BMC® Remedy® Action Request System® 7.0

Form and Application Objects
## Contents

**Preface** .................................................. 15
**Audience** ............................................. 15
**AR System documents** ................................. 16
**Learn about the AR System Developer Community** 18  
  Why should you participate in the Developer Community? 18  
  How do you access the Developer Community? 18

### Chapter 1 Defining applications 19

**Applications in AR System** ........................... 20  
  Local applications ................................. 20  
  Deployable applications ......................... 21  
  Creating applications—Overview ................... 23

**Creating an application object** ...................... 24  
  Specifying application permissions ................. 25  
  Including objects in an application ................ 26  
  Saving an application ............................. 28

**Deployable applications** ............................ 29  
  Understanding deployable application ownership 29  
  Working with deployable application states ........ 30  
  Working with deployable application access points 32

**Specifying additional application properties** ..... 35  
  General properties ............................... 35  
  Forms properties .................................. 36  
  Access Points properties .......................... 37
BMC Remedy Action Request System 7.0

Data properties .............................................. 37
Statistics properties ........................................ 37
Support Files properties .................................... 37
Dynamic Web Views properties ............................ 37
Permissions properties ..................................... 37
Subadministrator Permissions properties ............... 38
Change History properties ................................. 38
Help Text properties ........................................ 38

Presenting applications to users ......................... 38
Specifying General and Forms properties for application mode ............ 40
Specifying Help properties for application mode ..................... 42
Distributing the application to users as a shortcut ................. 44

Converting local applications to deployable applications .......... 44
Deleting applications ................................ 45

Chapter 2 Defining access control .................. 47
Understanding access control in AR System ............. 48
Groups in AR System .................................. 48
Reserved groups in AR System ......................... 50
Groups you create—Regular, computed, and dynamic ......... 52
Roles in AR System .................................. 53
Users in AR System .................................. 53
Subadministrators in AR System ....................... 53
Additive access control ................................ 56
Access to AR System objects .......................... 57
Form, active link guide, and application permissions ........... 57
Field permissions ........................................ 60
Active link permissions ................................ 65
Access to requests .................................... 66
Controlling access by using implicit groups—Row-level security ..... 67
Submitter and Assignee access .......................... 68
Assignee Group access ................................ 68
Dynamic group access ................................ 68
Using the Request ID field with implicit groups ............... 69
Using the Assignee Group and dynamic groups—Examples ......... 72
## Form and Application Objects

**Putting it all together** .......................................................... 76  
**Creating and managing groups** ........................................... 78  
  - Creating groups ............................................................ 78  
  - Managing groups ........................................................ 82  
**Creating and mapping roles** ............................................... 83  
**Assigning permissions—Four approaches** ............................ 86  
  - Defining default permissions ........................................ 87  
  - Defining permissions for individual or multiple AR System objects  90  
**Defining additional administrator permissions** ...................... 92  

### Chapter 3  Creating AR System forms .................................. 95  
**Types of forms** ...................................................................... 96  
  - Regular forms .............................................................. 96  
  - Join forms ................................................................... 96  
  - Display-only forms ....................................................... 98  
  - View and Vendor forms ................................................. 98  
**Creating and managing forms** ............................................. 99  
  - Creating forms ............................................................ 99  
  - Modifying forms .......................................................... 101  
  - Copying forms ............................................................ 102  
  - Renaming forms .......................................................... 102  
  - Deleting forms ............................................................ 103  
**Display-only forms** ............................................................. 104  
  - Creating display-only forms ........................................ 104  
  - Using a display-only form as a control panel .................... 104  
  - Using a display-only form as a dialog box ....................... 105  
**Join forms** .......................................................................... 110  
  - Understanding join forms .......................................... 110  
  - Using join forms in workflow ....................................... 117  
  - The Request ID field .................................................. 119  
  - Creating entries in join forms ...................................... 120  
  - Creating join forms ..................................................... 120  
  - Working with join forms in BMC Remedy Administrator .... 127
Contents

BMC Remedy Action Request System 7.0

Chapter 4 Types of fields ........................................... 143

Data fields ......................................................... 144
  Character fields .............................................. 144
  Diary fields ..................................................... 144
  Date and time fields .......................................... 146
  Currency fields ............................................... 147
  Integer fields ................................................... 150
  Real fields .................................................... 151
  Decimal fields ............................................... 151
  Selection fields ............................................... 151

Table fields ...................................................... 153
  List view table fields ......................................... 154
  Tree view table fields ....................................... 155
  Results list fields ........................................... 156
  Alert list fields ............................................. 157
  Table field structure ........................................ 157
  Table field function ......................................... 159
  Workflow considerations for table fields .............. 161

Attachment pools .................................................. 161
  Attachment pool structure .................................. 161
  Attachment pool function ................................... 163
  Workflow considerations for attachment pools ....... 164

View fields ......................................................... 164
  View field structure ......................................... 165
  View field function ........................................... 165
  Workflow considerations for view fields ................ 166

Data visualization fields ....................................... 166
<table>
<thead>
<tr>
<th>Form and Application Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application list fields</td>
</tr>
<tr>
<td>Workflow considerations for application list fields</td>
</tr>
<tr>
<td>Horizontal and vertical navigation fields</td>
</tr>
<tr>
<td>Workflow considerations for navigation fields</td>
</tr>
<tr>
<td>Button fields</td>
</tr>
<tr>
<td>Page fields</td>
</tr>
<tr>
<td>Page field structure</td>
</tr>
<tr>
<td>Page field function</td>
</tr>
<tr>
<td>Workflow considerations for page fields</td>
</tr>
<tr>
<td>Trim fields</td>
</tr>
<tr>
<td>Chapter 5 Creating fields</td>
</tr>
<tr>
<td>Determining what types of fields to use</td>
</tr>
<tr>
<td>Creating data fields</td>
</tr>
<tr>
<td>Creating currency fields</td>
</tr>
<tr>
<td>Defining default currency types</td>
</tr>
<tr>
<td>Creating a currency field</td>
</tr>
<tr>
<td>Defining currency ratios</td>
</tr>
<tr>
<td>Creating selection fields</td>
</tr>
<tr>
<td>Creating table fields</td>
</tr>
<tr>
<td>Table field column preferences</td>
</tr>
<tr>
<td>Defining advanced table field functionality</td>
</tr>
<tr>
<td>Creating attachment pools</td>
</tr>
<tr>
<td>Size factors to consider when creating attachment fields</td>
</tr>
<tr>
<td>Creating view fields</td>
</tr>
<tr>
<td>Creating data visualization fields</td>
</tr>
<tr>
<td>Creating application list fields</td>
</tr>
<tr>
<td>Executing entry points in HTML</td>
</tr>
<tr>
<td>Creating horizontal and vertical navigation fields</td>
</tr>
<tr>
<td>Creating button fields</td>
</tr>
<tr>
<td>Creating page fields</td>
</tr>
<tr>
<td>Creating trim fields</td>
</tr>
</tbody>
</table>
### Chapter 6  
**More field functionality**

- **Global fields**
  - Using regular global fields
  - Using window-scoped global fields

- **GUID fields**

- **Fields in join forms**

- **Managing fields**
  - Modifying fields
  - Copying fields
  - Deleting fields
  - Making data fields nonoperational
  - Finding fields in a form

### Chapter 7  
**Defining menus**

- Understanding character field menus
- Using the Menu window

- Creating and modifying menus
  - Creating menus
  - Modifying menus
  - Copying menus
  - Deleting menus

- Defining character menus
- Defining file menus
  - Menu file format
- Defining search menus
  - Defining advanced search menu functions
- Defining SQL menus
  - Performing the SQL operation
  - Database security issues
- Defining data dictionary menus
- Building and using menu change history
- Creating help text for menus
Chapter 8  Creating and managing form views  ........................................ 255
Creating and managing form views ....................................................... 256
Creating form views ........................................................................... 256
Changing which view is displayed ......................................................... 261
Modifying form views .......................................................................... 261
Renaming form views .......................................................................... 262
Deleting form views ............................................................................ 263
Setting view properties ....................................................................... 263
Managing fields in a form view ............................................................. 279
Spacing evenly vertically ...................................................................... 279
Spacing evenly horizontally ................................................................. 280
Spacing by length vertically ................................................................. 281
Arranging fields on a form .................................................................... 282
Setting the tab order for the fields in a form ....................................... 297
Including and excluding fields from form views .................................. 299
Preference settings for view selection .................................................. 301

Chapter 9  Defining entry points and home pages  ................................ 303
Overview of home pages and entry points .......................................... 304
Using entry points .............................................................................. 305
Types of entry points .......................................................................... 306
Using a home page ............................................................................. 308
Home page components ....................................................................... 309
Using home pages with web clients .................................................... 311
Creating form entry points ................................................................. 312
Creating entry point guides ................................................................. 316
Workflow issues with active links and entry point guides .................... 318
Overview of entry points in applications ............................................ 319
Defining application permissions ......................................................... 319
Creating entry points in applications .................................................. 320
Modifying or creating a home page ..................................................... 321
Modifying the default home page form ............................................... 321
BMC Remedy User—creating new home pages .................................... 321
Web clients—creating new home pages .............................................. 322
Using style sheets with application list and navigation fields .............. 323
Viewing a subset of entry points .............................................. 326
Home page navigation aids ..................................................... 329
   BMC Remedy User—adding a home page form action field to a form ........................................ 329
   Creating run process workflow that returns to the home page ...................................................... 330
Configuring home page preferences ........................................ 330
   Understanding how a home page appears in BMC Remedy User ................................................. 331
   Understanding how a home page appears in a browser .................................................................. 333
   Specifying a server and home page in the AR System User Preference form ................................... 335
   Specifying a home page on the server .............................................................................................. 336
   Specifying a server for the mid tier .................................................................................................. 336
   Suppressing automatic opening of home pages in BMC Remedy User .......................................... 337

Chapter 10 Defining packing lists ............................................... 339
   Using packing lists .......................................................................................................................... 340
   Creating packing lists .................................................................................................................... 340
   Working in the packing list window ............................................................................................... 341
      Specifying packing list basics ....................................................................................................... 342
      Defining packing list permissions ................................................................................................. 345
      Defining subadministrator permissions for packing lists .............................................................. 345
      Building and using packing list change history ............................................................................. 345
      Creating help text for packing lists .............................................................................................. 345
   Working with packing lists in BMC Remedy Administrator .......................................................... 346
      Viewing a packing list in a workspaces ......................................................................................... 346
      Including packing lists in a deployable application ........................................................................ 348
      Dragging and dropping packing lists between windows ............................................................... 349
   Saving packing lists in XML ........................................................................................................... 349

Appendix A Core fields ................................................................. 353
   Core fields ...................................................................................................................................... 354
      Core field characteristics ................................................................................................................ 357
   Short Description field .................................................................................................................... 358
   Working with the Request ID field .................................................................................................. 359
      Changing the next available ID for new requests ........................................................................... 360
      Changing the Request ID field length or prefix ............................................................................ 362
      Preserving existing Request ID field values .................................................................................. 363
Display properties .................................. 428
Functional Types properties (currency fields only) .... 437
Help Text properties .................................. 438
Image properties ..................................... 439
Navigation Items properties .................. 441
Pages properties (page holder fields only) .......... 443
Permissions properties ............................. 444
Related Workflow properties ............. 445
Results Color properties ........... 446
Shared Fields properties (page fields only) .... 447
Sort properties (table fields) ............. 448
Table Labels properties .................... 449
Table/Tree Property properties .... 453
Views properties .................................. 455

Appendix E Archiving data .......................... 457
Understanding data archiving .................. 458
Configuring data archiving for a form ........ 459
Saving the main form and deleting archive forms .... 462
Saving the main form to a different form name ... 462
Deleting an archive form ....................... 462
Properties of archive forms .................. 464
General properties ............................. 464
Archive forms created by the AR System ....... 465
Existing forms specified as archive forms ....... 466
The Distributed Server Option and archive forms 466
Configuring data archiving for a server ........ 466
Server events and logging ...................... 466

Appendix F Audit ..................................... 469
Understanding auditing ....................... 470
About audit styles ............................... 470
Form-style audits ............................... 471
Log-style audits .................................. 474
Form and Application Objects

Configuring auditing ........................................ 476
  Configuring a form for auditing. ......................... 476
  Specifying fields to be audited. ....................... 478
Auditing view and vendor forms ......................... 482
Considerations for join forms ......................... 482
  Form style ............................................ 482
  Log style ............................................ 484
Fields Changed flag fields .......................... 484
The Distributed Server Option and audit forms ...... 485
Audit processing and filters ......................... 485

Appendix G Importing and exporting object definitions and applications. 487
AR System object definitions ............................. 488
  The AR System definition (*.def) file type .......... 489
  The AR System XML (*.xml) file type ............... 489
Exporting and importing definitions .................. 490
  Exporting object definitions ....................... 490
  Importing object definitions ...................... 496
Exporting and importing deployable applications .... 500
  Exporting deployable applications ................ 500
  Importing deployable applications ............... 502
Exporting and importing data with deployable applications 504
  Including form data when exporting definitions .... 505
  Importing form data ................................ 507
Exporting and importing view definitions ............ 510
Locking objects ........................................... 511
  Modifying locked objects. ......................... 512
  Levels of object locking ......................... 513
  Locking objects using export operation .......... 515
  Importing locked objects ......................... 517

Appendix H Localizing AR System applications .................... 523
Distinguishing between data and display languages ... 524
  Data language ....................................... 525
  Display language ................................. 525
Important: The compatibility information listed in the product documentation is subject to change. See the compatibility matrix at http://supportweb.remedy.com for the latest, most complete information about what is officially supported.

Carefully read the system requirements for your particular operating system, especially the necessary patch requirements.

Audience

This guide is written for developers and administrators who create, customize, and maintains applications based on BMC® Remedy® Action Request System® (AR System®).

Note: You should be familiar with BMC Remedy User and BMC Remedy Alert before you begin.

This manual contains reference information and procedures for creating, modifying, and maintaining AR System components, including applications, forms, fields, menus, views, special forms, and packing list.
AR System documents

The following table lists documentation available for AR System products. Unless otherwise noted, online documentation in Adobe Acrobat (PDF) format is available on AR System product installation CDs, on the Customer Support site (supportweb.remedy.com), or both.

You can access product Help through each product’s Help menu or by clicking on Help links.

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
<th>Audience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concepts</td>
<td>Overview of AR System architecture and features with in-depth examples; includes information about other AR System products as well as a comprehensive glossary for the entire AR System documentation set.</td>
<td>Everyone</td>
</tr>
<tr>
<td>Installing</td>
<td>Procedures for installing AR System.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Getting Started</td>
<td>Introduces topics that are usually only learned when first starting to use the system, including logging in, searching for objects, and so on.</td>
<td>Everyone</td>
</tr>
<tr>
<td>Form and Application Objects</td>
<td>Describes components necessary to build applications in AR System, including applications, fields, forms, and views.</td>
<td>Developers</td>
</tr>
<tr>
<td>Workflow Objects</td>
<td>Contains all of the workflow information.</td>
<td>Developers</td>
</tr>
<tr>
<td>Configuring</td>
<td>Contains information about configuring AR System servers and clients, localizing, importing and exporting data, and archiving data.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Installing and Administering</td>
<td>Contains information about the mid tier, including mid tier installation and configuration, and web server configuration.</td>
<td>Administrators</td>
</tr>
<tr>
<td>BMC Remedy Mid Tier</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integrating with Plug-ins and Third-Party Products</td>
<td>Discusses integrating AR System with external systems using plug-ins and other products, including LDAP, OLE, and ARDBC.</td>
<td>Administrators /Developers</td>
</tr>
<tr>
<td>Optimizing and Troubleshooting</td>
<td>Server administration topics and technical essays related to monitoring and maintaining AR System for the purpose of optimizing performance and troubleshooting problems.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Title</td>
<td>Description</td>
<td>Audience</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>Database Reference</td>
<td>Database administration topics and rules related to how AR System interacts with specific databases; includes an overview of the data dictionary tables.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Administering BMC Remedy DSO</td>
<td>Server administration and procedures for implementing a distributed AR System server environment with the BMC Remedy Distributed Server Option (DSO).</td>
<td>Administrators</td>
</tr>
<tr>
<td>Administering BMC Remedy Flashboards</td>
<td>Flashboards administration and procedures for creating and modifying flashboards and flashboards components to display and monitor AR System information.</td>
<td>Administrators</td>
</tr>
<tr>
<td>C API Reference</td>
<td>Information about AR System data structures, C API function calls, and OLE support.</td>
<td>Administrators</td>
</tr>
<tr>
<td>C API Quick Reference</td>
<td>Quick reference to C API function calls.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Java API *</td>
<td>Information about Java classes, methods, and variables that integrate with AR System.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Administering BMC Remedy Email Engine</td>
<td>Procedures for installing, configuring, and using the BMC Remedy Email Engine.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Error Messages</td>
<td>List and expanded descriptions of AR System error messages.</td>
<td>Administrators</td>
</tr>
<tr>
<td>Master Index</td>
<td>Combined index of all books.</td>
<td>Everyone</td>
</tr>
<tr>
<td>Release Notes</td>
<td>Information about new features list, compatibility lists, international issues, and open and fixed issues.</td>
<td>Everyone</td>
</tr>
<tr>
<td>BMC Remedy Import Help</td>
<td>Procedures for using BMC Remedy Import.</td>
<td>Administrators</td>
</tr>
<tr>
<td>BMC Remedy Administrator Help</td>
<td>Procedures for creating and modifying an AR System application for tracking data and processes.</td>
<td>Administrators</td>
</tr>
<tr>
<td>BMC Remedy Alert Help</td>
<td>Procedures for using BMC Remedy Alert.</td>
<td>Everyone</td>
</tr>
<tr>
<td>BMC Remedy Mid Tier Configuration Tool Help</td>
<td>Procedures for configuring the BMC Remedy Mid Tier.</td>
<td>Administrators</td>
</tr>
</tbody>
</table>

* A JAR file containing the Java API documentation is installed with the AR System server. Typically, it is stored in \Program Files\AR System\Arserver\Api\doc\ardoc70.jar on Windows and /usr/ar/<server_name>/api/doc/ardoc70.jar on UNIX.
Learn about the AR System Developer Community

If you are interested in learning more about AR System, looking for an opportunity to collaborate with fellow AR System developers, and searching for additional resources that can benefit your AR System solution, then this online global community sponsored by BMC Remedy is for you.

In the Developer Community, you will find collaboration tools, product information, resource links, user group information, and be able to provide BMC Remedy with feedback.

The Developer Community offers the following tools and information:

- Community message board
- Community Downloads
- AR System Tips & Tricks
- Community recommended resources
- Product information
- User Experience Design tips

Why should you participate in the Developer Community?

You can benefit from participating in the Developer Community for the following reasons:

- The community is a direct result of AR System developer feedback.
- BMC Remedy provides unsupported applications and utilities by way of Community Downloads, an AR System application.
- BMC Remedy posts the latest AR System product information in the Developer Community to keep you up to date.
- It is an opportunity to directly impact product direction through online and email surveys.
- It’s free!

How do you access the Developer Community?

Go to supportweb.remedy.com, and click the Developer Community link.
This section describes the types of application objects and explains how to work with them. The following topics are provided:

- Applications in AR System (page 20)
- Creating an application object (page 24)
- Deployable applications (page 29)
- Specifying additional application properties (page 35)
- Presenting applications to users (page 38)
- Converting local applications to deployable applications (page 44)
- Deleting applications (page 45)

For information about determining your application needs and working with BMC Remedy Administrator, see the Getting Started guide.

For specific information about creating and managing web applications, see the Installing and Administering BMC Remedy Mid Tier guide.
Applications in AR System

An application is a server object that contains references to a set of forms. Based on the forms included in the application, other objects related to those forms (such as active links, filters, and menus attached to the forms) are also included in the application. Like other server objects, an application has properties such as a name, permissions, change history, and so on.

Use applications to group sets of objects to accomplish particular tasks. For example, you might create an employee setup application that contains forms and workflow related to setting up a new employee's telephone, computer, and office supplies. Users can run applications in BMC Remedy User or in a web client.

When you use an application object to associate a set of objects, you and your users can interact with the application as a functional unit. In BMC Remedy Administrator, you open an application in an Application window that displays only the objects included in that application. When you create a new object, such as a form, within an Application window, the form is automatically added to that application. You can open multiple Application windows at the same time and drag and drop objects between them. For more information about working in Application windows, see the Getting Started guide.

You can create two types of applications—local and deployable.

Local applications

Local applications are the same as applications created in pre-6.0 versions of AR System. They are intended for use on a single server or a small number of servers. They use permissions based on groups, which means that each local application is designed for one specific server environment. This makes it difficult to migrate local applications to different servers where permissions might need to be redefined for many objects, including forms, fields, and workflow.

Two or more local applications on the same server can include the same set of forms and workflow objects. This allows for efficient development and less maintenance. For more information about shared workflow, see the Workflow Objects guide.

Applications upgraded to version 6.0 or later are upgraded as local applications.
Deployable applications

Deployable applications are treated as specific collections of objects and data that are independent of the AR System server on which they are developed. All objects contained within a deployable application, the application’s permissions, and select data you want included with the application can be moved easily from server A to server B. A deployable application has extended functionality that makes it more portable. The application can be installed, licensed, and tracked on a variety of AR System servers.

You can convert local applications to deployable applications as explained in “Converting local applications to deployable applications” on page 44.

Retaining independence and integrity

To retain independence and integrity, overt sharing of forms and workflow is discouraged. For example, BMC Remedy Administrator does not allow you to include the same form in two or more deployable applications. Other objects and data can still be shared, but in a controlled way. There are also special considerations for shared active links and active link guides, which have permissions tied to a particular deployable application. For more information about shared workflow, see the Workflow Objects guide.

The integrity of a deployable application is guaranteed by the concept of application ownership. An object contained within a deployable application is owned by that application. This allows AR System to resolve conflicts between two deployable applications by simply examining the ownership property for each object in question. For more information, see “Understanding deployable application ownership” on page 29.

Exporting and importing deployable applications

Deployable applications use their own export and import mechanism. All objects and data necessary for the application to correctly install and execute on the target server are automatically compiled into a single definition file on export. This file is a location-independent archive of the application because it contains everything needed to install the application on an AR System server without any dependencies on the environment in which it was developed. You can also specify data to export along with the application, such as data from forms associated with the application. Workflow associated with the application, but not owned by the application, must be exported and imported separately.
On import, all objects and data are installed on the destination server in one step. For more information about exporting and importing data, see the Configuring guide. For information about exporting and importing object definition, see Appendix G, “Importing and exporting object definitions and applications.”

Roles and deployable applications

When assigning permissions for deployable applications, roles are used instead of groups. Roles are key to making the application location independent. Roles are a part of the application’s data, and as such, are exported with the deployable application.

On import, an application’s roles can be mapped to the target server’s groups. Therefore, the permissions for individual objects do not change; only the mapping changes when moving the application from server to server. This eliminates the task of reassigning permissions for potentially thousands of individual objects, and makes it relatively easy to install the same deployable application on a variety of target servers with differing permissions models.

Roles can also be mapped depending upon the application’s development state. For example, the role Management might be mapped to the group Test while the application is in a maintenance state. The same role might be mapped to the group Manager when in the production state. When you change the state of the application, you affect permissions for all objects within the application without modifying the objects themselves. For more information, see “Working with deployable application states” on page 30.

Additional features

The following functionality is available only for deployable applications:

- **Application access points**—Application access points are points of integration among deployable applications. For more information, see “Access Points properties” on page 37.

- **Application licensing**—You can require users to obtain a license from you before they can use a deployable application. You might use application licensing together with object locking, which prevents objects from being modified in BMC Remedy Administrator. Object locking can be used with both local and deployable applications. For more information about application licensing and object locking, see the Integrating with Plug-ins and Third-Party Products guide.
Form and Application Objects

- **Application statistics**—You can collect performance information such as how many times the application was accessed in a given period of time. You can collect similar statistics on individual forms. For more information, see the Optimizing and Troubleshooting guide.

Creating applications—Overview

The following steps explain how to create an application.

**Step 1** Create the groups or roles that will have access permission to the application object and to all of the objects it contains.

**Step 2** Create an application object and specify access permissions (as needed) and other properties for the application, as described in this section.

**Step 3** Optionally, define default permissions that apply to new objects you create. For more information, see “Defining default permissions” on page 87.

**Step 4** Create and include objects (such as forms and workflow) in the application, as described in this section and in the Getting Started guide. Apply permissions to these objects as needed.

**Step 5** Define and configure user access to the application:

- Define entry points for the application, as explained in “Creating form entry points” on page 312. For alternate user access methods, see “Presenting applications to users” on page 38.

- Create users as discussed in the Configuring guide

- Assign users to groups as discussed in the Form and Application Objects guide.

- Specify user licensing as discussed in the Integrating with Plug-ins and Third-Party Products guide.

**Step 6** Install the application in its target environment using export and import, as explained in the Appendix G, “Importing and exporting object definitions and applications.”
For deployable applications, perform the following tasks:

- Map roles to groups on the target server for each application development state. Set the application state property to enable a particular mapping. For information about roles, see “Roles in AR System” on page 53.
- Optionally, configure statistics for tracking application performance and usage. For more information, see the Optimizing and Troubleshooting guide.

Creating an application object

Use the following procedure to create an application object.

► To create an application object

1 In BMC Remedy Administrator, select the server where you want to create the application.

   In the Server window, select a server name. If an Application window is active, the new application is created on the same server as the open application.

2 Choose File > New Application to open the Create a New Application dialog box.

Figure 1-1: Create a New Application
3 Specify an application name.

Names are shared across applications, active link guides, filter guides, packing lists, and web services, so each name must be unique on each AR System server.

Names can be a maximum of 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name.

**Note:** Do not use the name `arforms` for the application name. It is a reserved name for the system.

The `Web Alias` field is populated automatically, and is not used in version 6.3 and later versions of the mid tier. In prior releases, the web alias was used in the directory structure for web applications deployed on the mid tier.

4 Select the type of application (Local Application or Deployable Application).

For more information about local and deployable applications, see “Applications in AR System” on page 20.

5 Click OK.

The Application window opens. For information about working in the Application window, see the Getting Started guide.

**Specifying application permissions**

You specify group or role permissions for the application object as well as for the objects contained within the application. If users do not have permission to an application, they will have limited access to the application’s entry points. For more information about entry points, see “Creating form entry points” on page 312.

The permissions you grant for the application object are not inherited by the objects in the application. Likewise, denying permission to the application does not deny permission to the objects within the application. You must grant or deny permissions for each form, field, and workflow object within the application.
Note: Newly created roles appear in the permissions tab of object properties in a deployable application after the server recaches (about 5 seconds, depending on your system).

For information about groups, roles, and applications, see Chapter 2, “Defining access control.”

Including objects in an application

The primary object in an application is the form. All other objects are included in an application based on their association with the forms in the application.

When you add a form to an application, all of the objects associated with that form, such as workflow and menus, are also added to the application. Likewise, when you remove a form, all of the objects related to it are removed from the application. Removing a form does not delete the form on the server. For information about deleting objects, see the Getting Started guide.

WARNING: When you add an existing form to a deployable application, AR System removes all explicit group permissions from the form and its associated objects. You must manually apply role permissions to every object, including the form and its fields, and to any active links and active link guides for which the form is the reference form.

You can include objects in an application in the following ways:

- By creating a new object in the context of an Application window, as explained in the Getting Started guide.
- By dragging and dropping objects to the Application window, as explained in the Getting Started guide.
- By including forms in the application using Application Properties, as explained in the following procedure.
Including forms in an application using forms properties

Use the Forms tab in the Application Properties to include forms in an application.

To include forms in an application using Forms properties

1. Open an application as described in the Getting Started guide.
2. Choose Application > Properties to open the Application Properties window.

Figure 1-2: Specifying application forms

3. Click the Forms tab.

For local applications, every form defined on the server appears. For deployable applications, only forms in local applications or forms not contained in other deployable applications appear.
4 Use the Add and Remove buttons to move forms to the appropriate Forms list.
   
   To select a form view other than the default administrator view, select the appropriate form, and then select the appropriate view from the View list below it.

5 Choose File > Save Application.

**Saving an application**

The application workspace is automatically updated to include new objects—you do not save the Application window to preserve its contents. For information about working in the Application window, see the Getting Started guide.

If you change any of the application’s properties, you need to save the properties to preserve your changes. For more information, see “Specifying additional application properties” on page 35.

**Note:** You cannot create copies of deployable applications using the Save As command. This is because different deployable applications cannot contain the same forms.
Deployable applications

The following sections will help you learn more about working with deployable applications.

Understanding deployable application ownership

All objects associated with a deployable application—except for menus—can be owned by that application. AR System uses the Application Owner property in the following ways:

- For active links and active link guides, AR System uses the owner application's role mappings to resolve permissions. The owner application is the application that contains the reference form for the active link or active link guide. For more information, see the Workflow Objects guide.

When workflow is used by multiple deployable applications, permissions for the workflow are determined by the owner application. For more information, see the Workflow Objects guide.

- For entry points, AR System uses the owner property to determine which application heading the entry points appear under in the home page. For information about entry points and home pages, see Chapter 9, “Defining entry points and home pages.”

- For application export, only objects owned by the application— and any menus and roles related to the application— are exported. For more information about application export and import, see Appendix G, “Importing and exporting object definitions and applications.”

You can identify the application owner (if any) in the following ways:

- Open the object’s properties and view the Permissions tab, as shown.
Chapter 1—Defining applications

BMC Remedy Action Request System 7.0

Figure 1-3: Application owner in the Permissions tab

- Display object details in the Server window or Application window, or view the Object Summary for the object, as explained in the Getting Started guide.

Working with deployable application states

In deployable applications, role permissions resolve to different groups depending on the application's state. State refers to the development state of the application, such as Maintenance, Test, or Production.

You can change an application's state in application properties, through the user client, or through workflow, as described in the following sections. You can also create custom states for use by all deployable applications on the server.
**Specifying deployable application states**

When you create or import a deployable application, the state is set to Maintenance. Only administrators and subadministrators have access to this state. For other states, such as Test and Production, access control depends on role-group mappings defined in the Roles form.

**To specify deployable application states**

1. Define role-group mappings for each state in the Roles form:
   a. In BMC Remedy User, open the Roles form and search for all roles defined for the application.
      If the search returns no roles, define roles for the application.
   b. For each role, make sure that the role is mapped to an explicit group for each state.
   c. Save the roles.
   
   For more information about roles, see “Roles in AR System” on page 53.

2. Specify the state for the application, as follows:
   a. In BMC Remedy Administrator, open the application as explained in the Getting Started guide.
   b. Choose Application > Properties.
   c. On the General tab, choose a state from the State list.
   d. Choose File > Save Application.

   The role-group mappings for the specified application state become effective after the server recaches.

**Using the user client or workflow to change deployable application states**

The state defined in application properties is stored in an entry for that application in the AR System Application State form. You can edit the entries in this form in the user client or create workflow that acts upon this form to change the application’s state.
When creating the workflow, remember the following tips:

- A state value of `NULL` is the same as the Maintenance (administrators only) state.
- State names are case-sensitive.
- Entries are removed from the AR System Application State form when applications are deleted.

**Creating custom states**

You can create custom application states by adding fields to the Roles form. Use field IDs in the range of 2003 to 2199.

**To create custom states**

1. In BMC Remedy Administrator, open the Roles form.
2. Under the Production field, add a character field.
3. Double-click the field to open the Field Properties dialog box.
4. In the ID field on the Database tab of the Field Properties dialog box, enter a unique field ID in the range from 2003 to 2199.
5. Save the form.

AR System automatically adds a menu to the field for selecting groups. The new state is available for deployable applications on the current server after the server recaches.

6. To make custom states available on another server, export the Roles form and import it on the target server.

For more information about exporting and importing forms, see Appendix G, “Importing and exporting object definitions and applications.”

**Working with deployable application access points**

You can identify specific forms and guides in a deployable application as access points, or points of integration, for use with other deployable applications. When creating table fields, join forms, and certain workflow actions (such as Set Fields or Call Guide actions), developers can choose to limit the scope of forms or guides listed to only those forms or guides within the current application, and those that have been identified as access points in other applications.
The access points you create for an application are recommended points of integration only—developers can still choose to work with any objects on the server.

Use the following procedures to define access points for your application, and to take advantage of access points in other applications during development.

▶ To define access points in a deployable application

1. Open a deployable application, as described in the Getting Started guide.
2. Choose Application > Properties.
3. Click the Access Points tab.

Every form, active link guide, and filter guide defined for the application appears.

Figure 1-4: Specifying application access points

4. Use the Add and Remove buttons to move objects to the appropriate Objects list.
5. Choose File > Save Application.
To use access points during development

1. Open a deployable Application window, as described in the Getting Started guide.

2. When you create one of the following objects, select the Restricted List check box to show the list of forms (or guides, where applicable) that are included in the current application, and those that have been defined as access points in other deployable applications:

<table>
<thead>
<tr>
<th>Object</th>
<th>Location of Restricted List check box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active link, filter, or</td>
<td>If Action and Else Action tabs for the following actions:</td>
</tr>
<tr>
<td>escalation</td>
<td>Call Guide</td>
</tr>
<tr>
<td></td>
<td>Open Window</td>
</tr>
<tr>
<td></td>
<td>Push Fields</td>
</tr>
<tr>
<td></td>
<td>Set Fields</td>
</tr>
<tr>
<td>Table field</td>
<td>Table Property tab in table field properties.</td>
</tr>
<tr>
<td>Join form</td>
<td>Join Wizard, Page 1</td>
</tr>
</tbody>
</table>

When you clear the Restricted List check box, you will see all of the forms (or guides) on the server.

**Note:** The Restricted List check box on the Basic tab for active links, filters, escalations, active link guides, and filter guides restricts the list of forms to forms within the current application only. Access points from other applications are not shown.

Also, for Web Services objects, you cannot use access points and there is no Restricted List check box. In deployable applications, you can only choose from among forms within the application. In local applications, you can choose from among all forms on the server.
Specifying additional application properties

Use the following procedure to define an application’s properties.

To define application properties
1 Open an application as described in the Getting Started guide.
2 Choose Application > Properties.
3 Select a tab such as General or Permissions and modify properties as desired.
   The following sections describe the contents of each tab.
4 Choose File > Save Application.

General properties

The primary properties on the General tab determine the application name, web alias, label, and development state. The following sections describe these properties in detail.

Note: The following additional properties on this tab are for configuring applications for direct access in BMC Remedy User:

- Description
- Run with Form Windows Maximized
- Show Only Forms in Application
- Custom Title Bar Icon
- Custom About Box

For more information about these properties, see “Specifying General and Forms properties for application mode” on page 40.

Name

Names are shared across applications, active link guides, filter guides, packing lists, and web services, so each name must be unique. Names can be a maximum of 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name.
Do not use the name \texttt{arforms} for the application name as it is a reserved name for the system.

**Web Alias**

In releases of the mid tier prior to version 6.3, the application web alias was used in the directory structure for web applications deployed on the mid tier. This field is not used in versions 6.3 and later.

**Label**

For deployable applications, if a Label is specified, it is used in place of the Name to identify the application corresponding to entry points in the home page. For more information about entry points and home pages, see Chapter 9, “Defining entry points and home pages.”

The Label also has a specific meaning in the context of applications accessed directly in Windows clients. For more information, see “Presenting applications to users” on page 38.

**State**

This property defines the application state (Maintenance, Production, or Test) and is for deployable applications only. Depending on the state you choose, different access permissions are applied to the application.

For more information, see “Working with deployable application states” on page 30.

**Forms properties**

The Forms tab determines the forms included in the application, as explained in “Including forms in an application using forms properties” on page 27.

The following Forms properties apply only to applications that users will not access through entry points on a home page:

- Primary Form
- Primary View

For more information about these properties, see “Presenting applications to users” on page 38.
Access Points properties

For deployable applications only, this tab defines which forms, active link guides, and filter guides are integrated with other deployable applications. For more information, see “Working with deployable application access points” on page 32.

Data properties

For deployable applications only, this tab defines the forms whose data will be included in an application export. It also defines the qualifications (if any) that select sets of records, and import options such as the handling of duplicate request IDs. For more information, see the Configuring guide.

Statistics properties

For deployable applications only, this tab defines the forms that will participate in statistics tracking for the application. For more information, see the Optimizing and Troubleshooting guide.

Support Files properties

The Support Files tab is for web applications only. This tab defines the names and locations of resources used in web views that are included in the application object. For more information, see the Installing and Administering BMC Remedy Mid Tier guide.

Dynamic Web Views properties

The Dynamic Web Views tab was used in releases prior to version 6.3. Dynamic web views are available only to clients using a pre-6.3 release of the mid tier. Please see your previous version’s documentation for more information.

Permissions properties

Use the Permissions tab to determine which access control groups can display the application in the user client. For more information, see “Specifying application permissions” on page 25.
Subadministrator Permissions properties

Use the Subadministrator Permissions tab to define subadministrator permissions for access control groups. For more information, see “Subadministrators in AR System” on page 53.

Change History properties

AR System automatically records the owner of an application, the developer who last modified the application, and the date of the modification. To display or add to this information, click the Change History tab in the Application window.

For general information about building and using change history, see the Getting Started guide.

Help Text properties

The Help Text property supplies help text about the application for administrators. It typically includes a description of the application, what it does, and how it is used. For general information about creating help text in AR System, see the Getting Started guide.

The Help properties also apply to applications configured for direct access in BMC Remedy User. For more information, see the following section.

Presenting applications to users

The typical or default method to present an application to users is to define entry points that appear in a home page, as explained in Chapter 9, “Defining entry points and home pages.” This section describes additional ways to present applications to users.

For web applications, you can provide links to forms using special URLs, such as encoded URLs, URLs that bypass the login page, or URLs that pass search parameters when a search form is opened. Include the URLs on a web page or form. For more information, see the Installing and Administering BMC Remedy Mid Tier guide.

You can configure Windows applications so that users open the applications within BMC Remedy User in application mode. This technique works for both local and deployable applications, as explained in this section.
In BMC Remedy User, users access an application in the Object List dialog box, locating the application by name or by a custom label that you provide. A description of the application (if defined) appears below the task list when the user selects the application, as shown.

**Figure 1-5: Object List dialog box**

Users can also open the application from an ARTask shortcut that you provide, as described in “Distributing the application to users as a shortcut” on page 44.
Specifying General and Forms properties for application mode

Use the following procedures to specify General and Forms properties for use in application mode. For more information about application mode, see “Presenting applications to users” on page 38.

To specify General properties for an application

1. Open an application, as described in the Getting Started guide.
2. Choose Application > Properties.
3. Click the General tab.
4. In the Label field, specify the label that you want to appear in the Object List dialog box in BMC Remedy User, and in the title bar in application mode.
   
   If you do not specify a label, the Name property will be used to identify the application.
   
