the change. |

The read-only **Current Approval Phase** field displays what approval phase the change is in.

# Configuring approval mappings

**Example**

You set up the approval mapping with the following values:

* **Approval for** - Individual, **First Name** - Allen, **Last Name** - Allbrook, **Approval Indicator** - Change, **Phase Name** - Review
* On the **Impacted Area/CI** tab: **Company** - Calbro Services, **Organization** - Information Technology,**Department** - Application Development

When the change reaches the Review approval phase, Allen Allbrook is listed in the Approvers table of the change only when one of the Impacted areas listed for the Change Request matches the following field values:

* Company = Calbro Services
* Organization = Information Technology
* Department = Application Development

The mapping is not applied if *any* of these values are different.

Configuring approval mappings

1. From the Application Administration console, click the Custom Configuration tab.
2. From the Application Settings list, select **Approval > Approval Mappings** of the application for which you want to configure approvals.
3. Click **Open**.
4. On the Approver Mappings form, select the appropriate value from **Approval Indicator**. For example for BMC Change Management, select **Change**.  
   The form refreshes, and the appropriate tabs and fields for the selected approval indicator are displayed.

**Note**

Select the criteria that should be met to map an approver to the approval phase. The approver is mapped to the selected phase only when *ALL* the defined criteria are met.

1. For **Phase Name**, select which approval phase needs an approval mapping in the application.   
   Approval phases are used to define when the approval is required during the lifecycle of the application request and maps the **Approval Process Name** that will be used by the Approval Server. The mapping between the application status, **Phase Name**, and **Approval Process** is done in the Approval Process Configuration form.  
   For example, Review phase for BMC Change Management indicates that the change will be sent for approval when it reaches the **Request for Authorization** status and will use the **Change Level CI-IA - Review**process.

**Note**

When you select a phase name, the **Phase Company** is automatically populated.

**Recommendation**

Select a user who has access to the company selected in **Phase Company**.

1. Enter information in the other fields as appropriate.

| **Field name** | **Description** |
| --- | --- |
| **Approval For** | The options are as follows:   * + **Individual** — If you select this option, enter the individual's last name and press the **Enter** key. The Approver ID (or logon name) is automatically populated.   + **Group** — If you select this option, the **First Name**, **Last Name**, and**Approver ID** fields are replaced by **Support Company**, **Support Organization**, and **Support Group Name**.  Select values for each field. The choices available in the **Support Organization** menu list depend on the option that you select for **Support Company**, and the choices available in the **Support Group** menu list depend on the option that you select for **Support Organization**.   + **Inherent Group/Role** — If you select this option, the **Inherent Group** and**Functional Role** fields are displayed.   **Note:** If you select the **Group** option, all the users who belong to the support group and have the corresponding approver functional role (for example, Infrastructure Change Approver for BMC Change Management) are set up as approvers. By default, only one approver in the group must sign off on the approval before it can move to the next stage. |
| **First Name** | If you select **Approval For Individual**, this field appears. Enter the first name of the approver. |
| **Last Name** | If you select **Approval For Individual**, this field appears. Use auto-fill to enter the approver's last name. |
| **Approver ID** | If you select **Approval For Individual**, this read-only field appears. The Login ID of the approver is automatically included when you select the approver. |
| **Assignment Availability** | This is a read-only field and is auto-populated and synchronized automatically.   * + For individuals the value displayed based on the Profile status on the People form. If the value on the People form is Enabled, this value is set to **Yes** to specify that the individual is available for the approval.   + For Support Group mapping, this value is set to **Yes** if the support group has at least one valid user with an Approval role (Infrastructure Change Approver for BMC Change Management). |
| **Submitter** | This field is auto-populated with the logon name of the person who is creating this approver mapping. |
| **Status** | Indicates the current status of the group or individual approving the request.   For the approver mapping to be available, select a status of **Enabled.** To disable the approval mapping, select a status of **Offline**. All other status options are for informational purposes only. |
| **Level** | Specify the level of approval for the individual or group. The level number defines the routing sequence within the approval hierarchy. You enter a number to determine the order in which the approvers are defined.   Approvers who are assigned lower numbers approve the requests before those who have higher numbers. |
| **Description** | Enter a description for this approver mapping. |
| **Support Company** | If you selected **Approval For Group**, enter the support company.   **Note:** If you select Change Implementer Group, the approval mapping is valid only until no tasks added to the change request. The Change Implementer Group is applicable only in the Classic view. |
| **Support Organization** | If you selected **Approval For Group**, enter the support organization. |
| **Support Group Name** | If you selected **Approval For Group**, enter the support group name. |
| **Inherent Group** | If you selected **Approval For Inherent Group/Role**, enter the inherent group name (for example, Change Assigned Group). |
| **Functional Role** | If you selected **Approval For Inherent Group/Role**, enter the functional role (for example, Infrastructure Change Approver). |

1. To specify the location, on the **Impacted Area/CI** tab, specify the **Company**, **Region**, **Site Group**, and **Site** fields. If required, specify the **Organization** and **Department** for the impacted area.   
   This maps the approval process when the location matches the change location specified for the **Requester** of the change request. For an example on adding IA information to approval mapping, see [Approvals generated based on Impacted Areas](https://docs.bmc.com/docs/display/change81/Approvals+generated+based+on+Impacted+Areas).
2. To specify the CI, on the **Impacted Area/CI**tab, specify whether the CI is selected by type or by name.
   * If you select **On CI Type**, select a specific CI type from the menu.
   * If you select **On CI Name**, enter the name of a specific CI.
3. To map approvers based on additional criteria, complete the fields on the additional tabs displayed. The tabs and fields displayed on each tab displayed depends on the **Approval Indicator** selected. For example, if you select**Change**, the **Categorization**, **Role**, and **Advanced Criteria** tabs are displayed.

For **Change**, you can map approvers based on the Change Implementer role *only* if you are still using the Classic View for BMC Change Management. This selection is not valid if you are using the Best Practice view.

For more information about the BMC Change Management views, see [Best Practice and Classic views](https://docs.bmc.com/docs/display/change81/Best+Practice+and+Classic+views).

The approval mapping now applies when values specified on a record exactly match the values selected in the approval mapping.

