This section introduces the three AR System workflow objects: active links, filters, and escalations. With workflow, you can define a set of processes to enforce business rules such as tracking company benefits, defects, assets, and so on.

**Note:** In this guide, the term *button* refers to a button that you place in a form to execute an active link; the term *toolbar button* refers to a button in a toolbar, which, in the case of active links, is the toolbar in BMC Remedy User. A *menu bar item* refers to the items displayed from a top-level menu in BMC Remedy User. Do not confuse it with a menu attached to a field.

**Workflow objects**

Workflow objects automate your organization’s business processes. You can create active links, filters, and escalations to perform actions on one form or several. When workflow is attached to multiple forms, it is considered shared workflow. See “Shared workflow” on page 22 for more information. Active links, filters, and escalations share many similarities, but also have several differences that are described in this section. See the *Concepts* guide for background information about workflow.

Workflow objects automate your organization’s business processes. You can create active links, filters, and escalations to perform actions on one form or several forms. When workflow is attached to multiple forms, it is considered shared workflow. See “Shared workflow” on page 32 for more information.

Active links, filters, and escalations share many similarities, but also have several differences that are described in this section.

**Active links**

An **active link** is an action or a series of actions that are conditionally interpreted after being triggered when a user performs an operation, based on the set of permissions assigned to that user. The interpretation occurs on the AR System *client* in the current form window. For example, you might

define an active link that displays a list of all problems reported for the current workstation whenever a user presses ENTER in the Workstation field.

When an active link is executed from a form, you can trigger multiple active links with a single user action. Because you can also specify the conditions under which an active link executes, you can define an active link where a single user action leads to different active link results. For example, you can design an active link that checks the group membership of the name in the Submitter field and, based on the group membership, attaches a different menu to the Short Description field. This would enable you to provide one set of descriptive options to someone in Human Resources and a different set

of options to someone in Shipping. Active links can be grouped into execution groupings called active link guides that allow procedural processing (discussed in the *Form and Application*

*Objects* guide).

**Note:** Active links cannot be triggered through the use of an API program.

**About active links**

An active link is an action or a series of actions that are triggered when a user performs an operation and are conditionally interpreted. The interpretation occurs on the AR System *client* in the current form window, and run with the permission of the user. For example, you might define an active link that displays a list of all problems reported for the current workstation whenever a user presses Enter in

the Workstation field. When an active link is executed, it can trigger varying actions as the result of a

single user action. When you design an active link, you specify the conditions under which the active link executes, and further conditions to determine which action it will take. For example, an active link could attach one menu to the Short Description field if the user is a member of the Human Resources department, or else attach a different menu if the user is a member of the Shipping department.

Active links can be grouped into execution groupings called active link guides that allow procedural processing.

**Filters**

You can design **filters** to implement and enforce your organization’s business rules because filters test every request transaction to see if certain conditions are met, and then respond to the conditions by taking specific actions. For example, you might define a filter that notifies support staff members when

they are assigned responsibility for a new request. Filters can act on virtually any condition that arises in a request. For example, filters can restrict how users create or modify a request. Filters can also check

for conditions in requests that are submitted by a network management system for a device that the system is monitoring. Then the filter can automatically call a program to control that device.

Filters execute on the AR System *server* and run with administrator permissions. This means that filters can access any field in the AR System database, even if the field is not available to the user (no view or change access). Filters can be grouped into filter guides.

**About filters**

Filters implement and enforce your organization’s business rules. A filter tests every request transaction to see if certain conditions are met, and then responds to the conditions by taking specific actions. For example, a filter can notify support staff members when they are assigned responsibility for a new request.

Filters can act on virtually any condition that arises in a request. For example, filters can restrict how users create or modify a request. As another example, a filter can check for conditions in requests that are submitted by a network management system for a device that the system is monitoring. Then, the filter can automatically call a program to control that device. Filters execute on the AR System *server* and run with administrator permissions. This means that filters can access any field in the AR System database, even if the field is not available to the user (no view or change access). Filters can be grouped into filter guides to control the order of processing.

**Escalations**

An **escalation** enables a condition to be checked on a regular basis and, depending on if and how it is met, performs one or more actions. For example, you might define an escalation that sets the priority to Urgent if the request is not closed within 24 hours. Or you might choose to page a support staff member if a Critical request has not been addressed in one hour. Escalations execute on the AR System *server* and run with administrator permissions. Escalations differ from filters in that escalations occur at a specific point on a time interval rather than in response to a specific operation. Additionally, escalations are scheduled to act on requests that meet a certain qualification, while filters act on the current request

(transaction). At the times specified within the escalation, AR System searches for requests that match the escalation qualification, and performs the specified escalation actions on requests that match.

**About escalations**

An escalation causes a condition to be checked on a regular basis and, depending on whether and how the condition is met, performs one or more actions. For example, an escalation can set the priority of a request to Urgent if the request is not closed within 24 hours, or send a page to a support staff member if a critical request has not been addressed in one hour.