   Labels can be as many as 255 bytes, including spaces.
5. In the Description field, specify the description that you want to appear below the task list in the Object List dialog box in BMC Remedy User.
   
   You can enter a maximum of 2000 bytes.
6. To specify a custom icon:
   a. Select the Custom Title Bar Icon check box.
   b. Click the Browse button and select the appropriate image file.
      
      You can add images in .bmp, .jpg, .jpeg, and .dib formats that are as large as 16 pixels wide by 16 pixels high. Images larger than these dimensions will be cropped. The image file size limit is 512 KB. Keep the file size as small as possible to avoid performance problems.
7. To display an image in the About box:
   a. Select the Custom About Box check box, and use the Browse button to locate the appropriate image.
      
      You can add images in .bmp, .jpg, .jpeg, and .dib formats. The image is cropped to fit within the About box. Keep the file as small as possible to avoid performance problems.
   b. To save the image to another area on the network, click Save To Disk.
8. Choose File > Save Application.
To specify the behavior of forms in application mode

1. Open an application, as described in the Getting Started guide.
2. Choose Application > Properties.
3. Click the Forms tab.
4. From the Primary Form list, select the form that will appear in BMC Remedy User when the application first opens.
   If you do not specify a primary form, no form appears when users open the application in BMC Remedy User.
5. From the Primary View list, select the view of the form to appear when the application opens.
   The views available in the list are defined by the view label. If no view is selected, the default view or the user preference view will be used.
6. Click the General tab.
7. Select or clear the Run With Form Windows Maximized check box. If this check box is:
   - Selected, the form window is maximized when it opens.
   - Cleared, the form window opens with the size that the developer defines.
8. Select or clear the Show Only Forms in Application check box. If this check box is:
   - Selected, users can access in BMC Remedy User only forms and guides within the application.
   - Cleared, users can access in BMC Remedy User any forms, guides, or applications to which they have access whether they are related to the application or not.
Specifying Help properties for application mode

You can create field-level help for users of your application, as explained in the Getting Started guide. For applications that run in application mode, you can also provide help in the following ways:

- Using help text that opens from the Help menu in a system-generated dialog box. Use this method if the application does not require extensive help text.

- Using external help files that open from the Help menu in a supporting program such as Microsoft Word. Use this method if the application is complex and requires detailed instructions. You can provide up to five external help files.

Specifying help text in a dialog box

Use the following procedure to specify help text that users can access by choosing Help > Help on <application_label> in BMC Remedy User.

Figure 1-6: Help text example

To specify help text in a dialog box

1. Open an application, as described in the Getting Started guide.
2. Choose Application > Properties.
3. Click the Help Text tab, and select the Help Text option button.
4. In the field, specify the help text that you want to appear.
5. Choose File > Save Application.
Specifying external help files

You can specify up to five different external help files for an application. Help formats that you can use include .htm, .html, .doc, .txt, .hlp, or .chm files. Your users must be able to access the help file locally on their machines. For example, if your help file is a web page, users must have browsers installed.

When users choose Help > Contents and Index (for a single help file) or Help > Help on <application_label> (for multiple help files) in application mode in BMC Remedy User, the help appears in the format that you specify. For example, if you created an HTML help system for your application, the user's default browser will open, displaying the HTML help file.

To specify external help files

1. Open an application, as described in the Getting Started guide.
2. Choose Application > Properties.
3. Click the Help Text tab.
4. Select the External Help Files option button.
5. Click Add to open the Add External Help File dialog box.
6. In the Label field, enter a label for the help file.
7. Click the browse button next to the Help File field.
   The Open dialog box appears.
8. Select a file of type .htm, .html, .doc, .txt, .hlp, or .chm.
9. Click Open.
   The Add External Help File dialog box appears, and the contents of the Help File field changes to identify the help file that you have selected.
10. Click OK.
   Continue working with application help as follows:
   - To test a help file, select the help file and click Test.
   - To change the label of a help file, select the help file and click Modify.
   - To remove a help file, select the help file and click Clear.
   - To export a help file from an application, select the help file and click Save to Disk.
11. Choose File > Save Application.
Distributing the application to users as a shortcut

You can send an application as an ARTask shortcut to your users, who can use the shortcut to start the application in BMC Remedy User. You can create the shortcut in BMC Remedy Administrator or BMC Remedy User.

**Note:** Creating a shortcut using the following procedure requires a MAPI-compliant email client.

To create an application shortcut in BMC Remedy Administrator

1. Open an application, as described in the Getting Started guide.
2. Choose Application > Properties.
3. Choose File > Send To > User(s) as Shortcut.
   
   Your email tool opens and displays your shortcut as an attachment.
4. Use your email tool to send the shortcut to one or more users.

To create an application shortcut in BMC Remedy User

1. In BMC Remedy User, open the Object List dialog box.
2. Right-click the application for which you want to create a shortcut.
3. Choose Create Shortcut to open the Save a Shortcut dialog box.
4. Specify a location, and click Save.

Converting local applications to deployable applications

You can convert local applications to deployable applications to take advantage of the extended functionality in deployable applications.

Since local applications and deployable applications use different permissions models, you will need to redefine all of the permissions for the converted application.
WARNING: Converting a local application to a deployable application removes all explicit group permissions for every object in the application (including forms, fields, active links, and active link guides). You must manually apply role permissions to every object.

To convert a local application to a deployable application

1 Create a new deployable application as explained in “Creating an application object” on page 24.

   The Application window for the new deployable application opens.

2 Define permissions for the new deployable application, as explained in “Specifying application permissions” on page 25.

3 Open the local application, as explained in the Getting Started guide.

4 Drag and drop the forms from the local application to the deployable application, as explained in the Getting Started guide.

   This process moves the forms and related objects from the local application to the deployable application, and removes all of the explicit group permissions defined for those objects.

5 In the deployable application, apply role permissions to all objects, including forms, fields, active links, and active link guides.

   You can apply permissions to multiple objects as explained in “Defining permissions for individual or multiple AR System objects” on page 90.

6 Optionally, delete the local application as explained in the following section.

Deleting applications

When you delete an application, it is removed from the database and from the list of applications in BMC Remedy Administrator. However, the forms, workflow, and data included in the application are not removed.

If you provided an application as a shortcut, tell your user community to delete the application shortcut from their desktops. If users try to start an application after it has been deleted from the server, they will receive an error message. For more information, see “Presenting applications to users” on page 38.
To delete an application

1. Click the Applications icon in the Server window, and select the application you want to delete from the Applications list.
   You cannot delete an application that is open in BMC Remedy Administrator.

2. Choose Edit > Delete Applications.

3. Click Yes to delete the application.
   The application is removed from the database and will no longer appear in the list of applications on the Server window. The objects in the application are not deleted.
This section describes how to define access control within AR System and how users can have different abilities to access an application or any of its parts. The following topics are provided:

- Understanding access control in AR System (page 48)
- Groups in AR System (page 48)
- Roles in AR System (page 53)
- Users in AR System (page 53)
- Subadministrators in AR System (page 53)
- Additive access control (page 56)
- Access to AR System objects (page 57)
- Access to requests (page 66)
- Putting it all together (page 76)
- Creating and managing groups (page 78)
- Creating and mapping roles (page 83)
- Assigning permissions—Four approaches (page 86)
- Defining additional administrator permissions (page 92)
Understanding access control in AR System

Access control is the mechanism used to control who can create, view, modify, and delete objects and requests. You can control the objects and data that users can access as well as the actions they can perform.

In defining access control within AR System, you must:

1. Identify and create the groups (or roles, for certain types of applications) that reflect key functions in your company.
2. Add the users of your AR System application into their respective groups, based on the type of information each user must access.

Group membership ultimately determines which operations individual users can perform and what they can access. AR System has various levels of security:

- **Server**—Each server has its own group of users defined so that a user has an account on an AR System server.
- **Application and Form**—Users who do not belong to a group that has permission to access an application or form cannot see or interact with it.
- **Request (or row level)**—Users can view only the requests that pertain to them, such as those requests that they submit or the requests available only to a group to which they belong.
- **Field (or column level)**—A field will not be visible to users who do not belong to any of the groups with permission to the field.
- **Active link or active link guide (workflow)**—Running certain types of workflow requires that a user belong to a group with permission to do so.

Groups in AR System

Access control groups are collections of AR System users and are the basis by which all access is granted. In addition, groups can be used in association with notifications. For example, you can designate an entire group to be notified in a filter action.
AR System includes a Public group and seven other reserved 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. For information about adding groups, see “Creating and managing groups” on page 78.

AR System provides two types of groups:

- **Explicit groups**—Groups to which you must manually assign users in the User form. When a user is assigned to a group, the user is given access to all items to which the group is granted access.

  Explicit groups that you create are defined for a particular server. If you move the objects to a new server with its own defined explicit groups, you might need to resolve permission conflicts. Consider using a deployable application, which uses role permissions that can be mapped to different groups on different servers. For more information, see “Roles in AR System” on page 53.

  For information about assigning users to groups, see the Configuring guide.

- **Implicit groups**—Groups that depend on specific user circumstances and situations. Users belong to these groups based on specific conditions, such as the contents of special fields within each request. You do not directly assign users to implicit groups.

  Any **dynamic groups** that you create are also implicit groups. For more information, see “Dynamic group access” on page 68.
Reserved groups in AR System

AR System reserves eight group IDs for special group definitions. The following table describes the access privileges for each of these groups.

<table>
<thead>
<tr>
<th>Group</th>
<th>ID</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public</td>
<td>0</td>
<td>Implicit</td>
<td>Provides general access. Access granted to this group is granted to all users. Every user who logs in to AR System is automatically a member of the Public group. This includes registered users (that is, listed in the User form) and guest users. For information about allowing guest users, see the Configuring guide.</td>
</tr>
<tr>
<td>Administrator</td>
<td>1</td>
<td>Explicit</td>
<td>Defines users who have full and unlimited access to AR System. Users must have a fixed license or this group assignment will be ignored.</td>
</tr>
<tr>
<td>Customize</td>
<td>2</td>
<td>Explicit</td>
<td>Grants users the ability to customize their form layout in BMC Remedy User. Use this group (with caution) to allow the users to change their own form views.</td>
</tr>
<tr>
<td>Submitter</td>
<td>3</td>
<td>Implicit</td>
<td>Provides access to the user whose login name is in the Submitter field (field ID 2) for a particular request. Users automatically belong to the Submitter group for their requests. For more information, see “Submitter and Assignee access” on page 68. See the Configuring guide for a special submitter access mode that enables you to override the need for users to have a write license before they can modify the requests they submit.</td>
</tr>
<tr>
<td>Assignee</td>
<td>4</td>
<td>Implicit</td>
<td>Provides access to the user whose name is in the Assignee field (field ID 4) for a particular request. Users automatically belong to the Assignee group when their name is in the Assignee field. For more information, see “Submitter and Assignee access” on page 68.</td>
</tr>
<tr>
<td>Subadministrator</td>
<td>5</td>
<td>Explicit</td>
<td>Defines users who have limited administrative access to AR System. Users must have a fixed license or this group assignment will be ignored. Users who belong to this group can perform the tasks described in “Subadministrators in AR System” on page 53.</td>
</tr>
</tbody>
</table>

* See “Groups in AR System” on page 48 for more information about explicit and implicit types.
**Groups in AR System**

In addition to the groups listed in the previous table, groups with IDs in the range of 60000 to 60999 are reserved for dynamic groups. For more information, see “Group Category” on page 80.

<table>
<thead>
<tr>
<th>Group</th>
<th>ID</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flashboards Administrator (obsolete)</td>
<td>6</td>
<td>Explicit</td>
<td>For versions of Flashboards prior to 5.0, defines users who have full and unlimited access to the Flashboards server. Users must have a fixed Flashboards user license, or this group assignment is ignored.</td>
</tr>
</tbody>
</table>
| Assignee Group                             | 7  | Implicit | Provides access to the user who is a member of one of the groups listed in the Assignee Group field (field ID 112) for a request. Users automatically belong to the Assignee Group group for requests in which the Assignee Group field exists and contains a group to which the user belongs. For more information, see “Assignee Group access” on page 68 and “Form, active link guide, and application permissions” on page 57.  
**Note** Do not confuse this group with the Assignee group, which gives permission to an individual. |

* See “Groups in AR System” on page 48 for more information about explicit and implicit types.
Groups you create— Regular, computed, and dynamic

You can create the following groups in the Group form.

- **Regular groups**— Explicit groups that you create and to which you assign a specific list of users. For information about assigning users to groups, see the Configuring guide.

- **Computed groups**— Explicit groups that you create and to which users are assigned based on the memberships of explicit groups included in an expression. For example, you can create a computed group definition such as \((A \text{ AND } B) \text{ OR } C \text{ AND NOT } D\). This computed group includes the list of users who are members of both groups A and B, or members of group C, but not members of group D.

  Computed groups make groups easier to manage, because you can create a limited number of regular groups for which you maintain lists of users, and then create computed groups based on these regular groups without the need to maintain additional lists of users.

- **Dynamic groups**— Groups similar to the reserved Assignee Group group in that they use the contents of special fields to determine group membership. For more information, see “Dynamic group access” on page 68.

For information about creating groups, see “Creating and managing groups” on page 78.
Roles in AR System

Roles

Roles are permissions similar to groups, except that they belong to a particular application, instead of a particular server. Roles are used exclusively in deployable applications.

Roles are defined for each deployable application and then mapped to explicit groups on the server. You can map a deployable application's roles to different groups on different servers, depending on how the groups are defined on each server. This allows you to develop and test the application on one server and deploy it to a number of other servers without having to redefine permissions on each server. You can also map roles to different groups for each development state, such as Test or Production. You can then switch between states in BMC Remedy Administrator or through workflow. For more information, see “Creating and mapping roles” on page 83.

Since roles are mapped to groups, the groups you define on the server and the users that belong to them are the foundation of access control.

For more information about creating and using deployable applications, see Chapter 1, “Defining applications.”

Users in AR System

A user is any person to whom you give permission to access AR System. Users can be members of multiple groups or no group at all. Users in AR System range from an administrator (who maintains the entire system) to employees (who submit requests or view data).

AR System includes one predefined user (Demo). You can use the User form in BMC Remedy User or a web client to rename this user, and you can create additional users in AR System. For information about defining users for AR System, see the Configuring guide.

Subadministrators in AR System

With subadministrator permissions, you can grant administrator access to a subset of existing forms, applications, and workflow, as shown in the following figure. Inside a form, application, or workflow definition, subadministrators have the same privileges and permissions as an administrator.
To make a user a subadministrator, you must:

1. Include the Subadministrator group in the Group list in the User form entry for every user who is to be a subadministrator. A member of the Subadministrator group must have a fixed license.

2. Give a group, of which the user is a member, subadministrator access to the appropriate applications, forms, and guides. For more information, see “Defining additional administrator permissions” on page 92.
Subadministrators can perform the following functions:

- Administer any application or form to which their group has subadministrator access.
- Create and administer filters, active links, and escalations connected to forms to which their group has subadministrative access.
- Create and administer menus.
- Create forms.
- Create applications (depending on the forms to which the subadministrator has access).
- Create active link and filter guides.
- View server information settings.

Members of the Subadministrator group cannot perform the following functions:

- Change server information settings.
- Release licenses of users currently accessing AR System.
- Administer applications and forms for which they are not subadministrators.

Although subadministrators can change certain properties of applications and change which forms are included in an application object, they cannot alter the forms themselves unless they have subadministrator access to them.
Additive access control

Access control in AR System is additive. This means that each user in AR System begins with no permissions. Administrators then add permissions as needed.

The server verifies the permissions of an object to determine if access to the object is granted. If access is granted at any step along the decision tree, as shown in Figure 2-3 on page 59, the user has permission to access the object. As you add permissions to various AR System objects, users have access to the object if they are members of a single group or role with access.

In the following figure, Lydia Lan is a member of two groups: Engineering and Engineering Managers.

Figure 2-2: Additive permissions
In this example, the Engineering group does not have access to Form1, but the Engineering Managers group does. Thus, although Lydia does not have access to Form1 through the Engineering group, she does have access through the Engineering Managers group.

You must assign permissions to every application, form, field, active link, active link guide, packing list, and web service. This can seem like a daunting task. Assigning default group permissions in the development process will save you time and prevent potential errors. Also, utilizing batch permissions or the permission objects themselves makes it easier to assign permissions after the fact. For more information, see “Assigning permissions—Four approaches” on page 86.

Access to AR System objects

You define permissions for applications, forms, fields, active links, active link guides, packing lists, and web services. Filters, filter guides, and escalations do not have permissions because these objects operate on the server. Menus also do not have (or need) permissions because they are attached to fields that have permissions.

Form, active link guide, and application permissions

Permissions determine which access control groups can access forms, active link guides, or applications in the user client. If a user does not have access to the object, it does not appear in the home page or in the Object List dialog box in BMC Remedy User.

When creating a form, active link guide, or application, you must decide the permission for each group or role:

- **Visible**—Members of the group or role can select and view the object in the user client.
- **Hidden**—Members of the group or role can access the object through workflow, but cannot select the form in the home page or in the Object List dialog box in BMC Remedy User, or open it in a web client.
- **None**—Members of the group or role have no access to the object.

Giving a user access to a form does not automatically give that user access to the fields or active links in that form. You must grant permission for each object individually.
When you log in as a member of the Administrator group, all objects are displayed in your client by default, whether they have Hidden or Visible access. BMC Remedy User includes a setting that enables you to display only those items for which you have Visible access. This setting is also available to subadministrators, although subadministrators can only manipulate the visibility of those forms for which they have access as a group. For more information, see the procedure “To change the visibility of hidden forms within BMC Remedy User for administrators” on page 93.

The following figure lists the questions that you can ask to determine the access that users have to forms in AR System. You can use this flowchart for guides and applications as well.
Figure 2-3: Accessing forms, guides, and applications

- **Are you an administrator?**
  - Yes: Full administrator access to all forms in the system
  - No: Continue

- **Are you a subadministrator?**
  - Yes: Full Administrator access to form
  - No: Continue

- **Are you in a group with subadministrator permissions to form?**
  - Yes: Full Administrator access to form
  - No: Continue

- **Is form hidden or visible?**
  - Visible: Can see form in the Open dialog box in Remedy User
  - Hidden: Can see form in Remedy User

- **Does Public group have permissions to form?**
  - Yes: Can see form in Remedy User
  - No: No access to form
Field permissions

Field permissions determine the types of access groups or roles have for individual fields in a form:

- **View**— Users can read the contents of the field.
- **Change**— Users can read and write the contents of the field.

If neither permission is selected, members of the group or role cannot view or change the field.

Groups and roles are defined with maximum privileges of View or Change, as explained in “To define default permissions” on page 88 and in the example on page 63. Groups or roles with maximum View permission can never be assigned Change permission for a field; groups or roles with maximum Change permission can be assigned Change, View, or no permission for a field.

Users must belong to a group or role with permission to view a form’s Request ID field (core field 1), or they cannot access any information from that request. After you give a group or role access to the Request ID field, or to any field in the form, the user does not automatically have access to the form or to workflow attached to the field. You must grant permissions to each object individually.

**Note:** In a Set Fields operation, because active links execute with the permissions of the user, field values set through an active link are updated only if the user has permission to change the field. Values retrieved must be accessible by the user. For more information, see the Workflow Objects guide.

The following figure lists the questions that you can ask to determine the access that users have to fields in AR System. Some of the questions are covered in the Configuring guide.
Figure 2-4: Accessing fields

1. Are you an administrator?
   - Yes: Full administrator access to all fields in all forms
   - No: Are you a subadministrator?

2. Are you a subadministrator?
   - Yes: Are you in a group with subadministrator permissions to form?
     - Yes: Full administrator access to all fields in form
     - No: Are you in a group with change permissions to field?
   - No: Are you in a group with change permissions to field?

3. Are you in a group with change permissions to field?
   - Yes: Do you have a write license?
     - Yes: Can change data in field
     - No: Can view data in field
   - No: Allow any user to submit?

4. Allow any user to submit?
   - Yes: Can change data in field during submit
   - No: Are you in a group with view permissions to field?

5. Are you in a group with view permissions to field?
   - Yes: Can view data in field
   - No: Without permissions, no access at all to field

6. Do you have a restricted read license?
   - Yes: View access only to data in field
   - No: Are you submitter and is submitter mode set to locked?

7. Are you submitter and is submitter mode set to locked?
   - Yes: Can change data in field
   - No: View access only to data in field
Advanced data fields

Advanced data fields require you to set permission on various levels. The advanced data field types are table fields, alert lists, results lists, page fields, and attachment pools. For example, a page field consists of three levels, each requiring consistent permission settings: the page holder, the page, and the fields on the page (so the user can see the complete tab set). See Chapter 4, “Types of fields,” for more information about the following advanced fields.

Table field permissions properties

Table field, alert list, and result list permissions are set in the same way as button field permissions, with the exception that you must set permissions at four levels. You must grant or deny a user access to the:

- Field or list.
- Columns in the field or list.
- Form from which rows are drawn.
- Fields from which each column draws its data.

The following examples explain the permission hierarchy:

- If a user does not have permission to view any columns, the field or list will appear blank in the user client.
- If a user does not have permission to access a field in the supporting form that contains column data, the user will see a blank cell.
- If the user has no permission to access any of the cells in a row, the row will not be displayed.

Page field permissions properties

Page field permissions are set at three levels. You must grant or deny a user access to:

- The page holder.
- The individual pages.
- Each field in each individual page.

To see an individual field, which is the lowest level of the hierarchy, the user must have permission to the upper levels of the hierarchy—that is, to the page holder and the individual pages.
Attachment pool permissions properties

For attachment pool field and attachment field permissions, you must grant or deny a user access to both.

To see an individual attachment field, the user must have permission to the attachment pool field.

If a user does not have permission to view any attachment fields, the attachment pool will appear blank in the user client.

Special submit setting

A special submit setting allows users to submit a new request without Change permission for fields that require a value. To use this feature, select the Allow Any User to Submit check box of the Field Properties dialog box for each applicable field.

If you do not select the Allow Any User to Submit check box and the field requires a value, a user must have a Write license and belong to a group with Change permission for the field to submit a request. For more information about using this feature, see “Defining default permissions” on page 87 and “Defining permissions for individual or multiple AR System objects” on page 90.

Field permissions example

Figure 2-5 illustrates how both permissions and field definitions work together to determine the access to a field. The example lists three groups: Browser, CS Staff, and Sales Staff. These groups have different maximum privileges of View or Change, as explained in “To define default permissions” on page 88.
Figure 2-5: Specifying field access control

At the field level, each group has been granted specific access to the Short Description data field:

- CS Staff group — Change
- Sales Staff group — View
- Browser group — View (Because the Browser group has a maximum access of View, Change access at the field level is not possible.)

John is a member of the CS Staff group and the Browser group. Although membership in the Browser group alone does not allow him to change the field, he can change it because of his group permission in the CS Staff group. When a user belongs to more than one group with different permissions to a field, the user has the highest level of permission granted by a group to which the user belongs.

Alice is a member of the Sales Staff group, which has maximum permission of Change. However, at the field level, members of the Sales Staff group can only view the contents of this field.

Rick also can only view the contents of the Short Description field because he is a member of the Browser group. Because the Browser group has maximum privileges of View, you can never give him Change permission for the Short Description field through the Browser group as it is currently defined.
Active link permissions

When you create an active link, you must define which groups or roles have access to it. A group or role needs permission to execute an active link.

After you give a group or role access to an active link, the user does not automatically have access to the field to which the active link is attached or to the form that contains the field.

The following figure lists the questions that you can ask to determine the access that users have to active links in AR System.

Figure 2-6: Accessing active links

- Are you an administrator? If yes, you can execute all active links. If no, proceed to the next question.
- Are you a subadministrator? If yes, you can execute all active links connected to forms you administer. If no, proceed to the next question.
- Are you in a group with subadministrator permissions to form? If yes, you can execute all active links connected to forms you administer. If no, proceed to the next question.
- Are you in a group with permissions to active link? If yes, you can execute active link. If no, you cannot execute active link.
Access to requests

Defining access to requests is important when you want to keep certain groups of users from knowing that certain requests exist. For example, if you use AR System as the outsource help desk for several companies, you can define access to requests so that only the company that submitted the request can see it.

You determine which groups or roles have access to a request through the Request ID field (field ID 1). If a group or role does not have access to that field, the group or role has no access to the request, even if it has access to other fields in that form.

You can grant access to members of explicit groups or roles. For example, you can give managers access to all requests. You can also grant access to members of implicit groups. For example, submitters can see their own requests but not those submitted by other users. For more information, see “Controlling access by using implicit groups—Row-level security” on page 67.

The following figure lists the questions that you can ask to determine the access that users have to requests in AR System.
Controlling access by using implicit groups—Row-level security

You can limit access to requests on a per-group or per-user basis. (This is often described as “row-level access.”) Membership in some groups (and their corresponding permissions) is implied when specific values are entered into certain AR System fields. The following table shows the differences and similarities among these implicit groups and their associated fields.

<table>
<thead>
<tr>
<th>Implicit Group</th>
<th>Group ID</th>
<th>Associated Default Field Name</th>
<th>Field ID</th>
<th>Core Field?</th>
<th>Associated Field Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submitter</td>
<td>3</td>
<td>Submitter</td>
<td>2</td>
<td>Yes</td>
<td>User name</td>
</tr>
<tr>
<td>Assignee</td>
<td>4</td>
<td>Assigned To</td>
<td>4</td>
<td>Yes</td>
<td>User name</td>
</tr>
<tr>
<td>Assignee Group</td>
<td>7</td>
<td>None</td>
<td>112</td>
<td>No</td>
<td>User, group, or role names</td>
</tr>
<tr>
<td>Dynamic groups</td>
<td>60000–60999</td>
<td>None</td>
<td>60000–60999</td>
<td>No</td>
<td>User, group, or role names</td>
</tr>
</tbody>
</table>
Note: In pre-6.0 clients, dynamic group fields behave as regular character fields, since these clients do not recognize the dynamic group reserved field ID range. Also in pre-6.0 clients, user names and role names are not recognized in the Assignee Group field.

Submitter and Assignee access

The Submitter and Assignee groups allow access to requests or fields on a per-user basis.

- To have access as a member of the Submitter group, the contents of Field ID 2 must be the user’s login name. Field ID 2 is usually set on submission by using the $USER$ keyword to define this field’s contents.

- To have access as a member of the Assignee group, the contents of Field ID 4 must equal the user’s login name. Field ID 4 is often set manually or by workflow to a user’s name when the status of the request changes.

Assignee Group access

The Assignee Group group allows access to requests or fields on a per-user or per-group basis.

To provide Assignee Group access, you must add the Assignee Group field (field ID 112) to your form and then specify which users, groups, or roles should have access to the request in this field. Although you can set the Assignee Group field manually, it is typically set by workflow. For more information, see “Using the Assignee Group and dynamic groups—Examples” on page 72.

Dynamic group access

Dynamic groups are similar to the Assignee Group (field ID 112), except that they are defined by the developer and include field and group IDs in the range of 60000 to 60999. For example, when you create a group with group ID 60000, its user list includes the individual user name or the members of any group or role that appears in field ID 60000. For more information, see “Using the Assignee Group and dynamic groups—Examples” on page 72.
**Form and Application Objects**

**Using the Request ID field with implicit groups**

Using implicit groups to control access to requests is a powerful method of access control within AR System. The Request ID field plays a key role in access control because a user cannot see a request unless the user belongs to a group with permission for its Request ID field.

**Defining access to requests at the user level**

You can link access control to a user’s login name:

- If you want submitters or assignees to have access to their requests on a single-user basis, grant the Submitter and Assignee groups permission to the Request ID field.
- If you want other users to have access, grant the Assignee Group or dynamic groups access to the Request ID field. Make sure that you also add Field ID 112 (the Assignee Group field) or the correct dynamic group fields to the form.

If you are entering a user’s login name to define access, remember these tips:

- In the Submitter or Assignee fields, enter the user’s login name without quotation marks.
- In the Assignee Group or dynamic group fields, enter the user’s login name in single quotation marks. Double any single quotation marks that are part of the login name (for example, ‘Dan O’Connor’).

**Defining access to requests at the group level**

Unlike Submitter and Assignee access, Assignee Group and dynamic group access can extend access control on a conditional basis by using explicit group and role membership.

You can enable the Enable Multiple Assign Groups setting in the Server Information window to allow multiple user, group, and role names in the Assignee Group field and dynamic fields. To enter users Dan O’Connor and Mary Manager, group ID 12000, role ID -9000, and role Managers, use the following syntax:

‘Dan O’Connor; Mary Manager; 12000; -9000; Managers’
**Note:** If a group and role have the same name, the role name will be assumed. For example, if a dynamic field contains **Managers; Sales**, AR System assumes the Managers and Sales roles, if they exist; otherwise, AR System assumes the Manager and Sales groups.

For more information about all settings in the Server Information window, see the Configuring guide.

Assignee Group and dynamic group permissions to the Request ID field, combined with the contents of the Assignee Group field or dynamic group fields, determines who can see the request. If a group or role to which the user belongs is in the Assignee Group or dynamic group field for a request, that user is given whatever access privileges you defined for the Assignee Group or dynamic group, as shown in the following figure.
AR System checks permissions for Request ID field

- Yes
  - Your name in Field ID 2?
    - Yes
      - Your name in Field ID 4?
        - Yes
          - Your name, group, or role in Field ID 112?
            - Yes
              - Assignee group has access?
                - Yes
                  - Can access request
                - No
                  - No
            - No
        - No
    - No
  - No
  - No

- No
  - Assignee group has access?
    - Yes
      - Can access request
    - No
      - No
  - No

- No access to request

* Unless a member of another group with access to Request ID field
** or dynamic groups if applicable
*** or dynamic group fields (ID 60000-60999) if applicable
Using the Assignee Group and dynamic groups—Examples

Use Assignee Group access, dynamic group access, or both to set up permissions so that users have conditional access to requests.

Imagine that you are working for Acme Outsource Help Desks, Inc. Three computer companies hire you to manage all of their help desk responsibilities. For security reasons, each computer company must not know about the existence of the other two. Therefore, you must create one form all three companies can use, but allows each company to see only the requests they create.

Explicit groups for each company have already been created, and each user belongs to one of these company groups.

To use the Assignee Group to control access to requests

1. Create the groups (or roles) and users to which you want to assign access.

   In this example, the four groups are:
   - Acme Help Desk Staff (who will have access to all requests)
   - Beta Computers
   - Gamma Computers
   - Delta Computers

   Beta Computers, Gamma Computers, and Delta Computers users must belong only to their company group. Acme employees can be members of more than one group.

2. Create a form, and give the appropriate groups Visible permission for it.

   For example, giving the Public group Visible permission for the form will enable all of the users to see it.

3. Add Assignee Group access to the form.

   The Assignee Group capabilities of AR System are activated when you add a character field to the form with Field ID 112 and a database input length of 255.

4. Restrict access to the necessary requests.
Because only groups or roles with permission for the Request ID field can access a request, restricting access to the Request ID field is the key to restricting access to a request. In this example, the Acme Help Desk Staff and the Assignee Group groups have the appropriate permissions for the Request ID field, as shown in the following figure.

**Figure 2-9: Field permissions for the Request ID field**

With Assignee Group access, a user from Beta Computers can submit requests, and anyone from Beta Computers can query them. However, no one from Gamma Computers or Delta Computers can query Beta Computer's requests.

**Note:** You might want to give permissions to a single group to begin with and submit a sample request to determine whether any group other than the designated group can access it.
5 Add workflow that inserts at least one explicit group, role, or user name into Field ID 112 according to the business rules at your site. If your configuration settings in the Server Information window are enabled to allow multiple assignee groups, you now can enter more than one explicit group, role, or user name into Field ID 112. For sample syntax, see “Defining access to requests at the group level” on page 69.

For more information about all settings in the Server Information window, see the Configuring guide.

**Note:** Running BMC Remedy User versions 4.5.1 or lower against a version 6.x or later server with the Enable Multiple Assign Groups option selected can cause groups to be erased from the field. Older BMC Remedy User clients do not understand multiple groups in Field ID 112.

Because Field ID 112 is designed for administrators and your help desk staff, deny access for most groups to this field. You can define a filter to set the contents of this field and use an active link Change Field action to allow your help desk staff to see and change the field as needed. If you must change the group or role in the field, Field ID 112 includes system-defined menus of all groups on the server and roles in the application (if the form is owned by a deployable application). Administrators can override these menus in BMC Remedy Administrator as needed.

In the example, Acme allows access to its service call database from the web but limits users to view only requests submitted by members of their company. An access control group was created for each different company name, and a filter that copies the company name into Field ID 112 (labeled Assignee Group in the following figure) executes when users submit requests.
When the filter executes, the Assignee Group for this request will be Beta Computers.

You also could have created individual filters, one that enables Beta Computers to see their requests, another that enables Gamma Computers to see their requests, and so on. Use appropriate filter qualifications to make sure that only users from the Beta Computers group can execute the filter, set Field ID 112 to “Beta Computers,” and so on. For more information about creating and using filters, see the Workflow Objects guide.

6 Change the permissions of other fields in the form to grant access as needed. Include the Assignee Group where appropriate.

As a result of carefully defining access control in your system, when members of Acme Outsource Help Desks search all open help desk requests, they will see a results list that includes requests submitted by Beta, Gamma, and Delta Computers. In contrast, if users from Delta Computers perform the same search, they will see only the requests where Delta Computers is the Assignee Group (that is, the requests submitted by anyone at Delta Computers).
To use a dynamic group to control access to requests

1. Create the groups (or roles) and users to which you want to assign access.
2. Create a dynamic group in the Group form.
   For example, create a group called Dynamic Access with a group ID of 60001.
3. Create a form, and give the appropriate groups Visible permission for it.
4. Add dynamic group access to the form.
   To activate the dynamic group, add a character field to the form with Field ID 60001, the same ID number as the dynamic group ID.
5. Restrict access to requests by specifying dynamic group access to the Request ID field.
6. Add workflow that inserts at least one explicit group name or ID, role name or ID, or user name into Field ID 60001 according to the business rules at your site. If your configuration settings in the Server Information window are enabled to allow multiple assignee groups, you can enter more than one explicit group, role, or user name into Field ID 60001. For sample syntax, see “Defining access to requests at the group level” on page 69.

   For more information about all settings in the Server Information window, see the Configuring guide.

   Like Field ID 112, dynamic group fields can be modified manually. They include system-defined menus of all groups on the server and roles in the application (if the form is owned by a deployable application). Administrators can modify these menus as needed.

Putting it all together

The following figure presents an overview of access control, and lists the questions that you can use to determine the access that users have to AR System.
Figure 2-11: Access control overview

- **Server Access**
  - Does login name match user definition? (Yes → Form Access, No → Are guests allowed to log in?)
  - Does password match? (Yes → Form Access, No → No access to server)

- **Form Access**
  - Do you have access to the form? (Yes → Request Access, No → No access to form)

- **Request Access**
  - Is the request viewable? (Yes → Field Access, No → No access to request)

- **Field Access**
  - Is the field viewable? (Yes → Form, request, and field accessible, No → No access to field)

- **Decision Points**
  - Are guests allowed to log in? (Yes → Server Access, No → No access to server)
  - Does login name match user definition? (Yes → Form Access, No → Are guests allowed to log in?)
  - Does password match? (Yes → Form Access, No → No access to server)
  - Do you have access to the form? (Yes → Request Access, No → No access to form)
  - Is the request viewable? (Yes → Field Access, No → No access to request)
  - Is the field viewable? (Yes → Form, request, and field accessible, No → No access to field)
Creating and managing groups

Use the **Group form** (shown in the following figure) in BMC Remedy User or a web client to create and manage the access control groups to which you grant or deny access to AR System objects.

Creating groups

This section provides the steps to create AR System access control groups. Although there is no limit to the number of groups that you can create, for maintenance purposes you might want to limit the number to avoid confusion. After you have created the necessary groups, use the steps described in the Configuring guide to assign individual users to the appropriate groups.

Figure 2-12: Group form— New mode
The following table lists the key fields in the Group form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>Name of the access control group. Use this name in the Group list field in the User form and in the Permission and No Permission lists when you are defining AR System object permissions. Every group name should be different. Use caution when creating group names that include spaces, because group names in the Group list field on the User form are separated by spaces. For example, if you have a group named CUSTOMER SUPPORT, you should not create a group named CUSTOMER or a group named SUPPORT.</td>
</tr>
</tbody>
</table>
| Group ID       | Integer ID that is the recognized identity of the group. The ranges are:  
  - 1000-14999 — For regular and computed groups  
  - 60000-60999 — For dynamic groups  
  - < 1000 — For AR System groups and current AR System applications  
  - >14999 (not including dynamic groups) — For future AR System applications  
  If you use the same ID with multiple group names, you must keep the Group type the same for each because you are creating aliases for the same group.  
  To make sure that you do not create duplicate Group IDs, use BMC Remedy Administrator to build a unique index on the Group ID field in the Group form. (For more information, see “To define indexes for a form” on page 140.) |
| Group Type     | Maximum permission type intended for the group. Your choices are None (no access), View (view field contents), and Change (modify field contents).  
  Specify None to disable all access for the group without deleting the group itself. The group remains as a placeholder (and can be restored in the future), but all permissions for the group are lost. Create a group with the type None if you want to define a group that will be used only for notifications.  
  For more information about field permissions, see “Field permissions” on page 60. |
| Long Group Name| Additional information about a group. The text should be descriptive of the group because it appears by default in the Results pane in BMC Remedy User when listing groups. |
Chapter 2—Defining access control

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Category</td>
<td>The group category, such as Regular, Dynamic, or Computed, which is described in “Groups you create—Regular, computed, and dynamic” on page 52. To define a dynamic group, use a group ID in the range of 60000 to 60999. On the form containing requests to which you want to define row-level access, add a field with a field ID that is the same as the dynamic group ID. You can populate a dynamic group field with a group name, role name, or the name of an individual user. Dynamic Groups are used only to control access to requests (row-level access). To define a computed group, select Computed Group as the group category and enter a qualification string in the Computed Group Definition field.</td>
</tr>
</tbody>
</table>
| Computed Group Definition | Qualification string that defines a computed group. Construct the string from any valid combination of explicit group IDs, explicit group names (in double quotation marks), user names (in single quotation marks), and operators such as AND, OR, and AND NOT. Optionally, use the AND, OR, NOT, Append Group, Append User, and parentheses buttons to build the qualification string. For example:  
  12000 OR 12001 includes all users in group ID 12000 or group ID 12001.  
  “A” OR “B” includes all users in group A or group B.  
  “A” AND “B” includes only those users in group A and group B.  
  (“A” OR “B”) AND NOT “C” includes all users in groups A or B, except for those users who are in group C.  
  “A” OR ‘Mary Manager’ includes all users in group A, and the user Mary Manager. |
| Floating Licenses   | Number of floating licenses reserved for the group. See the Installing guide for more information. If this field is missing from the Group form, you can add it in BMC Remedy Administrator. Use field ID 115. See “Creating data fields” on page 174. |
To create groups

1. Log in to BMC Remedy User or a web client as an administrator.
2. Open the Group form in New mode on the appropriate server.
3. Enter information in the appropriate fields, as described in the previous table.
   