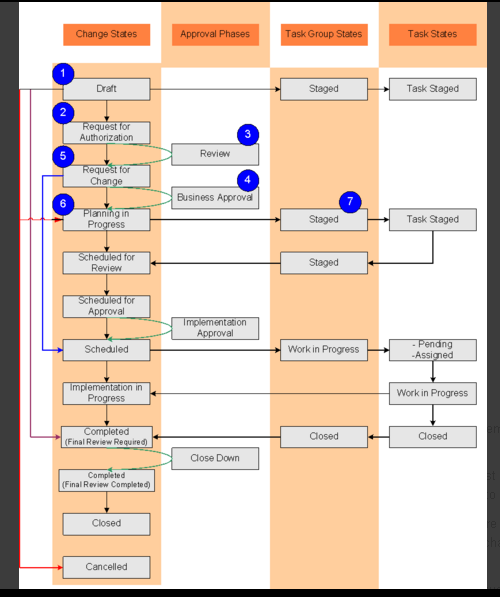
# Change request state transitions - Draft to Closed

You can use the following change request status transition information to follow the flow of a change request from Draft to Closed status.

**Note**

The Change Implementer is applicable only when using the Classic View.

1. A change request starts in the Draft status. Any related task groups or tasks are in Staged status.



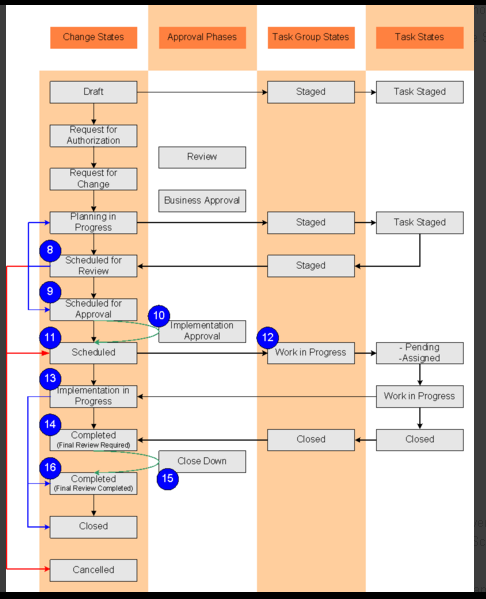
1. When the request moves through the process flow:
   * If approvers are mapped to the Review approval phase, the request remains in the Request For Authorization status.
   * If no approvers are mapped to the Review approval phase, the request moves to the Request for Change status and requires the change manager or the change coordinator to move it forward.
   * If no approvers are mapped to any of the approval phases and you are using only the standard approvals, the request moves to the Planning In Progress status and requires the change manager or the change coordinator to move it forward.
   * If the Timing of the request is Latent, it moves to the Completed status.
2. The Review approval phase requires that the request be approved before it can move forward. If the request is approved, it moves to the Request for Change status.
3. If no approvers are mapped to the Business Approval phase, the request moves to the Planning In Progress status and requires the change manager or the change coordinator to move it forward.  
   By default, No Impact changes follow the Business Approval - No Impact phase and move forward to the Scheduled status.
4. The Business Approval phase requires that the request be approved before it can move forward.
   * If the request is approved, it moves to the Planning In Progress status.
   * If the request is canceled, it moves to the Cancelled status.
5. When the change moves to the Planning in Progress status, the status reason of all Staged tasks is set to Staging in Progress.

**Note**

This is not applicable when Task Phase Management is enabled. For state transitions of the change with Task Phase Management enabled, see [Change status transitions - Draft to complete with task phase management enabled](https://docs.bmc.com/docs/display/change81/Change+status+transitions+-+Draft+to+complete+with+task+phase+management+enabled).

1. When the change manager or the change coordinator add dates to the request, or when the Status Reason of all the tasks has been marked as Staging Complete, the request moves to the Scheduled for Review status. If the Scheduled for Review status is bypassed in the change rules, the change moves to the next valid change status.
2. The change manager or the change coordinator can move the request to the Schedule for Approval status, cancel it, or move it back to the Planning In Progress status.

**State transitions - Steps 8 to 16**

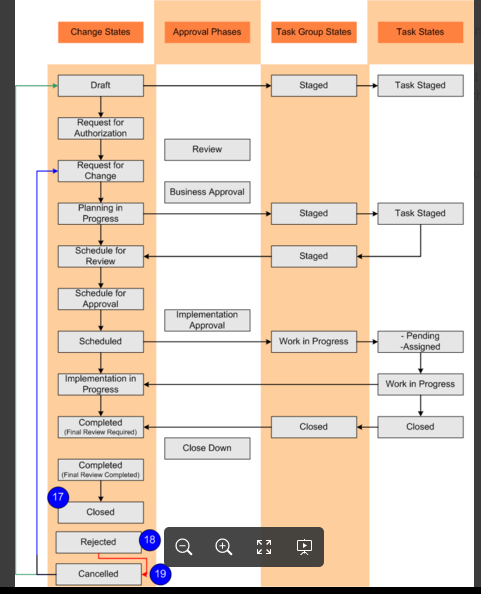


1. The request remains in the Scheduled for Approval status if there are approvers in the Implementation Approval phase. If no approvers are mapped to the phase, the request moves to the Scheduled status and requires the change manager or the change coordinator to move it forward.
2. If the request is approved, it moves to the Scheduled status. The change manager and the change coordinator can approve the request on behalf of approvers, if they are defined as alternates.
3. The change manager or the change coordinator must move the request to the Implementation in Progress status. For taskless changes only, the change implementer is notified.
4. When the change manager or the change coordinator move the request into the Implementation in Progress status, the task group is set to Work in Progress and the first task is set to Pending or Assigned.
5. Task implementers can start work on tasks. When all tasks are marked Closed, the change manager or the change coordinator enters information into Actual Start Date, Actual End Date, and Performance Rating fields to move the request to the Completed status.  
   If the Status Reason is set to Final Review Complete, the request moves to Completed status. If the Status Reason is set to Final Review Required, the request moves to Close Down Approval phase.
6. After the last task is marked as Closed, the task group that it belongs to is also set to Closed.   
     
   When all tasks are completed, the change request automatically moves to the Completed status with a status reason of Final Review Required. This happens regardless of whether you have configured an approval phase for the Completed status with a status reason of Final Review Required.  
     
   However, as of Service Pack 1 for version 8.1.00, the status reason is automatically selected, based on the approval process configuration:
   * If you have configured an approval phase for the Completed status, the status reason that you configured for the approval phase is selected. For example, the default status reason in the Close Down Approval phase is Final Review Required.
   * If you have not configured an approval phase for the Completed status, the status reason is set to Final Review Complete, by default.
7. The change manager is notified when final review is completed. The request moves to Completed (Final Review Complete) status if all tasks are completed or if there are *no*approvers for the Close Down Approval phase. The Requested For user (Requester tab) is then notified.