Escalations execute on the AR System *server* and run with administrator permissions. Unlike filters, which run in response to a specific operation, escalations run at a specific time or after a defined time interval. Also, when they run, escalations find and act on all requests that meet a qualification, while filters act on the current request if it meets a qualification.

At the time specified in the escalation, AR System searches for requests that match the escalation qualification and performs the specified escalation actions on those requests that match. Escalations can be assigned to pools so the escalations from each pool run in parallel on separate threads within the escalation queue. To use escalation pools, you must first configure multiple threads for the escalation queue as described in the *Configuration Guide*, “Queues,” page 27. If you assign an escalation to a pool

that has no thread configured, the escalation is run by the first thread. All escalations in a particular pool run on the same thread, so the execution of escalations within a pool is serialized. Escalations run in the order of their firing times, but an escalation is delayed if an escalation from the same pool is currently

running. If two or more escalations have dependencies and must not run at the same time, put them into the same pool to make sure they run in sequence.

**Workflow basics**

AR System workflow consists of active links, filters, and escalations that carry out business processes. For information about what these three main types of workflow, see “Workflow objects” on page 16.

All workflow objects include the following elements:

􀂄 **An associated form** is the basis for every workflow action. Sometimes a workflow object has more than one associated form, but one form is defined as the primary form and acts as the reference for fields and data used by the workflow. See “Configuring workflow forms and execution options” on page 35

and “Shared workflow” on page 32.

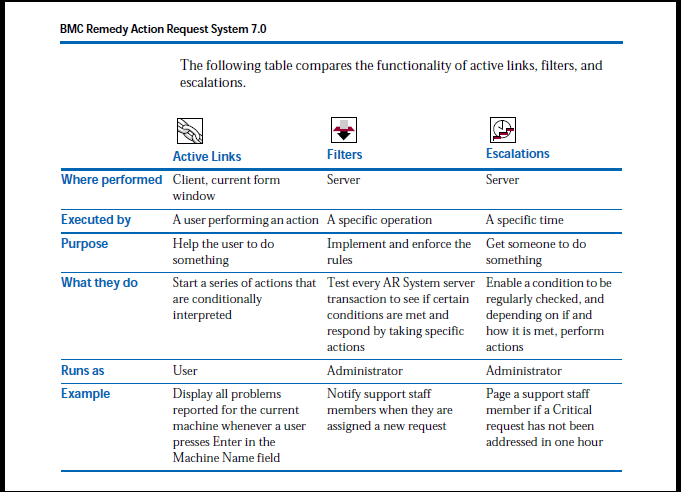
􀂄 **Workflow execution options** determine *when* the workflow runs. See“Configuring workflow forms and execution options” on page 35.

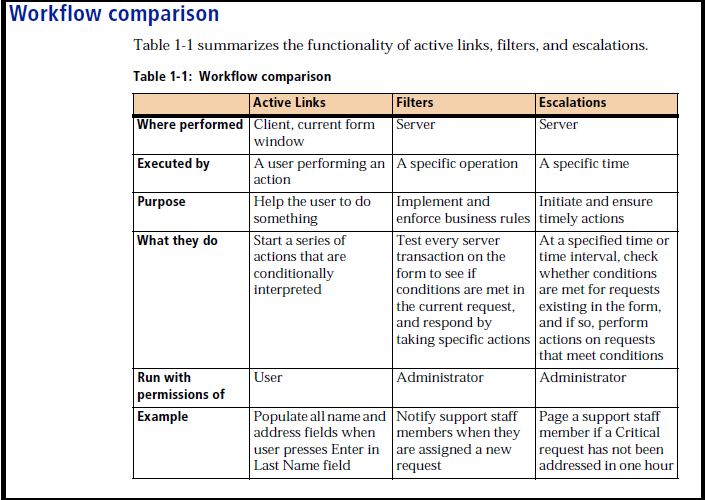
􀂄 **Run If qualifications** (optional) determine whether the workflow’s If Actions or Else Actions are carried out. See “Building qualifications and expressions” on page 49.

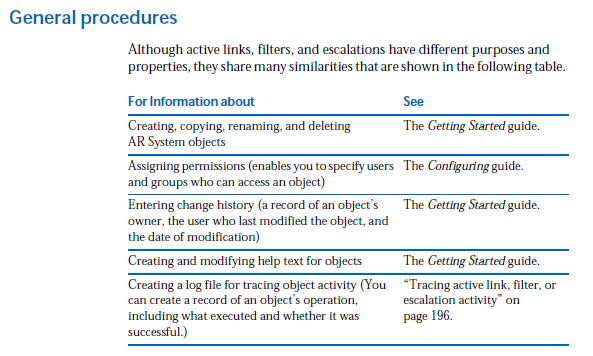
􀂄 **Workflow actions** determine *what* an active link, filter, or escalation does when

it runs. See “Specifying workflow actions” on page 63. You can use active link guides and filter guides to control the order of workflowactions and organize a related set of workflow objects. See “Defining guides and guide actions” on page 137.