   If attributes that you want to specify in the group definition are not represented in the Group form, you can use BMC Remedy Administrator to add the appropriate fields. However, be careful that you do not modify or delete any of the original fields or field IDs.
4. Save your changes.
5. For a regular group, assign users to it by using the User form in BMC Remedy User or a web client.

   After you save a new group, the server automatically recaches, and this group appears in the Group menu in the User form after a short amount of time without having to log in again. For more information about adding users, see the Configuring guide.
6. To enable a dynamic group, add a field to the form with a field ID that is the same as the group ID. For more information, see “Group Category” on page 80.
Managing groups

You can modify, delete, or search for groups in the Group form.

► To modify groups

1 In BMC Remedy User or a web client, open the Group form from the appropriate server in Search mode.
2 Perform a search to retrieve a list of currently defined groups.
3 Select the appropriate group from the list.
4 Modify information in the appropriate fields.

Note: Permissions for a user are determined by the list of groups in the Group list field of the user’s entry in the User form. If you later change the Group ID for a group, the users originally assigned to the group will still be attached to the old ID. If there is no group with the old ID, these users are no longer attached to any group.

5 Save your changes.

► To delete groups

1 In BMC Remedy User, open the Group form from the appropriate server in Search mode.
2 Choose Actions > Search to retrieve a list of currently defined groups.
3 Select the appropriate group from the list.
4 Choose Actions > Delete.
   A confirmation box appears to verify that you want to delete the group entry.
5 Click OK.

► To search for groups

1 In BMC Remedy User, open the Group form from the appropriate server in Search mode.
2 Enter values in fields, or use the Advanced Search Bar to specify search criteria.
For computed groups, enter one group ID or one user name (in single quotation marks) in the Computed Group Definition field. If you use the Advance Search Bar, use the LIKE operator to indicate that you are searching for a portion of a string (see the Workflow Objects guide). The search will return all computed groups that include the specified user or group in the definition.

You cannot search the Computed Group Definition field for group names, or for strings that include operators such as AND, OR, and NOT. This is because AR System converts group names to group IDs and encodes operators before storing them in the database. However, the search results will show the strings as they were originally entered, with group names and operators.

**Note:** Informix databases do not support searches on the Computed Group Definition field.

3. Choose Actions > Search to retrieve the list of currently defined groups that match your search criteria.

For more information about performing searches, see BMC Remedy User Help.

### Creating and mapping roles

Use the **Roles form** (shown in the following figure) in BMC Remedy User or a web client to create roles to which you grant or deny access to objects in deployable applications. In deployable applications, you assign permissions using implicit groups (including dynamic groups) and roles. You then map roles to explicit groups on the server. For more information about deployable applications, see “Deployable applications” on page 29.

This section provides the steps to create roles and map them to explicit groups. Although there is no limit to the number of roles that you can create, for maintenance purposes you might want to limit the number.

You can map roles to regular or computed groups for the Test and Production application development states. You can also create custom states and map roles for those states, as explained in the Forms and Applications guide. To enable a particular mapping, change the application’s state, as described in “Working with deployable application states” on page 30.
The following table lists the key fields in the Roles form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Name</td>
<td>Name of the deployable application for which the role is defined. You can define the same role for multiple applications, but you must create a separate Roles form entry for each.</td>
</tr>
<tr>
<td>Role Name</td>
<td>Name by which the role is known. Within each application, every role name should be unique. You can reuse the same role name-role ID pairs across a suite of applications.</td>
</tr>
<tr>
<td>Role ID</td>
<td>Integer ID that is the recognized identity of the role. The ID must be a negative number, such as -10001. Role IDs must be unique for each application name. You can reuse the same role name-role ID pairs across a suite of applications.</td>
</tr>
</tbody>
</table>
Creating and mapping roles

1. Log in to BMC Remedy User or a web client as an administrator on the server that contains the deployable application for which you are creating roles.

2. Open the Roles form in New mode.

3. Enter information in the Application Name, Role Name, and Role ID fields, as described in the previous table.

   If you save the role now, you can begin assigning permissions for this role to objects within the application. A role is only visible in the Application window for the deployable application to which the role belongs.

4. Enter a regular or computed group ID in each Mapped Group field to define access permissions for each application state.

5. Save your changes.

Note: Newly created roles appear in the Permissions tab of object properties after the server recaches (about 5 seconds, depending on your system).

To modify roles and role mappings

1. Log in to BMC Remedy User or a web client as an administrator on the server that contains the deployable application for which you are modifying roles.

2. Open the Roles form in Search mode.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test</td>
<td>Enter or select one group name for the regular or computed group to which you want to map this role for the Test application state. To enable this mapping, set the application’s State property to Test. For more information, see “Working with deployable application states” on page 30.</td>
</tr>
<tr>
<td>Production</td>
<td>Enter or select one group name for the regular or computed group to which you want to map this role for the Production application state. To enable this mapping, set the application’s State property to Production. For more information, see “Working with deployable application states” on page 30.</td>
</tr>
</tbody>
</table>

To create and map roles

1. Log in to BMC Remedy User or a web client as an administrator on the server that contains the deployable application for which you are creating roles.

2. Open the Roles form in New mode.

3. Enter information in the Application Name, Role Name, and Role ID fields, as described in the previous table.

   If you save the role now, you can begin assigning permissions for this role to objects within the application. A role is only visible in the Application window for the deployable application to which the role belongs.

4. Enter a regular or computed group ID in each Mapped Group field to define access permissions for each application state.

5. Save your changes.

Note: Newly created roles appear in the Permissions tab of object properties after the server recaches (about 5 seconds, depending on your system).
Chapter 2—Defining access control

3 Search the form to retrieve a list of currently defined roles for a particular application.
4 Select the appropriate roles and modify information in the appropriate fields.
5 Save your changes.

To delete roles

1 Log in to BMC Remedy User or a web client as an administrator on the server that contains the deployable application for which you are deleting roles.
2 Open the Roles form in Search mode.
3 Search the form to retrieve a list of currently defined roles for a particular application.
4 Select the appropriate role.
5 Choose Actions > Delete.
   A confirmation box appears to verify that you want to delete the role entry.
6 Click OK.

Assigning permissions—Four approaches

You define permissions for applications, forms, fields, active links, active link guides, packing lists, and web services. BMC Remedy Administrator includes four ways to define or modify specific permissions for objects:

- **Default permissions**—The permissions that you set for a new object. Once set, every time you create a new object, the groups or roles to which you gave default permission can access them without additional access modification. Defining default permissions is optional, but can be useful if you have many groups or roles. If you do not set default form permissions, only administrators (and subadministrators with access to the form that contains the active link or field) can access the object until you define specific permissions for it. The steps for this option are described in “Defining default permissions” on page 87.

- **Permissions for individual objects**—You can specify which groups or roles can access an object when you create or modify the object. This enables you to evaluate which users need access to an object. The steps for this option are described in “To define permissions for individual objects” on page 90.
Assigning permissions—Four approaches

- **Batch permissions**— You can specify permissions for multiple objects of the same type at the same time. For more information, see the Getting Started guide.

- **Group and role permissions**— You can give a group or role access to every applicable object within a server instead of opening each object and modifying the permissions individually. This method can be useful if you have added a new group or role after the objects were created. The steps for this option are described in “To define permissions for multiple objects” on page 91.

**Defining default permissions**

Use the Default Permissions dialog box to define initial or general permissions for new objects.

**Figure 2-14: Default permissions**

![Default Permissions dialog box](image)
When you create objects, the defined default permissions are automatically displayed in the Permissions tab of the Properties dialog box. If you have an existing object and want to reset modified permissions back to the defined default permissions, click the Set to Defaults button in the Properties dialog box. For additional information about permissions, see the following sections:

- “Form, active link guide, and application permissions” on page 57
- “Field permissions” on page 60
- “Active link permissions” on page 65
- “Defining permissions for individual or multiple AR System objects” on page 90

**To define default permissions**

1. In BMC Remedy Administrator, open the Default Permissions dialog box.
   - In the Server Window, select a server and choose File > Default Permissions.
     In the Server window, you define default permissions for the current administrator login, allowing each administrator to have unique default permissions for objects created.
   - In an Application Window, choose File > Default Permissions.
     In a local Application window, you define default permissions for the current administrator login.
     In a deployable Application window, you define the default permissions for that application for any administrator login. This keeps permissions consistent throughout the application’s development.

2. Click the Permissions tab for the appropriate object.
   All groups defined on the server (or roles defined for a deployable application on the server) are displayed.
For each group or role in the Permission list, an access icon might appear to the left of the group or role name, as described in the following table.

<table>
<thead>
<tr>
<th>Access Icon</th>
<th>Meaning</th>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Visible</td>
<td>Applications</td>
<td>Group or role has permission to view and select the object in the user client.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Forms</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Active link guides</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web services</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hidden</td>
<td>Applications</td>
<td>Group or role has access to the object through workflow, but it will not be present in the user client.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Forms</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Active link guides</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Web services</td>
<td></td>
</tr>
<tr>
<td></td>
<td>View</td>
<td>Fields</td>
<td>Group or role has permission to view the field.</td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>Fields</td>
<td>Group or role has permission to view and change the field.</td>
</tr>
<tr>
<td></td>
<td>No icon</td>
<td>Active links</td>
<td>The objects offer permission or no permission, so no icon is displayed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Packing lists</td>
<td></td>
</tr>
</tbody>
</table>

3 Use the Add and Remove buttons to move groups or roles between the Permission and No Permission lists.

4 To modify access, click the access icon to the left of a group or role, and choose the appropriate menu option.

5 For fields only, select or clear the Allow Any User to Submit check box on the Field tab.
   
   Use this mode to determine access control (security settings) for the field when a request is submitted. If the check box is:
   
   - Selected—Any user can assign a value to the field, regardless of whether the submitter belongs to a group with Change permission to the field.
   
   - Cleared (the default)—Only users who belong to one or more groups with Change permission to the field (or users who belong to groups mapped to roles with Change permission to the field) can enter data into the field.

6 Click OK.
Defining permissions for individual or multiple AR System objects

When you create an object in AR System, default permissions are assigned. You can change these permissions for individual objects or multiple objects, as described in the following procedures.

To define permissions for individual objects

1. In BMC Remedy Administrator, select a server to administer.
2. Open the object for which you want to change permissions.
   For forms and fields, open the Properties window, which contains the Permissions tab you need to change permissions.
   - For forms, choose Form > Form Properties, or double-click on the background of the form.
   - For fields, choose Form > Field Properties, or double-click on the field.
   For other objects, the Permissions tab is available when you open the object.
3. Click the Permissions tab.
   All groups defined on the server (or roles defined for the application that contains the objects) are displayed.
   The Submitter, Assignee, and Assignee Group groups are implicit based on field contents. Accordingly, these groups will appear only for Field Permissions. They do not apply for other objects. For more information about implicit groups, see “Controlling access by using implicit groups—Row-level security” on page 67.
4. Use the Add and Remove buttons to move groups or roles between the Permission and No Permission lists.
   To allow all users to see a form, active link guide, or application, add the Public group to the Permission list.
5. To modify access, click the access icon to the left of a group, and choose the appropriate menu option.
   For information about using the Visible and Hidden icons, see “Defining default permissions” on page 87.
6 For fields only, select or clear the Allow Any User to Submit check box. Use this mode to determine access control (security settings) for the field when a request is submitted. If the check box is:

- Selected—Any user can assign a value to the field, regardless of whether the submitter belongs to a group with Change permission to the field.
- Cleared (the default)—Only users who belong to one or more groups with Change permission to the field can enter data into the field.

7 To reset the object to the default permissions, click Set to Defaults.

For more information, see “Defining default permissions” on page 87.

8 Click OK (for forms) or click the Save toolbar button (for all others).

**To define permissions for multiple objects**

1 In BMC Remedy Administrator, expand the tree in the Object tab.

2 Click the Groups or Groups/Roles category.

   The Groups category is available in the Server window or in a local Application window. The Groups/Roles category is available only in the deployable Application window. For more information, see “Roles in AR System” on page 53.

3 Double-click the appropriate group or role to open the Group Permissions or Group/Role Permissions window.

4 Click the appropriate tab.

   Each tab lists the current permission settings that the selected group or role has for each object.

5 Use the Add and Remove buttons to move objects between the Permission and No Permission lists.

6 To modify access, click the access icon to the left of a group or role, and choose the appropriate menu option.

   For more information, see “Defining default permissions” on page 87.

7 Click Apply.
Defining additional administrator permissions

The following procedures explain how to define additional administrator permissions, such as granting subadministrator permissions to users, defining subadministrator permissions for forms and applications, and changing the visibility of forms for administrators.

After you have granted a group subadministrator permissions for an application, the members of that group can modify the application’s appearance, group permissions, help text, and change history. Subadministrators can also change which forms are included in an application object, although they cannot alter the forms themselves unless they have subadministrator access to them.

For more information, see “Subadministrators in AR System” on page 53.

To grant subadministrator capabilities to a user

1. In BMC Remedy User or a web client, open the User form from the appropriate server in Search mode.
2. Perform a search to find the person you want to give Subadministrator access to.
3. Make the following changes:
   - From the Group list menu, select Sub Administrator.
     - The list must include the Sub Administrator group to give the user the potential to be a subadministrator.
   - From the License Type option list, select Fixed.
     - You must assign subadministrators a Fixed Write license.
4. Save your changes.
5. Give subadministrator permission for the form to a group or role to which the subadministrator belongs, as described in the following procedure.

To define subadministrator permissions for forms and applications

1. In BMC Remedy Administrator, select a server to administer.
2. Open the form or application with which you want to work.
Defining additional administrator permissions

3 For forms, choose Form > Form Properties to open the Properties window, which contains the Permissions tab you need to change permissions.
   For applications, the Permissions tab is available when you open the object.
4 Click the Subadministrator Permissions tab.
5 Use the Add and Remove buttons to move groups between the Permission and No Permission lists.
   When setting permissions for an application, you must provide the same permission to the application as you do to the individual forms contained in the application.
6 Click OK (for forms) or click the Save toolbar button (for applications).
   The members of that group or role have the same privileges and permissions that an administrator has for that object.

To change the visibility of hidden forms within BMC Remedy User for administrators

1 Log in to BMC Remedy User as an administrator or subadministrator.
2 Choose Tools > Options.
3 Click the Advanced tab.
4 Select or clear the Display Hidden Forms check box. This check box is available only for administrators.

If the check box is:

- Selected (the default)—All forms are displayed in BMC Remedy User, even those with Hidden access.
- Cleared—Only those forms for which the Administrator or Subadministrator group has Visible access are displayed in BMC Remedy User.

5 Click OK.

For more information, see “Form, active link guide, and application permissions” on page 57.
You use forms to capture and display information. They typically include related components such as employee and department information. A form contains fields in which the information is entered and displayed. The collection of fields represents a record of information in AR System. While the entries comprise the rows of a database table, the fields comprise the columns.

This section discusses the types of forms available, and the tasks used to create them. The following topics are provided:

- Types of forms (page 96)
- Creating and managing forms (page 99)
- Display-only forms (page 104)
- Join forms (page 110)
- Setting form properties (page 131)
Chapter 3—Creating AR System forms

Types of forms

An administrator can create forms that serve as part of a unique workflow solution. Form types include regular, join, display-only, view, and vendor forms. These forms can be customized using form views, as explained in Chapter 8, “Creating and managing form views.”

Regular forms

Regular forms are generally the main forms of your applications. Within the AR System database, AR System builds and manages tables to store the data displayed on your forms.

When you create a regular form, you see the eight core fields, as shown in the following figure. All regular forms contain these fields. For information about core fields, see “Core fields” on page 354.

Join forms

AR System also enables you to create a join form, which is a form that combines information from multiple AR System forms. This composite form can consist of fields derived from other existing forms, and the resulting combined information is displayed in BMC Remedy User or a browser. Join forms enable you to avoid data redundancy (information is stored in only one form) and maintain data integrity (information updated through the join form is updated in all other places).
For example, you can combine the information from the Help Request and the Employee ID forms (as shown in the following figure) into a join form that displays information from both forms without duplicating employee information in every help request. You can combine a join form with other forms, or you can join a form to itself.

Figure 3-2: How joins work in AR System

For information about creating and using join forms, see “Join forms” on page 110.
Display-only forms

Display-only forms have no database table associated with them and have no associated requests. Display-only forms also have no core fields.

You can use display-only forms in various ways:

- **Control panels** (as shown in Figure 3-4 on page 105) — These provide an efficient way to organize and present users with specific tasks or objectives.

- **Dialog boxes** — These enable you to reuse specific groups of fields in a variety of ways. For example, you can create an employee information dialog box that contains generic fields (such as name and address) that could be used by multiple forms and applications.

- **Entry points to other forms that contain data** — You can add an OK or a Continue button to a display-only form to cause an active link to transfer data from the display-only form to the primary form and then submit a request.

View and Vendor forms

View and vendor forms allow you to access data outside of AR System. They can be used to:

- Execute workflow on creation and modification of data when the changes are performed using the view or vendor form.

- Execute escalations on external data.

- Access external data to populate search style character menus or table fields.

View forms are AR System objects that enable the AR System server to access external data sources through AR System forms. View forms allow AR System to point to, and access, data in an existing database table created outside AR System. The table can be located on the same server, or in any other database server accessible from the current AR System database server.

Vendor forms are AR System objects that enable the AR System server to access arbitrary external data sources through the use of an ARDBC plug-in. This type of form provides for easy integration with external data, without replicating the data.
Vendor forms connect to data sources that are exposed by an ARDBC plug-in. ARDBC plug-ins enable AR System to interface with external data sources such as LDAP directory services, legacy systems, spreadsheets, text files, or database tables.

For information about creating and using view and vendor forms, see the Integrating with Plug-ins and Third-Party Products guide.

Creating and managing forms

Forms are created and defined using the following steps. To make sure that all form components are properly defined, follow the steps in the order listed.

**Step 1** Create a form (see “Creating and managing forms” on page 99).

**Step 2** Set form properties (see “Setting form properties” on page 131).

**Step 3** Plan the layout of a form (see “Managing fields in a form view” on page 279).

**Step 4** Create fields on a form (see “Types of fields” on page 143).

**Step 5** Set field properties (see “Field properties” on page 397).

**Step 6** Create form views (see “Creating form views” on page 256).

Creating forms

When planning a form, sketch the layout before you begin creating fields so that you have an idea of the best field location and order. When deciding where to place fields that have menus, text editors, or diary editors associated with them, allow space for the icons that will appear next to the fields. Consider using trim (lines, boxes, or text) to group and label related fields. You can also add color to buttons and text labels. For information about form layout, see “Managing fields in a form view” on page 279 and the Getting Started guide.
To create forms in BMC Remedy Administrator

1. In BMC Remedy Administrator, select a server to administer.

2. Choose File > New Server Object to open the New Server Object dialog box.

3. Select the type of form from the list: Regular Form, Join Form, Display-Only Form, View Form, or Vendor Form.

4. Click OK.

   Depending on the type of form you selected, the following action occurs:
   - For regular forms, the Create Form window opens with the Core fields as displayed in Figure 3-1 on page 96.
   - For join forms, the Join Wizard opens. To continue, see “Creating join forms” on page 120.
   - For display-only forms, a blank form opens. To continue, see “Display-only forms” on page 104.
   - For view and vendor forms, other dialog boxes open. For information about view and vendor forms, see the Integrating with Plug-ins and Third-Party Products guide.

5. Choose File > Save Form to open the Save Regular Form As dialog box.

6. Enter the name of the new form in the Form Name field.

   Form names must be unique on each AR System server. Names can be as many as 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name.
When you enter a name in the Form Name field, that name is also entered into the Web Alias field. This field was used in versions prior to 6.3 for administrators who wanted to create and deploy web views of a form. As of version 6.3 of the mid tier, forms are no longer “deployed”; all forms can be viewed on the web without special aliases. For more information, see the Installing and Administering BMC Remedy Mid Tier guide.

7 Click OK to save the form.

The name of the new form appears in the title bar of the form and in the Forms list in the Server Window.

Be aware of the following issues when creating forms:

- Users who are logged in to BMC Remedy User when you are creating a new form will not see the new form until they log in again.
- The size of the window when you save the form is the initial size that users will see when they open the form in BMC Remedy User.

Modifying forms

The following procedure describes how to open all form types when changes or additions are necessary.

To modify forms in BMC Remedy Administrator

1 Click the Forms icon in the Server window, and double-click the form name in the Forms list.

   The default view of the selected form opens in the Modify Form window. For information about default views, see “Setting view properties” on page 263.

2 Choose Form > Manage Views to open the Manage Views dialog box.

3 Select the view you want to modify, and click the Display button.

   Forms defined using the Standard option open in a Modify Form window.

4 Make the necessary changes to the view.

   For information about the types of modifications you can make to a form view, see “Modifying form views” on page 261.

5 Choose File > Save Form to save the changes.

   Users who have a form open in BMC Remedy User when you are making form modifications must reopen the form to see your changes.
Copying forms

The procedure that follows describes how to create a duplicate of a form.

**Note:** The User and Group forms contain several reserved fields that make these forms unique. Do not copy these forms, or you might introduce unintended access control functionality into your environment. For information about the Group form, see “Creating groups” on page 78. For information about the User form, see the Configuring guide.

**To copy forms in BMC Remedy Administrator**

1. Click the Forms icon in the Server window, and double-click the form name in the Forms list.
   
   The default view of the selected form opens in the Modify Form window.

2. Choose File > Save Form As to open the Save Form As dialog box.

3. Enter the new name of the form in the Form Name field.
   
   When you enter a name in the Form Name field, that name is also entered into the Web Alias field. This field was used in versions prior to 6.3 for administrators who wanted to create and deploy web views of a form. As of version 6.3 of the mid tier, forms are no longer “deployed”; all forms can be viewed on the web without special aliases. For more information, see the Installing and Administering BMC Remedy Mid Tier guide.

4. Click OK.
   
   The new form is listed in the Forms list in the Server window.

   The new form retains all of the fields, views, and properties of the original form. Workflow objects (such as active links, filters, and escalations) associated with a form are not copied.

Renaming forms

Renaming a form enables you to change the name of an existing form without having to create a duplicate form under a different name. The renamed form retains all of the settings of the original form. Whenever you rename a form, any workflow that references the form is automatically updated with the new name of the form.
The procedure that follows describes how to rename a form without having to create a new copy.

**To rename forms in BMC Remedy Administrator**

1. Click the Forms icon in the Server window, and double-click the form name in the Forms list.

   The default view of the selected form opens in the Modify Form window.

2. Choose File > Rename Form to open the Rename Form dialog box.

3. Enter the new name of the form in the Form Name field, and click OK.

   The form is listed under the new name in the Forms list in the Server window.

---

### Deleting forms

The following procedure describes how to delete a form from a server.

**To delete forms in BMC Remedy Administrator**

1. Click the Forms icon in the Server window, and select the form you want to delete from the Forms list.

2. Choose Edit > Delete - Form(s) from the menu.

   The form is deleted from the Forms list in the Server window.

   When you delete a form, all associated data and workflow that are not associated with any other form are deleted. However, if the workflow is shared by multiple forms, it will not be deleted until the last form that uses it is also deleted. Menus and applications must be deleted separately because they are independent of forms.

   If you delete a primary or secondary form of a join, the join form is also deleted.

**WARNING:** Do not delete the User or Group forms, or you will lose the ability to add or modify users and groups. For more information about the Group form, see “Creating groups” on page 78. For information about the User form, see the Configuring guide.
Display-only forms

You can create display-only forms for various purposes. This section provides tips for creating these forms, and examples of how you can use them.

Creating display-only forms

You can create display-only forms that work in both New mode and Search mode in BMC Remedy User (as shown in Figure 3-4 on page 105). Creating display-only forms is similar to creating regular forms. For general information, see “Creating and managing forms” on page 99.

Be aware of the following issues when you create a display-only form:

- Unlike regular forms, display-only forms do not have Results List Fields or Indexes tab properties.
- By definition, all fields that you add are display-only.

Using a display-only form as a control panel

A display-only form can be used as a centralized entry point from which users choose the tasks they want to accomplish. When a form is used in this way, it is referred to as a control panel. A control panel might include tasks encompassing a variety of functional areas such as Help Desks, Employee Services, and Asset Tracking. Users select a functional area from the main control panel and fill in data on the form related to the specific task.
In this figure, the buttons on the display-only form act as entry points to multiple underlying forms.

**Note:** When using a display-only form as a control panel in an application, set the control panel form as the primary form. Consider hiding the Details Pane Banner of control panels so that users are not distracted by banner buttons. For information about how to hide the Details Pane Banner, see “Pane Banner Visibility” on page 267.

**Using a display-only form as a dialog box**

Dialog boxes require user interaction and are useful when you want to:

- Prompt users for confirmation.
- Implement a main-detail (or parent-child) relationship between forms where users can edit the main form using a details dialog box.
- Reuse a form in a variety of ways.
- Embed a table that lists options from which users can choose.
Provide a way for users to edit or view a rarely used set of fields, and thus avoid cluttering the main form.

Provide a way for users to view or edit a set of fields that handle complex calculations of multiple components.

To define a dialog box, set Window Type to Dialog mode and use the following active link actions:

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open Window</td>
<td>Sets the Open Window action to open a dialog box from a parent form. This action also defines what data will be transferred from the parent form to the dialog box when the dialog box opens, and what data will be transferred from the dialog box back to the parent form when the Commit Changes action occurs—usually when the user clicks an OK button or the dialog box closes.</td>
</tr>
<tr>
<td>Commit Changes</td>
<td>Changes the fields in the parent form to the values that the user specifies in the dialog box. The data from the dialog box is written to the parent form based on the mapping you created for the On Close mode in the Field Mapping region when setting the Open Window active link action. Changes are usually committed with the OK button on the dialog box. You might also want to create an Apply button that commits changes without closing the dialog box.</td>
</tr>
<tr>
<td>Close Window</td>
<td>Closes the active dialog box. The Close Window action usually occurs immediately after the Commit Changes action or as the active link action associated with Cancel occurs. For information about active links, see the Workflow Objects guide.</td>
</tr>
</tbody>
</table>

Creating dialog boxes

The following procedure provides a general overview for creating an application that uses dialog boxes. This particular example is a confirmation dialog box that returns a value to the parent form. When a user clicks a button on the parent form, the dialog box appears. The user enters information into a field on the dialog box, or the field might be populated as a result of an active link.
To create a simple dialog box

1. Create a button on the parent form from which you want to open the dialog box.

2. Save your parent form.

3. Create a display-only form with a character field representing the entry field. For example, if the user must enter the serial number of a product, you would create a field labeled “Serial Number.”

4. Create OK and Cancel buttons on your display-only form.

5. Save your display-only form.

You can add additional fields to a dialog box, although fields will always be added in display-only mode, and you can run active links from a dialog box.

Note: You cannot give a default value to fields in dialog boxes through the Field Properties window. Instead, use the On Open mode in the Open Window action to set any default values you want to appear in a dialog box.

6. Create an active link to launch the dialog box with the following conditions and actions:

   a. Open the Create Active Link dialog box.
   b. In the Basic tab, enter a name for your active link in the Name field.
   c. From the Form Name list, select the parent form.
   d. From the Execute On section, select Button/Menu item.
   e. From the Field list, select the name of button you want your users to click to open the dialog box.
      - In this example, the button name is Enter Serial Number.
   f. Click the If Action tab.
   g. From the New Action menu list, select Open Window.
   h. From the Window Type menu list, select Dialog.
   i. From the Form Name menu list, select your display-only form.
From the Field Mapping mode menu list, select On Open. The fields in your display-only form will be listed in the Fields in Dialog Form column. In this example, the field listed is Serial Number.

If you want the field to be automatically populated when the dialog box opens, enter a value in the Value column.

From the Mode menu list of the Field Mapping section, select On Close.

From the Fields in Parent Form column, select the field from the parent form that you want populated from the dialog box’s Serial Number field.

Click the Value pane at the selection point, click the down arrow and select from the Fields list the field in the dialog box whose value you want transferred to the parent form when the dialog box closes. In this example, the field is Serial Number.

Click Add Action to add your action to the Current Action field.

Save your active link.

Create an active link to execute when the user clicks the OK button on the dialog box. Creating this workflow in this example transfers information from the Serial Number field on the Dialog Box to a field on the parent form.

Open the Create Active Link dialog box.

In the Basic tab, enter a name for your active link in the Name field.

From the Form Name list, select the parent form.

From the Execute On section, select Button/Menu item.

From the Field list, select the name of button you want your users to click. In this example, the button name is OK.

Click the If Action tab.

From the New Action menu list, select Commit Changes.

Click Add Action to add your active link action to the Current Actions field.

From the New Action field, select Close Window.

Click Add Action to add your active link action to the Current Actions field.

Save your active link.
8 Create an active link that executes when the user clicks Cancel. This action simply closes the dialog box without returning any values.
   a Open the Create Active Link dialog box.
   b In the Basic tab, enter a name for your active link in the Name field.
   c From the Form Name list, select the display-only form.
   d From the Execute On section, select Button/Menu.
   e From the Field list, select the name of button you want your users to click.
      In this example, the button name is Cancel.
   f Click the If Action tab.
   g From the New Action menu list, select Close Window.
   h Select Close All Windows.
   i Click Add Action to add your active link action to the Current Actions field.
   j Save your active link.
9 Set the correct permissions for the forms and active links so that your users can operate them successfully.

For additional examples of using buttons that open dialog boxes, open the Sample:ClassCentral form that is installed with AR System, and click the Enroll tab.

For additional information about active links, see the Workflow Objects guide.
Join forms

Join forms are composite forms that consist of fields derived from other existing forms. This section helps you understand more about join forms.

Understanding join forms

A join form in AR System is similar to joining tables in a relational database. A join form uses searches to combine fields from two forms based on join criteria (see the “Join criteria” section that follows). The data in a join form comes from the database tables of the forms that comprise the join form.

After the join form is created, it behaves similarly to non-join forms. Users can submit data for creation or modification, report from it, select entries from it, use it in workflow requests, define workflow on it, and so on. From the user’s perspective, there is no difference between join and non-join forms.

You can use a join form as a member of another join form. For more information, see “Joining three or more forms” on page 114.

Primary and secondary forms in the join form

When creating join forms, you designate one of the underlying forms as the primary form and the other as the secondary form. Primary forms are used when determining which extra entries are included in outer join forms (see “Inner and outer joins” on page 112) and the execution order in workflow (see “Filters and join forms” on page 118).

Join criteria

Join criteria define the link between the two underlying forms. Join criteria are values common to the forms that you want to join. For example, if a help desk form and an employee record form both have an employee ID field, the two could be joined by the equality relationship between them, as shown in the following figure. In database terms, this is an equal join.
You can join forms by using any combination of fields.

**Note:** Try to use indexed fields in the join criteria. A join, like any other query of the database, should be optimized for best performance results. For information about indexing fields, see “Defining indexes” on page 139.

**Including fields in the join form**

You choose the fields from the primary and secondary forms that will be part of the join form. If you choose two fields that have the same field ID, the system provides a new mapped field ID for one of the fields because duplicate field IDs are not allowed in a form. The new field ID is mapped to the actual field ID in the underlying form when operations are performed.
Note: In a join form, BMC Remedy Administrator will try to preserve the name and field ID of fields from the primary form.

You can change the display properties for fields in a join form and set permissions for the join form itself.

After creating a join form, you can add display-only fields to the form. For information, see Chapter 4, “Types of fields.”

Inner and outer joins

AR System enables you to create join forms by using inner or outer joins. An inner join selects entries (or rows) only when there are corresponding values in both of the forms. For example, you would use an inner join if you want to retrieve only the entries from one form that have matching entries in another form. If there is an entry in one of the forms without a corresponding entry in the other form, the data would be omitted.

Outer joins include all of the entries from the form that you select as primary, even entries for which there are no matching entries in the secondary form. For example, to see all submitted help requests, including those that have no specific employee information connected with them, create an outer join.

Note: An outer join in AR System is what relational database administrators call a left outer join. Selecting the left (or primary) form includes all of the entries associated with that form.

The following figure illustrates the concept of inner joins. The Library Catalog form is the primary form. The Customer Checkout form is the secondary form. The join criteria is the ISBN (International Standard Book Number).

Because an inner join creates a form that contains only the entries in which the join criteria exists in both the primary and secondary forms, the join form produces a report that shows only the titles that are actually checked out.
Figure 3-6: Example of an inner join

Because an inner join contains only the entries in which the join criteria is present in both forms, ISBN items 3 and 5 (which are not in the secondary form) are omitted from the join form.

If the library had produced the same report using an outer join, it would be a comprehensive listing of all the catalog items in the library, whether or not they had corresponding entries in the other form. The following figure shows an example of an outer join.
When determining whether to create an inner join or an outer join, one approach is to base your choice on how much data you want to see. Inner joins are more useful for ad hoc queries and selection lists, while outer joins are more useful for special reports that are comprehensive by nature.

**Joining three or more forms**

To join three forms, you must first join two forms and then join the resulting form to the third, creating a hierarchy of joins. Joining multiple forms in a hierarchical order makes it easier to provide a consistent workflow.

---

**Figure 3-7: Example of an outer join**

<table>
<thead>
<tr>
<th>Primary form</th>
<th>Secondary form</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Library Catalog Form</strong></td>
<td><strong>Customer Checkout Form</strong></td>
</tr>
<tr>
<td>ISBN</td>
<td>Title</td>
</tr>
<tr>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>B</td>
</tr>
<tr>
<td>3</td>
<td>C</td>
</tr>
<tr>
<td>4</td>
<td>D</td>
</tr>
<tr>
<td>5</td>
<td>E</td>
</tr>
<tr>
<td>6</td>
<td>F</td>
</tr>
</tbody>
</table>

An outer join contains all of the entries from the primary form even if there is no corresponding join criteria in the secondary form.
If you need to combine data from three or more forms, you can do so by creating a series of two-way joins. As shown in the following figure, you can join two regular forms, a regular form to a join form, or two join forms.

**Figure 3-8: Joining several forms**
Add only as many join layers as you need, and make sure that your join criteria is efficient. The practical upper limit for combining forms is about six layers. This is because each join form is created by querying the database—which ultimately affects system performance. In addition, the workflow attached to each form in multiple layers of joins can also severely impact performance.

**Self-join forms**

In AR System, you can join a form to itself. This is also known as a “Cartesian Join.” This functionality is useful when comparing data from the same form or when preparing reports. Suppose that you want a report of all of the managers, the managers’ phone numbers, the employees they supervise, and the employees’ phone numbers. Assume also that the employees and managers both exist in this form.

In this example, DemoHD:Staff is the primary (designated as A) and the secondary form (designated as B), and it has the data shown in the following table.

**Figure 3-9: Self-join example 1**

<table>
<thead>
<tr>
<th>Request ID</th>
<th>Employee ID</th>
<th>Employee Name</th>
<th>Employee Phone Number</th>
<th>Manager ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>111</td>
<td>Bob</td>
<td>111-1111</td>
<td>555</td>
</tr>
<tr>
<td>002</td>
<td>222</td>
<td>Steve</td>
<td>222-2222</td>
<td>555</td>
</tr>
<tr>
<td>003</td>
<td>333</td>
<td>John</td>
<td>333-3333</td>
<td>555</td>
</tr>
<tr>
<td>004</td>
<td>444</td>
<td>Sue</td>
<td>444-4444</td>
<td>666</td>
</tr>
<tr>
<td>005</td>
<td>555</td>
<td>Doug</td>
<td>555-5555</td>
<td>666</td>
</tr>
<tr>
<td>006</td>
<td>666</td>
<td>Nancy</td>
<td>666-6666</td>
<td>NULL</td>
</tr>
</tbody>
</table>

If you join the form to itself and specify A.Employee ID = B.Manager ID as the join criteria, you could then add and rename the following fields in the resulting join form:

- A.Employee Name (from the primary form) renamed to Manager.
- A.Employee Phone Number (from the primary form) renamed to Manager Phone Number.
- B.Employee Name (from the secondary form) renamed to Employee.
B. Employee Phone Number (from the primary form) renamed to Employee Phone Number.

The join form with four fields (or five if you include the composite request ID) contains the following results from an unqualified search.

Figure 3-10: Self-join example 2

<table>
<thead>
<tr>
<th>Composite Request ID</th>
<th>Manager</th>
<th>Manager Phone Number</th>
<th>Employee Name</th>
<th>Employee Phone Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>005</td>
<td>001</td>
<td>Doug</td>
<td>555-5555</td>
<td>Bob</td>
</tr>
<tr>
<td>005</td>
<td>002</td>
<td>Doug</td>
<td>555-5555</td>
<td>Steve</td>
</tr>
<tr>
<td>005</td>
<td>003</td>
<td>Doug</td>
<td>555-5555</td>
<td>John</td>
</tr>
<tr>
<td>006</td>
<td>004</td>
<td>Nancy</td>
<td>666-6666</td>
<td>Sue</td>
</tr>
<tr>
<td>006</td>
<td>005</td>
<td>Nancy</td>
<td>666-6666</td>
<td>Doug</td>
</tr>
</tbody>
</table>

You can include phone numbers for each manager and employee in one entry, even though they come from the same column in the same table. The self-join logically joins two separate forms that contain identical information.

Self-join forms are useful for certain parent-child relationships. In the previous example, the manager-employee relationship is a type of parent-child relationship in which child entries (the employees) belong to a parent entry (the manager). A self-join form also enables you to compare entries. For example, you can create a join form with side-by-side comparisons of long description fields to verify that the entries are duplicates of one another.

Using join forms in workflow

Connecting objects (filters, escalations, active links, active link guides, and filter guides) to a join form enables you to use the form in workflow. All of the typical execution conditions that apply to objects in non-join forms also apply to objects in join forms.

To enable the “data entry” functionality of a join form, data created or updated for a particular field must be associated with the appropriate database. The administrator must create workflow that defines how data should be pushed into the database when entered through join forms.
Filters and join forms

When you use filters with join forms, there are special issues to consider. Filters are used to enforce the integrity of the system. The filters for the join form execute first, and then the filters for the underlying forms execute.

The following figure illustrates the execution order of the join forms and their primary and secondary forms.

Figure 3-11: Execution order of filters with join forms
Observe that as you go down the execution order of the filters, all of the forms on the left side (the primary forms) execute first, and then forms on the right side (the secondary forms) execute. For information, see the Workflow Objects guide.

**Transaction control in database operations**

All of the operations performed on join forms, such as querying, displaying, and workflow operations, are executed as a single database transaction. For example, if you update data in one form that affects the data in multiple forms, all of the changes are performed as a single transaction in the database. This process ensures the integrity of the information.