**Note**

At this stage, the change manager can change the status of the request back to Planning in Progress or Pending, if required.

**State transitions - Steps 17 to 19**



1. After the final review is completed, the Change Manager or the Change Coordinator closes the change request.

However, as of Service Pack 1 for version 8.1.00, if you enable the Auto-Close feature, the change request is closed automatically based on the [Auto-Close Days rule](https://docs.bmc.com/docs/display/change81/Configuring+change+rules) that you can configure.  
  
After the change is closed, users with Change Master permissions can update information within the change record. They cannot however update the following fields:

* + Class
  + Status
  + Any date fields  
      
    They also cannot add new tasks or ad hoc approvers.

**Note**

After a change request is closed, it cannot be reopened.

1. If a change has been rejected:
   * It can be resumed. When the change is resumed, it moves back to the approval status in which it was rejected.
   * It can be moved to Cancelled status. It can then be moved back to the Draft status to start the process over.
2. If change request is cancelled, the Requested For user is notified.
   * If a change request is cancelled, all the task groups and tasks associated with the request are also cancelled.
   * If a task group is cancelled, all its tasks are cancelled.

# Enabling email-based approvals

You can use configuration options to enable or disable email-based approvals for companies. Out of the box, email-based approvals for the Global company record are enabled.

**Note**

When you upgrade to version 8.1.00 of BMC Remedy IT Service Management, email-based approvals are disabled. You must manually enable email-based approvals after the upgrade.

## To enable or disable email-based approvals for a company

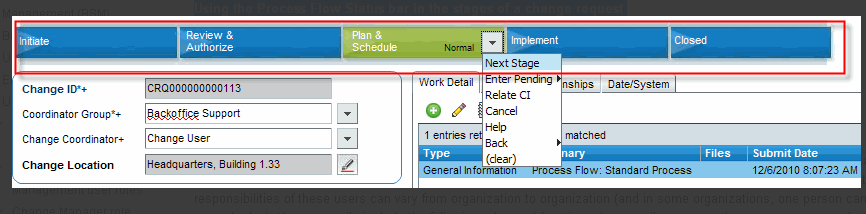
1. On the **Custom Configuration** tab of the Application Administration Console, choose **Foundation > Notification Engine > Approval Email Configuration**.
2. In the Configure Email Based Approval dialog box, select the name from the **Company** field.  
   To enable or disable email based approvals for all companies, select **Global** from this list.
3. In the **Status** field, select the appropriate status

# User roles in the change request lifecycle

BMC Change Management is a role-based application. What applications you can access, and how much information you can view or modify depends upon your role. As the Process Flow Status bar steps you through the stages of a change request, different roles perform different tasks. The change manager creates the change request at the Initiate stage, the task implementer works on the tasks that are assigned to the change at the Implement stage, and so on.

**Using the Process Flow Status bar in the stages of a change request**

Click the following figure to expand it.



A user with the Change Manager functional role can view functionality in the application that other users cannot.

Change requests track the progress of a change through its entire lifecycle, from the Initiation stage to the Closed stage. To manage a change request from start to finish, the main user roles listed in the following table are required. Although the responsibilities of these users can vary from organization to organization (and in some organizations, one person can fulfill several roles), they generally include the following roles and functions. In a small company, the same person can function as change manager, approver, and task implementer.

**IT Support user roles**

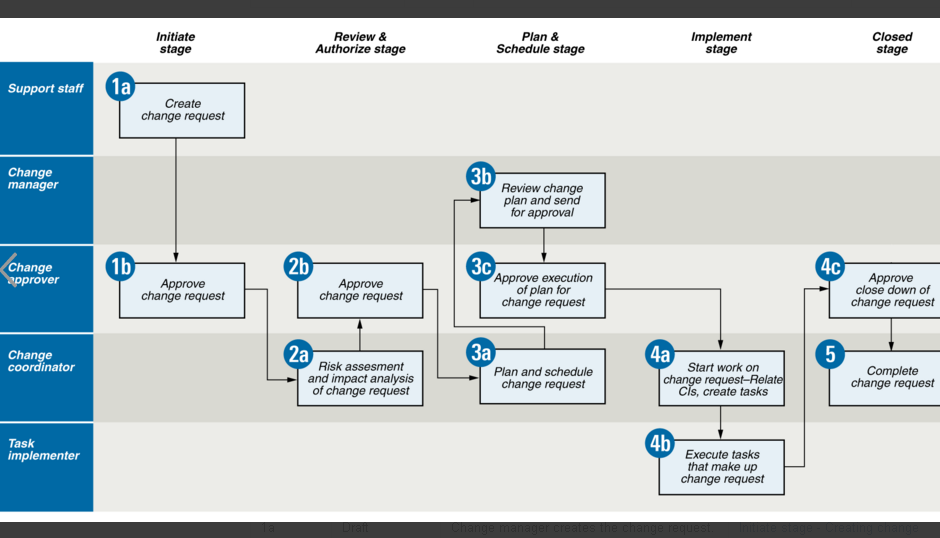
| **Change management role** | **Calbro user** | **Function** | **More information** |
| --- | --- | --- | --- |
| **Change management role** | **Calbro user** | **Function** | **More information** |
| Change coordinator   **Note:** This role is a combination of change supervisor and assignee roles of the previous release. | Allen Allbrook | Member of a support group responsible to provide one or more services. The change coordinator is responsible to create, plan, implement, track, and close changes related to services that the support group is responsible for.   * Assesses requests for change that originated from Incident Management, Problem Management, Release Management, or Continuity Management * Registers changes as needed to handle requests for change * Determines the risk & impact for requested changes * Prepares implementation plans by creating tasks * Monitors the progress of changes | For more information, see:   * [Change coordinator role](https://docs.bmc.com/docs/display/change81/Change+coordinator+role) * [Requester console users](https://docs.bmc.com/docs/display/change81/Requester+console+users) |
| Change manager | Mary Mann | Operations manager who has a complete overview of the infrastructure and must know details of what is to be changed.   * Reviews the risk & impact analysis to ensure that this has been performed thoroughly * Ensures that appropriate actions have been planned to minimize both the risk of failure and the impact on users during change implementations * Ensures that the timing of planned implementations does not conflict with other planned changes or events * Obtains approval for changes | [Change Manager role](https://docs.bmc.com/docs/display/change81/Change+Manager+role) |
| Task implementer   **Note:** This role is sometimes called the task assignee. | Ian Plyment | Support staff member or group who performs the tasks associated with a change request. For example, a change request for upgrading a mission-critical server might include backing up data in the server, uninstalling the old hard drive, and installing the new hard drive.   * Member of Front Office Support or Back Office Support * Uses the Change Management Console to perform the tasks associated with a change request | [Working with BMC Change Management as a task implementer](https://docs.bmc.com/docs/display/change81/Working+with+BMC+Change+Management+as+a+task+implementer) |

For more information about permissions and roles, see [User permissions](https://docs.bmc.com/docs/display/itsm80/User+permissions). For information about other roles important in BMC Change Management, see [Additional Change Management user roles](https://docs.bmc.com/docs/display/change81/Additional+Change+Management+user+roles).