Active links allow you to create workflow designed for user interaction. You can use buttons and field menus with active links to assist the user. See “Using buttons and menu bar items to execute active links” on page 165. For information about how AR System processes active links, filters, and escalations, see “Workflow processing” on page 175.







**Using the workflow windows**

You can open multiple windows for creating or modifying the active links, filters, and escalations that you have permission to administer. The title bar of each window tells you whether you are working in the Active Link, Filter, or Escalation window. Use the Active Link window, shown in the following figure, to create and modify active links.



Use the Filter window, shown in the following figure, to create and modify filters.

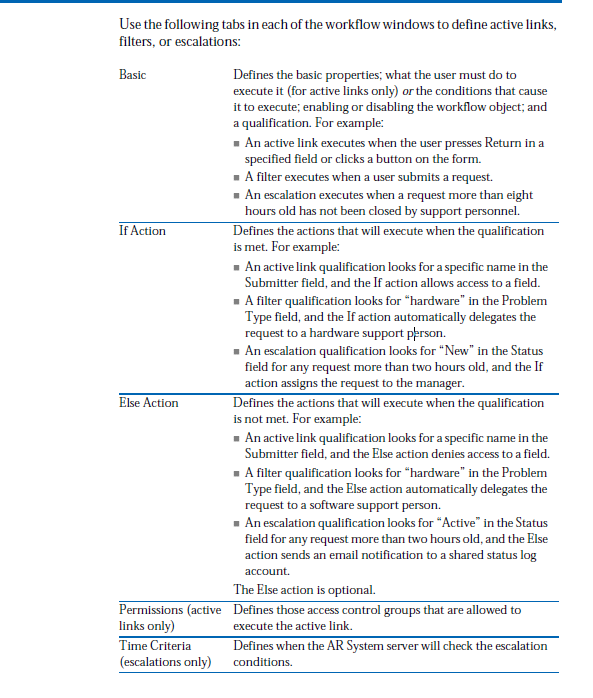
**Figure 1-2: Filter window**



Use the Escalation window, shown in the following figure, to create and modify escalations.

**Figure 1-3: Escalation window**





Change History Records the owner, the user who last modified it, and the

date of the modification. You can also enter a description of

your changes.

Help Text Supplies help text. In most cases, this help text is a

description of what the workflow object does, and how it is

used.

**Shared workflow**

You can create workflow (active links, filters, and escalations) that can be attached to one or multiple forms. For example, you could create an employee information active link that would populate generic identification and address fields anytime a user enters a name, and use this on multiple

forms. Shared workflow lets you efficiently build, maintain, and troubleshoot versions of forms and applications. Fewer workflow objects need to be stored on the server because any changes you make only need to be made once for all forms that use the objects.

**WARNING:** Sharing active links and active link guides among forms in different deployable applications requires caution. Role permissions are resolved based on which application has ownership. The owner

application contains the form that is the primary or reference form in the active link or active link guide. The non-owner application might have identical roles mapped to different groups, but these mappings are

ignored. If only implicit groups have permission (no role permissions), there are no conflicts. For more information, see the *Form and Application* *Objects* guide The way you define shared active links, filters, or escalations is similar to the way you define workflow for an individual form. The main difference is that instead of attaching the workflow to one form, you attach it to multiple forms. If you do not want the workflow to be shared, select only one form. See “Defining workflow basics” on page 26 for instructions on creating workflow. Workflow actions interact with fields based on field ID (not the field name) on each of the forms to which the field ID is attached.

Because shared workflow uses any valid field ID, plan how you want to use shared workflow before attaching it to multiple forms. Multiple forms that share workflow can show different field names for fields with the same ID. To make it easier to administer shared workflow, create fields with the same ID

*and* the same field name on each form to help you remember the purpose of the field. Otherwise, the workflow might not fire, or the shared workflow actions will still be triggered but might not use the expected field. In either case, the action you expect to happen—because you have connected the workflow to a non-existing field or the wrong field—will have no effect (depending upon the workflow).

Finally, if fields have matching IDs but are different data types, AR System will convert them appropriately. After you have created a form with which you want to share workflow, you

can:

\_ Create a new workflow object and then attach it to forms.

\_ Select an existing workflow object and then attach it to forms.

**WARNING:** When you create a copy of a field to put on another form, the new field retains all of the properties of the original field including the field ID.Use caution when using different field names that share the same field ID since shared workflow uses the field ID to trigger actions. When exporting definitions, you can choose whether to maintain an association between the selected workflow and all related forms. See the *Form* *and Application Objects* guide for more information. When you delete a form that uses non-shared workflow, the workflow is deleted along with the form. However, if workflow is shared by multiple forms, it will not be deleted until the last form that uses it is deleted.

**Sample uses of shared workflow**

Possible ways to use shared workflow include:

\_ Populating common fields used to store employee data.

\_ Automatically assigning cases (based on tasks and applications).

\_ Accessing common forms such as bulletin boards, reporting, preferences,

reminders, and so on.

\_ Navigating, for example, in a configuration tool.

\_ Displaying a message or warning dialog box that appears whenever a user

submits a form or enters invalid information