The completion of the database transaction for a hierarchy of forms is all or none. For example, in Figure 3-8 on page 115, if a filter returns an error on form I, none of the updates for the other forms is written to the database.

**The Request ID field**

The AR System tracks the entries in the underlying forms that comprise a join form through the request ID of each entry. When you look at the Request ID field in a join form entry, you see that the field contains the Request ID of each underlying entry separated by a vertical bar. For this reason, if you create a join form from a non-join form and a join form (see join form H in Figure 3-8 on page 115), the join form’s Request ID field will contain three request IDs separated by vertical bars.

In all fields in a join form except the Request ID field, the permissions of a join form are inherited from the underlying forms and cannot be changed. The default permissions of the Request ID field in a join form are defined as Visible for the Assignee, Public, and Submitter groups.

However, you can remove the Public permissions of the Request ID field to make row-level security work in AR System. For information, see “Controlling access by using implicit groups—Row-level security” on page 67.
Creating entries in join forms

When creating entries through a join, AR System does not actually initiate a database operation. Because creating an entry in a join is not a determinate action, the system cannot automatically perform it. However, all filter operations defined for a join are performed, which requires you to define workflow that appropriately creates or modifies entries in the primary and secondary forms.

For example, if you create a join between a customer and an item purchased, creating an entry through the join could be defined through workflow. With push fields filter actions, the workflow could create an entry for an item purchased or an entry for a customer.

Creating join forms

The following procedure guides you through the process of creating a join form. A sequence of five windows takes you through the following tasks:

- Specify general information about the join that you want to create, including the names of the primary and secondary forms, whether you want to keep the existing field help text, and whether you want to create an inner or outer join.
- Specify join criteria.
- Select fields from the primary form.
- Select fields from the secondary form.
- Select a view for each member form, and the method to use to combine the views.

To create a join between forms

1. In BMC Remedy Administrator, select a server to administer.
3. Select Join Form from the list.

The Join Wizard appears.
Join forms

Figure 3-12: Choosing forms and join types

The following pages will guide you through the process of creating a Join Form. Once you have selected the two forms for the join, you can press Next to go to the next page, or press Finish to create the join using defaults.

4 From the Primary Form list, select a primary form.

5 From the Secondary Form list, select a secondary form.

6 For deployable applications only, select the Restricted List check box (default option) to restrict the join form only to the forms in the application.

Clearing the Restricted List check box lets you access all the forms on the server.

7 If you do not want to include the help text that you already created for the fields in the primary and secondary forms, clear the Inherit Field Help Text check box.

You can create field help text directly in the join form at a later time.

8 From the Join Type option list, select the appropriate join type.

The default is Inner Join. For more information about inner and outer joins, see “Inner and outer joins” on page 112.
Click Next to open Page 2 of the Join Wizard.

**Figure 3-13: Choosing join criteria**

A join line appears between two forms.

```
To select join criteria, drag a field from one form and drop it on a field of the same data type in the other form. Indexed fields for each form are shown in bold.
```

To graphically join criteria:

a. Select one field from the primary form list (left), and one field from the secondary form list (right).

b. Click Join.

You can only join fields if they have the same data type. If you try to join fields with different data types, the Join button is disabled and cannot be selected.
Join lines connect fields that are associated. Connecting two fields with a join line performs an equal operation on the selected fields. If multiple selections are made, an AND operation is applied to the selections.

To unjoin two fields, select the appropriate criterion, and click Unjoin.

c Repeat step a and step b until you finish specifying join criteria.

**Note:** For optimal performance, use indexed fields in the join criteria. For information about indexing fields, see “Defining indexes” on page 139.

To apply an OR operation, compare to static values, use arithmetic operations, or use a comparison other than equal:

a Select the Use Qualification Bar box.

b Select the appropriate fields from the primary and secondary forms by clicking the menu button to the right of the Qualification field. You can join fields only if they have the same data type.

c Add the appropriate operators by selecting them from the qualification bar.

For more information about entering qualifications, see the Workflow Objects guide.

For more information about join criteria, see “Join criteria” on page 110.

11 Click Next to open Page 3 of the Join Wizard.
Use the Add and Remove buttons to move fields to the appropriate Include list.

Click Next to open Page 4 of the Join Wizard.

Add all of the fields from the secondary form that you want to include in the join form, and click Next.

Page 5 of the Join Wizard appears. Use this window to define the overall appearance of the join form by specifying how the fields will be displayed. The colors are different for each contributing form.
15 From the Views from `<primary form name>` list, select the appropriate view of the primary form.

16 From the Views from `<secondary form name>` list, select the appropriate view of the secondary form.

17 From the Method to Combine Views option list, select the option that gives the appropriate appearance to the resulting join form.

**Stacked Vertically**

Arranges the fields of the primary form above the fields of the secondary form. This option (the default) keeps the fields of the underlying forms separate from one another.
Click Finish. The new join form appears with a tool palette in a Create Form window.

Arrange the fields as you want them to appear in the join form. For example:

- Add trim, buttons, page fields, or table fields.
- Create views for the join form.

For information about fields in join forms, see “Fields in join forms” on page 220. For information about arranging fields on a form, see “Managing fields in a form view” on page 279.

To access the Form properties dialog box, choose Form > Form Properties or double-click an empty part of the form.

Join form properties are set the same way as regular form properties. For information about setting form properties, see “Setting form properties” on page 131.

Choose File > Save Form.
In the Form Name field, enter a name. Form names must be unique on each AR System server. Names can be as many as 80 characters, including spaces.

Click OK. The name of the new join form appears in the title bar and the Forms list.

Working with join forms in BMC Remedy Administrator

When you create a join form, it appears in the Forms list, as shown.

Figure 3-16: Join forms in Server window

The name “Join” in the Type column differentiates a join form from non-join forms.

When a join form is open in a Modify Form window, the following additional menu items appear in the Form > Create a New menu:

- **Field From <primary form>**: Adds an existing data field from the primary form.
- **Field From <secondary form>**: Adds an existing data field from the secondary form.
Modifying join form properties

After you create a join form, you can modify properties that determine the characteristics of how that join form will look and perform during operations performed in BMC Remedy User.

The properties that non-join forms and join forms have in common (Results List Fields, Permissions, Subadministrator Permissions, Change History, and Help Text) are set the same way for both types of forms, as described in this section. The properties that are unique to join forms (Join Forms and Join Criteria) are described in the following sections.

Primary and secondary forms and join types

After you create a join form, you can use the Join Forms tab to “swap” which form is primary and which is secondary. You can also change the type of join from inner to outer, or from outer to inner. Depending on whether you are working with an inner join or outer join, swapping forms can result in completely different criteria. For example, suppose your primary form (A) has three fields (1, 2, 3) and your secondary form (B) has three fields (3, 4, 5). In this situation, an inner join would retrieve the field that the two forms have in common (field 3), and an outer join would retrieve this field and the remaining primary form fields for a total of fields 1, 2, and 3. If you swap forms so that form B becomes your primary form and form A becomes your secondary form, an inner join will yield the same results (field 3), but an outer join now retrieves the fields 3, 4, and 5. For more information about inner and outer joins, see “Inner and outer joins” on page 112.

To modify primary and secondary forms and join type

1. Open the join form with which you want to work.
2. Choose Form > Form Properties to open the Form Properties dialog box.
3. Click the Join Forms tab.
4 To switch the primary and secondary forms, click Swap Forms.

5 To change the join type, from the Join Type option list, select Inner or Outer. For information about inner and outer join forms, see “Inner and outer joins” on page 112.

6 Click OK.

7 Choose File > Save Form.

Modifying join criteria

Use the Join Criteria tab to add, change, or delete join criteria for a join form that you have already created. For details about join criteria, see “Join criteria” on page 110.
To add, change, or delete join criteria

1. Open the join form with which you want to work.
2. Choose Form > Form Properties, or double-click an empty part of the form to open the Form Properties dialog box.
3. Click the Join Criteria tab.

Figure 3-18: Form Properties dialog box—Join Criteria tab

4. To modify your join criteria in the Selection boxes, use the Join and Unjoin buttons and redefine the join criteria, as appropriate.
   For more information about Join and Unjoin buttons, see step 9 on page 122.
5. To modify your join criteria in the Qualification edit box, select the Use Qualification Bar check box and modify the Qualification edit field.
   For information about entering qualifications, see step 9 on page 122.
6. Choose File > Save Form.
Setting form properties

For each form, you can define properties that determine how that form will look and perform during operations performed in BMC Remedy User. Form properties are defined in the Form Properties dialog box accessed by choosing Form > Form Properties.

When using the procedures that follow, note that the properties defined for a form apply to all views associated with that form. To set properties for a specific view, select the view in the Manage Views dialog box and click the Properties button. For more information about setting view-specific properties, see “Setting view properties” on page 263.

Detailed instructions for defining each of the form properties are provided in the following sections.

**Vendor Information**
For vendor forms, provides information about the vendor and table name.

**View Information**
For view forms, provides information about the database table associated with the form.

**Entry Points**
Defines the order in which entry points appear in a Application List field. For information, see “Entry Points—Defining entry points” on page 133.

**Results List Fields**
Defines which of the form’s fields appear when a user performs a Search operation in BMC Remedy User. For information, see “Results list fields—Defining search results” on page 135.

**Sort**
Defines the order in which requests appear in the matching table list when clicking the Search button in BMC Remedy User. For information, see “Setting up sort order” on page 138.

**Indexes**
Defines form indexes to reduce the database search time for frequently searched fields. For information, see “Defining indexes” on page 139.

**Archive**
Defines settings for periodically backing up or deleting form data. For information, see Appendix E, “Archiving data.”

**Permissions**
Defines which access control groups can access the form. For information, see “Form, active link guide, and application permissions” on page 57.
Defining vendor information

The Vendor Information tab contains the Vendor Name and Table Name fields, which contain the vendor and table names used to create the vendor form. If you modify these fields and specify a vendor name that is not associated with a valid ARDBC plug-in or a table name that the plug-in does not support, you will receive errors when you try to access data from the vendor form.

For more information about vendor forms, see the ARDBC LDAP discussion in the Integrating with Plug-ins and Third-Party Products guide.

Figure 3-19: Form Properties dialog box—Vendor Information tab
Displaying view information

The View Information tab displays the names of the table and key field used to create the view form. You cannot edit these fields.

Figure 3-20: Form Properties dialog box—View Information tab

Entry Points—Defining entry points

The Entry Points tab enables you to determine the numeric order of your form’s New and Search entry points in an Application List field:

- New entry points open your form in create mode.
- Search entry points open your form in query mode.

Entry point values are especially important to create a logical order of all the entry points that will appear on your Home Page form.

An entry point with a value of 301 will appear in the Application List field after an entry point with value of 300. As a result, you can order the entry points according to their relative importance. Entry point values must be equal to or greater than zero.

Use these settings in connection with the form’s View Properties. For complete information about using entry points, see Chapter 9, “Defining entry points and home pages.”

To define entry points for a form

1. Open the form with which you want to work.
2. Choose Form > Form Properties to open the Form Properties dialog box.
3. Click the Entry Points tab.
4 For New Mode entry points, perform the following tasks:
   a Select the Enable Entry Point check box to activate the new mode entry point.
   b In the Application List Display Order field, enter a value to order the new mode entry point in the application list.

5 For Search Mode entry points, perform the following tasks:
   a Select the Enable Entry Point check box to activate the search mode entry point.
   b In the Application List Display Order field, enter a value to order the search mode entry point in the application list.

6 Click OK.
Results list fields—Defining search results

Use the Results List Fields tab to customize which fields appear in the results pane when a user performs a search operation in BMC Remedy User. If you do not define a results list for a form, the default is to display the contents of the Short Description field when a user performs a search operation. If you add any other field to the results list, the Short Description field is no longer automatically included and will not be part of the results list unless you add it.

You cannot include diary fields in a results list. You can include an attachment field, but all you will see in the results list is the attachment file name. You can include fields that are greater than 255 characters, but only 128 characters will appear in the column.

Note: A performance issue can be created when using long characters fields in a Results List. Even though 128 characters or less will be displayed, all of the field’s data is returned to the client.

Modifying the contents of the results list also affects the contents of the Selection List window in BMC Remedy User. The Selection List window appears in BMC Remedy User when an active link performs a Set Fields operation based on a search and returns more than one result.

The settings in the Results List tab specify the default set of fields returned for API programs that do not override the fields returned.

For a web client, a special Results List field can be used for the results of a search. You can set the columns for the Results List field in the Table Properties tab of the Results List field properties dialog box. For more information, see “Table fields” on page 153.

Note: As of version 6.3 of the mid tier, a special Results List field is not needed for forms viewed in a browser. Similar to BMC Remedy User, a dual pane layout contains a top pane that contains the results of a search. If you include a Results List field, the dual pane layout will not appear.
To define fields returned in a search

1. Open the form with which you want to work.
2. Choose Form > Form Properties to open the Form Properties dialog box.
3. Click the Results List Fields tab.

The following figure shows an example of how the Results List Fields tab might look after you complete the remaining steps in this procedure.

Figure 3-22: Form Properties dialog box—Results List Fields tab

4. From the Fields in Form list, select the first field to include in the results list pane.

The Fields in Form list includes all fields with a database length of 255 characters or less. Therefore, diary fields are not available for selection. Also, table fields and page fields do not appear in the list.

If you have purchased the full text search option, you can also select WEIGHT from the Fields in Form list to display the weighted value of retrieved requests when you perform a search in BMC Remedy User. See the Configuring guide for more information about full text search.

136 Chapter 3—Creating AR System forms
In the Separator field, enter a character or string to separate this field from the next field in the results list. The default is a single blank space, but you can enter any character or string of characters, and the character string can include any of the following special characters:

<table>
<thead>
<tr>
<th>Character:</th>
<th>Enter:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backspace</td>
<td>\b</td>
</tr>
<tr>
<td>Return</td>
<td>\n</td>
</tr>
<tr>
<td>Tab</td>
<td>\t</td>
</tr>
<tr>
<td>Backslash</td>
<td>\</td>
</tr>
<tr>
<td>ASCII character</td>
<td>&lt;nnn&gt; (where nnn is the decimal designator for the ASCII character that you want to use)</td>
</tr>
</tbody>
</table>

In the Width field, enter a number to limit the width of this field in the results list. For example, if you set a width of 20, only the first 20 characters of the field value will appear in the list. Separator characters and field widths are used when a Set Fields action of an active link causes a selection list to appear if a multiple match is found. If this limit is exceeded, an AR System error message is generated when attempting to add the field.

Click Add.

The field appears in the Field Name list, along with its separator and width settings. You can also select an entry from the Field Name list and use the following buttons to perform the following actions:

- **Remove**: Removes the results list entry.
- **Modify**: Modifies the results list entry according to the changes you make in the Fields in Form field, Separator field, or Width field.
- **Move**: Moves the results list entry up or down in the results list.

To add additional fields, repeat steps 4 through 7 for each field.

Click OK.
To set the color of requests that appear in the results list, after a search is accomplished in the Results List Color tab, use the View Properties dialog box. For information, see the procedure “To define view properties in the Results List Color tab” on page 276.

**Setting up sort order**

The Sort tab enables you to define the order in which requests appear in the matching results list when clicking the Search button in BMC Remedy User.

► **To set up the sort order for the results list pane**

1. Open the form with which you want to work.
2. Choose Form > Form Properties to open the Form Properties dialog box.
3. Click the Sort tab.

![Figure 3-23: Forms Properties dialog box—Sort tab](image)

4. From the Non-Sorted Fields list, select the fields you want to sort, and click Add to move them to the Sorted Fields list.

To remove a field from the Sorted Fields list, select an entry in the Sorted Fields list, and click Remove.
Multiple selections for both the Non-Sorted Fields and Sorted Fields lists can be made at one time by using the SHIFT or CTRL keys.

5 To set the sorting order for each field, click its Arrow icon. An up arrow indicates ascending order, and a down arrow indicates descending order.

6 To reorder items in the Sorted Fields list, select the field and click the up or down arrow at the top of the list.

7 Click OK.

Defining indexes

Indexing can greatly reduce database search time. Indexes can be defined for regular forms only. You cannot create indexes for other form types for the following reasons:

- Join forms use the indexing defined for the forms from which they are constructed.
- Because display-only forms have no database table, they need no indexing.
- Because view and vendor forms are owned outside of AR System, any indexing they support must be managed outside of AR System.

Only data fields can be indexed. The Request ID field is already indexed, so you need not build a separate index for this field. Good candidates for indexing include fields that users search on frequently.

If you define an index for a character field, you might save search time by using a QBE Match setting of Leading or Equal, not by using a QBE Match setting of Anywhere. For information, see “Database properties” on page 420.

If you are creating or modifying indexes in a form for which a large amount of data exists, this process can take a significant amount of time and disk space because the index must be built or rebuilt. Therefore, you might want to avoid defining indexes during normal production hours.

More time is required to modify a form (for example, adding new fields) when indexes have been defined for the form. The greater the number of indexes defined for the form, the more time and disk space is required. Submit and modify operations in BMC Remedy User also take longer on forms with many indexes.
For additional information about maximizing index performance, the AR System Performance Tuning and Troubleshooting course is an excellent resource.

To define indexes for a form

1. Open the Form with which you want to work.
2. Choose Form > Form Properties to open the Form Properties dialog box.
3. Click the Indexes tab.

If you are creating a new index, “No Indexes” appears above the Previous and Next buttons. If you are modifying indexes, you can click these buttons to find the index you want to modify.

4. From the Fields in Form list, select a field to include in the index, and click Add.
   The selected field is added to the Index On list.
5 To combine multiple fields into a composite index, repeat step 4 for each field you want to add.

Enter the fields in the order that you want them indexed. You can enter as many as 16 fields. Each indexed field must be less than or equal to 255 character, but the composite Index can have a total length greater than 255.

Note: AR System does not verify if the size violates the databases rules, and a database error will be returned if the size is too large.

6 Select or clear the Unique check box.

If the check box is selected, the index is treated as a unique index, and AR System requires all values (existing and new) in the field or combination of fields to be unique. For example, you could define a unique index on a phone number field because each phone number is unique; however, you would not define a unique index on a field that contains first names, because more than one person might have the same name.

7 To create additional indexes for this form, click New Index, and repeat steps 4 through 6 for each index.

8 To remove an index from the database, move to the correct index, and then click Delete Index.

You will not receive a confirmation, so make sure that you want to delete the index.

9 Click OK.

For more information about performance issues and tuning in AR System, especially analyzing forms for performance, see the Optimizing and Troubleshooting guide.
Fields are objects that enable you to control how information is captured and displayed on a form. Fields contain properties that determine their structure within AR System.

The following topics are provided:
- Data fields (page 144)
- Table fields (page 153)
- Attachment pools (page 161)
- View fields (page 164)
- Data visualization fields (page 166)
- Application list fields (page 166)
- Horizontal and vertical navigation fields (page 168)
- Button fields (page 169)
- Page fields (page 170)
- Trim fields (page 172)

For a detailed discussion of each of the tabs in the Field Properties window, see Appendix D, “Field properties,” on page 397.
Data fields

Data fields are the fields that contain data and can be any one of the following data types:

- Character
- Diary
- Date/Time
- Date
- Time
- Currency
- Integer
- Real number
- Decimal
- Selection (radio button, drop-down list, check box)

The following sections describe these types of fields. To learn how to create them, see Chapter 5, “Creating fields,” on page 173.

Character fields

Character fields are useful when there is significant variation in the field contents or length of the content; for example, descriptive text, names of people, addresses, and keywords.

You can attach menus to character fields or fill them with default text. For more information, see “Character field attributes” on page 406.

**Note:** If users want to include a tab in the text they enter into a character field’s dialog box, they must press CTRL+TAB.

Diary fields

Diary fields capture the history of a request over time. Whenever users enter comments in the diary field, the new entry is appended to the previous diary entries. Every diary entry is stamped by the AR System server with a timestamp and user name. After they are entered, diary entries cannot be modified.

**Note:** If users want to include a tab in the text they enter into a diary field’s dialog box, they must press CTRL+TAB.
By default, AR System inserts a diary icon to the right of each diary field that users can click to open a Diary Editor dialog box. When the diary field contains entries, the icon changes from a blank book to an image of a book containing text.

The default upper size limit of data contained in diary fields is different for each of the following databases:

- For Sybase/MSSQL and Informix databases, 2 GB
- For DB2, 10 MB
- For Oracle, 4 GB

**Note:** Use the `ar.conf` or `ar.cfg` files to configure a different upper limit for Oracle and SQL Server databases using the `Db-Max-Text-Size` configuration option. For more information about the AR System configuration files, see the Configuring AR System guide.

To search a diary field when using Oracle, you must configure `ar.conf` or `ar.cfg` to allow searching on `clob` data types. For more information, see the Configuring guide.

Consider the effect of searching on system performance. You might want to use FTS to create a search index for diary fields. Note that you cannot search the timestamp or the name of the user who submitted an entry.

You cannot use the Indexing form property to create an index for a diary field. However, if you are licensed for full text search, you can use the Index for FTS field described on page 281 to create a search index for the field. For more information about FTS, see the Configuring guide.
There are three types of date and time fields:

- **Date/time fields**— Store calendar dates and time together. You can set the display type to Date Only or Time Only so that the user only sees the date or time. Users can enter dates from January 1, 1970 to January 1, 2038. If the user enters a time only, then the current date is assumed. If the user enters the date only, then the time will default to 12:00:00 AM.

- **Date fields**— Store date information only, as the number of days from the beginning of its range. Use a Date field when you want to compare two dates or perform calculations based on the date, such as calculating the number of days between two dates. Users can enter dates from January 1, 4713 B.C. to January 1, 9999 A.D. in the Date field.

- **Time fields**— Store time information only, as the number of seconds from 12:00:00 AM. Use a time field to compare two times or perform calculations based on time, such as the number of seconds elapsed. The value in a time field is independent of the time zone. While a date/time field will adjust the displayed value to reflect the user’s time zone, the time value in a time field will remain unchanged when displayed on the client.

**WARNING:** Because values stored in date/time, date, and time fields are not equivalent, setting, pushing, or merging values among these fields might produce unexpected results.

The format for these fields matches the locale specified in user preferences. If there is no user preference for locale, then user environment settings are used (for example, Regional Settings Properties in the Windows Control Panel).

For more information about how date and time settings are determined in AR System, see the Getting Started guide.

For information about workflow considerations for date/time, date, and time fields, see information about keywords and assigning values using function results in the Workflow Objects guide.
Currency fields

A currency field stores currency data, including information the user enters (a decimal value and a currency type), one or more functional currency types (whose values are filled in when the currency data is submitted), and the date on which functional values were converted.

Currency fields require special configuration using AR System forms. For more information, see “Creating currency fields” on page 176.

Currency field structure

The currency types that users can enter are selected from the list of allowable currency types, which are visible in menus or drop-down lists in BMC Remedy Administrator and in user clients. For each currency field, you can specify a primary allowable currency type, which is the currency type that is assumed if the user enters only a decimal value in the field. If you do not specify a primary allowable currency, the first allowable currency type is used.

When the user saves a request, the value in the currency field is converted to each functional currency type defined for the field. Functional currency decimal values are stored as part of the field’s value. Having these pre-converted functional currencies allows you to search, report, and run qualifications without using the additional server processing time required to convert the values. Every currency field must have at least one functional currency type. There is no maximum number of functional currencies, but it is recommended that you have no more than five, to avoid delays when you submit requests.

You can set default allowable and functional currency types and decimal precisions in the Currency Types tab of the Server Information window. These default currency types then appear in the field properties for all new currency fields. You can also define unique allowable and functional currency type properties for individual currency fields. For more information, see “Creating currency fields” on page 176.

You can define localized currency labels in the AR System Currency Label Catalog form. For more information, see “Localizing currency type descriptions” on page 549.
Currency exchange rates

You define currency exchange rates using the AR System Currency Ratios form. Create entries that define exchange ratios between every allowable and functional currency type. Define one exchange ratio for converting from one currency to another, and a second ratio for converting back. For example, for United States dollars and British pounds, create one entry for converting from USD to GBP, and another entry for converting from GBP to USD. When a user saves a request, functional currency values are calculated according to ratios stored in the AR System Currency Ratios form.

Add new entries to the AR System Currency Ratios form as currency exchange rates change over time. During currency conversions, the system uses the value in the Create Date field to determine the latest available ratio for new currency values, or the latest available for historical conversion, when the date specified is something other than current.

Keeping currency ratios up to date

There is no mechanism in AR System that automatically updates records in the AR System Currency Ratios form. However, you can use AR System to design your own update mechanism. For example, you can:

- Create a web service that consumes a currency conversion web service.
- Use an ARDBC plugin that interfaces with a rate service.
- Create an escalation that submits new values.

You can define the client update interval by which clients query the server for the latest currency ratios for new currency values. (Currency ratios are stored in the AR System Currency Ratios form on the server.) Specify this interval in minutes in the Timeouts tab of the Server Information window. For more information about server settings, see the Configuring guide.

Currency field function

The user enters a decimal value and selects an allowable currency code from the menu next to the field. The currency type that was used at the time of submission is stored as part of the currency value, and this stored currency value is used when the record is displayed again. Users can view functional currency values by clicking on the expand box next to the currency field.
Decimal values are displayed according to the user’s locale. For example, on German systems, thousands are separated by periods.

The user can define a preferred currency type, or you can define an initial currency type in field properties, which loads if the currency field is empty when the user creates a new request. If you set a default value for the field, it overrides any initial currency type. For more information, see “Currency field attributes” on page 409.

In table fields and results lists, when a user sorts on a column, records are grouped by currency type and then sorted within each group. This allows meaningful comparisons among currencies of the same type.

**Workflow considerations for currency fields**

You can use currency fields in active link, filter, and escalation actions. Currency fields behave like other fields in workflow actions, with the following exceptions:

- The Change Field active link action cannot access the menu attached to the field.
- The Set Fields and Push Fields actions allow only the overall value of the field to be set. You can use the overall value or any portion of the value (such as the date) as a data source.

There are four functions for currency fields—`CURRCONVERT`, `CURRSETDATE`, `CURRSETTYPE`, and `CURRSETVALUE`. For more information, see the Workflow Objects guide.
Because the currency field is a complex type, it has some special data conversion rules for different situations:

### From Currency to <other data type>:

<table>
<thead>
<tr>
<th>Data conversion rule</th>
<th>From Currency to &lt;other data type&gt;:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character or Diary</td>
<td>The decimal value, with the currency code to the right.</td>
</tr>
<tr>
<td>Decimal, Real, or Integer</td>
<td>The decimal value only, dropping the fraction as necessary.</td>
</tr>
<tr>
<td>Date/time, Date, or Time</td>
<td>The decimal value converts to a long date value.</td>
</tr>
<tr>
<td>Selection</td>
<td>The decimal value converts to an integer value.</td>
</tr>
</tbody>
</table>

#### Integer fields

### From <other data type> to Currency:

<table>
<thead>
<tr>
<th>Data conversion rule</th>
<th>From &lt;other data type&gt; to Currency:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character or Diary</td>
<td>Parses the string to get a number and symbol. If the symbol is valid, the following rules are applied:</td>
</tr>
<tr>
<td></td>
<td>✓ If the symbol is part of allowed currencies for the field, the symbol and number are used as is. For example, if (100\ \text{ARS}) is entered, the data is converted to (100.00\ \text{ARS}).</td>
</tr>
<tr>
<td></td>
<td>✓ If the symbol is not part of the allowed currencies for the field:</td>
</tr>
<tr>
<td></td>
<td>✓ If a currency ratio exists between the symbol and primary allowed currency for the field, the value is converted to the primary allowed currency.</td>
</tr>
<tr>
<td></td>
<td>✓ If no ratio exists between this symbol and primary allowed currency for the field, the data is set to NULL.</td>
</tr>
<tr>
<td>Decimal, Real, or Integer</td>
<td>Converts the numeric value to Decimal, then appends the primary allowable currency type.</td>
</tr>
<tr>
<td>Date/time, Date, or Time</td>
<td>Converts the numeric value of the timestamp to Decimal, then appends the primary allowable currency type.</td>
</tr>
<tr>
<td>Selection</td>
<td>Converts the numeric value to Decimal, then appends the primary allowable currency type.</td>
</tr>
</tbody>
</table>
Integer fields accept integer values between –2147483647 and 2147483647. You can use integer fields to process statistical information in reports.

**Real fields**

Real number fields accept and contain floating-point numbers, which are useful for displaying very small and very large numbers. You can use real fields to process statistical information in reports.

**Decimal fields**

Decimal fields accept and contain fixed-point decimal numbers. Real number fields and decimal number fields differ in the following ways:

- Real numbers are inherently defined to be approximations, and decimal numbers are defined to be exact.
- Decimal numbers can be higher precision.
- The administrator has control over the fractional portion in a decimal field.

The total number of values in a decimal number field can be as many as 28 places long. This number includes the decimal places (up to 9) you define in the Precision field. You can control the precision of the number by defining where the decimal point is placed. Decimal number fields appear right-justified. Decimal, digit grouping, and negative sign symbols are based on the user’s locale settings, which are found in Regional Settings Properties window.

You can use decimal fields to process statistical information in reports.

**Selection fields**

Selection fields provide for a small number of choices. Selection fields are displayed as one of three types:

- Drop-Down List—Users can select a list of choices.
- Radio Button—Users can select from a visible set of choices.
- Check Box—Users can select only one choice.
The user cannot enter choices that are not included in the definition of the selection field. (This is one distinction between selection fields and character fields with menus.)

Data for a selection field is stored in the database as an integer that relates to the order of the choices. AR System can automatically number IDs, or you can choose the ID for each.

If you choose the IDs, the IDs can contain gaps between the numbers, enabling you to insert new values at a later time. For example, you might create a list of values as follows:

<table>
<thead>
<tr>
<th>Numeric value</th>
<th>String value</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Open</td>
</tr>
<tr>
<td>20</td>
<td>In Development</td>
</tr>
<tr>
<td>35</td>
<td>In QA</td>
</tr>
<tr>
<td>39</td>
<td>Closed</td>
</tr>
</tbody>
</table>

At a later time, you can add a set of numeric and string values, such as 15 and Pending, which would be placed between Open and In Development.

If AR System numbers the IDs for you, do not change the order of existing selection field items, or the meaning of data previously entered in the database will be changed. For example, in a Status field, if the current choices are New, Assigned, and Closed, and you add a choice labeled Fixed before the Closed selection, existing database entries with a status of Closed will change to a status of Fixed.

If you must add a new choice to a selection field on an active form, add it only as the last choice, as in the following example:

```
New = 0
Assigned = 1
Closed = 2
Fixed = 3
```

**WARNING:** Altering the choices in a selection field might require explicit modifications to every request in a form.

For check boxes, you can define more than one choice, but users only have access to the first value.
You can define searches using the numeric value of choices in the selection field. For example, the search string ‘Status < 2’ will search for all New and Assigned requests. The search string ‘Check Box = $NULL$’ will search for requests where the check box was not selected.

Use selection lists only in cases where you do not expect the options available to users to change over time. For information about creating or changing choices in a selection field, see “Creating selection fields” on page 179.

Table fields

The AR System supports two types of regular table fields: list view table fields and tree view table fields. Two other special table fields are results list fields and alert list fields. These fields can be used by the client and the server. When using table fields, remember the following important distinction:

- Active links work on table fields on the client.
- Filters work on table fields on the server.

Depending on your needs, you might use the same table field in one context as a server-side table field, but in another as a client-side server table. The choice depends on where you want to process the information: the client or the server.

Client-side table fields enable users to view specific fields and requests from a form in tabular format. Requests are displayed in the field based on a search of the supporting form.

A server-side table field is any table field (table, results list, or alert list) that appears in server-side workflow in filters or filter guides. Server-side table fields let you use filters in performing calculations on a list of records.

With a server-side table field, the goal is not to display the entire list of records to the client, as you would with a client-side table field; you just want to return the final result. Accordingly, you can create filters to find a specific row in a table field (for example, find the last Entry ID), and then perform actions based on specific criteria. You could also use functions that compute how many records there are, for example, for a given user, or even how many records simply exist.

Data in a server-side table field is read-only.
The most important reason for using server-side table fields is network performance. If you are evaluating and performing actions on large amounts of data (for example, Push Fields actions), you will find a great benefit in performance because you are not using API calls from the client to the server. Using filters with server-side table fields creates less network traffic than a client-side table field where the client must communicate with the server, or where a browser must communicate with the mid tier to the server, and back to the browser.

For information about server-side table fields and filter guides, see the Workflow Objects guide.

**List view table fields**

Use a list view table field in a form to display the results of a search. You can define which fields from the supporting form are displayed in the table. You can have multiple table fields on one form.

Controls for a table field in BMC Remedy User are contained in a right-click menu, as shown.

![Figure 4-2: Table field—Windows view with menu](image)

Similar controls appear as links and buttons below the table in a browser, as shown.
Tree view table fields

Use a tree view to display data from forms in a hierarchical manner. To view or select data, a user simply navigates the tree by clicking the tree nodes to expand them. Then, workflow that the administrator builds determines what happens when a node is selected.

Unlike list view table fields, tree view fields do not have controls such as Refresh or Select All. Users can simply open and close nodes by double-clicking on a node or selecting the plus (+) or minus (-) sign next to the node.

**Note:** Be aware of how NULL values are displayed in tree view fields. When users select a parent node and any of its child nodes have data, all nodes (including NULL nodes) will be displayed. If an entire branch is NULL, the first child node will appear, but the rest of the nodes will be collapsed. If a parent has one child, which is NULL, that child will not be displayed. NULL data is represented with the words No Value in brackets ([No Value]), unless you specify a different string on the Display tab of the Field Properties dialog box for the tree view field.
You might want to create a tree view field to enable users to look at information within a hierarchical relationship of data. For example, a user could use a tree view that lists assets to find assets associated with a particular laptop. The asset list is a dynamic list that is pulled from the asset database. The user opens the appropriate nodes to narrow down to the laptop in question and lastly clicks on the laptop name. The components can be displayed in the fields next to the tree view, which is populated through workflow that you set from actions associated with the tree.

A tree view can obtain data from only one form, but you can have multiple tree views on a form.

**Note:** To make sure a list view or tree view field appears with data when a user opens a form, you must create a ChangeField action that refreshes the tree view when the window is loaded. For information about the Change Field action, see the Workflow Objects guide.

---

**Results list fields**

Forms viewed in BMC Remedy User and in a browser include a results list pane that shows a list of requests after a search.

For forms on a pre-6.3 mid tier, you can set a preference in BMC Remedy Administrator to add a results list field to every web view, as described in the Getting Started guide.

In the Table Property tab of the results list Field Properties dialog box, you can define which fields appear as columns. You define which fields appear as columns in the results list pane in BMC Remedy User in the Form Properties dialog box. For more information, see “Setting form properties” on page 131.

The results list field displays details of a record in a table (or “drill-down”) on the web. To support table drill-down, you do not need to define any particular columns in the results list field. If you are only using the results list field to support a table, and not to display the results of a search, consider hiding the results list field.

You can have only one results list field on a form; however, each view of that form can display different columns.
The results list field is visible only during a query. When the user switches to submit mode, the results list field becomes hidden. Consider where to position the field in the view so that you achieve the desired layout in different modes.

Alert list fields

Use an alert list field to display an alert list in a web view, as shown. You can insert only one alert list field on a form. The supporting form for an alert list is the Alert Events form, but when users drill-down, they see the request on the form that generated the alert. To support drill-down from the alert list field, the forms originating the alerts must contain results list fields.

Figure 4-5: Alert list field

![Alert list field](image)

Each AR System server installation includes the Alert List form, which includes an alert list field in a web view. You can add this form to your web-based applications, or use your own form. See the Configuring guide for more information about implementing alert lists on the web.

Table field structure

This section discusses the structure of table fields. Because list views (which include results list and alert list fields) and tree views have somewhat different structure, the two types are discussed separately.

List view table fields (including results list and alert list fields)

In list view tables, each column represents a field from the supporting form, and each row represents a request from that form.

For list views and results list fields, you can define which fields will appear as columns. Almost any field can be defined as a field column; however, you cannot list attachment fields, diary fields, or character fields with a database length of more than 255.
The columns themselves are also fields, and you can specify their properties. In BMC Remedy Administrator, double-clicking a column causes the Field Properties dialog box for that column to appear. You can alternate between viewing column properties (by clicking a column) and table field properties (by clicking inside the table field).

**Tree view table fields**

In tree view tables, each level represents a field from the supporting form, and each node represents a request or set of requests from that form. (Several rows might be associated with a node.) If a user selects a node, the first row that matches the node (and its parent nodes) is selected. Be aware of this functionality when creating workflow with tree views.

For tree view tables, you can define which fields will appear as levels. Note that you cannot list attachment fields, diary fields, or character fields with a database length of more than 255.

The levels themselves are also fields, and you can specify their properties. In BMC Remedy Administrator, double-clicking a level causes the Field Properties dialog box for that level to appear. You can alternate between viewing level properties (by clicking a column) and table field properties (by clicking inside the table field).

The Displayed Levels property in the Sort/Levels tab of the tree view properties determine what is displayed in a tree view. Each column in the table becomes one level in the tree view.

**Note:** Display-only columns cannot be tree levels.

**Comparing list view and tree view table fields**

The following figure illustrates a simple comparison of a list view table and a tree view table that are derived from the same source fields. You can see that the list view displays the field names as column headers, and the rows display the field’s data. In the tree view, the field’s data is all that appears.
Table field function

When the table field is refreshed, information that meets the field’s search criteria (such as parts ordered, delivery status, and price) appears beneath the column headings. If new entries that meet the table field’s search criteria are made to the supporting form, the user will see them when the table is refreshed.

You can set the Refresh on Entry Change property (see page 432) to automatically refresh data whenever a request is displayed, or to allow manual refresh. If this property is not set, the table or tree is cleared when the user switches from one entry to another.
You can also set the Data Chunking property to “chunk” (or display in parts) the requests returned (see page 402). For example, you set the size of chunks to 5, and up to 5 requests are initially displayed. If more than 5 requests are returned, you can determine how users proceed to the next chunk—by clicking a hyperlink that you define in Advanced Properties for the table, or by clicking a button that triggers an active link. For more information about configuring chunks, see “Advanced Display properties (for tables)” on page 400.

**Note:** For tree views, chunking is turned off in BMC Remedy Administrator.

If you set the chunking through the ar.conf (ar.cfg) file as described in the Configuring guide, users will see only the first chunk in the tree, and there will be no way to go to the next chunk.

You can allow users to dynamically define the search criteria by using the EXTERNAL() operator. For more information, see the Workflow Objects guide.