The following figure illustrates the different Change Management support staff roles. It also shows where each role fits into the stages of the change request lifecycle.

**BMC Change Management support and management roles**

Click the following figure to expand it.



**Best practices for managing change requests**

| **Request stage** | **Request status** | **Role and task** | **For more information** |
| --- | --- | --- | --- |
| **Request stage** | **Request status** | **Role and task** | **For more information** |
| 1a  Initiate | Draft | Change manager creates the change request. | [Initiate stage - Creating change requests](https://docs.bmc.com/docs/display/change81/Initiate+stage+-+Creating+change+requests) |
| 1b  Initiate | Request For Authorization | Change approver approves the change request. | * [Approval processes provided out-of-the-box for Change Management](https://docs.bmc.com/docs/display/change81/Approval+processes+provided+out-of-the-box+for+BMC+Change+Management) * [Handling approvals for emergency change requests](https://docs.bmc.com/docs/display/change81/Handling+approvals+for+emergency+change+requests) |
| 2a  Review & Authorize | Not applicable | Change coordinator performs risk assessment and impact analysis of the change request. | [Review and Authorize stage - Risk and impact analysis](https://docs.bmc.com/docs/display/change81/Review+and+Authorize+stage+-+Risk+and+impact+analysis) |
| 2b  Review & Authorize | Request For Change | Change approver approves the business case for the change. | [Approval processes provided out-of-the-box for Change Management](https://docs.bmc.com/docs/display/change81/Approval+processes+provided+out-of-the-box+for+BMC+Change+Management) |
| 3a  Plan & Schedule | Planning In Progress | Change coordinator plans and schedules the details associated with the change.   1. Reviews change calendar. 2. Relates CIs to the change request. | [Plan and Schedule stage - Planning the change request](https://docs.bmc.com/docs/display/change81/Plan+and+Schedule+stage+-+Planning+the+change+request) |
| 3b  Plan & Schedule | Not applicable | Change manager reviews the change plan and sends it for approval. | [Approval processes provided out-of-the-box for Change Management](https://docs.bmc.com/docs/display/change81/Approval+processes+provided+out-of-the-box+for+BMC+Change+Management) |
| 3c  Plan & Schedule | Scheduled For Approval | Change approver approves the execution of the change before it can be scheduled. | [Approval processes provided out-of-the-box for Change Management](https://docs.bmc.com/docs/display/change81/Approval+processes+provided+out-of-the-box+for+BMC+Change+Management) |
| 4a  Implement | Implementation in Progress | Change coordinator starts work on the change request.   1. Relates CIs, incidents, and services to the change request. 2. Creates tasks. | [Implement stage - Implementing the change](https://docs.bmc.com/docs/display/change81/Implement+stage+-+Implementing+the+change) |
| 4b  Implement | Implementation in Progress | Task implementer executes tasks. | [Implement stage - Working on task assignments](https://docs.bmc.com/docs/display/change81/Implement+stage+-+Working+on+task+assignments) |
| 4c  Implement | Completed | Change approver approves that the request can now be closed. | [Approval processes provided out-of-the-box for Change Management](https://docs.bmc.com/docs/display/change81/Approval+processes+provided+out-of-the-box+for+BMC+Change+Management) |
| 5  Closed | Closed | Change coordinator closes the change request. | [Closed stage - Completing change requests](https://docs.bmc.com/docs/display/change81/Closed+stage+-+Completing+change+requests) |

**Note**

Except in the Closed status, the Change Manager and Change Coordinator can change the status of the request back to Pending or cancel the change request.

The primary objective of Change Management is to enable beneficial Changes to be made, with minimum disruption to IT Services.

Release Management are like air traffic controllers; they package together bundles of Change into a single Release to reduce periods of downtime and inconvenience to the rest of the business.

he primary objective of Release Management is to ensure that the integrity of the live environment is protected and that the correct components are released.

### ****Bringing out the big guns****

Let’s get back to basics and talk ITIL for a second. Both Change and Release Management sit in the Service Transition stage of the ITIL lifecycle so are part of the value stream that delivers effective business change;

Change; The addition, modification or removal of anything that could have an impact on IT services.

Release; Collection of hardware, software, documentation, processes or other components required to implement one or more approved changes to IT Services. The contents of each release are managed, tested and deployed as a single entity.

In other words, Change is about installing, modifying or retiring things safely without setting anything on fire. Release Management is a holistic process that bundles together multiple Changes into a single deployment. So now that we’ve got that sorted; let’s talk about how to make sure Change and Release Management play nicely together.

### ****Separate the roles of Change Manager and Release Manager****

Change Management is a governance process, the role of the Change Manager is to review, authorise and schedule the Change. Release Management is an installation process. It works with the support of Change Management to builds, tests and deploy new or updated services into the live environment. Both are equally important so you need subject matter experts for both.

### ****Agree the level of documentation required****

A Change is a single record containing:

* Dates
* Change description
* Approval details & audit trail

Release documentation is much more involved and as a starter for ten will contain**:**

* Scope
* Release details
* Implementation plan
* Back out plan
* Contact details
* Release note

### ****Ensure the Release Manager is present at CAB****

If your Release Manager isn’t attending CAB invite them immediately! It’s really important that the Release Manager is there to explain the Release content and any dependencies, communicate business approval and advise the Service Desk and Problem Management of any defects; working with them to ensure any known errors with any workaround details are raised where appropriate.

By having Change and Release Management working closely together, your effectiveness rates should improve and unforeseen incidents, problems and defects should be reduced. How do you manage Change and Release Management? Let us know in the comments!

# Creating a release request at the initiate milestone - Best Practice view

[Skip to end of banner](https://docs.bmc.com/docs/display/public/change81/Creating+a+release+request+at+the+initiate+milestone+-+Best+Practice++view#page-banner-end)

[Go to start of banner](https://docs.bmc.com/docs/display/public/change81/Creating+a+release+request+at+the+initiate+milestone+-+Best+Practice++view#page-banner-start)

[Skip to end of metadata](https://docs.bmc.com/docs/display/public/change81/Creating+a+release+request+at+the+initiate+milestone+-+Best+Practice++view#page-metadata-end)

* Created by [**Confluence Admin**](https://docs.bmc.com/docs/display/~admin), last modified by [**Kelly Holcomb**](https://docs.bmc.com/docs/display/~kholcomb) on [**Oct 30, 2012**](https://docs.bmc.com/docs/pages/diffpagesbyversion.action?pageId=233046397&selectedPageVersions=8&selectedPageVersions=9)

[Go to start of metadata](https://docs.bmc.com/docs/display/public/change81/Creating+a+release+request+at+the+initiate+milestone+-+Best+Practice++view#page-metadata-start)

The section describes how to create a release request at the Initiate stage when using the Best Practice view of Release Management.

## To create a release request at the Initiate milestone

1. Review the RFCs to make sure you understand their requirements.  
   This is a manual step you should perform before you open Release Management. You should understand the scheduled start and end dates, the CIs involved, the manifests necessary to complete the release, the CAB approvers, and so on.

**Recommendation**

Enter information into Release Management as soon as it is available to you. You can revise it at later milestones.

1. On the Release Management Console, click **Create** to open the Release form.  
   The **Release ID** field is automatically filled with an ID number for the release request.  
   In the Initiate stage, the release request initially appears in Draft status. The release request has not yet been submitted to the Release Management process.
2. Use the following fields to specify the release coordinator:
   * **Coordinator Group** — Specify the group of people with the Release Coordinator functional role. This list is populated with groups that have at least one user with a Release Coordinator functional role.
   * **Coordinator** — Specify the user responsible for the release. The list is populated with people with the Release Coordinator functional role and who are included in the Coordinator Group selected.
3. From the **Service** field, select the business service configuration item (CI) that relates to the release request that you are creating.  
   The **Service** field relates business service configuration items (CIs) to the release request at the time it is created. Business service CIs are related either to the customer directly or to the customer's company, organization, or department.

**Note**

The business service CI is not a physical CI (such as a printer or a router); it is a logical CI. In this context, a logical CI is a business service that can be provided from one business, or organization within a business, to another. Service CIs can include customer support, employee provisioning, web farms, storage, and so on. When business service CIs are created and made available on the **Service** field menu, they are related either to a customer directly or to the customer's company, organization, or department. If you need to have a new business service CI added to the **Service** field menu, you must notify a system administrator with Asset Administrator privileges.

1. (Optional) Select a template to complete part of the release request.  
   Release templates are especially useful in any release request that follows well-defined methods for specific and repeated requirements. Release templates do more than simply fill out fields for you; they can also include CIs and manifests to the release request. For more information, see [Selecting release templates](https://docs.bmc.com/docs/display/change81/Selecting+release+templates). The template is attached to the release request.

**Notes**

* + The release template can be applied in the Draft state, that is, while creating the release request.
  + If you start to enter fields in the release request and then select a release template, the release template overwrites any field values that are already present in the release request. Any relationships or manifests included with the release request are not overwritten. Any additional manifests from the template are added as peers, and additional relationships (for example, CIs) are included with the release request.

1. (Optional) The target date is the date by when the release must be completed, according to the applicable service level target. Alternatively, the target date can be a date that is agreed to on an ad hoc basis, per release.
2. Complete the following fields:

| **Field** | **Description** |
| --- | --- |
| **Summary** | Provide a brief description of the release. |
| **Business Justification** | Specify the business requirement for the request. Select an option based on the policies defined by your company. By default these values are for informational purpose only. You have the following options:   * + Corporate Strategic   + Business Unit Strategic   + Maintenance   + Defect   + Upgrade   + Enhancement   + Customer Commitment   + Sarbanes-Oxley |
| **Impact** | Specify the extent to which the release affects the business. The default value is 4-Minor/Localized. Impact is often directly related to the extent to which the service has degraded from agreed service levels. Impact can be measured by the number of people affected, the criticality of the system affected, and the loss of revenue as a result of the service degradation or disruption. |
| **Urgency** | Specify a value that indicates the importance of the release request, and reflects how quickly a release must be implemented, or the time available to reduce the impact of the release on the business. The default value is **Low**.   Use the following factors to determine Impact and Urgency:   * + Number of customers affected by associated Releases   + Duration and scope of the service disruption   + Availability of a solution or workaround   + The type of service being disrupted, usually based on the CI involved   + Awareness of the future impact on the business |
| **Priority** | (Optional) Specify the importance that you (as support staff) assign to the release request.   Priority indicates the relative order in which to address the releases. It is influenced by considerations of risk and resource availability, but is primarily driven by the combination of Urgency and Impact. The default value is **Low**. |

1. Select the Risk Level to indicate the relative risk associated with the release.  
   The default risk level is Level 1, which is the lowest level. The highest risk level is Level 5. The Risk Level is used as a criterion to determine required approvals.
2. Select the Release Type for the release request.  
   You use this field to further categorize the releases that are released into the IT infrastructure.

| **Option** | **Description** |
| --- | --- |
| **Option** | **Description** |
| Full | All components of the release are built, tested and deployed together. You typically use the Full release type if you want to make sure that the release or version of your application, plus the necessary CI components to run the application, are all linked throughout the entire process. |
| Delta | Includes only incremental releases and only those components or CIs that need to be included in this release. You typically use the Delta release type if your application was moving from version 1.0.00 to 1.1.00. |
| Package | All individual Full and Delta releases are grouped together and form a packaged release. You typically use the Package release type if you combine several minor and major updates together. |
| Backlog | Method to identify and review multiple changes, incidents, and problems that are candidates to target to implement in a release. The Backlog release type is a grouping mechanism that the CAB uses as their candidates to review for future releases. |

1. **Note**
2. Do not manually set the **Milestone**, **Status**, and **Status Reason** fields in the Release form.   
     
   When you use the Process Flow Status bar the release request to the next milestone, the value in the**Status** field automatically changes, based on the options you select from the Status bar menus.
3. Click **Save** to create the release request.  
   This is the minimum information needed to create the release.