Users can view and edit row data (see “Display Type” on page 434). If you give users permission, they can change the data in a table or alert list field by “drilling down” to the source request. For results list fields, users can display and modify a record in the current form by selecting that record in the results list field. For tree views, users must double-click the leaf node to open the supporting form, which might show one or more requests. Users must have permission to the supporting form, to the fields on the supporting form, and to the table field. For more information, see “Table field permissions properties” on page 62.

On all table types (except tree views), you can set properties to make rows appear in different colors based on the value of a selection field item. For example, as shown in Figure 4-7, a table field might use data showing various types of engine parts that have been ordered from multiple vendors, with Backordered in red to alert the assembly manager. For more information, see “Results Color properties” on page 446.

**Figure 4-7: Table field—Windows view**

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Part</th>
<th>Status</th>
<th>Vendor</th>
<th>Backordered</th>
<th>Assembly Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456</td>
<td>Engine Block</td>
<td>In Stock</td>
<td>ABC Automotive</td>
<td></td>
<td>Mark Scott</td>
</tr>
<tr>
<td>789012</td>
<td>Brake Pad</td>
<td>Backordered</td>
<td>XYZ Manufacturing</td>
<td></td>
<td>Jane Johnson</td>
</tr>
<tr>
<td>345678</td>
<td>Radiator</td>
<td>In Stock</td>
<td>AutoPlus Inc</td>
<td></td>
<td>Lee Chan</td>
</tr>
<tr>
<td>901234</td>
<td>Cylinder</td>
<td>Available</td>
<td>More Parts</td>
<td></td>
<td>Jane Smith</td>
</tr>
</tbody>
</table>
Workflow considerations for table fields

You can use active links and filters to allow users to modify data in the supporting form of a table field as follows:

1. Create a Set Fields action to pull request data from the row into a set of fields. The user can then modify the values in this set of fields.
2. Create a Push Fields action to push the new values into the request on the supporting form.
3. Create a Change Field action to refresh the table.

You can create guides (active link and filter) that loop through rows in a table. The guide selects each row in a table field (without highlighting) and performs a series of workflow actions on the row. This action is also known as “walking” a table field. For more information, see the Workflow Objects guide.

You can also select rows and create statistics on columns such as sums, averages, maximum and minimum values, and both total number of rows or total number of non-NULL values in a column. For more information about functions you can use with table fields, see the Workflow Objects guide.

Attachment pools

An attachment pool contains attachment fields that enable users to store text, graphics, audio, or video with a request. Attachment data is compressed and stored in the database with each request.

Attachment pool structure

Each form can contain any number of attachment pools, and each attachment pool can contain any number of attachment fields, subject to database limitations. You also define the maximum size for each attachment field (in bytes). Depending on field setup, the attachment pool shows the names of attached files, or displays columns of information about each attachment. Attachment pools do not have labels, but you can use trim to label the attachment pool on the form.
You set the properties (including permissions) for the attachment pool, and for each of the attachment fields within the pool. View permission allows users in a group to view attachments. Change permission allows users in a group to add and remove attachments in an attachment pool.

**Attachments viewed in BMC Remedy User**

In BMC Remedy User, attachments can be viewed as icons or with details, such as the file size. When creating a new request or modifying an existing request, the user right-clicks in the attachment pool to open a menu with commands to change the view; to add, delete, or display attachments; or to save them to a local disk. Users can also add attachments by dragging files into the attachment pool. The following figure shows an attachment pool with its menu displayed in BMC Remedy User.

**Figure 4-8: Attachment pool—Windows view**

![Attachment pool—Windows view](image1.png)

**Attachments viewed in a browser**

For forms that will be viewed in a browser, you define the columns that appear at the top of the attachment pool, as well as the add, save, delete, and display commands, which appear as buttons below the pool. When creating a new request or modifying an existing request, the user clicks an available button to perform the specified function, as shown.

**Figure 4-9: Attachment pool displayed in a browser**

![Attachment pool displayed in a browser](image2.png)
You can define labels for attachment pool functions, such as Add, Delete, Display, and Save to Disk. These functions appear as buttons in web clients (as shown in the previous figure), and as right-click menu items in BMC Remedy User.

If a label is left blank, the button, or right-click menu item will not be displayed. Depending on the labels you define for attachment pools on the web, you might want to adjust the size of the pool to accommodate them.

**Attachment pool function**

Attachment pools function slightly differently in BMC Remedy User and in forms viewed in a browser.

**Attachments viewed in BMC Remedy User**

In BMC Remedy User, when a user drags or adds a file to the attachment pool, the file will be added to the next available attachment field. If no empty attachment field is available, the user is prompted to replace the file in the first attachment field. When a user selects Add or Delete, attachments are added to or deleted from the database when the request is saved.

---

**Note:** When users perform the Copy to New operation in BMC Remedy User, the new request does not include the attachments and diary fields. Users must save attachments to files and then attach them to the new request. Web clients do not have a Copy to New operation.

---

**Attachments viewed in a browser**

Users viewing a form in a browser can add, delete, open, or save attachments by using buttons defined by the administrator in Attach Fields properties for the attachment pool. If the user selects an attachment field that does not contain an attachment file, only the Add button appears. For information about attachment field buttons, see “Attach Fields properties” on page 405.
Workflow considerations for attachment pools

The attachment pool and attachment field names you specify in the Database tab can be used as field references when creating workflow. You can use all attachment properties in active link, filter, and escalation actions. You can use the following supported actions:

- The Change Field active link action to show or hide an attachment pool and to set focus to it.
- The Message action to display the name of an attached file in a message.
- The Set Fields and Push Fields actions to transfer attachment data from one attachment field to another attachment field. You can also use the Set Fields action to display an attachment in a View field.
- The Set Fields and Run Process active link actions to add, save, open, or delete attachments. The available Run Process commands are:
  - PERFORM-ACTION-ADD-ATTACHMENT
  - PERFORM-ACTION-DELETE-ATTACHMENT
  - PERFORM-ACTION-OPEN-ATTACHMENT
  - PERFORM-ACTION-SAVE-ATTACHMENT

For more information about these commands and actions, see the Workflow Objects guide.

View fields

A view field is a display-only field that displays any type of document that a typical browser can display, including:

- URLs for locally stored HTML files or published web pages (BMC Remedy User and Internet Explorer clients only).
- Attachment field contents, such as HTML files, image files, and Microsoft Word documents.
- Views of AR System forms.
- HTML source code with embedded field references.
View field structure

You set the initial value of the view field by setting a Display property or by using a Set Fields active link action. You can add multiple view fields to a form to display various pieces of data from different sources. You can also use different initial display values for every view of a form to customize the presentation for any language, platform, or user role.

View field function

The contents of a view field are displayed in BMC Remedy Administrator after you specify a URL, file, AR System form view, or HTML code as an initial value and close the Display Properties window. If the view field is too small to display the entire document, scroll bars appear on the right side and bottom of the field depending on how you define Advanced display properties values. See “Creating view fields” on page 199.

View fields cannot be resized.

Figure 4-10: View field with scroll bars

The contents of a view field do not load automatically when the user opens a form. In New mode, users can see the contents if the Set To Defaults user preference is set. In New or Search mode, you can set the initial value with an active link that executes on Window Loaded and sets the value of the view field.
You can also make the contents of the view field persist across actions in the same window by making the view field a window-scoped global field. Create the view field with a field ID in the range of 3000000 to 3999999 and use workflow to set the initial value. For more information, see “Global fields” on page 218.

To print a document in a view field in BMC Remedy User, right-click on the field and choose Print from the menu that appears. (On a web client, you cannot print the contents of a view field.)

**Workflow considerations for view fields**

A view field participates in active links and filters as a display-only character field. It cannot participate in escalations, and it cannot trigger active links. You can use workflow to set focus to view fields.

**Data visualization fields**

The data visualization field provides a framework and services for BMC Remedy Mid Tier-based graphing solutions. It provides an efficient way to add graphical fields such as Flashboards to AR System forms. To learn how to create a data visualization field on a form, see “Creating data visualization fields” on page 201. For more information about developing modules for data visualization fields, see the Integrating with Plug-ins and Third-Party Products guide.

**Application list fields**

An application list field displays a list of available entry points. The contents of this list are automatically generated from AR System. It is composed of the available applications, forms, and entry point guides on a given server.
Any form that contains an application list field can be used as a Home Page form, as shown in Figure 4-11. You can insert only one application list field on a form.

For more information, see Chapter 9, “Defining entry points and home pages.”

Workflow considerations for application list fields

You might want to display only a subset of applications and entry points to users. Accordingly, you can create sophisticated workflow using a reserved character field ID that lets users see only a subset of entry points. Here you use a Run Process command (PERFORM-ACTION-HOME-FIELD-REFRESH) to display dynamically a subset of servers and applications.
Horizontal and vertical navigation fields

Horizontal and vertical navigation fields allow users to navigate to the correct screen quickly and easily. A horizontal navigation field might allow a user to move from application to application, and a vertical navigation field might give the user access to common functions and application entry points within an application.

Figure 4-12: Horizontal and vertical navigation fields

Horizontal navigation fields can have only one level in their structure, but vertical navigation fields can have an unlimited number of levels. (No more than three levels is recommended to keep your application’s navigation simple.)

Navigation fields can act as an anchor for other menu fields, and you can have multiple navigation fields with menus attached to them.

To show and hide menus and items in a navigation field, you must set the proper permissions for each menu and item. For more information, see "To create a horizontal or vertical navigation field" on page 205.
Workflow considerations for navigation fields

Remember the following workflow considerations when building workflow for navigation fields:

- You cannot define one navigation field that is shared across several forms. To overcome this limitation, you can create common workflow for the navigation fields on several forms.
- You can show or hide individual navigation field items or the entire field.
- You can enable and disable items on a navigation field.
- You cannot set focus or change the label, color, or font on a navigation field.
- You can set workflow to fire on a leaf node when it is selected, but workflow cannot fire on parent nodes.

Button fields

Button fields are control fields that are used to execute active links. For more information, see the Workflow Objects guide.

Figure 4-13: Button

Buttons can be displayed as URLs and then associated with an Open Window active link action to simulate a hyperlink that opens a new window. For more information, see “Display Type” on page 434. To add a real URL that links to a web page, see “Adding a URL to a text field” on page 214.

You can add an image to a button to enhance the look of forms displayed in BMC Remedy User. For more information, see “To add an image to a button” on page 209.
Page fields

A page field enables you to organize fields on a set of tabbed pages. When fields are grouped onto pages, the form is easier to use because users do not have to scroll through long forms to find a particular field.

Figure 4-14: Page field

Page field structure

Page fields are composed of a page holder containing one or more tabbed pages. You add fields to each page and arrange the page order and field layout. You can hide the page holder border and page tabs, and use workflow to display each page. You can specify some fields as shared fields, meaning that they appear on every page.

When creating page fields, you set properties for each individual page (including field ID) and for the page holder. In addition, you must provide permissions for each of the following levels:

- Page holder
- Individual pages
- Each field on each page

Users without permission to the page holder cannot see the pages or the fields. Users with permission to the page holder but not a page, cannot see any fields on that page. For more information, see “Page field permissions properties” on page 62.
Page field function

For page fields with a border and tabs, users click on a tab to display a page. If you choose not to display the page holder border and the page tabs in a form that will be viewed in a browser, you must create workflow to allow users to change which page is displayed. See the procedure “To create a page field” on page 209 for information about removing the border and tabs.

Workflow considerations for page fields

Remember the following workflow considerations when building workflow for navigation fields:

- When executing active links, a specific page gains focus when any of the following conditions occur:
  - A user clicks a tab.
  - A user tabs to a page.
  - An active link sets focus to a page.
- A page in a page holder loses focus when any user or workflow operation causes the focus to move from a page tab. Page visibility is not always dependent upon field focus. You can use the following workflow actions with page fields:
  - Use the Set Fields action on a page holder to bring a specific page (by using its database name) to the top of the page holder without setting focus.
  - Use the Change Field action to change focus to a page field, to make a page or page holder hidden or visible, or to set the page label color.
- You can use a page holder as a data source in workflow. For example, you can use a Run If qualification such as 'PageHolder' = 'Page3'.
- If users have permissions to a table field on a page field or page holder but they do not have permissions to the page field or page holder that contains a table field, active links and active link guides will not be able to access that table. To work around this issue, give users permissions to the page field holder, and then hide the holder.

For more information, see the Workflow Objects guide.
Trim fields

Trim fields are lines, boxes, and text (including URLs) that enable you to modify the appearance of a form. You can use vertical or horizontal lines, rectangles, or text to group similar fields together or to emphasize parts of the form. Types of trim are shown in the following figure.

Figure 4-15: Lines, boxes, and text

Each piece of trim is treated as a field by AR System, meaning that it has a unique field ID, a name, and display information. (No database information is associated with trim.) Treating trim as a field enables you to:

- Address each piece of trim in your workflow. For example, you can use a Change Field active link action to make a box hidden or visible.
- Change how or if the trim appears in different form views.
- Define permissions for the trim to match the permissions of associated fields so that the trim is not visible if the fields are not visible.
- Add and change trim in the same way that you add and change other fields.
- Create URLs that link the form or application to the Internet. For more information, see “Adding a URL to a text field” on page 214.

For forms used in BMC Remedy User, you can place a transparent trim box field over existing fields to visually group the fields. This is not recommended for forms that will be viewed on the web. For more information, see “Creating trim fields” on page 213.
You can create new fields or modify existing fields in a form at any time. All changes take effect as soon as you save them to the database, but if a user has a form open during modification, the user must close all instances of the form and reopen it to see your changes. The following topics are provided:

- Determining what types of fields to use (page 174)
- Creating data fields (page 174)
- Creating currency fields (page 176)
- Creating selection fields (page 179)
- Creating table fields (page 182)
- Creating attachment pools (page 194)
- Creating view fields (page 199)
- Creating data visualization fields (page 201)
- Creating application list fields (page 202)
- Creating horizontal and vertical navigation fields (page 205)
- Creating button fields (page 207)
- Creating page fields (page 209)
- Creating trim fields (page 213)

**Note:** For a detailed discussion of each of the tabs in the Field Properties window, see Appendix D, “Field properties,” on page 397.
Determining what types of fields to use

When creating a form, determine what type of information your form will contain. Adding new fields should be motivated by the planned use of the fields. Some possible uses include:

- Data fields that all users need. These fields should be grouped together.
- Data fields used by selected groups of users. Consider grouping these fields on separate tabbed pages.
- Data fields that contain information not presented to users. Consider hiding them from all views.
- Temporary workflow fields. These fields store temporary, working values used during workflow processing. Consider hiding them from all views because users do not need to interact with these fields.
- Visual cue fields. Trim fields, page fields, view fields, and images on button fields provide cues to users on how best to use each form.
- List-oriented fields. Use table fields when presenting data lists on forms.

Add fields carefully since you might find it impractical to eliminate a field after users have come to rely on it. In addition, how you administer fields can affect performance. For more information about system performance and fields, see the Optimizing and Troubleshooting AR System guide.

Creating data fields

Use the following procedure to create all data fields, except for selection and currency fields, which are discussed in “Creating selection fields” on page 179 and “Creating currency fields” on page 176.

To create a data field

1. Open the form with which you want to work.
2. Choose Form > Create a New > <field type>.
   The new field appears in the upper-left corner of the Form window.
   You can create a field by selecting it from the field palette that is displayed when you choose View > Toolbars > Palette.
3 Open the Field Properties window by double-clicking the field. The Field Properties dialog box appears.

**Figure 5-1: Field Properties—Character Field**

You can set your preferences to open the properties window whenever a field is created. (Modify the Preferences dialog box as described in the Getting Started guide.)

If the Keep Window Open check box is selected, you need not close the properties window to modify other fields in the form. The properties window displays the properties of the selected field.

4 Specify the field properties, which are listed in Appendix D, “Field properties,” on page 397.

5 Select the field, drag it to a position in the form, and adjust its size.

See “Arranging fields on a form” on page 282 for more information.
Choose File > Save Form to save your changes. For some fields, you can set Attributes properties. For more information, see the next section.

Creating currency fields

To create a currency field, see the following sections.

Defining default currency types

Before you create a currency field, you should specify default allowable and functional currency types in the Server Information window to predefine these currency types for all new currency fields.

You can also define allowable and functional currency types and precisions for individual currency fields. See “Allowable Types properties (currency fields only)” on page 403 and “Functional Types properties (currency fields only)” on page 437.

To define default currency types

1. In BMC Remedy Administrator, select a server.
2. Choose File > Server Information.
3. Select the Currency Types tab.
4. Select a currency under Choose Default Allowable Types, and click Add.
5. Select a currency under Choose Default Functional Types, and click Add.

For more information, see “Currency fields” on page 147.

6. To specify a decimal precision, click the number under the Precision column and edit as appropriate.
7. Repeat steps 4 through 6 to define all default allowable and functional currency types and precisions.
8. To delete a default currency, select the currency under the Currency Type column and click Remove.
9. Click Apply.
Creating a currency field

This procedure explains the basic steps for creating a currency field using BMC Remedy Administrator.

To create a currency field

1. Open the form with which you want to work.
2. Choose Form > Create a New > Currency.
   A currency field appears in the upper-left corner of the Form window.
3. Open the Field Properties window for the new currency field.
   The Field Properties dialog box appears.

Figure 5-2: Field Properties—Currency Field
Define allowable and functional currency types in the Allowable Types and Functional Types tabs.

You must have at least one functional currency. For more information, see “Allowable Types properties (currency fields only)” on page 403 and “Functional Types properties (currency fields only)” on page 437.

Specify the other currency field properties, which are listed in Appendix D, “Field properties,” on page 397.

Select the currency field, drag it to position it on the form, and adjust its size with the resize handles.

See “Arranging fields on a form” on page 282 for more information.

Choose File > Save Form to save your changes.

**Defining currency ratios**

The following procedure describes how to create currency exchange ratios for converting allowable currencies to functional currencies. You must create currency ratios from every allowable currency to every functional currency.

**Note:** If you do not supply a complete set of currency ratios, your applications might not work properly. For example, sorting on currency fields might produce unexpected results.

**To define currency ratios**

1. On the server where you defined currency types, open the AR System Currency Ratios form in New mode in BMC Remedy User.

2. Create two entries for each pair of currencies to which and from which you want to convert.

For example, you might create one entry for converting from USD to GBP, like this:

<table>
<thead>
<tr>
<th>Field</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conversion Date</td>
<td>$TIMESTAMP$</td>
</tr>
<tr>
<td>From Currency</td>
<td>USD</td>
</tr>
</tbody>
</table>
Creating selection fields

To create a selection field

1. Open the form with which you want to work.
2. Choose Form > Create a New > <selection field>.
   Selection field choices are Drop-Down List, Radio Button, or Check Box.
   A selection field appears in the upper-left corner of the Form window.
3. Open the Field Properties dialog box for the new selection field.

Creating selection fields

Use the following procedures to create an attachment pool.

To define the currency ratio refresh interval

1. In BMC Remedy Administrator, open the Server Information window.
2. Click on the Timeouts tab.
3. In the Client Refresh Interval field, enter the interval in minutes.
   The default setting is 60. A setting of 0 disables refresh.
   This defines the interval by which clients query the server for new currency ratios from the AR System Currency Ratios form.
4. Click Apply.
Chapter 5—Creating fields

4 Specify Attributes properties, as described in the following procedures.

5 Specify other field properties, which are listed in Appendix D, “Field properties,” on page 397.

6 Choose File > Save Form to save your changes.

To add selection values

1 In the Field Properties window for a selection field, select the Attributes tab.

Figure 5-3: Field Properties—Status, Attributes tab

2 If you want to create custom IDs, complete the following steps:
   a Select the Custom option under ID Enumeration.
   b In the ID field, enter any value from 0 through 2147483647. Negative values are not permitted.

   If you select Custom, you must number all of the IDs manually. If you select Linear, AR System numbers the IDs beginning with 0. You cannot have custom IDs and automatically generated (Linear) IDs in the same selection field.
From the Selection Value list, select where you want to position the new value.

4 In the Value field, enter the value of the new item.

5 If you are localizing the form, enter localized text in the Alias Value field.

   For more information about manually localizing field labels, request aliases, and selection fields, see Appendix H, “Localizing AR System applications.”

6 Click Add Before or click Add After to add the item to the selection field.

   If you selected Linear under ID Enumeration, do not add choices in the middle of a selection field in an existing form, or the meaning of the data in existing requests will change, as described in the introduction to this section.

   For selection fields with a display type of Check Box, add only one item because the user will only have access to the first item.

7 Repeat steps 3 through 6 for each value you want to add.

8 In the Default Value field, enter the value that you want to appear whenever users load default values before performing a search or submitting a new request.

9 Choose File > Save Form to save your changes.

To modify selection values

1 From the Selection Values list, select a value.

2 In the Value, Default Value, or Alias Value fields, edit the information.

3 Click Modify.

4 Choose File > Save Form to save your changes.

To delete selection values

1 From the Selection Values list, select a value.

2 Click Delete.

   Do not delete choices in the middle of a selection field in an existing form.

3 Choose File > Save Form to save your changes.
Creating table fields

This section contains procedures for creating table (list view or tree view), results list, and alert list fields.

The following procedure explains the basic steps for creating a table, results list, or alert list field using BMC Remedy Administrator.

- Client-side table fields can be used in forms viewed in BMC Remedy User or in a browser, but results list and alert list fields are specifically for forms viewed in a browser.
- Server-side table fields are not used to propagate results to the client; they are used exclusively to manipulate data.

See “Advanced Display properties (for tables)” on page 400 and “Table Labels properties” on page 449 for property settings used for table, results list, and alert list fields in web views.

To create a table (list view or tree view), results list, or alert list field

1. Open the form with which you want to work.
2. Choose Form > Create a New > Table > <List View or Tree View>.
   To create a results list, choose Form > Create a New > Results List. To create an alert list, choose Form > Create a New > Alert List.
   The new field appears in the upper-left corner of the Form window.
3. Open the Field Properties window for the new field.
The Auto Fit Columns and Fixed Header check boxes do not appear when you create a tree field.

4 Specify Display properties, as described in “Display properties” on page 428. You can set display, database, permissions, help text, and change history properties for each column separately from those of the entire field. See “To define Column properties” on page 186.

5 Select the Table/Tree Property tab and specify table properties, as described in “Table/Tree Property properties” on page 453.

6 Specify Advanced Display and Table Labels properties.

   See “Advanced Display properties (for tables)” on page 400 and “Table Labels properties” on page 449 for descriptions of these properties.

7 Set the column or tree level properties (column width, ID, column field database name, the permissions, and so on) for each column.

   To open properties for a column, double-click on the column title or tree node.

Creating table fields ▶ 183
Note: When you are creating a column and you need to define a column ID (for example, to control the ID for purposes of shared workflow), make the changes to the ID before saving the table field.

See the procedure “To define Column properties” on page 186 for more information.

8 Specify Sort/Levels properties.
See the procedure “To define the sort order for table fields” on page 187 for more information.

9 Specify the other field properties, which are listed on page 397.

10 Select the field, drag it to position it on the form, and adjust its size with the resize handles.
On the web, tables (except tree views) have space allocated at the top and bottom for buttons such as Refresh and Select All, so make your table field a little taller in BMC Remedy Administrator to leave room for the buttons.

11 Choose File > Save Form.

To define Table Property properties

1 Open the Table Field Properties dialog box.

2 Select the Table/Tree Property tab.

3 For list view and tree view tables, select the server that has the supporting form you want to use for the table.

4 For list view and tree view tables, choose the form containing the data to be used.
Fields On Form is populated with field names. The following fields cannot be listed and do not appear in Fields On Form:

- Fields with a database length of more than 255.
- Display-only fields from the supporting form. (You can use display-only fields from the current form.)
- View fields.

5 (For deployable applications only) From the Restricted List field, select if the table field is restricted only to the forms in the application.
6 Use the Add and Remove buttons to move the fields between columns until the Field As Column shows the fields you want to appear.

You can add as many as 128 columns to a table.

7 To edit or delete a column title, right-click on any text under the Column Title heading, and select Edit or Delete.

Text beneath the Field heading matches the original field text in the supporting form. Text that you change beneath the Column Titles heading appears in the table field as the column title.

8 Use the up and down arrows next to Field As Column to position the order in which the columns will appear.

For tree views, the order of the fields is ignored.

9 For list view, tree view, and alert list table fields, enter a qualification statement to further define what data will appear in the table.

**WARNING:** If you do not include a qualification, all requests in the supporting form will appear. For information about using qualifications, see the Workflow Objects guide.

Click the Qualification down arrow to include a keyword or field (from the current or supporting form) in the qualification. See the Workflow Objects guide for information about using keywords.

10 To limit the maximum number of rows that will be displayed in a list view, tree view, or results list table field, enter a number for the Max Rows field.

The default is 0, which means that the maximum number of rows is unlimited. Because users can override the maximum row limit you set by selecting Refresh All in BMC Remedy User, use an appropriate qualification to control the rows of data that can appear.

Alternately, use chunking to display all records while limiting the number of rows displayed at one time. If you enable chunking, the Max Rows setting is ignored. For more information, see “Advanced Display properties (for tables)” on page 400.
To supply a dynamic server or form, select the Advanced check box, and then define a sample server and form.

For example, you could select the value of a server and a form that will function as the “sample” form whose entries will be displayed at runtime. For more information, see “Defining advanced table field functionality” on page 191.

Choose File > Save Form.

To define Column properties

1 Double-click the column title in the list view, results list, or alert list table field (not in the field itself) to set the properties for each column.

For tree views, double-click the level. Tree nodes (columns) that are not part of the sort order are disabled. For more information about sort order, see “To define the sort order for table fields” on page 187.

The Column Properties dialog box appears.

Figure 5-5: Field Properties—Column, Display tab
2. Under the Display tab, select a display type from the Display Type menu. For more information, see “Display properties” on page 428.

Note: For columns in tree view table fields, the Editable and Read Only-HTML options are ignored.

3. To set the width of the selected column, enter a number in the Column Width field of the Column Properties dialog box. This field is not relevant for tree views.

4. To wrap multiple lines of data in table column fields or to allow carriage returns in row data, select Wrap Text. This option is not relevant for tree views.

5. For display-only fields, optionally specify a default value in the Default Value field. For more information, see “Default Value” on page 412.

6. To hide the current column, select the Hidden check box. This option is not relevant for tree views.

7. To rename what this column is called in the database, select the Database tab, and change the default in the Name field.

8. Choose File > Save Form.

To define the sort order for table fields

1. Open the Field Properties window for a table field.

2. Select the Sort/Levels tab.

3. Select the fields from the Available Fields field by which you want to sort, and click Add to move them to the Sorted Columns field. (For a tree view, select the fields that you want to be visible.)

By default, if you do not define a sort order for list view, results list, or alert list, the data will appear in ascending order based on the Request ID number.

If you do not define a sort level for tree views, AR System will sort the tree by the first column chosen on the Table/Tree Property tab.

On Table/Tree Property tab, if you added character fields with a data length of 0 or more than 255, those fields will not appear on the Sort/Levels tab.
4 Use the up and down arrows at the top of the Sorted Columns field to set the sort precedence of each column in the table or level in the tree. The field name that appears at the top of the Sorted Column list has the highest precedence. Requests matching the field setting with the highest precedence appear at the top of the table list or tree level.

For example, in list view table fields, if you set the Assembly Manager field before the Part Number field, the requests will appear alphabetically by Assembly Manager. In tree views, if you set Assembly Manager before Part Number, Assembly Manager values would simply be higher in the tree hierarchy (further to the left in the tree display).

5 Use the up and down arrows next to each column title (within the Sorted Columns field) to set the sort direction of each field.

Clicking an arrow toggles the field between ascending and descending order. Ascending order means that values like lower ID numbers or earlier dates appear at the top of the table list. Ascending order for character fields means that requests are sorted alphabetically.

6 Choose File > Save Form.

▶ To define Advanced Display properties

1 Open the Table Field Properties dialog box.
2 Select the Advanced Display tab.
3 Select properties for the table or results list field.
4 Choose File > Save Form.

▶ To define Table Labels properties

1 Open the Field Properties dialog box.
2 Select the Table Labels tab.
3 Select properties for the Header, Body, and Footer.

You can write strings in different languages for localized form views.

You can remove a button, function, or message string by clearing the appropriate edit box. For example, if you clear the contents of the Refresh Button field, the Refresh button will not appear below the table in web clients, and the Refresh menu item will not appear when you right-click on the table in BMC Remedy User.
Table field column preferences

You can enable preferences that let users customize column settings in table fields. Users can add and remove columns, resize columns, change the order in which columns appear in a table, and change the order by which data is sorted in a table. When logged into a preference server, users can save these settings to the preference server, making them available for future logins in both BMC Remedy User and the Web.

Users can set preferences for any table columns that are visible.

To enable preferences, enter a label in the Preferences field of the Table Labels tab of the Properties dialog box for the table field. For more information, see “To define Table Labels properties” on page 188.

The following sections describe the table column preference options available and how they work on the web and in BMC Remedy User.

Web

When column preference settings are enabled for a table field, a button appears in the table heading with the label Preferences (or another label name if you change it).

Figure 5-6: Preference button and menu items in web table header

This button, when clicked, displays a drop-down menu with the following options:

- **Add Column**—Displays a list of available columns that can be added to the table. The list includes only those columns made visible by the administrator, and which have a width of zero (previously hidden by the user).
Note: If a table is set to AutoFit Columns, the width of all columns will be adjusted so that they will fit into the table. As a result, when a column that was previously removed is added back to the table, its width might not be the same as the administrator-defined width.

- **Remove Column**— Displays a list of columns that the user can remove from the table. This list includes only columns that the administrator has made visible, and whose width is greater than zero.

- **Reset**— Restores the column width, column order, visibility, and sort order to their administrator-defined default values.

- **Save**— Saves the settings to the user’s preference server, making them available from a centralized location for future logins. If the user is not logged into a preference server, this option is disabled.

**BMC Remedy User**

When column preference settings are enabled for a table field, a menu appears with the label Preferences (or another label name if you change it) when the user right-clicks anywhere in the table.

**Figure 5-7: Menu with table preferences submenu in BMC Remedy User**

This menu appears with a submenu that includes the following options:
Creating table fields

- **Add Column**—Displays a list of available columns that can be added to the table. The list includes only those columns made visible by the administrator, and which have a width of zero (previously hidden by the user).

- **Remove Column**—Displays a list of columns that the user can remove from the table. This list includes only columns that the administrator has made visible, and whose width is greater than zero.

- **Column Order**—Displays a dialog box from which users can change the order in which columns appear in the table.

- **Reset**—Restores the column width, column order, visibility, and sort order to their administrator-defined default values.

- **Save**—Saves the settings to the user’s preference server, making them available from a centralized location for future logins.

**Note:** The standard Save operation for a request will not save table field preferences. Users must select Save from the table preferences menu to save their table preferences. Also, users must be logged into a preference server to save table preferences.

---

**Defining advanced table field functionality**

You can define advanced table field properties that provide enhanced functionality beyond retrieving and displaying requests from a hard-coded server and form defined in the table field. Based on values that workflow or users enter into certain fields, you can dynamically select which server and form you are reading the table field data from.

For example, the Dynamic Table Field form contains two fields (server and form) that can be hidden or display-only. The values you enter into these fields (for example, values could come from a Set Fields action) determine at runtime which server and form are used as sources in the table field. When you refresh the table field, it is dynamically refreshed with values from the source server and form. Administrators might prefer to hide this functionality from users altogether and use the Window Loaded execution condition instead.
Figure 5-8: Using dynamic table fields

In this example, two records were created in the Source form on the server cordova. If you opened the Dynamic Table Field form in BMC Remedy User, entered cordova as the server and Source as the form, and then refreshed the table field, the contents of the table field would be refreshed by the underlying active link workflow. The table field displays the entries from the dynamically defined server and form.

To define advanced table field functionality

1. Select the Advanced check box on the Table Property tab.

The window changes in the following ways:

- Two icons appear next to the Server and Form fields. The first icon opens a dialog box for entering a long string of data into the Server or Form field. The second icon opens a menu that allows you to enter field references or keywords for the server and form names. These fields come from the current form on which the table field resides and are typically hidden.
The Sample Server Name and Sample Form Name fields appear (as shown in Figure 5-9). Default values listed are the current server and form. You use these "sample" fields to map the column fields. The "sample" form must contain fields that exist on any dynamic source forms that are used at runtime. But the sample server and form are not actually used at runtime. The fields you choose are only used to build the table field. Then, they are deleted.

**Figure 5-9: Advanced functionality of table fields**

2. In the Server and Form fields, enter a field or variable (for example, $server$) that will dynamically specify which server and form are used. The server and form that a user (or workflow) enters into those fields will be used as the source server and form whose data the table field will display.

3. From the Sample Server Name and Sample Form Name lists, select a reference server and form. The Fields on Form list updated with fields from the “sample” form.
The “sample” server and form are used as a temporary reference to create and save the dynamic table field settings. You can even delete the sample form after saving the table field. What is important is being able to select the field columns used in step 5.

4 Enter a qualification in the Qualification field to specify the request that contains the values that you want to retrieve.

The fields from the sample form can be used in the qualification. Conversion rules do not apply in this case.

5 Define the field columns you are including in the table field. Make sure you select fields that correspond to field IDs available on any dynamically selected form at runtime.

In this example, the table field includes two columns, Request ID and Short Description, which are AR System fields that are required on every form. To display the required entries in your table field properly, make sure that you have corresponding (or matching) fields included among all your forms. Otherwise, when you try to refresh the table field, you will receive an error that the field does not exist, and no data will appear in the table field.

6 Choose File > Save Form to save your changes.

Creating attachment pools

Use the following procedures to create an attachment pool.

To create an attachment pool

1 Open the form with which you want to work.

2 Choose Form > Create a New > Attachment Pool.

An attachment pool appears in the upper-left corner of the Form window.

3 Open the Field Properties window for the new attachment pool.
4 Specify Database properties as discussed in “To define database properties for attachment fields” on page 196.

5 Specify Attach Fields properties to add attachment fields to the attachment pool, as described in “To create an attachment field in an attachment pool” on page 196.

6 For web views, define button and column labels for attachment pool, as described in “Attach Fields properties” on page 405.

7 Specify the other pool properties, which are listed in Appendix D, “Field properties,” on page 397.

8 Select the attachment pool, drag it to position it on the form, and adjust its size with the resize handles.

**Note:** For web views, make sure to leave enough space on the form to accommodate the Add, Delete, Display, and Save to Disk buttons that appear below the attachment pool in the browser.
See “Arranging fields on a form” on page 282 for more information.

9 Choose File > Save Form to save your changes.

10 Specify properties for each attachment field within the attachment pool.

11 Choose File > Save Form to save your changes again.

To define database properties for attachment fields

1 Select the Database tab.

2 Click the ID field to edit the field ID for the attachment field.

3 Click the Name field to edit the database name for the attachment field.

4 Click the Input Length field, and enter the maximum size (in bytes) for the attachment field.

   The input length appears under the Max Size column in the attachment pool.

5 Set Entry Mode to Optional or Display-Only. (See “Entry Mode” on page 425 for more information.)

6 Repeat steps 2 through 5 for each additional attachment field.

7 To delete an attachment field, see the procedure “To delete an attachment field” on page 226.

8 Choose File > Save Form to save your changes.

To create an attachment field in an attachment pool

1 In the attachment pool Field Properties window, select the Attach Fields tab.

2 Enter a label for the attachment field in the Enter Attachment Field Label field.

   This label can be up to 80 characters, and will be used as the display label and database name for the attachment field. After saving the form, you can change the display label by modifying the attachment field label in this dialog box, or by defining Display properties for the attachment field. You can change the database name by defining Database properties for the attachment field. See Appendix D, “Field properties,” for information about defining properties for attachment fields.

3 Click the Add button to add the attachment field to the attachment pool.

   The name of the new attachment field appears in the Attachment Field List box.
Repeat steps 2 and 3 for each attachment field you want to add to the attachment pool.

The number of fields you add determines the number of attachments the user can add (assuming permissions are set for that particular user). You can specify as many attachment fields as you want, depending on your underlying database. If you exceed the limit your database sets, you will get a database error. See your database documentation for the maximum number of columns allowed in each database.

There is no automatic sorting of the fields in the attachment pool. Rearrange the fields by using the up and down arrows next to the Attachment Field List box.

Choose File > Save Form to save your changes.

To modify an attachment field in an attachment pool

1 In the attachment pool Field Properties window, select the Attach Fields tab.
2 Select the attachment field you want to modify in the Attachment Field List.
3 Click the Enter Attachment Field Label field and change the label for the attachment field.

This label can be up to 80 characters, and will be used as both the display label and database name for the attachment field. After saving the form, you can change the display label by modifying the attachment field label in this dialog box, or by defining Display properties for the attachment field. You can change the database name by defining Database properties for the attachment field. See Appendix D, “Field properties,” for information about defining properties for attachment fields.

4 Click the Modify button to apply changes to the attachment field.

The new name of the attachment field appears in the Attachment Field List box.

Choose File > Save Form to save your changes.
To delete an attachment field from an attachment pool

1. In the attachment pool Field Properties window, select the Attach Fields tab.
2. Click the name of the attachment field in the Attachment Field List box.
3. Click the Delete button to delete the attachment field from the attachment pool.
   The name of the attachment field disappears from the Attachment Field List box.
4. Choose File > Save Form to save your changes.

To define display labels for attachment pools viewed in a browser

1. In the attachment pool Field Properties dialog box, select the Attach Fields tab.
2. In the Display Labels in View section, click the field next to the label you want to modify.
3. Enter a new label, or clear the existing label.
   If you clear a label for a button, the button will not be displayed. If you clear a label for a column, the default column names will be used. When a user adds attachments, the file names will always appear in the attachment pool, even if you do not display any of the columns or buttons.
4. Choose File > Save Form to save your changes.

Size factors to consider when creating attachment fields

When setting properties for each attachment field within an attachment pool, set a maximum size that is large enough to accommodate typical attachments, but small enough to conserve database space. If you do not allocate enough space in the Input Length field and the size of the attachment exceeds what you have allocated, users will not be able to add attachments, even when there are empty attachment fields available in the attachment pool. If you set the maximum field size to 0 (the default), users can add attachments up to whatever size your DBMS will allow. Any other number you enter represents actual bytes.

The size of your attachments can also be limited by the available memory on your client machine. For example, your DBMS might allow attachments up to 2 GB, but in fact be limited to attachments of only a few MB at any given time due to memory constraints on the client side.
Creating view fields

Use the following procedure to create a view field.

**To create view fields**

1. Open the form with which you want to work.
2. Choose Form > Create a New > View.
   The new field appears in the upper-left corner of the Form window.
3. Click Field Properties.
   The Field Properties—View Field window appears.
4 In the Text field on the Display tab, specify an initial value for the view field. You can enter any value that can be read by a browser. For example, you can enter the URL of your company, the URL for a form on the web, an HTML snippet, or JavaScript. The web page, form, or interpreted code appears in the view field in BMC Remedy Administrator when you close the Field Properties window.