# Recommendation for managing release requests

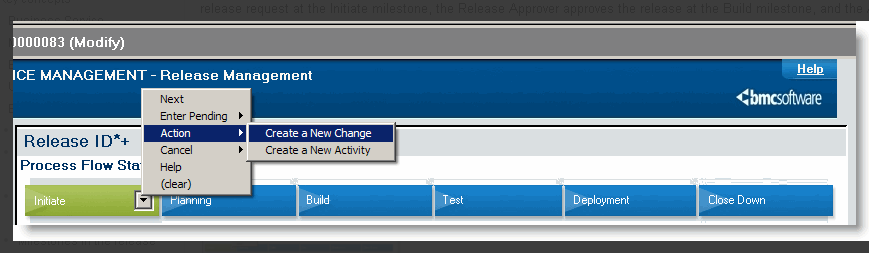
As with BMC Change Management, the Release Management module is role-based. What applications you can access, and how much information you can view or modify is dependent upon your role. As the Process Flow Status bar steps you through the milestones of a release request, different roles perform different tasks. The Release Coordinator creates the release request at the Initiate milestone, the Release Approver approves the release at the Build milestone, and the Activity Assignee completes the activity at the Deployment milestone.

**Note**

Defining approvals at the Build milestone is a recommendation. However, approvals can be defined at each milestone of the Release Management process.

**Using the Process Flow Status bar in the milestones of a release request**

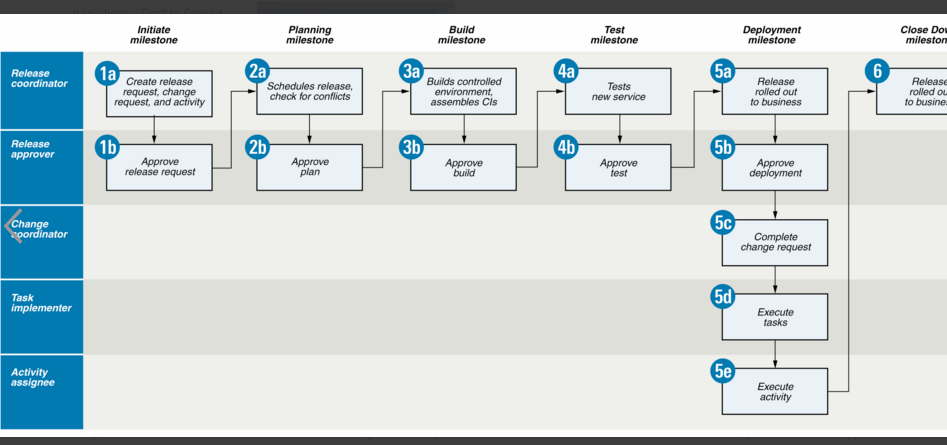
Click the following figure to expand it.



How each person uses Release Management depends on their role. The following figure illustrates the different Release Management support staff roles. It also shows where each role fits into the milestones of the release request lifecycle.

**Release Management support and management roles**

Click the following figure to expand it.



**Recommendation**

To simplify the release request process, the following table navigates you through the BMC Change Management documentation. It provides a map of BMC recommendation that you should follow, when you use the Release Management module.

**Note**

Approvals might not apply to all milestones in a release request. For example, you see a Build approval if the application administrator mapped an approver to the Build approval phase.

**Recommendation for managing release requests**

| **Milestone** | **Role and task** | **For more information** |
| --- | --- | --- |
| **Milestone** | **Role and task** | **For more information** |
| 1a  Initiate | Release Coordinator creates the release request, change request, and activity. | [Initiate milestone - Creating release requests](https://docs.bmc.com/docs/display/change81/Initiate+milestone+-+Creating+release+requests) |
| 1b  Initiate | CAB approves the release request. |  |
| 2a  Planning | Release Coordinator plans release request, uses Change Calendar to check for conflicts.   1. Reviews Change Calendar. 2. Enter start and end dates. | [Planning milestone - Planning and scheduling the release request](https://docs.bmc.com/docs/display/change81/Planning+milestone+-+Planning+and+scheduling+the+release+request) |
| 2b  Planning | CAB approves the release request. |  |
| 3a  Build | Release Coordinator oversees the release builds. | [Build milestone - Building a controlled environment for the release](https://docs.bmc.com/docs/display/change81/Build+milestone+-+Building+a+controlled+environment+for+the++release) |
| 3b  Build | CAB approves the release request. |  |
| 4a  Test | Release Coordinator oversees the testing of the new service, to make sure CIs meet specifications and requirements. | [Test milestone - Testing the release](https://docs.bmc.com/docs/display/change81/Test+milestone+-+Testing+the+release) |
| 4b  Test | CAB approves the release request. |  |
| 5a  Deployment | Release Coordinator rolls release out to the business. Starts phased or non-phased deployment of release. |  |
| 5b  Deployment | CAB approves the release request. |  |
| 5c  Deployment | Change Manager executes change request in its Deployment phase. | [Deployment milestone - Rolling out the release to the business](https://docs.bmc.com/docs/display/change81/Deployment+milestone+-+Rolling+out+the+release+to+the++business) |
| 5d  Deployment | Task Implementer executes task in its Deployment phase. |  |
| 5e  Deployment | Activity Assignee executes task in its Deployment phase. |  |
| 6  Closed | Release Coordinator completes the change request. | [Close Down milestone - Completing release requests](https://docs.bmc.com/docs/display/change81/Close+Down+milestone+-+Completing+release+requests) |

The intentions behind the processes are where the main differences lie. While Change Management is intended to be used with most minor, individual and general changes, Release and Deployment Management is primarily aimed at being used with large sets of grouped changes (also known as a release). A release can include changes to hardware, software, documentation, processes and other components.

# Filter Phases for on submit, modify and merge

##### **An Overview**

·         Remedy-AR System Work Flow Objects -Introduction

·         Filter Phases for on Submit and On Merge Actions

·         Filter Phases for on Modify Action

·         Filter Phases for on Get Entry Action

·         Filter Phases for on Delete Action

#### ****Remedy-AR System Work Flow Objects -Introduction****

In AR System, form is an AR System object which is used to capture and display the information. The information captured in the forms will be processed by Remedy workflow objects before storing it in the database.

AR System has following three types of work flow objects to implement the Business Logic:

1. Active Links
2. Filters
3. Escalations

#### Active Links:

An active link is an action or a series of actions that are being triggered when a user performs an operation, based on the set of permissions assigned to that user. The action(s) is triggered on the AR System client (either web or Windows) in the current form window.