To display the view field initial value in BMC Remedy User or web clients, perform one of the following tasks:

- Configure user preferences to Set Fields to Default Values in new and search modes.
- Use a Set Fields active link action to set the view field value to $DEFAULT$ when the form opens. In this case, set the execute on condition to Window Loaded.

You can also leave the Text field blank and use workflow to set the initial display value.
WARNING: Do not use HTML snippets that contain tables or frames. Display this type of content using a URL instead.

5 Specify border and scroll bar properties on the Advanced tab, which is discussed in detail on page page 398.
   For more information, see “Advanced properties” on page 398.

6 Specify the other field properties, which are listed in Appendix D, “Field properties,” on page 397.

7 Select the field, drag it to a position in the form, and adjust its size.
   If the contents of the web page or HTML snippet are too large for the field, scroll bars will appear at the top and right side of the field in BMC Remedy User or web client.
   See “Arranging fields on a form” on page 282 for more information.

8 Choose File > Save Form to save your changes.

9 Use a text trim field to add a label as needed because view fields do not have labels.

Creating data visualization fields

Use the following process to create a data visualization field. (See the Integrating with Plug-ins and Third-Party Products guide for the first three steps.)

Step 1 Create a module on the mid tier.
Step 2 Register the module.
Step 3 Deploy a custom data visualization module.
Step 4 Add a data visualization field to a form (See the following procedure.)
To add a data visualization field to a form

1. In BMC Remedy Administrator, open the form for the data visualization field.
2. Choose File > New Server Object > Regular Form
3. Choose Form > Create a New Field > Data Visualization.
4. Click Field Properties.
5. In the Form Properties dialog box, click the Advanced tab.
6. In the data visualization Properties area, enter the following information:
   a. In the Module Type field, select the module type for the data visualization. This is defined on the Module Registration form, which is discussed in the Integrating with Plug-ins and Third-Party Products guide.
   b. In the Server field, select the AR System server that contains the data visualization module.
   c. In the Definition Name field, select the definition name for the data visualization module.
7. Choose File > Save Form.

You can now view the form with the data visualization field in BMC Remedy User or a browser.

Creating application list fields

Use the following procedure to create an application list field.

To create an application list field

1. Open the form with which you want to work.
2. Choose Form > Create a New > Application List.
   The new field appears in the upper-left corner of the Form window.
3. Click Field Properties.
   The Field Properties—Application List window appears.
4 In the Display tab, specify the Label.

5 Specify other field properties, which are listed in Appendix D, “Field properties,” on page 397.

6 Select the field, drag it to a position in the form, and adjust its size. For more information, see “Arranging fields on a form” on page 282.

7 Choose File > Save Form to save your changes.
Executing entry points in HTML

If you want to design custom HTML on a form, create a view field. This field should contain HTML that uses the arInvoke commands to execute entry points in BMC Remedy User and on the web:

- **arInvokeEntryPoint\(\text{"<servername>"","<guidename>"}\)** — Executes the specified entry point guide.
- **arInvokeGuide\(\text{"<servername>"","<guidename>"}\)** — Executes the specified guide.
- **arInvokeForm\(\text{"<servername>"","<guidename>"","<mode>"}\)** — Executes the specified form in New or Search mode. The options are mode are New and Search.

You can use single or double quotation marks in these commands. (JavaScript conventions allow for strings to be single- or double-quoted.)

**To execute entry points in HTML**

1. Create a view field.
   
   For more information, see “Creating view fields” on page 199.

2. Open the Field Properties window for the view field.

3. In the Text field on the Display tab, enter your commands.

   For example:

   ```html
   <html>
   <script>
   function doInvokeForm(server, form, mode)
   {
      if("external" in window && "arInvokeForm" in window.external)
         arInvokeForm(server, form, mode);
      else
         parent.arInvokeForm(server, form, mode);
   }
   </script>
   <body>
   <a href="javascript:doInvokeForm('ServerA','HelpDesk','search')">Search HelpDesk form</a>
   <br>
   </form>
   </body>
   </html>
   ```
Creating horizontal and vertical navigation fields

Use the following procedure to create a horizontal or vertical navigation field.

To create a horizontal or vertical navigation field

1 Open the form with which you want to work.
2 Choose Form > Create a New > Horizontal Navigation, or choose Form > Create a New > Vertical Navigation.
   The new field appears in the upper-left corner of the Form window.
3 Click Field Properties.
   The Field Properties—View Field window appears.
4 Click the Navigation Items tab.

Figure 5-13: Field Properties—Navigation Items tab

5 Add menus and items as described in “To add a menu to a navigation field” on page 206.
Click the Active Links tab, and add the appropriate active links to the field. For information about creating workflow, see the Workflow Objects guide.

As needed, specify other field properties, which are listed in Appendix D, “Field properties,” on page 397.

Select the field, drag it to a position in the form, and adjust its size.

Make sure that the field is large enough to hold the menus and items you created. (Horizontal and vertical scrollbars will appear in a navigation field in BMC Remedy User or a browser if the field is not large enough.)

Choose File > Save Form to save your changes.

(Optional) To change the styling of menus and items in forms on the web, create properties in the application’s cascading style sheet (CSS).

For information about CSSs, see the Installing and Administering BMC Mid Tier guide.

To add a menu to a navigation field

1 In the Field properties window for a horizontal or vertical navigation field, click the Navigation Items tab.

2 Click the Edit Navigation Items button.

3 Use the Add Menu, Add Menu Items, and Add Separator buttons to create the menus and items for your navigation field.

4 Click the Attach Orphaned Items button to attach orphaned navigation field and menu items that are not attached to the form.

Orphaned items come from:
- Menu items from the Edit Menu Bar dialog box (choose Form > Edit Menu Bar).
- Items from a navigation field that you deleted.

5 If you want a particular menu item selected when the user opens the form, select the item from the Navigation Initial State list.

Note: If you disable a parent item in a vertical navigation field, the parent item is collapsed and then disabled, so the user will not be able to access the children. (To disable an item, use the Change Field action.)
6 For each menu item, attach an active link that define the action that should occur when that item is clicked.
   a Select the menu item.
   b Click the Active Links tab.
   c Move the appropriate active link to the Selected Active Links column.
   d Click the Menu Bar Layout tab, and repeat these steps for each menu item.
7 For each menu item, click the other tabs and set the item’s properties as needed.
   For example, if you click the Permissions tab, you can set permissions for a menu item. Users will see only the menu items to which they have access.
8 Click OK.

Creating button fields

Use the following procedure to create a button field.

To create a button field
1 Open the form with which you want to work.
2 Choose Form > Create a New > Button.
   The new field appears in the upper-left corner of the Form window.
3 Click Field Properties.
   The Field Properties—Button window appears.
4 In the Display tab, specify the Button Label and Display Type. For more information, see “Display properties” on page 428.

5 In the Active Links tab, specify active links to execute when the button is clicked.

Active links define operations that are executed on the client machine when the user performs a specific action. Use the Active Link(s) tab in the button’s Field Properties window to connect active links to buttons or menu items, as explained in the Workflow Objects guide.

6 Specify other field properties, which are listed in Appendix D, “Field properties,” on page 397.

7 Select the field, drag it to a position in the form, and adjust its size. For more information, see “Arranging fields on a form” on page 282.

8 Choose File > Save Form to save your changes.
To add an image to a button

1. In the Field Properties window for the button, select the Image tab.
   For more information, see “Image properties” on page 439.
2. Click the Change Image button.
3. In the dialog box that appears, open the file for the image you want displayed on the button.
4. Change the image properties as needed.
5. Choose File > Save to save your changes.

Creating page fields

Use the following procedure to create a page field. For more detail about each tab of the Field Properties window, see Appendix D, “Field properties,” on page 397.

To create a page field

1. Set up the page holder.
   a. Open the form with which you want to work.
   b. Choose Form > Create a New > Page Holder.
      The page field appears in the upper-left corner of the Form window.
   c. Click the Field Properties button to open the Field Properties dialog box for the page holder.
Under the Database tab, enter a name for the page holder in the Name field of the Database. See “Display properties” on page 428 for more information.

Under the Permissions tab, set permissions for the entire page holder. Users must have permission to the page holder to see or change the fields on the pages. See “Page field permissions properties” on page 62 for more information.

Select the Pages tab to specify the number of pages you want and whether to display the border and tabs.

The Available Pages field lists the pages in the page field.

To remove a page in a form view, clear the check box to the left of the page in the Available Pages field.

If there is only one view, you will delete the page.
Note: By contrast, if you delete a page (or field in a page) from a view, the page (or field) is deleted in all views.

To add a new tab, click Create a New Page.

In a browser, only one row of tabs for a page field is displayed. If users will be viewing your form in a browser, size the page field appropriately or reduce the number of tabs so that users can view all the tabs.

See “Pages properties (page holder fields only)” on page 443 for more information.

Name and set properties for the tabbed pages.

In the form, open the Properties dialog box for first page tab you want to label.

The Field Properties dialog box for that tab appears.

Figure 5-16: Field Properties— Page
Under the Display tab, enter the label you want to appear for this page tab. The name you enter for the label will also appear as the Name field on the Database tab.

When you are creating a page and you need to define an ID (for example, to control the ID for purposes of shared workflow), make the changes to the ID before saving the page.

(Optional) Specify a font for the page tabs in the Color/Font tab.

You can select a different font for each tab in a page field. You can also specify font preferences for tabs by changing the Optional Field font setting. For information about changing font preferences on forms, see the Getting Started guide.

Set permissions for the page.

Users must have permission to see or change the fields on this page. For more information, see “Page field permissions properties” on page 62.

Specify the other field properties, which are listed in Appendix D, “Field properties,” on page 397.

Repeat these steps until you have named all the pages you want to appear in the tab set.

Add fields to each page, and set field permissions.

When adding fields to page fields, select the tab of the page field, and then add the field.

If you want a set of fields to appear on every page in a view, specify Shared Fields properties for the page holder, as needed.

See “Shared Fields properties (page fields only)” on page 447 for more information.

Specify the other field properties, which are listed in Appendix D, “Field properties,” on page 397.

Select the page holder, and drag it to position it on the form.

See “Arranging fields on a form” on page 282 for more information.

Choose File > Save Form to save your changes.
Creating trim fields

Use the following procedure to create a trim field.

► To create trim fields
  1 To create one of the trim elements, open the form with which you want to work.
  2 Choose Form > Create a New.
  3 Choose Horizontal Line, Vertical Line, Box, or Text.
     The new field appears in the upper-left corner of the Form window.
  4 Click Field Properties.
     The Field Properties dialog box appears.

Figure 5-17: Field Properties—Text window
5 Specify the field properties, which are listed in Appendix D, “Field properties,” on page 397.

By default, boxes and text boxes are opaque and use the background color of the form. To show the area on the form beneath a box (for example, when using a background image on a button and you want the text on top of the image), make the box or text transparent (see “Color/Font properties” on page 417), and bring it to the front (choose Layout > Bring to Front).

**Note:** In some browsers, users cannot click through a transparent trim box to the fields underneath. For applications that will be viewed on the web, place box trim fields under fields.

6 Select the field, drag it to a position in the form, and adjust its size.

When adding a box to a form, drag it by its interior to change its position or drag it by the corners or edges to change its size. Similarly, you can move and resize the box that surrounds text that you add.

- To change the length of a line, drag the anchors that appear at either end.
- To control where the box appears, use the toolbar buttons or the Layout Menu.

For more information, see “Arranging fields on a form” on page 282.

7 Choose File > Save Form to save your changes.

**Adding a URL to a text field**

Using a text field (trim), you can insert URLs into a form that will link your users to the Internet or intranet, or will open the file in the associated program.

If the user views the form in BMC Remedy User, the URL opens in a browser window. If the user views the form in a browser, the URL opens in a new window.
To add a URL to a text field

1. Open the form with which you want to work.
2. Create a text field using the procedure in “To create trim fields” on page 213. The new field appears in the upper-left corner of the Form window.
3. Double-click the new field to open the properties window. The Display properties tab is displayed.
4. In the Text field, select New Text, and replace it with the URL label that you want to appear on the form. For example, you can enter the name of your company.
5. Select the text that is to be converted to a URL, and click the Insert URL Link button. The Insert URL Link dialog box appears.
6. In the URL field, enter the URL destination to which you want to connect. From the URL field list menu, you can select any URL protocol identifier. For example, select one of the protocol identifiers (http://) and enter the URL, as in http://www.bmc.com.
7. Click the Test URL button to make sure that the URL destination is valid. Your default browser will open to the URL destination you entered, or the associated program will open the file with a known extension.
8. Drag the text URL field to a position in the form.
9. Choose File > Save Form to save your changes.

To change the URL color (text trim fields)

1. Select the Color/Font tab.
2. Clear the Default URL Color check box. The field under URL Color is enabled.
3. Select a color from the palette attached to the field. You can also select Other from the palette to create a custom color.
4. Choose File > Save Form to save your changes.
This chapter discusses other tips regarding using fields on your forms. The following topics are provided:

- Global fields (page 218)
- GUID fields (page 220)
- Fields in join forms (page 220)
- Managing fields (page 223)
Global fields

Global fields are a class of display-only fields that share values across multiple windows and forms (regular global fields) or across multiple records in the same window (window-scoped global fields).

You can make a field into a global field by setting the field ID to a value within a reserved range. The following field types can be global fields:

- Character
- Date/Time
- Date
- Time
- Diary
- Integer
- Real Number
- Decimal Number
- Selection List (radio button, drop-down list, or check box)
- Currency
- View (window-scoped global fields only)

You cannot create default values for global fields. The values in global fields begin as NULL until they are initialized through active links, or the user enters the information. The values in global fields are not affected if the user chooses Edit > Clear All or Edit > Set to Defaults.

For more information about display-only fields, see “Entry Mode” on page 425.

In forms viewed in a browser, global fields are implemented as encoded values in cookies, which have a 4k limitation. (With other cookie data, you can estimate that 3,500 bytes can likely be stored.) Global values are checked and fields are updated when a window receives focus.
Using regular global fields

Use a regular global field to share data across multiple forms. The field value stays in BMC Remedy User memory until the next login. On web clients, the value in the global field persists across forms for the user session, but is not shared across sessions.

Use regular global fields for information that is expected to be shared across multiple forms and windows in an application. For example, on several forms, you might include a global field for the user’s license information.

▸ To create a regular global field
1. Add one of the valid global field types to a form.
   For a list of valid field types, see “Global fields” on page 218.
2. Set the field ID to a value in the range from 1000000 to 1999999.
   See “Database properties” on page 420 for more information.
3. Copy the global field to multiple forms.
   Make sure that each copy of the global field uses the same field ID number.

Using window-scoped global fields

Use a window-scoped global field to share data between records in the same window. The initial field value is set by the user or by active links and stays in the field until the window is closed. This value must be reinitialized when the window opens again.

Using a window-scoped global field, for example, you could display information about weather conditions in a view field. This information would remain the same while the user performs searches on other fields in the same window. If you displayed the same information in a regular global field, you would have to refresh it every time a new record is displayed. With a window-scoped global field, you set the value once for that window and update the value using active links when you choose to do so. For example, you could update the information about weather conditions on an interval.
To create a window-scoped global field

1. Add one of the valid global field types to a form.
   For a list of valid field types, see “Global fields” on page 218.
2. Set the field ID to a value in the range from 3000000 to 3999999.
   See “Database properties” on page 420 for more information.

GUID fields

You can use a character field to automatically generate a globally unique identifier (GUID). GUIDs are useful in a multi-AR System server environment where you need unique IDs across AR System servers. Using GUIDs in this case guarantees unique, transactionally safe IDs for your requests. GUIDs do not replace Request IDs. Each request will still have a Request ID.

Note: Prior to AR System 7.0, you used the $PROCESS$ command to generate GUIDs. Using a GUID field provides better performance than the $PROCESS$ command.

To auto-populate a field with a GUID, define a character field using field ID 179. A GUID is available through all filter phases. By contrast, the value of Request ID field is not available until an entry has been successfully committed to the database. You can set the attributes of the GUID field, except for field type, length, and ID.

Fields in join forms

Field properties function somewhat differently in a join form than in a non-join form.

For each field in a join form, you can perform the following tasks:

- View information about the underlying form and field for a specific join field, including:
  - Database properties such as field ID and data type.
  - The field ID in the underlying form.
  - The underlying form from which the field came.
The name of the field in the underlying form. (This information is helpful for keeping track of field names when they are changed in a join form.)

Group permissions.

Modify a join field’s display, database, and color/font, help text, and change history properties.

Add data fields to the join form, or create display-only or trim fields. (Display-only fields can be inherited or created on the join form.)

Remove fields from the join form.

The following sections provide instructions for performing these tasks. For important background information, see Appendix D, “Field properties,” on page 397.

► To view or modify field properties in a join form

1 Open the join form with which you want to work.

2 Open the Field Properties window of the field.

3 Modify the field’s display properties by changing any of the values under the Display and Color/Font tabs.

   See “Display properties” on page 428 and “Color/Font properties” on page 417 for information about the values you can enter.

   Other tabs in the Field Properties window let you also view important read-only information. For example, the Database tab shows the underlying form that the field came from.

4 To modify the field’s help text or change history, select the Help Text and Change History tabs.

   See the Getting Started guide.

5 Choose File > Save Form to save your changes.

Adding fields to join forms

If you want to add a data field to a join form after it has already been created, the field must already exist in one of the underlying forms. This is because, from a database point of view, a join form is only a temporary composite database table.
You can add new trim fields (lines, boxes, or text), buttons, page fields, table fields, and display-only data fields directly to the join form in the same way that you add them to non-join forms.

The following procedure only provides instructions for adding data fields from the primary or secondary form to your join form.

To add a field from a primary or secondary form to a join form

1 Open the join form with which you want to work.
2 Choose Form > Create a New > Field From <xxx>, where xxx is the primary or secondary form that contains the appropriate field.
   The Add Field dialog box appears. It contains a list of the fields that you have not yet added from the underlying form to the join form.
3 From the Select field list, select the appropriate fields, and click OK.
   The new fields appear in the upper-left corner of the form window.
4 Drag the new fields to the appropriate position in the Join form.
5 To modify a field’s display properties:
   a Double-click the field to open the Field Properties window.
   b Edit the field’s display properties, help text, and change history as needed.
      See Appendix D, “Field properties,” for information about the values that you can enter.
6 Choose File > Save Form to save your changes.

Removing fields from join forms

You can remove inherited data fields from a join form at any time. Removing a data field from a join form removes it only from the join form view. To delete a data field displayed in a join form from the database, you must delete the field from the underlying non-join form.

If you remove a display-only field or trim from a join form, it is actually deleted.
To remove fields from a join form

1. Open the join form with which you want to work.
2. Select the appropriate fields.
3. Choose Edit > Delete.
4. Click Yes to delete the field, or click Yes to All to delete multiple fields.
5. Choose File > Save Form to save your changes.

Managing fields

The following sections explain how to modify, copy, delete, disable, and find fields on a form.

Modifying fields

Use the following procedure to modify an existing field. If you modify the display properties of a field, the modifications apply only to the form view in which you are working. If you modify any other field properties, the modifications apply to all form views.

To modify a field

1. Open the form with which you want to work.
2. Double-click the field to open the properties window.
   Alternatively, select the field from the Find Field drop-down list or select its ID from the ID field, and click the Field Properties button at the top of the window.
3. Modify the field properties, which are listed in Appendix D, “Field properties,” on page 397.
4. Select the field, drag it to a position in the form, and adjust its size.
   See “Arranging fields on a form” on page 282 for more information.
5. Choose File > Save Form to save your changes.
Copying fields

You can copy all field types, except page fields. When copying attachment fields, you can copy only one at a time.

When you create a copy of a field, most properties for the new field are the same as the original, with the following exceptions:

- If you copy a field to the same or different view of the original form, the only field properties that change are the ID and the Name.
- If you copy a field to a different form, its properties (including the field ID) remain the same. However, if you want to use this field in shared workflow (for example, in an active link), you must modify the workflow to include the form to which the field was copied.

Regardless of the form or form view to which you copy a field, menus that are attached to a field are copied with the field and need not be reattached.

**WARNING:** Use caution when using different field names that share the same field ID since shared workflow could use the field ID. You might want to use the same field name to help you remember what the field’s purpose is on each form if you are attaching shared workflow to multiple forms.

To copy a field

1. Open the form from which you want to copy a field.
2. Select the field.
   - You can also select and copy multiple fields.
3. Choose Edit > Copy.
4. Open the form to which you want to copy the field.
5. Choose Edit > Paste.

The new field appears on the form. If the form has multiple views, the following conditions apply:

- If your system preferences are set to copy fields to all form views (the default setting), the field will be added to all form views. For more information about form preferences, see the Getting Started guide.
If these preferences are cleared, the field will appear only in the selected form view. However, if you want the field to appear also in a different form view, select the view, and then choose Form > Current View > Fields in View to add the field to that view. For more information, see “Including and excluding fields from form views” on page 299.

If you copy the field to a different view on the same form (choose Form > Select a View to select the view), the Paste to View Options dialog box appears with the following prompt: “Do you wish to add existing fields to this view, or create new fields?” Click one of the following buttons to complete the action:

- **Add to View**— Adds the selected field to the current view. The added field has the same display properties as the original field, and no new fields are created. See “Views properties” on page 455 for another way to perform this operation.

- **Create New**— Creates and adds a copy of the field to the current view. The copied field has the same properties as the original field, except for the field ID and field Name.

6 Click Field Properties to open the properties window.

7 Modify the field properties, which are listed in Appendix D, “Field properties,” on page 397.

8 Select the field, drag it to a position in the form, and adjust its size.

   See “Arranging fields on a form” on page 282 for more information.

9 Choose File > Save Form to save your changes.

### Deleting fields

The delete operation is permanent and cannot be undone. When the field is deleted, it is removed from all form views. The field and its associated data for each request are also deleted from the database and space is freed. The operation might take several minutes to complete. Because your database might be unavailable while the deletion is occurring, users will not be able to access the server and might receive time-out messages. To minimize user inconvenience, perform deletions during off-peak hours.

If you want to make data fields nonoperational without removing them or their associated data from the database, see “Making data fields nonoperational” on page 227.
To delete a field

1. Open the form with which you want to work.
2. Select the field (or fields).
   You cannot delete the core fields. See “Core fields” on page 354 for more information.
3. Choose Edit > Delete.
   A delete field warning appears.
4. Click Yes to delete the field.
   The field is removed from the form.
5. Choose File > Save Form to save your changes.
   A final warning appears alerting you that the fields will be deleted from the database.
6. Click OK.

**Note:** When deleting a page field, the data fields on the page are not deleted. They are removed from the view, but are still present. For more information, see “Including and excluding fields from form views” on page 299.

To delete an attachment field

1. Open the Properties window for the attachment pool.
2. Select the Attach Fields tab.
3. Select the attachment field you want to delete.
4. Click the Delete button.
5. Choose File > Save Form to save your changes.

**Note:** You can select the attachment pool and press the Delete key to delete the attachment pool and all its fields. At the prompt in the dialog box, click the Yes button.
Making data fields nonoperational

If you do not want to use a data field but do not want to delete it, you can make it nonoperational. The following procedure can be useful for fields that contain necessary data that you do not want users to access or for fields that you plan to delete when you restructure the database.

To make a data field nonoperational

1. Open the form with which you want to work.
2. Double-click the field to open the properties window.
3. Select the Database tab.
4. From the Entry Mode list, select Optional.
   The Entry Mode list is active only for required and optional fields. You cannot change the entry mode of a display-only field, table field, or a page field. See “Entry Mode” on page 425 for more information.

5. If there are multiple views, use the Views tab to remove the field from all views other than the current one.
   See “Views properties” on page 455 for more information.

6. Close the properties window.

7. Remove the field from the current view by using the Fields in View dialog box, as described in “Including and excluding fields from form views” on page 299.

8. Choose File > Save Form to save your changes.

   The field will not be hidden to current users of the form until they reopen the form or log in again.

Note: You can also set permissions to make a field available to selected users only. See “Field permissions” on page 60.
Finding fields in a form

Use the Find Fields feature to locate a field in a specific form view. This feature is useful when you:

- Have a large number of fields in a form.
- Need help locating a field that does not have a label.
- Need help locating a field that is behind another field.
- Know the field ID, but do not know where it is on the form.

To find a field

1. Open the form with which you want to work.
   If the field you are trying to locate is in another view of the form, use the Form > Select a View menu to choose the form view. This menu selection is active only if more than one view is associated with the form.

2. From the Find Field menu list beneath the toolbar, select the field that you want to locate.
   Each field in the list is displayed in the forms field_database_name (field_display_label [field_ID]). If the field that you are trying to locate does not have a field label, AR System displays field_database_name [field_ID].
   The selected field's ID is listed in the ID field, and the field is outlined with a selection box.
   To find a field by its ID, select the ID from the ID field.

3. To open the properties window, click the Field Properties button or double-click the selected field.
This section describes how to create and modify character field menus. The following topics are provided:

- Understanding character field menus (page 230)
- Using the Menu window (page 230)
- Creating and modifying menus (page 232)
- Defining character menus (page 236)
- Defining file menus (page 238)
- Defining search menus (page 240)
- Defining SQL menus (page 246)
- Defining data dictionary menus (page 251)
- Building and using menu change history (page 253)
- Creating help text for menus (page 253)
Understanding character field menus

Menus provide organized lists of choices for quick selection and field entry. You can attach menus to any character field in any form on the server, and you can use the same menu for as many different fields as you want.

In BMC Remedy User, users can set a preference to determine how menus are displayed, such as hierarchical levels (“popup” style) or tree views (list box style). On web clients, menus are always displayed as lists.

Note: Do not confuse character field menus with the menus in the menu bar used to execute active links on the enumerated field type (drop-down list field). For more information about active link menus, see the Workflow Objects guide.

You should familiarize yourself with the information in Chapter 3, “Creating AR System forms,” and Chapter 4, “Types of fields,” before you perform the operations described in this section.

Using the Menu window

Use the Menu window (shown in the following figure) to create and modify menus.

Note: In this section, Menu window refers to a Create Menu or Modify Menu window.
Menu names must be unique on each AR System server. The maximum number of characters that can be displayed in BMC Remedy User for a menu item regardless of type is 80 characters.

The Menu window contains the following tabs:

- **Menu Definition**: Defines the basic properties of the menu, including refresh mode and menu type.
- **Change History**: Records the owner of a menu, 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 for the menu. In most cases, this help text is a description of the menu, what it does, and how it is used.
Creating and modifying menus

You can create a menu and attach it to any character data field. The following figure shows a menu with two levels.

Figure 7-2: Character field with a menu

You can define as many as 14 menu levels, but, for good usability, limit menus to as few as possible.

Creating menus

You can create menus in a Server Window or an Application Window. When you create a menu in an Application Window, the menu initially appears in the Server Window. After you add the menu to a character field on a form (or to a Change Field action) that belongs to an application, the menu appears in the Application Window for that application. For more information, see “Attributes properties” on page 406 and Chapter 7, “Defining menus.”

To create a menu

1. Open a Server Window or an Application Window.
2. Click the New Server Object toolbar button.
3. From the New Server Object list, select Menu, and then click OK to open the Create Menu window (Figure 7-1 on page 231).
4. In the Menu Name field, enter a name.

Names can be as many as 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name.
5 From the Menu Type region, select one of the following menu types.

<table>
<thead>
<tr>
<th>Menu Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character</td>
<td>Defines the menu as a series of label and value entries.</td>
</tr>
<tr>
<td>File</td>
<td>Specifies a file that contains a formatted character menu. The file can reside on the client or the server and can be updated at any time. After you update the file, the changes are automatically applied when the menu is refreshed.</td>
</tr>
<tr>
<td>Search</td>
<td>Uses a search that specifies a form from which to draw labels and values. The search runs when a user accesses the menu so that you can create a dynamic menu that updates based on current conditions. Because permissions are verified at the time of the search, users will only be able to see items for which they have permission.</td>
</tr>
<tr>
<td>SQL</td>
<td>Uses an SQL command to pull values from a database table. You specify which column of the retrieved data to use as the label and which to use as the value. You can create a dynamic menu that updates based on current conditions.</td>
</tr>
<tr>
<td>Data Dictionary</td>
<td>Uses names, labels, or IDs of forms or fields in the data dictionary as menu labels and values. You specify the name of the server and form (as appropriate), and the types of forms or fields to use.</td>
</tr>
</tbody>
</table>

The Menu Definition tab changes to display the fields required for the menu type that you have selected.

6 From the Refresh region, select the appropriate refresh mode.

<table>
<thead>
<tr>
<th>Refresh Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>On Connect</td>
<td>Retrieves the menu when the user opens the menu after selecting the form. To update the menu, the user must reopen the form.</td>
</tr>
<tr>
<td>On Open</td>
<td>Retrieves the menu each time the user opens it. Frequent menu retrieval can slow performance, so select this option only when it is critical that the menu be up to date.</td>
</tr>
<tr>
<td>On 15 Minute Interval</td>
<td>Retrieves the menu when the user first opens it, and when 15 minutes have passed since the last retrieval. This choice provides a balance between the need to be current and the expense of constant menu retrieval. For web clients, this option behaves the same as On Open.</td>
</tr>
</tbody>
</table>
Refresh modes affect only a menu’s contents, not its definition. The definitions of all menus are updated every time you reconnect to a form.

7 Specify values for the remaining fields for the menu type that you have selected, as described in the appropriate procedure:

- “Defining character menus” on page 236.
- “Defining file menus” on page 238.
- “Defining search menus” on page 240.
- “Defining SQL menus” on page 246.
- “Defining data dictionary menus” on page 251.

8 Choose File > Save Menu to save your changes.

Modifying menus

Use the following procedure to modify an existing menu.

To modify a menu

1 Select a server to administer.
2 Click the Menus category.
3 Double-click the appropriate menu to open it in a Modify Menu window.
4 To modify the menu name, enter a new name in the Menu Name field.
5 To modify the refresh mode, select the appropriate option from the Refresh region. For more information, see “Creating menus” on page 232.
6 Modify the remaining fields for the menu type that you have selected.
7 Choose File > Save Menu to save your changes.
Creating and modifying menus

When you save a menu under a different name, the new menu contains all of the properties of the original menu. The only difference is the name.

- **To copy a menu**
  1. Select a server to administer.
  2. Click the Menus category.
  3. Double-click the appropriate menu to open it in a Modify Menu window.
  4. Choose File > Save Menu As.
  5. In the Menu Name field, enter a new name.
  6. Click OK.

Deleting menus

The delete operation is permanent and cannot be undone. You cannot delete a menu that is open in BMC Remedy Administrator.

- **To delete a menu**
  1. Select a server to administer.
  2. Click the Menus category.
  3. Select the appropriate menu.
  4. Choose Edit > Delete - Menu(s).
    
    A confirmation message appears (if your preferences are set to display a message, as described in the administrator preferences discussion in the Getting Started guide).
    
    You can also right-click on the menu, and choose Delete from the context menu that appears.
  5. Click Yes to delete the menu.
    
    The menu is deleted from the database and will no longer appear in the list of menus.
Defining character menus

A character menu is a series of label and value entries. Plan your menu structure before creating a character menu. The following figure shows an example of a plan, including the parent and child menu entries.

Figure 7-3: Design for parent and child menus

<table>
<thead>
<tr>
<th>First-Level Menu (parent)</th>
<th>AR System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Second-Level Submenu (child)</td>
<td></td>
</tr>
<tr>
<td>Utilities</td>
<td></td>
</tr>
<tr>
<td>Clients</td>
<td></td>
</tr>
<tr>
<td>Third-Level Submenu for Clients (child)</td>
<td></td>
</tr>
<tr>
<td>Remedy User</td>
<td></td>
</tr>
<tr>
<td>Remedy Import</td>
<td></td>
</tr>
<tr>
<td>Remedy Alert</td>
<td></td>
</tr>
</tbody>
</table>

To define a character menu

1. From the Menu Type region of the Create Menu window, select Character. The fields required to define a Character menu appear, as shown in Figure 7-1 on page 231.

2. Click Add Menu to add a new menu entry. An item is created in the Menu Definition box (Figure 7-4 on page 237) and contains the default text, New Menu. The Value field becomes enabled when you click Add Menu.

3. Replace New Menu with the appropriate menu item label. The maximum length is 80 characters.

4. To add a submenu, select a menu or submenu, and click Add Menu Item.

5. Edit the menu item label. You can specify as many as 99 submenu items for each menu item.

For example, to create the AR System menu hierarchy shown in Figure 7-3:

a. Click Add Menu, and replace the text New Menu with AR System.
b Select AR System, click Add Menu Item, and replace New Menu with Utilities.

c Click Add Menu, and replace New Menu with Clients.

d To add the third-level items, select Clients, click Add Menu Item, and replace New Menu with BMC Remedy User. Click Add Menu, and replace New Menu with BMC Remedy Import, and so on.

Figure 7-4: Menu definition list with parent and child menus

6 If the value that you want to appear in the field when the user selects this menu item differs from the label text, enter the appropriate text in the Value field.

The maximum length is 255 bytes. The label text is used if you do not enter a value in this field. Because this text appears when a user selects the menu item, the Value field applies to selectable menu items only (that is, menu items that are not parents for other menu items).
7 To add menu entries, repeat steps 2 and 3.

You can also perform the following actions:

- To modify a menu item, select and edit its label in the Menu Definition list. You can also select an item and click F2 to edit the menu label. You can edit the menu item value in the Value field as well.

- To rename, delete, cut, copy, or paste a menu item, right-click the item and choose the appropriate option. Perform the action as needed.

- To expand or contract all menu hierarchies, click Open All or click Close All.

These buttons control your view of the menu definition, not how the menu appears in the User Tool. Menus are saved with child menu items hidden.

- To delete a menu entry, select the appropriate menu entry in the Menu Definition list, and then click Delete.

Child menu entries are deleted when you delete the parent menu entry.

- To change the order of the menu items in the Menu Definition list, right-click the item that you want to move, drag it up or down in the menu hierarchy, and then release the mouse button. A context menu appears, providing you several options where to move or copy the menu item.

In addition, you also can select a menu item, and then click the up or down arrows.

8 Choose File > Save Menu to save your changes.

9 Attach the menu to a character field as described in “Attributes properties” on page 406.

Defining file menus

A file menu lets you display a formatted character menu by referencing an external plain text file. After you update the file, the changes are automatically applied when the menu is refreshed.

To define a file menu

1 From the Menu Type region of the Create Menu window, select File.

The following figure shows how the Menu window might look after you complete the remaining steps in this procedure.
Figure 7-5: File menu window

2 From the File Location region, select the file location:

**Client**  
File is on the system where the BMC Remedy User client is running. This location is not supported for web clients.

**Server**  
File is on the AR System server.

3 In the File Name field, enter or browse for the path and name of the plain text file that contains the menu definition.

For information about how this file is formatted, see the following section, “Menu file format.”

4 Choose File > Save Menu to save your changes.

5 Attach the menu to a character field as described in “Attributes properties” on page 406.

**Menu file format**

A menu file is a plain text (.txt) file that contains a formatted menu structure and is used with file menus. You cannot use MS Word (.doc) or Rich Text Format (.rtf) files as menu files.
Each line in the file contains a definition of a menu entry in the following format:

`label\value`

To create submenu items under a specific item, use tabs (not spaces) to indicate a child menu. (Omit the value specification for any label that has sub-items.)

The following example shows a formatted menu file with three main items, two of which have sub-items:

```
Consulting Services\Consulting Services Training
  Administrator\Administrator Training
  User\User Training
Support
  Standard\Standard Support
  Priority\Priority Support
```

You can use a pound sign (\#) in the far-left column, and the line of text that follows the pound sign will be ignored.

For labels in BMC Remedy User, the maximum number of characters that can be displayed is 80 characters. For values, the maximum number is 255 bytes.

## Defining search menus

When you create a search menu, you specify a form from which to draw field names and values. This method enables you to create a dynamic menu that updates based on current conditions. Because permissions are verified at the time of the search, users will be able to see only the items for which they have permission.

1. **To define a search menu**

   From the Menu Type region of the Create Menu window, select Search.

   You can create a hierarchical search menu by defining up to five levels.

   You could build a menu that displays the names of everyone who has an open help desk request. To do so, select the server and form that contain the information. Then, select the Submitter field as both the Label and Value. Finally, build a qualification that searches for requests with a Status not yet Closed. Figure 7-6 shows how the Menu window might look if you created this type of menu.
Figure 7-6: Search menu with qualification for open requests

2 From the Server list, select the server that contains the form that you want to search.

3 From the Form Name list, select the form that you want to search.

4 Define the fields whose values will be used as menu items.
   a From the Label Fields list on the left, move the fields that you want to appear as menu items to the right side. Use one of the following methods:
     - Select a field name in the list on the left and click the >> button.
     - Double-click a field name in the list on the left.
   To remove a label from the menu:
     - Select a field name in the list on the left and click the << button.
     - Double-click a field name in the list on the right.
b If you add more than one field to the list on the right, you can change the order of the fields to determine their position in the hierarchy by clicking the Up and Down arrows.

The first field's values will be the first level, the second field's values will be the second level, and so on.

Only the first 80 characters of the field will be displayed. Menu items within the first four levels that do not have a value will appear blank. If the fifth level menu item does not have a value, it will not appear.

5 From the Value Field list, select the field whose value will be assigned to the lowest level menu item selected in step 4.

Only the first 255 bytes of the field will be displayed.

6 Select or clear the Sort On Label check box:

- Selected—Menu labels appear alphabetically.
- Cleared (the default)—Menu labels appear in the order in which they are retrieved (that is, the form default sort order).

**Note:** If there are multiple levels in the menu, Sort On Label is enabled, even if you do not select it.

7 In the Qualification field, build a qualification to define which requests from the form that you are searching will be included in the menu.

You can type the qualification or you can build it by using the qualification bar and list, as described in the Workflow Objects guide. You can use the menu button to choose from a list of fields in the form that you are searching and from a list of keywords. Make the qualification as specific as possible to avoid building a list with an unmanageable number of items.

In addition, BMC Remedy Administrator assists you in building qualifications by performing error checking against certain invalid qualifications, for example, referencing a field that does not exist on the form, misspelling the field name, and so on.

When referencing a value from the current screen, you must use the \$<field_ID>\$, not the field name, as in the following qualification example:

'Problem Summary' = $8$
Note: If you see a field on the current screen in the qualification, do not use the pattern $MENUS$ for any field to which the menu is attached. The server cannot resolve the field references, so the value will always be rejected.

8 (Optional) To supply a dynamic server or form, select the Advanced check box.

For more information, see “Defining advanced search menu functions” on page 243.

9 Attach the menu to any field in any form as described in “Attributes properties” on page 406.

You might use the menu created in this example to help perform a search on a caller’s open requests, as shown in Figure 7-7.

Figure 7-7: Menu items generated from search menu

Defining advanced search menu functions

You can define advanced Search Menu functions that provide alternate functionality beyond the hard-coded server and form defined in the search menu. Based on values that workflow or users enter into certain fields, you can dynamically change which server and form you are querying to create the search menu.

For example, the Dynamic Search Menu form (as shown in Figure 7-8) contains two fields (server and form) that can be hidden or display-only. The values you enter into these fields (for example, they could come from a Set Fields action) determine at runtime which source server and form are used as sources in the Search Menu.

When you click the menu on the Short Description field, a query is performed on the Help Desk Request form on the source server (cordova), and the returned values are used to create the search menu. Administrators might prefer to hide this functionality from users altogether and use the Window Loaded execution condition instead.
In this example, records were created in the Help Desk Request form on the server cordova. If you opened the Dynamic Search Menu form in BMC Remedy User and entered cordova as the server and Help Desk Request as the form, when you click the search menu on the Short Description field, the search menu queries for all open tickets ('Status' < 'Closed') on the Help Desk Request form. The values returned create the dynamic entries in the search menu (Joe User and Josephine User) that users can choose.

Figure 7-8: Using dynamic workflow with search menus

To design advanced search menu functions

1. Click the Advanced check box in the Menu window.

The window changes in the following ways:

- The Server and Form Name fields have boxes that allow you to enter field IDs or select keywords for the server and form names. These fields must exist on the form where the menu will be attached.

- The Sample Server Name and Sample Form Name fields appear (as shown in Figure 7-9). Default values listed are the current server and form. You use these "sample" fields to specify the Label and Value fields. The "sample" form must contain fields that exist on any dynamic source forms that are used at runtime. But the sample server and form are not actually used at runtime; the fields you pick are only used to build the menu.
2 In the Server and Form Name fields, enter a field ID (or choose a keyword from the drop-down list) that will “dynamically” specify which server and form are used.

The server and form that a user (or workflow) enters into those fields at runtime will be used as the source server and form from which to query for records when the search menu executes.

You must enter the field IDs for the server and form in their variable format, for example, $336870916$, as shown in Figure 7-9. When you attach the search menu to a form, make sure that the form you are querying contains the appropriate field IDs.

3 From the Sample Server Name and Sample Form Name lists, select a reference server and form.

By selecting these, you can map the labels and values in step 4. The labels and values are populated with fields from the “sample” form. Select field values that correspond to fields available on any dynamically selected form at runtime.
The “sample” server and form are used as a temporary reference to create and save the Search menu. You can even delete the sample form after saving the action.

4 Map the Label Field and the Value that you are setting with this action.

   In this example, the label and value for the first level of the search menu are built from querying the dynamic server and form.

5 Choose File > Save Menu to save your changes.

Defining SQL menus

An SQL menu pulls values from a database table by using an SQL command. The database values make up the SQL menu item labels and define the value of each SQL menu item. The ability to submit SQL commands to create a menu enables you to:

- Select data from databases other than AR System databases.
- Issue complex queries to the database. This is useful for customers who want to use database features specific to a particular database platform.

For the most effective use of SQL commands, you must have a general understanding of relational databases and a specific understanding of your relational database. If an SQL command is specific to a database instead of generic, you might have difficulty moving the definition to another environment.

Before you create an SQL menu, determine what information you want to search from the database and what information will be returned from your SQL command.

To define a menu that uses SQL commands

1 From the Menu Type region of the Create Menu window, select SQL.

   The following figure shows how the Menu window might look after you complete the remaining steps in this procedure.
2 From the Server list, select the database server to which the SQL command will be issued.

3 In the Label Index List field, enter the number that is the numerical index of the database column that contains the information you want to display as menu item labels.

   You can create a hierarchical menu by entering up to five index numbers separated by commas. The first number becomes the first level, the second number the second level, and so on.

   Only the first 80 characters will be used. Menu items within the first four levels that do not have a value will appear blank. If the fifth level menu item does not have a value, it will not appear.

4 In the Value Index field, enter the number that is the numerical index of the database column that contains the information you want to load in the field when the user chooses the menu item.

   Only the first 255 bytes will be used.
In the SQL Command field, enter the SQL command that you want to issue to the database to create the menu.

You can enter only one SQL command for each menu. If you need to run more than one SQL command, you must use stored procedures or functions. Do not end the SQL command with `run` or `go`.

**Note:** The syntax that you use must be recognized by the underlying SQL database. The AR System does not verify the validity of your SQL command.

Attach the menu to a field in any form as described in “Attributes properties” on page 406.

Suppose the SQL command from Figure 7-10 on page 247 returned the results shown in the following figure.

**Figure 7-11: Sample results returned from SQL command**

```
SELECT BUG_ID, FIRST_NAME, TECHNCN FROM CUSTMR_INFO
```

<table>
<thead>
<tr>
<th>BUG_ID</th>
<th>FIRST_NAME</th>
<th>TECHNCN</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000</td>
<td>Mary</td>
<td>Zan</td>
</tr>
<tr>
<td>5001</td>
<td>John</td>
<td>Fran</td>
</tr>
<tr>
<td>5002</td>
<td>Mark</td>
<td>Tran</td>
</tr>
</tbody>
</table>

Each column has a corresponding numerical index: 1 corresponds to the `BUG_ID` column, 2 to the `FIRST_NAME` column, and 3 to the `TECHNCN` column.

Entering a 2 into the Label Index List field creates a menu with the contents of the `FIRST_NAME` column as the menu items, as shown in the following figure.

**Figure 7-12: Menu created with an SQL command**

Menu created from contents of column 2.
Entering a 1 in the Value Index field inserts the value from the **BUG_ID** column into the field. If a user chooses John from the menu in BMC Remedy User, the field would be filled with the **BUG_ID** value associated with John, as shown in the following figure.

**Figure 7-13: Sample field value if value index set to 1**

[Diagram of SQL Menu with Value inserted and Menu selection]

Entering a 3 into the Value Index field inserts the value from the **TECHNCN** column into the field, as shown in the following figure.

**Figure 7-14: Sample field value if value index set to 3**

[Diagram of SQL Menu with Value inserted and Menu selection]

Entering 2, 3 into the Label Index List field creates a hierarchical menu as shown.

**Figure 7-15: Hierarchical menu**

[Diagram of SQL Menu with hierarchical structure]
Performing the SQL operation

Remember the following tips for using SQL commands:

- You can issue only one SQL command per menu. You cannot enter two commands separated by a semicolon and have both commands run. If you need to run a set of commands, you must create a stored procedure and run that.

- Turn on SQL logging in the database to debug the SQL syntax if it returns unexpected values or results. An additional debugging strategy is to start an SQL interpreter (for example, `isql` for Sybase, SQL*Plus for Oracle, Command Center for DB2, or Query Analyzer or Microsoft ISQL/w for SQL Server), and enter the same SQL command directly into the database to see what results occur.

- Because AR System does not perform error checking on the SQL statement, run the SQL statement directly against the database (as a test) before you enter it into the SQL Command field. You can then copy and paste the tested SQL command into the SQL Command field.

- If the SQL operation fails, an AR System error message and the underlying database error message appear.

- You can greatly affect database performance by how an SQL search is written. If the row has many columns, a `SELECT *` SQL command can have a greater performance impact than if you select specific columns. For more information, see your relational database documentation.
Database security issues

To use SQL commands, familiarize yourself with the specific features of your underlying database.

All SQL commands are issued by the following users:

- For Oracle and Sybase databases, run as ARAdmin.
- For Informix databases, run as the user that controls the arserverd process.
- For DB2 databases, run as the user who installs AR System.

If you are running AR System as one of these users without permission to access the database, you cannot issue the SQL command.

To access non-AR System databases, use the database name as part of the SQL command syntax, for example, for a Sybase/MS SQL database:

```
<DatabaseName.owner.table>
acme.ARAdmin.CUSTMR_INFO
```

Defining data dictionary menus

A data dictionary menu pulls values for field and form objects from the data dictionary. For field objects, you specify the server and form, and the type of fields to use, such as data fields or control fields. For form objects, you specify the server and the type of forms to use, such as regular forms or join forms, and whether to include hidden forms.

To define a data dictionary menu

1. From the Menu Type region of the Create Menu window, select Data Dictionary.

The following figure shows how the Menu window looks when you are defining a data dictionary menu.
2 In the Server Name field, enter the name of the server from which the objects will be selected.

You must have administrator permissions to the server specified.

You can also specify a field ID, such as $<field_ID>$, so that the value in that field will be used as the server name at run time.

3 Select a format for menu item names in the Label Format field.

The following formats are available:

- **Name**—The name of the object in the database.
- **Label**—The label displayed for the object in the client.
- **ID**—The ID of the object.

4 Select a format for displaying values in the Value Format field.

You can specify the value to be the name, label, or ID in various formats, such as plain (Name), or surrounded with single quotation marks ('Name') or dollar signs ($Name$). You can specify pairs of values separated by semicolons, for example, $ID;Label$ or $Name;Label$. 

---

252 Chapter 7—Defining menus
5 Select the Object Type.

If you select Field, perform the following tasks:

a In Field Details, select a form name in the Form Name field.

b In the Field Type field, select one or more types of fields.

The menu will be constructed using fields that match the types of fields selected.

If you select Form, perform the following tasks:

a In Form Details, select a form type in the Form Type field.

The menu will be constructed using forms that match the type of form selected.

b To display hidden forms matching the form type, select the Show Hidden Forms check box.

6 Choose File > Save Menu to save your changes.

7 Attach the menu to any field in any form as described in “Attributes properties” on page 406.

Building and using menu change history

AR System automatically records the owner of a menu, the user who last modified the menu, and the date of the modification. To display or add to this information, click the Change History tab in the Menu window.

For more information about building and using change history, see the Getting Started guide.

Creating help text for menus

You create help text for menus the way you create help text for other objects. For more information, see the Getting Started guide.
Creating and managing form views

This section covers topics related to managing and defining form views. The following topics are provided:

- Creating and managing form views (page 256)
- Setting view properties (page 263)
- Managing fields in a form view (page 279)
- Setting the tab order for the fields in a form (page 297)
- Including and excluding fields from form views (page 299)
- Preference settings for view selection (page 301)

For additional information specific to web views, see the Installing and Administering BMC Remedy Mid Tier guide.
Creating and managing form views

A form view is what users see when they open a form in BMC Remedy User or on the web. Form views are versions of an original form.

As an administrator, you can provide users with multiple views of the same form, each defined for a unique purpose, locale, or view type. You can set default views to determine which view of a form the user will open in BMC Remedy User or in a browser. An administrator might define views for the following environments:

- Different locale—Views for specific languages or regions.
- Different user roles—Customized views for requesters, frontline staff, backline staff, or managers.
- Hardware dependencies—Views that allow for the size of a monitor, such as laptop or desktop. This is useful for legacy systems without scalable fonts.
- Purposes other than a form—Views that are used only when invoked as a dialog box using an active link.

Form views are associated with a particular form and can exist only after the initial creation of a regular form. Every form has one default view, and you can create additional views. For information about how to create regular forms, see “Creating and managing forms” on page 99.

Creating form views

Use the following procedure to create a new form view. Because one view exists with every new form, view creation is a copy operation. When you copy a form view, you get a “clone” of that view. Only the label name and ID of the view are changed in the copy. After the copy is made, you can change view-specific properties without affecting other views.

For information about views that are displayed in a browser, see the Installing and Administering BMC Remedy Mid Tier guide.

To create an alternate view for a group of users that is invoked using active links, append an extension to the view label. This view label extension is then specified in user preferences to identify the extension to be appended to the view label. The primary view must be specified in the active link and use of the alternate view is controlled by user preferences. For more information about user preferences, see the Configuring guide.
To create a form view

1. Open a form in BMC Remedy Administrator.
2. Choose Form > Manage Views to open the Manage Views dialog box.

![Manage Views dialog box](image)

3. From the list of views, select the view that you want to use as a base for your new view, and click Copy. (Your new view will inherit the view properties and fields of the original.)

The Manage Views - Copy View To dialog box appears.

![Manage Views - Copy View To dialog box](image)

**Note:** To create a blank form view with no core fields, click the New button (instead of the Copy button) in the Manage Views dialog box.
Chapter 8—Creating and managing form views

4. From the View Type list, select Standard (Recommended) to create a view that can be displayed in BMC Remedy User and in a browser.

The other option, Web - Alternate (Fixed), is included for special purposes or for backward compatibility with previous releases. A view that uses a fixed layout positioning for the Web. This type of view is not needed in version 6.3 and later versions. It would be used only for compatibility with older versions of the mid-tier or for a situation where you need to create a view that is different on the Web than what you want on Windows. If there is a view present of type Web - Alternate (Fixed), that view will be selected first when running on the Web, unless the Standard view is the default preference set in BMC Remedy Mid Tier Configuration Tool. See the Configuring guide for more information.

5. Click OK.

The new view is created and listed under the view that was selected as the template in the Manage View dialog box.

Figure 8-3: Manage Views dialog box
6 In the Manage Views dialog box, modify the following fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ID</strong></td>
<td>A unique identifier for a form view. This value is automatically generated by AR System and cannot be modified.</td>
</tr>
<tr>
<td><strong>Label</strong></td>
<td>The label name for the view. The Label, Locale, and View Type fields together define a unique identifier for the view. The client uses the label name to identify the user’s preferred view. For more information about user preferences, see the Configuring guide. When a view is created, a label name is assigned automatically to the new view. You can change the label name. There is no enforced convention for specifying label names, but it is helpful to make the name descriptive and indicative of the view's function consistently across forms. For example, you might create a &quot;Manager&quot; view for all forms in an application. Label names can be as many as 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name. <strong>Note:</strong> If you want a specific view (other than the default set in BMC Remedy Administrator) to be displayed in a browser by using a URL, make sure the label name is unique.</td>
</tr>
<tr>
<td><strong>Locale</strong></td>
<td>Defines the view’s locale. When a view is created, the Locale field is empty. If left blank, the locale of the user’s operating system is assumed for the view. To specify a locale for a view, click on the cell for the Locale field and select the preferred locale from the list. The Locale field associates a language and country dialect with a view following the format: <code>&lt;language&gt;_&lt;country&gt;</code>. Selecting only the language will include all variations of that language. Conversely, selecting <code>fr_CA</code> (French_Canadian) defines a view for Canadian French speakers, while selecting <code>fr</code> includes all French speaking countries. For more information about localization, Appendix H, “Localizing AR System applications.” Different views for different locales should have the same Label, and the correct locale for the user will be automatically opened.</td>
</tr>
</tbody>
</table>
The Label, Locale, and View Type fields do not need to contain unique values, but the combination of the three fields must be unique to each view.

7 To set a default view, select a view from the Choose Default View list.

The default view is the view that is displayed to a user when a request for a form is made and no user preferences have been set. An exception to this rule is seen in a web client; if no user preference has been set but the configuration preference is set to a web view, the web view is chosen.

8 To open your view, select it in the Manage Views dialog box, and click Display.

9 Close the Manage Views dialog box.

10 Choose File > Save Form to save the new form view created.
Changing which view is displayed

Use the following procedure to change which view is currently displayed.

> **To change which view is displayed**

1. Open a form in BMC Remedy Administrator.
   The default view of the selected form appears in the Modify Form window.

2. Choose Form > Select a View.
   The Select View dialog box opens with options for selecting and displaying views. To create or delete a view or to set the default view, use the Manage Views dialog box accessed by choosing Form > Manage Views from the menu.

3. Double-click the view you want to open.

Modifying form views

Use the following procedure to modify existing form views.

> **To modify a form view**

1. Open a form in BMC Remedy Administrator.

2. Choose Form > Manage Views.
   You can also use choose Form > Select a View to open or change which view is currently displayed.

3. Select the view you want to modify from the list of views in the Manage Views dialog box.

4. Click Display.
   The Modify Form window in BMC Remedy Administrator opens.

The following are examples of modifications that you can make to a form view:

- Modifying view properties. See “Setting view properties” on page 263.
- Adding or modifying fields. See “Types of fields” on page 143 and “Field properties” on page 397.
- Specifying which fields to include or exclude. See “Including and excluding fields from form views” on page 299.
Arranging fields. See “Managing fields in a form view” on page 279.

Modifying field properties of fields. See “Field properties” on page 397.

Resizing a Web - Alternate (Fixed) view for use as a dialog window. See the Installing and Administering BMC Remedy Mid Tier guide.

Localizing the view. For more information, see Appendix H, “Localizing AR System applications.”

5 Close the Manage Views dialog box.

6 Choose File > Save Form to save changes to the view.

Renaming form views

Renaming a form view enables an administrator to change the name of an existing form view without having to create a duplicate under a different name. The renamed view retains all of the settings of the original form view.

To rename a form view

1 Open a form in BMC Remedy Administrator.

2 Choose Form > Manage Views.

3 Select the view, and click the Properties button to open the View Properties dialog box.

4 In the Basics tab, enter the new name in the Name field.

5 To change the view’s label, enter the new name in the Label field.

6 Click OK to close the View Properties dialog box.

7 Click Close to close the Manage Views dialog box.

8 Choose File > Save Form to save changes to the view.

262 Chapter 8—Creating and managing form views
Deleting form views

You cannot delete all views for a form. Each form must have at least one view.

If the view designated as the default view is deleted, the first view in the list in the Manage Views dialog box is then designated as the default view.

To delete a form view

1. Open a form in BMC Remedy Administrator.
2. Choose Form > Manage Views from the menu.
3. Select the view you want to delete.
4. Click the Delete button.
   
   A warning message opens to confirm that the view you selected is the one you want to delete. The name of the view in the message dialog box is taken from the entry made in the Name field in the View Properties dialog box for the selected view, rather than from the label in the Manage Views dialog box.
5. Click Close to close the Manage Views dialog box.
6. Choose File > Save Form to save changes to the view.

Setting view properties

In the View Properties dialog box, you can specify pane layout, create aliases that describe how requests are named, predefine searches that you think users will want, disable user access to selected features, and define the color of rows of a results list pane.

The following option tabs are available in the View Properties dialog box for Standard views:

<table>
<thead>
<tr>
<th>Option Tab</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic</td>
<td>Defines the variable names that uniquely identify a form view.</td>
</tr>
<tr>
<td>Appearance</td>
<td>Defines how a view appears in BMC Remedy User and on forms viewed in a browser.</td>
</tr>
<tr>
<td>Aliases and Labels</td>
<td>Defines how form aliases and entry point labels are displayed in AR System.</td>
</tr>
<tr>
<td>Defined Searches</td>
<td>Defines common searches for users.</td>
</tr>
</tbody>
</table>
To define view properties in the Basic tab

1. Open a form in BMC Remedy Administrator.
2. Choose Form > Manage Views.
3. Select the view, and click the Properties button.

The View Properties dialog box opens with the Basic tab selected.

**Menu Access**

- Allows you to enable or disable menu items and toolbar items in BMC Remedy User and toolbar buttons on forms viewed in a browser.

**Results List Color**

- Defines the color of selection field items that appear in a results list pane.

**Advanced Results List**

- Defines special results list characteristics for BMC Remedy User.

**Dynamic Web Views**

- Used in version 6.0 of AR System, and is available only to clients using version 6.0 of the mid tier. Please see that version's documentation for more information.
NOTE: You can also open the View Properties dialog box by opening a view and choosing Form > Current View > Properties.

4 Enter the appropriate information in the following fields:

**Master View for Server Processing**
- If selected, this view will be used for processing to retrieve field labels and selection value aliases (not database values). For more information, see the description for "Master" on page 260.

**Name**
- Defines a unique identifier for a form view. This value is initially generated by AR System, but you can modify it.

**Label**
- Defines the label name for the view. When a view is created, a label name is automatically assigned for the new view. The assigned label can be changed.
- There is no enforced convention for specifying label names, but it is helpful to make the name descriptive, indicative of the view’s function, and consistent across forms. Label names can be as many as 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name.
- **NOTE:** If you want a specific view (other than the default set in BMC Remedy Administrator) to be displayed in a browser by using a URL, make sure the label name is unique.

**Locale**
- Defines the view’s locale. When a view is created, the Locale field is empty. If left blank, the locale of the user’s operating system is assumed for the view.
- To specify a locale for a view, click on the cell for the Locale field and select the preferred locale from the list.

**Web Alias**
- This field is not supported in version 6.3 and later versions of the mid tier. The field was used in earlier versions for administrators who wanted to create and deploy web views of a form. As of version 6.3, forms are no longer “deployed”; all forms can be viewed on the web without special aliases.
- If you created a form in a mid tier prior to version 6.3, and a web alias is associated with the form, you can also edit the web alias by choosing Form > Compatibility > Set Web Alias, and editing the Set Web Alias dialog box that appears.

**View Type**
- Defines the type of view associated with the selected view.
5 Click OK.
6 Close the Manage Views dialog box.
7 Choose File > Save Form.

To define view properties in the Appearance tab
1 Open a form in BMC Remedy Administrator.
2 Choose Form > Manage Views.
3 Select a view and click the Properties button.
4 Click the Appearance tab.

Figure 8-5: View Properties dialog box—Appearance tab

Note: When creating graphics in a form, use images that employ a “web-safe” palette, which is a palette of 256 colors.
5 Enter the appropriate information in the following fields:

### Background
Defines the background color and the use of images in the Details pane. Choose from the following options:

- **Default Color**—Defines the background color, usually gray.
- **Color**—Enables you to choose a background color from the palette.
- **Image**—Enables you to add an image (.jpeg, .jpg, .bmp, or .dib format) to the Details pane. The Horizontal and Vertical menu lists give you options for positioning the graphic in the view. For example, select Horizontal and Vertical Fill to cover the entire form, or select Horizontal Right and Vertical Bottom to move it to the bottom right side of the view. Click Browse to locate the image you want to add. It is then saved with the form (not referenced in its original location). Click Save to Disk to save the image to another area on the network.

**Note:** Alternatively, you can use the App-specific Image Path for Mid-Tier View field for forms viewed through applications on the web. In this field, insert the relative path from an application's Support Files top directory to the background image. If the file is not found in the application resource files, then the path in this field is ignored. For more information about managing resource files, see the Installing and Administering BMC Remedy Mid Tier guide.

The size of an image file used for a background should be considered. Although view definitions are cached when initially loaded, images associated with a view must first be downloaded. Changes to a form will also force the image to again be downloaded. A large image file can ultimately have a negative affect on response time.

### Pane Banner Visibility
Enables you to define visibility of the banners in the view.

- **Hidden**—Does not appear when BMC Remedy User is opened.
- **Visible**—Appears when BMC Remedy User is opened.

For the Web Toolbar, you can choose:

- **Hidden**—Does not appear in a form opened in a browser.
- **Visible**—Appears in a form opened in a browser.

### Title Bar Icon
Enables you to add an icon image (.jpeg, .jpg, .bmp, or .dib format) to the title bar of the form view.

Select the Use Custom Icon check box. Click Browse to locate the image you want to add. Images larger than 16x16 pixels will be cropped. Click Save to save the image to another area on the network.
### BMC Remedy Action Request System 7.0

<table>
<thead>
<tr>
<th>Layout</th>
<th>Defines the position of the panes and banners in the view. Choose from the following options:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>R/D (Results/Details)</strong></td>
<td>Enables you to select from four predefined choices for pane layout.</td>
</tr>
<tr>
<td><strong>Prompt Bar</strong></td>
<td>Defines the location of the prompt bar, at the top or bottom of the window.</td>
</tr>
<tr>
<td><strong>User Can Change Layout</strong></td>
<td>Enables users to define the position of panes and banners in their view. If this check box is cleared, the positions are “locked,” and the corresponding menu items in BMC Remedy User are disabled.</td>
</tr>
</tbody>
</table>

| Edit Web Header Content | Enables you to add HTML code that you want to include in the header of a Web Alternate (Fixed) view. For more information about editing the web header and footer content, see the Installing and Administering BMC Remedy Mid Tier guide. |

| Edit Web Footer Content | Enables you to add HTML code that you want to include in the footer of a Web Alternate (Fixed) view. For more information about editing the web header and footer content, see the Installing and Administering BMC Remedy Mid Tier guide. |

6. Click OK.  
7. Close the Manage Views dialog box.  
8. Choose File > Save Form.

#### To define view properties in the Aliases and Labels tab

For each view, use the Aliases and Labels tab to define the alias names to be used for the view, including the labels you want to appear for entry points. The alias might be a local language version of the view name, or it might be a more user friendly version of the name. When aliases and labels are defined, the user will work with the view as the alias and not see the base view name.

For views displayed in a browser, only the Singular option is available, and the name is displayed in the title bar of a browser.

1. Open a form in BMC Remedy Administrator.  
2. Choose Form > Manage Views.  
3. Select a view and click the Properties button.  
4. Click the Aliases and Labels tab.

---

268 Chapter 8—Creating and managing form views
5 Enter the appropriate information in the following fields:

**Singular**
Defines the alias that appears in the Details banner and the title bar of the window.

**Plural**
Defines the aliases that appear in the Results and Details banners in Search mode and the Open dialog box in BMC Remedy User and the title bar of the window.

**Short Singular**
Defines the shortened version of the singular alias that appears when the title does not fit in the Details banner. The Short Singular alias is not used in the title bar of the window. The web does not use the short alias and precedence is as follows:
- If long and short aliases are defined, the title bar displays the long alias.
- If only the short alias is defined or if no aliases are defined, the title bar displays the form name.
Click OK.

Close the Manage Views dialog box.

Choose File > Save Form.

To define view properties in the Defined Searches tab

For each view, use the Defined Searches tab to define the searches that appear in the Search menu list on the Details banner and in the Actions > Defined Searches menu in BMC Remedy User.

Use this feature to create predefined searches that you think will be helpful to your users; for example, finding all open requests or entries. You can define as many as 18 searches.

1 Open a form in BMC Remedy Administrator.

2 Choose Form > Manage Views.
3 Select a view and click the Properties button.
4 Click the Defined Searches tab.

**Figure 8-7: View Properties dialog box—Defined Searches tab**
5 Enter the appropriate information in the following fields:

**Name**
If you do not enter a Description, defines the name of the predefined search that users will see in the Search menu in BMC Remedy User. This name will appear in the Results pane banner after users perform a search.

For example, if you enter “Open Requests” in the Name field and “Search for open requests” in the Description field, users will see “Search for open requests” when they view the list of Defined Searches.

**Note:** You can create parameterized searches for users who use browsers, but these searches will return an error in Remedy User. To avoid this error, create parameterized searches only for the Web view of the form and not the Remedy User view. For more information about parameterized searches, see the Installing and Administering BMC Remedy Mid Tier guide.

**Description**
Defines the name of the predefined search that users will see in the Search menu in BMC Remedy User.
If you do not enter a Description, the text in the Name field will be used.

**Qualification**
Creates search criteria that you want users to access from the Search menu. Use it for any search that your users frequently perform; for example, 'Status' < "Closed by QA", to find all requests that are not closed.

You can right-click in the Qualification field to access a menu of fields and keywords for help in building the qualification. For information about building qualifications, see the Workflow Objects guide.

6 Click OK.

7 Close the Manage Views dialog box.

8 Choose File > Save Form.
To define view properties in the Menu Access tab

For each view, use the Menu Access tab to control which menu and toolbar items users can access in BMC Remedy User and which buttons appear in the toolbar of a form viewed in a browser. For example, if you do not want users to have Modify All capability, you can disable it.

1. Open a form in BMC Remedy Administrator.
2. Choose Form > Manage Views.
3. Select a view and click the Properties button.
4. Click the Menu Access tab.

Figure 8-8: View Properties dialog box—Menu Access tab
5  Select or clear the following Accessible Menu Items options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>BMC Remedy User menu item</th>
<th>browser toolbar button</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify All</td>
<td>If cleared, Modify All is disabled.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Options</td>
<td>If cleared, Options is disabled.</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Auto-Refresh</td>
<td>If cleared, all user actions to invoke Auto-Refresh (polling refresh) of the results list are disabled.</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Reporting</td>
<td>If cleared, the ability to generate a report on data in this form is disabled. This includes disabling the report option of table and results list fields on a form.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Home</td>
<td>If cleared, the menu, button, and keyboard actions for opening the Home page are disabled. If selected and a user clicks the button in a browser, the Home page opens in a new window.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Help</td>
<td>If cleared, help is disabled. If selected and a user clicks the button in a browser, the help opens in a new window.</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Delete</td>
<td>If cleared, Delete is disabled.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Save</td>
<td>If cleared, the menu, button, and keyboard actions for saving a request are disabled. This is useful when users do not need to save a form, such as in control panels. This option can also force the user to save by using a button you create on your form.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Print</td>
<td>If cleared, the ability to print the entry is disabled. This includes disabling the print option of table and results list fields on a form.</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Clear All</td>
<td>If cleared, Clear and Clear All are disabled.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Set to Default</td>
<td>If cleared, Set to Defaults is disabled.</td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>
Setting view properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>BMC Remedy User menu item</th>
<th>browser toolbar button</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Search</td>
<td>If cleared, the Advanced Search capability is disabled. If workflow shows or hides an Advanced Query bar form action field (ID 1005), the toolbar button should toggle 1005 to the opposite state. (In such a case, you should remove the Advanced Search button from the toolbar for that view.)</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Search</td>
<td>If cleared, the menu and button actions for searching for requests are disabled. This is useful when users do not need to search a form, such as in control panels. This option can also force the user to search by using a button you create on your form.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>New Search</td>
<td>If cleared, New Search is disabled.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>New Request</td>
<td>If cleared, New Request is disabled.</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Status History</td>
<td>If cleared, Status History is disabled. If selected, when the user clicks the button in a form viewed in a browser, the status history opens in a new window.</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

6 Close the Manage Views dialog box.

7 Choose File > Save Form.
To define view properties in the Results List Color tab

For standard views, use the Results List Color tab to control the color of requests that appear in the results list pane after a search in BMC Remedy User. The colors you choose are based on Selection Field items. For example, you can choose to color all New requests red and all Assigned requests green.

For information about defining fields to be included in the results list pane, see “Results list fields—Defining search results” on page 135. For information about results lists on web clients, see “Table fields” on page 153.

1 Open a form in BMC Remedy Administrator.
2 Choose Form > Manage Views.
3 Select a view and click the Properties button.
4 Click the Results List Color tab.

Figure 8-9: View Properties dialog box—Results List Color tab

5 Select a field from the Selection Field list.
6 From the Value list, select one of the values (including NULL or none) for which you want to define a color for rows with that value.
From the Color section, specify the color for rows with the selected value. You can select Default Color, or you can select Custom Color and choose a color from the color palette.

Repeat step 5 through step 7 for each selection field for which you want to define a color.

You will see only one value at a time, but if you return to a value, you will see that the color you specified has been entered. If you do not specify a color for a value, rows with that value will use the default color.

Click OK.

Close the Manage Views dialog box.

Choose File > Save Form.

To define view properties in the Advanced Results List tab

For each view, use the Advanced Results List tab to specify advanced options for managing the results list pane in BMC Remedy User.

Open a form in BMC Remedy Administrator.

Choose Form > Manage Views.

Select a view and click the Properties button.

Click the Advanced Results List tab.
5 Enter the appropriate information in the following fields.

**Initial Row Selection**
- Defines the item you want selected when the results list is loaded. The options are:
  - **Select First, Fire Workflow**—The first item is shown selected and any workflow defined will be activated. This is the default setting.
  - **No Selection**—No item is selected.

**Size of Chunk**
- Defines how many rows of data to return at one time (or “chunk”) if a search yields a long list of requests in the BMC Remedy User results list pane. Buttons are provided for displaying previous and next chunks.
  - A value greater than zero in this field enables the chunking feature. The default value is 0.

6 Click OK.

7 Close the Manage Views dialog box.

8 Choose File > Save Form.
Managing fields in a form view

You can use field spacing to arrange the position of fields in a form. This feature enables you to specify the amount and type of spacing for fields in a form. Using field spacing can help improve the appearance and usability of a form by providing greater consistency in the placement of fields, and can save development time because fields do not need to be spaced manually.

You can configure horizontal or vertical field spacing in a form through the Field Spacing selection from the Layout menu, or through buttons in the toolbar. You can select even spacing horizontally or vertically, or specify an amount of spacing (in pixels) between fields, from an edit field on the toolbar.

Figure 8-11: Spacing toolbar and edit box

The following sections describe how field spacing is calculated for each option.

Spacing evenly vertically

For even vertical spacing between fields, the topmost and bottommost fields in each group are identified. The distance between the top edge of the topmost field and the bottom edge of the bottommost field is calculated; this is the total height covered.

The total height of all the fields in the group is then calculated. This total is subtracted from the total height covered to get the amount of clear space. The length of clear space is divided by the number of fields, minus one, to determine the spacing between each field.

The fields are then placed in order from top to bottom according to the calculated spacing.
Spacing evenly horizontally

For even horizontal spacing between fields, the leftmost and rightmost fields in each group are identified. The distance between the left edge of the leftmost field and the right edge of the rightmost field is calculated; this is the total width covered.

The total width of all the fields in the group is then calculated; this is the total width of all fields. This total is subtracted from the total width covered to get the amount of clear space. The length of clear space is divided by the number of fields, minus one, to determine the spacing between each field.

The fields are then placed in order from left to right according to the spacing calculated.

Note: When the spacing is adjusted, the position of the field at the top or the left of the group remains stationary. Because fractional space measurements are not allowed, the bottom or right field in the group will shift to allow even spacing.

You can specify an amount of horizontal spacing (in pixels) between two or more fields by entering a number in the toolbar edit field. Only a positive, whole number of 1 or greater is valid. If the edit field is left blank or has a value of zero, the spacing action cannot be performed.
Note: For both horizontal and vertical spacing, edges pertain to the bounding box and not to the actual elements that are drawn for the field.

Figure 8-14: Spacing by length horizontally

Spacing by length vertically

You can specify an amount of vertical spacing (in pixels) between two or more fields by entering a number in the toolbar edit field. Only a positive, whole number of 1 or greater is valid. If the edit field is left blank or has a value of zero, the spacing action cannot be performed.

Figure 8-15: Spacing by length vertically

To space fields evenly

1. In BMC Remedy Administrator, open the form for which you want to specify field spacing.
2. Select the fields you want to space.
   You must select at least three fields.
3. Choose Layout > Field Spacing > Evenly Vertically or Evenly Horizontally.
   The fields are arranged with even spacing.
To specify the length of vertical or horizontal spacing between fields

1 In BMC Remedy Administrator, open the form for which you want to specify field spacing.

2 Select the fields whose spacing you want to specify.
   You must select at least two fields.

3 In the toolbar edit field, enter the number of pixels to be used for spacing the selected fields.
   You must enter a whole number; decimal values are not allowed.

4 Choose Layout > Field Spacing > By Length Vertically or By Length Horizontally.
   The selected fields are spaced by the amount specified.

Arranging fields on a form

The following sections describe how to arrange fields on forms.

The font preferences that you, and your users, set affect the layout of the form. Use the same font settings that you expect your users to use. For more information about form font preferences, see the Getting Started guide.

To change the position of a field in a form, you can do any of the following steps:

- Drag it to a different position.
- Use the Layout menu and corresponding toolbar buttons.
- Set the X and Y coordinates in the Display tab of the Field Properties window.

The following sections describe how to move a field by dragging it into position or by using the Layout menu. For information about aligning a field by using the field's X and Y coordinates, see “Display properties” on page 428.
Fields consist of two regions (as shown in the following figure): the label and the data entry region (which might include a menu or edit icon). When selected, fields contain resize handles.

Figure 8-17: Field regions you can align to the grid

When you use the toolbar buttons (shown in the following figure) or the Layout menu to move groups of fields, you can move them based on the field label or the data entry region. That is, you can align fields so that all of the field labels line up or so all of the data entry regions line up. After you have lined up the fields by the field label or data entry region, you can adjust each manually by dragging the handles to the appropriate position.

Figure 8-18: Form window—Layout menu

Too lb ar buttons available for aligning objects
The toolbar buttons and the Layout menu enable you to align groups of fields:

- When you align fields by using a grid, you must choose an align-to-grid option before you start moving fields. You execute the selected movement, such as Left, when you select and drag the appropriate field.

- When you align fields to a specified field, the Align Fields menu items and corresponding toolbar buttons become active as soon as you select two or more fields to move. You execute the selected movement, such as Right, by selecting this option from the Layout > Align Fields menu, or by clicking the corresponding toolbar button.

The Layout menu options are listed in the following tables.

<table>
<thead>
<tr>
<th>Align Fields</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>Aligns the left edges of the fields.</td>
</tr>
<tr>
<td>Right</td>
<td>Aligns the right edges of the fields, including any menu, edit, or diary icons.</td>
</tr>
<tr>
<td>Value Left</td>
<td>Aligns the left edges of the data entry region (the part of the field where you enter information) of the fields.</td>
</tr>
<tr>
<td>Value Right</td>
<td>Aligns the right edges of the data entry region (the part of the field where you enter information) of the fields.</td>
</tr>
<tr>
<td>Center</td>
<td>Centers the selected fields horizontally relative to one another. This option can be selected only from the menu; it does not have a corresponding toolbar button.</td>
</tr>
<tr>
<td>Top</td>
<td>Aligns the top edges of the fields, enabling you to align a horizontal group of fields.</td>
</tr>
<tr>
<td>Bottom</td>
<td>Aligns the bottom edges of the fields, enabling you to align a horizontal group of fields.</td>
</tr>
</tbody>
</table>

Note: The toolbar icons become active when you select multiple fields. However, the icons are only shortcuts to the Align Fields menu functions. This means that the icons will move the field according to the alignment field even if you have the grid turned on.
**Size Fields**

Defines how fields are sized when you want multiple fields to have the same field label length, data entry length, or both. When you select multiple fields, one of the fields (the alignment field) will have a darker outline. All other fields are sized based on the dimensions of this field.

You can adjust the size of the selected fields by:

| ![Label icon] | **Label**—To make all selected field labels the same size. If a field label is longer than the alignment field, it will remain the same size. |
| ![Value icon] | **Value**—To make all selected field values the same size as the alignment field. |

**Align to Grid**

Defines what portion of the alignment field is aligned to the grid when you select and drag a field across the form (see “Using the grid to align form fields” on page 287). Pull to the right to select from the following options:

- **None**—The grid is ignored.
- **Left**—Aligns the left edge of the field labels to the grid.
- **Value Left**—Aligns the left edges of the data entry region of the fields to the grid.
- **Right**—Aligns the right edges of the fields, including any associated field icons, to the grid.
- **Value Right**—Aligns the right edges of the data entry region of the fields to the grid.

A check mark appears to the left of the enabled menu selection.

**Expand**

Proportionally increases or decreases the distance between all of the fields in the form. The Expand Control Spacing dialog box appears so that you can specify a percentage of the expansion horizontally, vertically, or both. Enter percentages greater than 100 to move the fields farther apart. Enter percentages less than 100 to move the fields closer together.