#### ****Filters:****

Filters are meant to implement and enforce business rules because they test every request transaction to see if certain conditions

are met, and then respond to the conditions by taking specific actions. These filters are executed for all the transactions occur in the Database. Filters will run from the Server. So Administrator permission is defined for the Filters.

#### ****Escalations:****

An escalation enables a condition to be checked on a regular basis or time bound basis and, depending on if and how it is met, performs one or more actions. Escalations differ from filters in that escalations occur at a specific point on a time interval rather than in response to a specific operation. Escalations will run from the Server. So Administrator permission is defined for the Filters.

Here we are going to discuss about the processing of Filter in the AR Server.

As we already discussed, Filter Object executes on the Server with Administrator permission. So whenever the following actions are made, the filter objects are executed:

* Submit
* Modify
* Delete
* Get entry
* Merge

If any one of the action is triggered for a Request, then all the filter for the corresponding execute on condition will be executed. While executing these filters, they will be executed based on the execution order. But the actions present may not be executed sequentially as Active link or in escalation actions.

The actions present in the Filters are classified under the following three categories:

* Phase 1
* Phase 2
* Phase 3

The following table shows actions that are processed under each phase:

|  |  |  |
| --- | --- | --- |
| Phase 1 | Phase 2 | Phase 3 |
| Call Guide  Exit Guide  GoTo  Go To Guide Label  Log to File  Message  Set Fields | Push Fields  Direct SQL | Notify  Run Process  Distributed Server Option |

This implementation is meant for ensuring that notifications are sent and that processes are run onlyafter the database operations are successful. If any database operation fails, all subsequent actions are suppressed, and database changes are rolled back. So, the system defers to a final phase the operations that should not run until database transactions have been committed and there is no chance that a rollback will occur.

The subsequent section will detail the order in which the filter actions are executed in more detail.

**Filter Phases for on Submit and On Merge Action**

When the On submit or On merge execute action is triggered, the filter actions are executed in the following procedure:

* First all the phase 1 actions in all the filters executed.
* A new entry in created in the Database with created record committed into the data base.
* All the Phase 2 Actions are executed.
* Then at last Phase 3 actions are executed.

Here if there are any errors in any one of the phase 2 actions, then all the phase 2 actions will be rolled back. Then the created record will be deleted from the data base.

If any phase 2 action triggers any more transactions such as CREATE or MODIFY subsequently, the phase 2 action will not be committed until all the phase 1 and phase 2 actions are completed. If there are any errors in these, the transaction is rolled back and created entry is deleted from the data base.

The same principle applies to On Merge Option also.

**Filter Phases for on Modify Action**

When the On Modify execute action is triggered, the filter actions are executed in the following procedure:

* First all the phase 1 actions in all the filters executed.
* All the Phase 2 Actions are executed.
* Entries are modified into the database(committed)
* Phase 3 actions are executed at last.

Here if there are any errors in any one of the phase 2 actions, then all the phase 2 actions will be rolled back.

If any phase 2 action triggers any CREATE transactions subsequently, the phase 2 action will not be committed until all the phase 1 and phase 2 actions are completed. If there are any errors in these, the transaction is rolled back and created entries are deleted from the data base.

**Filter Phases for on Get Entry Action**

When the On Get Entry execute action is triggered, the filter actions are executed in the following procedure:

* Data is Retrieved from the database
* All the Phase 1 and phase 3 Actions are executed.
* Phase 2 actions are executed at last.

Here if there are any errors in any one of the phase 2 actions, then all the phase 2 actions will be rolled back. But by the time the **Notification** would have been sent. So be careful in using this option.

For getEntry operations, there is generally no database change, so the actions are not be phased.

**Filter Phases for on Delete Action**

When the On Delete execute action is triggered, the filter actions are executed in the following procedure:

* All the Phase 1 and phase 3 Actions are executed.
* Phase 2 actions are executed at last.
* Request is deleted from the Database.

Here if there are any errors in any one of the phase 2 actions, then all the phase 2 actions will be rolled back. But by the time the **Notification** would have been sent. So be careful in using this option.

QUES-How to call filter using button click without using active link

You can create button using below ids:

Ans- 1001/1003. FIrst one will result in submit action and second one will result in Modify action.

And you can write a filter on Submit/modify to call your API

***for that it's quite simple. It's one filter with the following actions:***

1. Set Field from Form A (Phase 1)
2. Push Field to Form B (Phase 2)
3. Set Field (current transaction) (Phase 1)

If you enable the Error Handler filter on your first filter, no Retry will never Occur.

In that case and if error occurs, the following actions are executed (And you can't see that in the filter logs ):

1. Set Field from Form A (Phase 1)
2. Push Field to Form B (Phase 2) => Error
3. Error Handler action fire Set the status to "Retry". (Phase 1)
4. Set Field (current transaction) (Phase 1)

But in that case you can see the following in the filter logs. Have some fun...

1. Set Field from Form A (Phase 1)
2. Set Field (current transaction) (Phase 1)
3. Push Field to Form B (Phase 2) => Error
4. Error Handler action fire Set the status to "Retry". (Phase 1)

So for that kind of scenario, you must change the order of the actions of the first filter if you want the error handling works as expected.

1. Set Field from Form A (Phase 1)
2. Set Field (current transaction) (Phase 1)
3. Push Field to Form B (Phase 2)

Ques-Why we have don’t have activelink phases:

Ans-The phases as a workaround on the server (FLTR), not something that is missing from the client (ACTL).   
  
The phases is there because on a submit, you can and want to do certain things before you commit things to the database.   
  
For example, it would not be efficient to commit something to the database and then through an ERROR, followed by a rollback.   
  
Some things needs to be done after the ticket has been successfully stored, for example sending a notification email to the assignee, this is not something to do if the transaction fails, and we also need the request-id in the notification, which originally was not reserved until a later phase.   
  
What then about the ESCL? They do not have phases, because they are not committing anything to the database. They just search the database and performs actions on the tickets found. An ESCL can only update the database by a Set-Fields or Push-Fields, and in this case the different phases are processed exactly as a Push-Fields or Save from the client, i.e. they trigger submit/modify filters.   
  
The ACTL actually has phases of sorts. For example you have both the (Before) Submit and the After Submit actions. One is performed before sending the data to the server, and the other after the record has been committed to the database.   
  
Another way of looking at it is that all filters are performed within a single transaction to the database. If it fails, everything is rolled back (restored). Active Links on the other hand, and their actions, work one at a time. If an ACTL action fails, it stops ACTL processing, but nothing that happened before is rolled back.

Misi's notes are the best description of the situation.   
  
Escalations are searches and all work is done with Filters and Filters have phasing so Escalations have phasing.   
  
Active links have no phasing as they do not have the operations and side effect  issues (see below) that Filters can have.  Additionally, active links have no concept of a "transaction" so there is no rollback issue (see below).   
  
Filters have phasing as an API call to the AR System server is considered a single unit of work that either succeeds or fails.  And, when you have workflow firing, you want a consistent unit of work to occur.  So, phasing causes operations that are needed to prepare the data for the transaction to occur immediately while operations that you want to run only if the operation was successful run after the   
operation to verify it was successful.   
  
For example,   
 You don't want a notification sent out saying something happened if an error occurs and the operation really didn't happen   
 You don't want an independent process not used to return data to run if the  operation doesn't succeed as a retry would run that process again   
  
So, phasing is the way we defer the operations that you don't want to run when there is an error as if they ran "in-line" they would have happened and could not be called back on an error.   
  
In addition, we provided a mechanism to override the phasing and RUN NOW rather than being deferred if that is important to your logic.

# Ques- [Difference between ARsystem and ITSM](https://communities.bmc.com/message/314588#314588)

Ans-AR System is a platform where you install applications provided by BMC or other vendors or you can develop your own application. ITSM is an application developed by BMC which runs on AR System ( ITSM is desinged from AR System)

When you create/modify a workflow or an object, you need to flush the mid tier cache so that changes gets reflected to the mid tier.

# Ques-[Difference between Member and Associate Member](https://communities.bmc.com/message/171893#171893)

This is something for the identification purpose only.From the System point of view there is no such difference in these two Association Role.

However from the information point of view the person who is a member is actually the one who belongs to that group permanently.

And the Associate Member is the one who just need temporary access to this Support Group.

**"The relationship roles are Member and Associate Member. Member means that the person belongs to the support group. Associate Member means that the person has access to modify requests assigned to that support group. Also, for BMC Remedy Incident Management, if you are an associate member, you cannot own an incident ticket."**

Only users that belongs to support group with respective functional role and Relationship role as "member" can re-open the ticket from Closed status.

# Mapping Incident Management roles to permission groups

The following table maps Incident Management user roles to permission groups:

| **Role** | **Permission groups** |
| --- | --- |
| **Role** | **Permission groups** |
| Service Desk Analyst | * Incident Master * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge User |
| Specialist | * Incident User * Problem User * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge User |
| Group  Coordinator | * Incident User  **Note:** You can give a user Master permissions if full access to Incidents is required. * Problem User  **Note:** Give this individual Master permissions if they need full access to Problems, Known Errors or Solutions. * Infrastructure Change User * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
| Operator | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
| Operations  Manager | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
| On-Duty  Manager | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge User |

# Typical permissions combination setups

The following table is organized by application, role, and the permissions combination recommended for the role.

| **Application** | **Role** | **Permissions combination** |
| --- | --- | --- |

|  |  |  |
| --- | --- | --- |
| ncident Management | Service Desk Analyst | * Incident Master * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge User |
|  | Group Coordinator | * Incident User   Note: You can give a user Master permissions if full access to Incidents is required.   * Problem User   Note: Give this individual Master permissions if they need full access to Problems, Known Errors or Solutions.   * Infrastructure Change User * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
|  | Specialist | * Incident User * Problem User * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge User |
|  | On-Duty Manager | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge User |
|  | Operator | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
|  | Operations Manager | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
|  | Service Level Manager | * Incident User * Problem Viewer * Infrastructure Change Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |
|  | Service Owner | * Incident User * Problem Viewer * Infrastructure Change * Viewer * Release Viewer * Contract Viewer * Asset Viewer * Knowledge Viewer |

# Foundation objects

The Foundation contains the following objects, which are needed to support the applications:

* [Company](https://docs.bmc.com/docs/display/public/itsm81/Foundation+objects#Foundationobjects-Company)
* [People](https://docs.bmc.com/docs/display/public/itsm81/Foundation+objects#Foundationobjects-People)
* [Organization](https://docs.bmc.com/docs/display/public/itsm81/Foundation+objects#Foundationobjects-Organization)
* [Support groups](https://docs.bmc.com/docs/display/public/itsm81/Foundation+objects#Foundationobjects-Supportgroups)

### Multi-tenancy

Multi-tenancy defines who has access to what data on a row-level basis. For example, in a service provider environment a single application might be used by multiple companies, with the data for each company hidden from other companies using that application.

To provide a user with access to data for multiple companies, add more companies to the Access Restrictions list. If a user needs to access data for all companies, you can set the user's access to Unrestricted. After **field 112** and **field 60900** are populated, any query to BMC Remedy AR System shows only rows of data that a user has permission to see, based on their own permissions and the permissions in **field 112** and **field 60900**.

In BMC Remedy IT Service Management (BMC Remedy ITSM), multi-tenancy is defined using companies. Companies are defined as operating companies and vendor companies, and users are associated with these companies to define their access rights. A user is associated with a company through the People form.

# About floating licenses in a license pool

A license pool consists of a number of floating licenses reserved for a group, subject to the number of floating licenses available in the database. When a member of a group logs in, a license from the license pool for that group is granted. When the user finishes using the license, it is released back into the pool.   
  
If no licenses are available in the pool, a check is made to see if the user is a member of any other group that has a license pool. If no licenses are available in any pool the user is a member of, a check is made for floating licenses not associated with any pool. A user is never granted a floating license from a pool of which the user is not a member.   
  
License pools enable you to give priority to a group that needs licenses more urgently. The group with the smallest group ID has the highest priority. When a non-reserved floating license becomes available, it is granted to the next user who needs it, regardless of the priority of that user's access to the system.

# Groups in BMC Remedy AR System

Access control groups are collections of BMC Remedy AR System users. A user gains access to an object, a field, or a request if a group the user is in has access, or a role mapped to such a group has access. Notifications also can use groups. For example, you can designate an entire group to be notified in a filter action.   
  
BMC Remedy AR System includes a Public group and eight other special groups that are essential for access control within the system. You can define additional groups based on a common profile and assign access accordingly. For example, you might create a Sales group and allow members to view the status of a request but not to change it. A group can also be a general category, such as Browsers.