**Bring to Front**

Placed the selected object in front of the objects covering it.
Whether you choose to align a field to a grid, or to a specified field, you must first select the field. When you select multiple fields, one of the fields will have a darker outline and dark points marked on the corners and sides (see the Button field in the following figure). This is called the **alignment field**. Other fields in the group are aligned based on the position of this field.
Using the grid to align form fields

You can use a grid for precise alignment of fields along a network of lines. If you want to adjust the spacing between grid cells, see the procedure “To adjust grid size” on page 289. You can use the Layout menu or a toolbar icon to display the grid, but it is not necessary to display the grid to use it for field alignment. This is because the grid is not enabled by displaying it; it is enabled by selecting an Align to Grid menu option other than None.

You can override the grid by specifying X and Y coordinates in the Field Properties window. If one of the Align to Grid options is activated, setting X and Y field coordinates overrides it. For information, see “Arranging fields on a form” on page 282.

You can also use the arrow keys to move a field by the amount shown in the Keyboard Move Step Size field in the Form tab of the Preferences dialog box (choose File > Preferences), thus overriding the grid.
To align fields to the grid

1. If you have not already done so, use “To move and space selected fields” on page 290 to select the fields that you want to move, and set the alignment field.

2. To display the grid, click the Show Grid toolbar button.

   The grid appears, as shown.

Figure 8-20: Displaying a form grid

3. Choose Layout > Align to Grid, and choose the appropriate option.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>Fields align the left edges of their field label with the grid.</td>
</tr>
<tr>
<td>Value Left</td>
<td>Fields align the left edges of their data entry region with the grid.</td>
</tr>
<tr>
<td>Right</td>
<td>Fields align the right edges of any icons associated with the field with the grid.</td>
</tr>
<tr>
<td>Value Right</td>
<td>Fields align the right edges of their data entry region with the grid.</td>
</tr>
</tbody>
</table>

   A check mark appears to the left of the enabled grid option.

4. Drag the selected fields to where you want them positioned in the form.
The alignment field (and all other selected fields) is aligned to the grid according to the selected option. For example, if you choose Layout > Align to Grid > Left, the left edge of the Button field (and the Status and Short Description fields) will align to the left edge of the grid, as shown in the following figure.

Figure 8-21: Aligning left

Aligning Left aligns fields on their left edges, according to position of the alignment field.

5 Choose File > Save Form to save changes to the form.

To adjust grid size

You can control the incremental movement of a field within a grid by adjusting the size of the grid cells.

1 Open the form with which you want to work.

2 To display the grid, click the Show Grid toolbar button.

3 Choose Layout > Grid Size.

4 In the Width and Height fields, enter the appropriate pixel size (2 to 100) for the grid.

The smaller the number of pixels, the smaller the grid and the more precisely you can align your fields.
5  Click OK.
6  Choose File > Save Form to save changes to the form.

**Moving and spacing form fields**

The following sections describe how to select the fields that you want to move, change the alignment field, and increase or decrease the amount of space between all of the fields in a form.

**To move and space selected fields**

1  Open the form with which you want to work.
2  Select the fields that you want to align by using one of the following methods:
   - Use the mouse to “lasso” the appropriate fields.
     - If you select multiple fields, the topmost field will be the alignment field.
     - If there is more than one topmost field, the topmost field on the far left will be the alignment field.
     - To lasso fields in a page field, hold down the Control key while you drag the mouse.
   - Click the appropriate field to select or deselect it.
     - To work with multiple fields, hold down the Shift key as you click each field. The last field that you select will become the alignment field.
3  If necessary, use one of the following methods to change the alignment field:
   - Press the Ctrl key, and click the appropriate field.
   - Extend the selection by pressing the Shift key and clicking on more fields; the last field selected will become the alignment field.
   - Drag a group of fields; the field with which you drag becomes the alignment field.
Move the selected fields by using one or more of the following methods:

- Drag the fields into position.
- Align the fields to the alignment field as described in “Using the alignment field to align form fields” on page 292.
- Align the fields to the grid as described in “Using the grid to align form fields” on page 287.
- Align the fields by using the X and Y coordinates in the Display tab of the Field Properties window as described in “Display properties” on page 428.

If necessary, adjust the label size, or the data entry region, using one or more of the following methods:

- Select a field and drag the appropriate anchor to the appropriate place.
- Use the Layout > Size Fields options to make field labels, values, or both, the same size.

The Width and Height display properties reflect the new field width and height. The Length (Char) display property indicates the approximate number of characters that might fit into the data region.

Adjust the appropriate field properties.

For example, you could change the justification of the field label by adjusting the Justify display property, described in “Justify” on page 429.

Choose File > Save Form to save changes to the form.

To adjust the spacing between all form fields

1. Open the form with which you want to work.
2. Choose Layout > Expand to open the Expand Control Spacing dialog box.
   The horizontal and vertical percentages always appear as 100, even if you previously increased or decreased the spacing between fields.
3. In the Horizontal Expansion and Vertical Expansion fields, enter the percentages by which you want to increase or decrease the spacing between all fields in the form.
   Percentages less than 100 decrease the distance between fields, and percentages greater than 100 increase the distance between fields.
4. Click OK.
5. Choose File > Save Form to save changes to the form.
Using the alignment field to align form fields

The following procedure describes how to use an alignment field as the reference against which all other selected fields are aligned.

To align multiple fields to an alignment field

1. If you have not already done so, use the procedure “To move and space selected fields” on page 290 to select the fields that you want to move and to set the alignment field.

2. Choose Layout > Align Fields, and choose the appropriate option.

Figure 8-22: Aligning multiple fields to an alignment field

Aligning Left aligns fields on their left edges, according to position of the alignment field.

The selected fields line up with the alignment field according to the selected option. For example, if you choose Layout > Align Fields > Left, the Status and Short Description fields line up with the left edge of the Button field, as shown.

You can position the field label region manually, and then use Layout menu options to adjust how the data entry regions align.

3. Choose File > Save Form to save changes to the form.
Using Set Multiple Field properties to align form fields and labels

The following procedures describe how to set alignment properties for multiple fields, and how to set multiple field alignment properties to match a selected field.

To set multiple field properties

1. In the Server Information Window, open the form with which you want to work.

2. Select Form > Multiple Field Properties.

   The Set Multiple Field Properties dialog box appears with the field on the form listed, including hidden fields.

   Figure 8-23: Set Multiple Field Properties dialog box

3. Select the fields for which you want to set the properties.
4 Select the Clear Property Settings button to clear any unwanted settings in the fields.

5 Set the properties for all the selected fields.

The options are:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label Width</td>
<td>Defines the width of the label in pixels. If you enter fewer characters than the label requires for full display, the setting increases the number in this field to the minimum required to accommodate the label.</td>
</tr>
<tr>
<td>Data Length</td>
<td>Defines the width of the field’s data entry region in pixels. If you enter more characters than can be displayed, the text will scroll off the end of the field.</td>
</tr>
<tr>
<td>Label Alignment</td>
<td>Defines the vertical position of the label in the space available. The options are:</td>
</tr>
<tr>
<td></td>
<td>- Top (default for labels positioned at the left of the field)</td>
</tr>
<tr>
<td></td>
<td>- Center</td>
</tr>
<tr>
<td></td>
<td>- Bottom (default for labels positioned above the field)</td>
</tr>
<tr>
<td>Label Justification</td>
<td>Defines where the label is positioned relative to the left and right edges of the region available. The options are:</td>
</tr>
<tr>
<td></td>
<td>- Left (default)</td>
</tr>
<tr>
<td></td>
<td>- Center</td>
</tr>
<tr>
<td></td>
<td>- Right</td>
</tr>
</tbody>
</table>
Form and Application Objects

Managing fields in a form view

6 Click OK.

7 Save your form.

The changes will not be saved until you save the form. You can view the settings for a field after you have saved the form by selecting the field in the Fields in View pane and clicking the Like Selected Field button.

You can also set the properties for several fields to the same property specifications as a selected field, as described in the following procedure.

To set multiple field properties the same as a selected field

1 In the Server Information Window, open the form with which you want to work.

2 Select Form > Multiple Field Properties.

The Set Multiple Field Properties dialog box appears with the field on the form listed, including hidden fields.

3 Select a field that has the required properties.

<table>
<thead>
<tr>
<th>Label Location</th>
<th>Defines where the data field's label will appear in relation to the field. The options are:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top— The label is positioned above the field.</td>
<td></td>
</tr>
<tr>
<td>Left (default)— The label is positioned to the left of the field.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Audit Option</th>
<th>Defines how auditing will be conducted. The options are:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unchanged—Selected by default. Use this option if you are setting other field properties and you do not want to change the Audit Option value.</td>
<td></td>
</tr>
<tr>
<td>None—Changes to the field are not recorded by any audit processing.</td>
<td></td>
</tr>
<tr>
<td>Audit—Changes to this field trigger audit processing and its new value will be recorded in the audit form or log form, depending on the audit style you specified at the form level.</td>
<td></td>
</tr>
<tr>
<td>Copy—Either the database value or the value in the current transaction of present will be recorded during an audit, but will not trigger audit processing.</td>
<td></td>
</tr>
<tr>
<td>Audit and Copy—Changes to this field trigger audit processing. If the value has not changed, then the value from the database will be recorded (similar to the behavior of the Copy option).</td>
<td></td>
</tr>
</tbody>
</table>
4 Click the Like Selected Field button. The fields on the dialog box are filled with data relating to the selected field.

5 Select the fields to assume this data.

6 Click OK.

**Tips for aligning and resizing fields**

Use the following tips for aligning and resizing fields in a form:

- Use the grid. Setting the grid width and height to 10 should allow you enough control to align the fields in your form precisely.

- Keep the grid on for accurate alignment of your fields.

- Align the field labels before you align the data entry regions, as shown in the following figure. To help with alignment, set the Align to Grid option to Left.

- If you select multiple fields, the fields are grouped and cannot be modified individually. After you align the field labels, click anywhere in the form. Then select the field and resize its data region individually. After selecting a field, the resize handles appear for the field, as shown in the following figure. Select the resize handle for the data region and stretch it to the appropriate alignment and size. For example, you could stretch data regions so that the fields align. Then you could stretch the data region again so that the menu boxes align.

**Figure 8-24: Aligning data regions**

![Diagram showing alignment and resizing of form fields.](image-url)
Setting the tab order for the fields in a form

The Set Tab Order dialog box determines which field is selected in what order when the user presses the Tab key.

If you do not define the tab order for a form view, the default tab order is based on the X and Y positions of the field in the form and moves from left to right, then down (in a zigzag-like pattern). A field added to the top of a form view would receive the cursor before a field added to the bottom of a form view, regardless of the order in which they were added to the form.

You can define any tab order and override the field’s position default so that users will go to specific fields in the order that you specify, despite their location on the form. For example, you can define a tab order that includes all required fields first, and then all optional fields. Also, because different views of the same form can have different tab orders, you can define tab orders so that different groups of users are first taken to the fields that are the most significant to them.

You cannot use tab order control to move from a field in one form to a field in a different form or view. You also cannot use tab order to move on to, off of, or between pages.

To set the tab order of form fields

1. Open the form with which you want to work.
2. To change to a form view other than the one displayed, choose Form > Select a View, and select the appropriate view from the Select View dialog box.
3. Choose Form > Current View > Set Tab Order.

The Set Tab Order dialog box appears, as shown in the following figure. A tab order list with the fields in default tab order is displayed. The first field in the tab order list will be first in the tab order, the second field will be second, and so on.
### Figure 8-25: Set Tab Order dialog box

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Move Up" /></td>
<td>Moves the selected field to the top of the tab order list.</td>
</tr>
<tr>
<td><img src="image" alt="Move Up" /></td>
<td>Moves the selected field one space up in the tab order list.</td>
</tr>
<tr>
<td><img src="image" alt="Move Down" /></td>
<td>Moves the selected field one space down in the tab order list.</td>
</tr>
<tr>
<td><img src="image" alt="Move Bottom" /></td>
<td>Moves the selected field to the bottom of the tab order list.</td>
</tr>
<tr>
<td><img src="image" alt="Restore Default" /></td>
<td>Resets the tab order for this form view to the default settings (from top-to-bottom, left-to-right).</td>
</tr>
</tbody>
</table>

4. Select the name of the field you want to move, and click the appropriate button.

5. Click OK when the fields are in the correct tab order.

6. Choose File > Save Form to save changes to the form.
Note: Fields that are on pages appear as “nested” under the page on the Set Tab Order window. Shared fields on pages appear under each page on which it is shared. This enables you to set the tab order differently on different pages.

Including and excluding fields from form views

Use the Fields in View dialog box to specify which fields will appear in a form view. This feature can be used to exclude fields used for internal workflow that users do not need to see. Even if you remove a data field from a user’s view, it is still available for workflow.

Note: Any field not in the current view does not affect the change flag, even if the Disable Change Flag field property is not selected. You can associate a SET-CHANGE-FLAG Run Process action with a field not in the view to set the change flag as needed. For more information, see “Disable Change Flag” on page 431 and the special Run Process discussion in the Workflow Objects guide.

To add and remove fields in a form view

1. Open a form in BMC Remedy Administrator.
2. Choose Form > Manage Views.
3. Select a view and click the Display button.
   The appropriate editing environment opens, depending on the view type defined for the selected view.
4. Choose Form > Current View > Fields in View.
   The Fields in View dialog box appears.
To add fields to the current form view, select the appropriate field from the Fields Not in View (By Name) list, and click Add.

To remove fields from the current form view, select the appropriate field from the Fields in View (By Label) list, and click Remove.

Use the Display Type list to change the appearance of the selected fields in the Fields in the View (By Label) list.

The Display Type list applies only to fields that can be displayed in multiple ways (such as selection fields, which can be radio buttons or a list).

Click OK to close the Fields in View dialog box and return to the form view when finished.

Choose File > Save Form to save your changes.
Preference settings for view selection

You can use preference settings and selection criteria to define which view of a form to display when opening a form in BMC Remedy Administrator, BMC Remedy User, or the web client.

Use the Default Locale and Default View Type settings on the Form tab in the BMC Remedy Administrator Preferences dialog box to decide which view type and locale version of a form view is displayed when an administrator opens a form.

When opening a form, rules are applied in selecting which view to open initially. The Choose Default View selection in the BMC Remedy Administrator Manage Views dialog box corresponds to the Label field for a view. When multiple views match the selection in the Choose Default View list, then criteria for view selection is applied. The process works as follows:

**Step 1** BMC Remedy Administrator attempts to select a view based on the client type (for example, Windows or the web).

For views that match the default view selected in the Choose Default View list in the Manage Views dialog box, BMC Remedy Administrator filters out views that do not match the client type specified for the default view type.

If a default view type is not selected as a preference, the Standard (Recommended) view is used.

**Step 2** From the list of remaining views, BMC Remedy Administrator attempts to select a view based on the default locale. Locale is selected as follows:

- For default locale, the system first attempts an exact match following the format: `<language>_country`. If a default locale is not selected, or cannot be found in the list of views, then the first view that matches the View type is selected.

- If default view type and default locale are not set, then the first view that matches the selection in the Choose Default View list in the Manage Views dialog box is selected.

- If a default view is not selected in the Manage Views dialog box, then the first view in the list is selected.

User preference settings for default view, locale, and the client view type (Standard or Web) control which view of a form is opened in the client.
Although a user can set view and locale preferences, an administrator should set these options for the user because the administrator defines form views. For information about the view selection process for the web client, see the Installing and Administering BMC Remedy Mid Tier guide.

For information about setting user preferences in the BMC Remedy User Options dialog box, see BMC Remedy User Help.
This section describes how to use BMC Remedy Administrator to create and modify entry points for use in home pages. The following topics are provided:

- Overview of home pages and entry points (page 304)
- Using entry points (page 305)
- Using a home page (page 308)
- Creating form entry points (page 312)
- Creating entry point guides (page 316)
- Overview of entry points in applications (page 319)
- Modifying or creating a home page (page 321)
- Viewing a subset of entry points (page 326)
- Home page navigation aids (page 329)
- Configuring home page preferences (page 330)
Overview of home pages and entry points

The home page lets you turn a form into a single point of access for users into AR System. A home page contains the following characteristics:

- Is any form that contains an Application List field
- Appears automatically (unless you disable it) when users log in to BMC Remedy User or when users access the home page URL from a browser (http://<host_name>/arsys/home)
- Is designated by user preferences, server, or mid tier configuration settings to function as a “portal” into AR System
- Contains a list of entry points inside the Application List field
- Could be any form that you designate as a home page

Figure 9-1 illustrates a sample home page form that contains an Application List field with multiple entry points.

Figure 9-1: Sample home page
Using entry points

Entry points are links in an Application List field that users can click to open, for example, a Search window. The use of entry points in AR System simplifies how different groups of users—for example, application designers, administrators, users—interact with applications, forms, and entry point guides.

**Application designers**

Entry points allow application designers to package common starting points that all users can easily access. Application designers are responsible for the following tasks:

- Designing the application.
- Defining the entry points used to interact with the application.
- Setting the correct permissions within the application so that the appropriate users see the appropriate entry points.
- Performing localization tasks, if any.

On the other hand, application designers do not need to have knowledge of the actual home page form that the entry points will be displayed on.

**Tip:** You do not need to make entry points for every starting point, only for those most frequently used. If you have a large number of entry points (for example, over 30), consider using the subset mechanism (see “Viewing a subset of entry points” on page 326) to limit the number of entry points displayed.

**Administrators**

Entry points make access to tasks much easier to control, maintain, and distribute for administrators who buy and install BMC Remedy applications. Administrators are responsible for the following tasks:

- Installing the 7.0 application.
- Configuring the server and mid tier to use the correct home page form.
- Specifying user preferences (or BMC Remedy User options) to behave correctly for each user. (BMC Remedy User only)
- (Optional) Specifying a home page form for each user (both BMC Remedy User and web clients)
(Optional) Designing a custom homepage form if the default Home Page is insufficient for their needs.

(Optional) Changing the Home Page settings in the mid tier Configuration Tool or Server Information window to specify the server and form for the home page.

As a help to administrators, AR System installs a default home page form automatically. However, they can create a customized home page by inserting an Application List field on to the form.

Users

Users do not need to know which applications or forms are available on a given server. Entry points simplify how users interact with AR System forms and applications. Entry points also make it easier for web users to keep track of multiple servers and URLs.

Users can access their tasks through a home page form, which is any form that contains a special Application List field listing all the entry points that a user has permission to access.

Types of entry points

AR System provides three different types of entry points:

- Default new entry points to open a form in New mode.
- Default search entry points to open a form in Search mode.
- Entry point guides that allow you to use all the functionality of a guide. For example, you can use an entry point guide to run reports or Modify or Display windows, or to run AR System workflow. This guide functionality is hidden from users. They do not know the entry point is a guide; they simply see this advanced functionality as another entry point.

Figure 9-2 illustrates a home page form with multiple entry points.
Detailed procedures for creating entry points are described in “Creating form entry points” on page 312 and “Creating entry point guides” on page 316.

When the user clicks an entry point to an application, the appropriate form appears in the specified mode (for example, New or Search). You can also create an entry point to a guide that exposes users to more advanced functionality, for example, a guide that displays a report or opens a modify window.

BMC Remedy User and web clients differ slightly in how entry point guides behave:

- In BMC Remedy User, you can access an application without knowing which form performs which task, but you must know which servers to log in to.
- In a web client, you can access an application without having to log in to a particular server, or without knowing which form performs which task.
Note: Initially, web client display entry points differently than BMC Remedy User does. For more information, see “Understanding how a home page appears in a browser” on page 333.

Entry points are designed to hide underlying forms, applications, and guides and give users one point of access. In BMC Remedy User, the home page functions like a control panel and opens automatically after you log in (unless you disable it).

Users can still access the “classic” Open dialog box by choosing File > Open > Object List. In addition, BMC Remedy User lists the most recently accessed entry points by choosing File > Recent Entry Points. For more information, see “Understanding how a home page appears in BMC Remedy User” on page 331.

Finally, you can localize entry points, just like other AR System components. For more information, see Appendix H, “Localizing AR System applications.”

Using a home page

When using a home page, keep the following terminology in mind:

- “Home Page” is the name of a default form that is installed automatically with AR System, as a convenient starting point for administrators. You can customize both the content and the look and feel of this form.

- The home page is any form that contains an Application List field.

In both cases, your home page functions as a console that is designed to open by default when users log in to BMC Remedy User or when they access the home page URL (http://<host_name>/arsys/home).

In both cases, the common element is the presence of the Application List field on a form. This field is a list of exposed entry points that are dynamically generated from AR System, for example, when the cache of the web client is updated and a new copy of the home page is displayed.
When you open the Object List in BMC Remedy User or a browser, the Object List contains every form, application, and guide on the server that the user has permission to. Not all of these forms are intended to be starting points. A home page form makes access to AR System much less complicated for Windows or web users. They no longer need to scroll through a long list to find the particular form they are looking for. (For web users, see “Using home pages with web clients” on page 311.)

Finally, you can suppress the automatic opening of the home page in BMC Remedy User as needed. For more information, see “Suppressing automatic opening of home pages in BMC Remedy User” on page 337.

**Home page components**

With a home page, users no longer need to know which server and form to open to perform a given task. When users open the home page in BMC Remedy User, they see a list of available entry points they can select from. The home page lets all users access multiple entry points in a single list. Figure 9-3 shows the default home page form that comes installed with AR System and some sample entry points.

*Figure 9-3: Default home page in BMC Remedy User*
The default home page form displayed in BMC Remedy User contains the following components:

- System-generated heading—Shows the default Home Page title and name of user logged in to AR System.

- Application list field, which consists of the following items:
  - Application heading—When you create a deployable application and include any forms with entry points, the application name appears as the heading in the home page by default. But if you add a label to your application, the label appears as its heading.
  - Entry point for a New (Submit) window—When you create a form, you can specify an entry point that opens a New window.
  - Entry point for a Search window—When you create a form, you can specify an entry point that opens a Search window.
  - Entry point guides—When you create an active link guide, you can specify an entry point that opens the guide. This guide functions exclusively as an entry point. Users will not know that this link actually points to a guide. For them, it is simply an entry point.

When you are creating a home page form, you use AR System objects and their properties you are already familiar with. The heading (This heading is the label for the deployable application) is the label of a deployable application. You create the labels for a form’s entry points in the Aliases and Labels tab in the View Properties dialog box. Finally, the label for an entry point guide is actually the label of the active link guide.

You should customize the labels of the entry points to make them task-specific and easily identifiable, for example, Generate a New Request or New Employee Set Up. You can use all the resources of AR System to create multiple views, define background images and color, display additional fields, and so on, on your home page.
Using home pages with web clients

A home page form makes access to AR System much less complicated for web clients. If you are using a web client, the home page appears automatically when users access the home page URL.

**Figure 9-4: Home page in web client**

The home page is published as a simple URL:

```
http://<host_name>/arsys/home
```

For example:

```
http://polycarp/arsys/home
```

If you enter a URL incorrectly, the mid tier Object List appears (Figure 9-5), listing all the forms and applications on accessible servers.
Creating form entry points

The following section describes how application designers can define a form entry point. These steps are the minimum (and simplest) for displaying entry points in the home page to open New or Search windows. You can mix and match entry points on your form as needed. For example, you might need only a Search entry point but not a New entry point.

To create form entry points, you enable entry points for the form, and then create custom labels for each view of the form, as described in the following procedures.
Note: Based on user preferences for locale and default view, different views of a form might be selected as entry points for each user login. For more information about how views are selected for users, see “Preference settings for view selection” on page 301.

To create form entry points

1. Create a form (for example, Customer Order Form).

2. Open the Form Properties dialog box (Form > Form Properties) and perform the following steps on the Entry Points tab:
   a. Enable the New or Search modes for the entry points you want to create.
   b. Set the values of the entry points for the New and Search windows. For this form, we set the New entry point to 100 and the Search entry point to 101.

Figure 9-6: Creating entry points
These values must be greater than or equal to 1. They determine the order of entries in the application list. For example, an entry point with a value of 101 will appear in the list after an entry point with value of 100. You should plan to logically group your most important entry points at the top of the application list.

**Tip:** If the entry point orders are the same for multiple entry points, the order they appear in the Application List field is random. Plan the order of your entry points carefully. Also, your numbering scheme does not have to be sequential; you can have gaps in the order numbers.

3 Make sure that your form and its fields have adequate permissions for users to view the correct information. The only entry points that users will see in the form’s Application List field are those they have permission to access.

4 (Optional) Use the Form Action Controls dialog box to add a home page form action button to your form. Adding this button on your form allows users to return easily to the home page with one click. For information, see “Home page navigation aids” on page 329.

**To configure custom view labels for entry points**

1 Select a view of the form, open the View Properties dialog box (Form > Current View > Properties) and click the Aliases and Labels tab.

2 Define the label string that you want to appear for each entry point.
Creating form entry points

Only define labels for each entry point defined in the Form Properties dialog box, such as New or Search.

If the view is selected for the current user login, the defined text will appear as the label for the form entry point in the home page, for example, *This is the entry point for a New window.*

3 Repeat step 1 and step 2 for each view of the form.

Make sure you define a label string. If you do not, the Application List field will display the entry points by their form name.
Creating entry point guides

The following section describes how application designers can define entry point guides. You use them to run complex workflow, for example, to open a Report window.

For other considerations when creating workflow, see “Workflow issues with active links and entry point guides” on page 318.

To create entry point guides

1 Create your starting active link.

You will be using this starting active link when you create your active link guide.

a In the Basic tab, specify the form and the Run If qualification for your active link.

b Click the If Action tab and create an Open Window active link action. You can use additional active link actions but you should create at least one Open Window action to open the window from your entry point.

This starting active link will execute on your home page form.

2 Create an entry point guide.

You create an entry point guide just like a regular guide, except that an entry point guide belongs to the form that is being opened and the starting active link that opens the form. If you do not plan your active links carefully, you might see some unexpected behavior. (For more information, see “Workflow issues with active links and entry point guides” on page 318.)

a Specify the name, label, form, entry point, display order, and starting active link (created in step 1) for the guide.
Here you specify the active link that opens the form (which you created in step 1). For more information, see the Workflow Objects guide.

b Define your permissions. Permissions are crucial for determining what groups of users can access your guide.

c In the Active Links tab, add any other active links you want to use that run after the form is opened. The form is opened from the Open Window action in the starting active link. After the form opens, the active links specified in the Active Links tab are run.

These active links will execute in the window you are opening with the starting active link. Therefore, these active links in their Form Name list must contain all the names of the forms that could be opened in the starting active link action.

d Add a descriptive label that you want to appear in the home page. Otherwise, the home page will simply display the guide name.
Workflow issues with active links and entry point guides

You should be aware of the following issues regarding how active links run inside entry point guides:

- When you specify the starting active link inside the entry point guide, remember that both the Run If qualification and the If and Else actions of the active link are run inside the home page form. You might not expect this behavior in the active link. To test this, create an active link that sets a field value to use inside the guide (for example, setting the Short Description field with Hello World!). Also, make the Short Description field visible on the home page form. After you run the entry point guide, you will see that Hello World! appears in the home page form, not the primary form that owns the active link.

As a result, while you can use any type of active link action inside the starting active link, not all operations make sense. Also, make sure your active link guide Run If qualification is valid inside every home page form that it can be run in.

- After the starting active link is run on the home page, the active links specified in the Guide list are run automatically for you. As a result, you do not need to create a Call Guide action to run these active links. The active links run as defined in the context of the form opened in the starting active link of the guide.

Typically, you should define the starting active link only with an Open Window action, to avoid unexpected behavior. You would add other active links in the Active Links in Guide list. You could design your application to perform tasks like opening a URL. For example, you could create a starting active link with a Run Process action, as in the following example:

PERFORM-ACTION-OPEN-URL http://www.google.com

This active link opens the Google search engine after the user clicks the link in the Application List field.
Overview of entry points in applications

The following section describes how application designers can add entry points to applications. It builds upon everything you just learned about entry points up to now, defining which order entry points are displayed in the Application List field (reserved field ID 1575).

Defining application permissions

The entry points that a user can see in an application depend on the following factors:

- The Application List field displays only those applications to which a user has a license.
- The Application List field displays only those deployable applications to which a user has state permissions. You should also factor in the permissions associated with Groups/Roles.

Plan all your permissions carefully. If users do not have permissions to a deployable application, then they will not see any entry points for that application on the home page, even if they have the correct permissions to the entry point. On the other hand, if users have permissions to the application but not the underlying objects (for example, a form) in the application, they will not see the entry points. You can easily set up default permissions for objects you create, such as applications, forms, fields, and workflow. For more information about access control, see “Defining default permissions” on page 87.

Note: Even without permissions to the application, users can access objects to which they have permission from the File > Open > Object List menu in BMC Remedy User.
Creating entry points in applications

The following procedures describe how to create entry points in applications. For detailed information about creating local and deployable applications, see Chapter 1, “Defining applications.”

To create application entry points

1. Create an application that requires entry points, or open an existing application.
2. Create form and guide entry points.
3. Choose Application > Properties to open the application properties page.
4. (Optional) In the General tab, enter a descriptive header in the Label field.
   - For deployable applications, this label text will appear in the home page as the application heading for your entry points, as shown in Figure 9-9. Otherwise, the application name appears by default as the heading.

Figure 9-9: Entry points

You do not need to create a deployable application for the entry points of an application to appear in the home page. However, putting your forms into a deployable application lets you include a descriptive heading to categorize the entry points.

- For local applications, no headings appear. Only the list of entry points is displayed in the application list.
5. Click the Forms tab and add your forms to the application.
6. For deployable applications, make sure you use the “Standard” form view, unless you have some special reason to use a web view, for example, backward-compatibility with old applications and forms.

Note: For more information about views, see Chapter 8, “Creating and managing form views.”
Create as many applications with their forms and entry points as needed. Applications appear in the home page in alphabetic order.

Modifying or creating a home page

The following section describes how administrators can modify the default Home Page form or create their own home page. It also describes how you can restrict users to interact with AR System only through the home page. For example, you now can restrict all access to AR System with BMC Remedy User to the home page. You also can allow them a “back door” to access applications, forms, and guides through the “classic” task list, by choosing File > Open > Object List in BMC Remedy User.

Modifying the default home page form

This section describes how to modify the default home page form that comes installed with the AR System server.

To modify the default home page form

1. Open the default home page form in BMC Remedy Administrator.
2. Open the Application List field.
3. In the Field Properties window, click the Advanced tab.
4. Define how you want the Application Field to appear, for example, if you want a border or scroll bars to be displayed.
5. Click the Permissions tab to define which groups should have access to the home page form.
6. Change the permissions of the home page form as needed.
7. Save your changes.

BMC Remedy User—creating new home pages

This section describes how to create a new home page form to use with BMC Remedy User.
To create a new home page form for BMC Remedy User

1. In BMC Remedy Administrator, open or create the form you want to use as your alternate home page form.

2. Add an Application List field to display your entry points.
   You can include an Application List field in any AR System form.

3. Set your preferences.
   The following steps are crucial to configuring which entry points will appear in the Application List field on your home page form. For more information, see “Understanding how a home page appears in BMC Remedy User” on page 331.
   a. Open the AR System User Preference form in BMC Remedy User, and then click the Home Page tab.

   You can also use define BMC Remedy User settings by choosing Tools > Options to open the Options dialog box.
   b. Specify the server the home page form resides on.
   c. Specify the default home page form.
   d. In the Object List field, select how you want users to interact with the home page. For example, selecting Disable hides the menu item and icon to access the Object List.

4. Display the home page by logging out, then logging back in to BMC Remedy User.

   In BMC Remedy User, the home page will be automatically displayed. You can also click the home page icon to display the home page, choose File > Open > Home, or click ALT+HOME.

   You can also set a preference in the AR System User Preference form to disable the “classic” task list, so that users are forced to use the home page.

Web clients—creating new home pages

This section describes how to create a new home page form for use with web clients.
To create a new home page form for web clients

1. In BMC Remedy Administrator, open or create the form you want to use as your alternate home page form.

2. Add an Application List field to display your entry points.
   
   You can include an Application List field in any AR System form.

3. Set your preferences.
   
   This step is crucial to configuring which entry points will appear in the Application List field on your home page form. For more information, see “Understanding how a home page appears in a browser” on page 333.

4. Display the home page.
   
   In a web client, enter the following URL:

   http://<host_name>/arsys/home

   If your mid tier server was named polycarp, the URL would be:

   http://polycarp/arsys/home

Using style sheets with application list and navigation fields

You can use style sheets to customize the appearance of the Application List and navigation fields in BMC Remedy User or a web client.

A style sheet named arsystem.css is located in the BMC Remedy User installation folder. Each user using that installation folder will have the styles in the style sheet.

For the Web, another arsystem.css style sheet determines the appearance of all fields (not only the Application List and navigation fields). This file is in the <mid_tier_install_dir>\resources\ folder.

Note: The arsystem.css file has replaced the appfield.css (or ARSystemApplist.css) file, which was used prior to version 7.0. If you have made customizations in the appfield.css file, you must copy these customizations to the arsystem.css file after you upgrade BMC Remedy User.
Using style sheets with BMC Remedy User

Users can edit the `arsystem.css` style sheet in their installation folder to change fonts, colors, attributes, and other properties of the Application List field.

**Note:** You should be familiar with HTML and Cascading Style Sheet (CSS) files before editing style sheets. An excellent background resource is the MSDN Library website at [http://msdn.microsoft.com/library/default.asp](http://msdn.microsoft.com/library/default.asp). See especially the sections on HTML, Dynamic HTML, and CSS Attributes.

After users save their changes to the style sheet, they will not see changes to the Application List field immediately because it is generated at runtime. The next time they open the Home Page—they do not have to log out and log back in to BMC Remedy User—the edited style sheet will be applied to the Application List field.

**Note:** You cannot rename the style sheet. BMC Remedy User expects the `arsystem.css` file to be present in the installation folder. The style sheet is updated with every installation of BMC Remedy User. If you customize the file, you must reapply your changes after an upgrade of BMC Remedy User. The installer will create a backup of your previous style sheet, so the customizations are not lost.

> **To use style sheets in BMC Remedy User**

1. Open the Home Page in BMC Remedy User to view the default settings of the style sheet.
2. Open the `arsystem.css` style sheet in a text editor.
   
   The `arsystem.css` file for each user is located in your BMC Remedy User installation directory folder.
3 Edit any of the attributes as needed.

You must use the defined style classes, for example,
.HomeListAppFirstStyle. You can add any available style to these classes.
Make sure you use the correct syntax, adding semicolons after an attribute,
valid font families, correct color or RGB settings, and so on. Edit the font
family, font size, and add a color to the style, for example:
.HomeListAppFirstStyle {font-family:Century Gothic; font-size:10pt;
font-weight:bold; color: red;}

4 Save your changes.

5 Close your Home Page (or the form containing the Application List field).

6 Open the Home Page.

The new attributes you defined will appear in the Application List field.

**Using style sheets with web clients**

Most of the common styles displayed in web clients are defined in
<mid_tier_install_dir>/resources/standard/stylesheets/ARSystem.css.
AR System provides individual arsystem.css style sheets for the MS Internet
Explorer (.resources/stylesheets/resources/ie/stylesheets) and
Mozilla (.resources/stylesheets/resources/moz/stylesheets) browsers if there are browser-specific styles to define. When you edit these
style sheets, you are making changes to the entire mid tier, not just to a
particular application.

When you open the Home Page in a web client, your browser automatically
applies the appropriate style sheet. Users can then edit this style sheet to
change fonts, colors, attributes, and so on.

**Note:** You should be familiar with HTML and CSS files before editing the
style sheets. An excellent background resource is the MSDN Library
especially the sections on HTML, Dynamic HTML, and CSS Attributes.
Also, MS and Mozilla attributes, while similar, are not exactly the same.
Not all attributes work with every browser.

For more information about ARSystem.css for the mid tier, see the Installing
and Administering BMC Remedy Mid Tier guide.
To use style sheets with application list fields for web clients

1. Open the Home Page in your web client.
2. Open the ARSystem.css style sheet for browsers in a text editor:
3. Edit any of the attributes as needed.
   You must use the defined style classes, for example, td.FIRSTAPPNAME. You can add any available style to these classes. Make sure you use the correct syntax, adding a semicolon after an attribute, valid font families, correct color or RGB settings, and so on. Edit the font family, font size, and add a color to the style, for example:

   td.FIRSTAPPNAME {
     font-family: 'Century Gothic';
     font-size: 10pt;
     font-weight: bold;
     color: deeppink;
   }

4. Save your changes.
5. Refresh your Home Page in the web client.
   The new attributes you defined will appear in the Application List field.

Viewing a subset of entry points

You can display only a subset of applications and entry points to users by creating sophisticated workflow that uses a reserved field ID (Displaying a subset of entry points works for deployed applications only.)

---

Note: In the default home page form installed with AR System, a hidden reserved field named AppSubset (field ID 1576) has already been created for you. You just need to add the necessary workflow to use it.

---

You might want to hide the reserved field on most form views. If you set its field values with the Window Loaded execute condition, the Application List field will be populated automatically with the subset of entry points when the window becomes loaded.

The following procedure adds a special character field (the reserved field used to configure the list of entry points) and a button to a form. It then uses Run Process workflow to display dynamically a subset of servers and applications.
Form and Application Objects

To test this functionality, create multiple applications and entry points, or use multiple AR System servers.

Note: If you use multiple servers on the Web, you must add servers using the BMC Remedy Mid Tier Configuration Tool. For more information, see “Specifying a server for the mid tier” on page 336.

To create sample workflow to view a subset of entry points

1. Add a character field to your home page and name it Configuration Field.
2. In the Database tab in the Form Properties dialog box, enter the value of 1576 in the ID field.
   This value creates a special reserved field ID. If this field contains a value, only the entry points for these applications and servers are displayed. Most likely, you will want to set this value dynamically through workflow.
3. Add a button field to your form and name it Button.
4. Save your changes to the form.
5. Create an active link with the following Basic conditions:
   - Form Name: your home page form
   - Execute On condition: Button
6. Create a Run Process If action with the following Run Process syntax:
   PERFORM-ACTION-HOME-FIELD-REFRESH
7. Save your active link.
8. Display your home page form.
   All the entry points in the Application List field are displayed. But when you enter a value into the Configuration Field and click Button, you will see a subset of entry points.
   For this example, two servers were used (polycarp and mirepoix). The Application List field on this sample home page contains a list of all the entry points this user can see that are available on these two servers.
Enter the special syntax into the configuration field to display a subset of servers and applications. Separate applications and servers by commas.

\(<\text{server\_name}>.<\text{application\_name}>, <\text{server\_name}>.<\text{application\_name}>\)

You must enter a backslash (\) escape sequence in the syntax if your application name uses a comma or period. For example, if the application is named Help.Desk on server polycarp, you must enter the following escape sequence:

```
polycarp.Help\..Desk
```

Enter \(<\text{server\_name}>\) to display all the entry points and applications that the user has permissions to access on \(<\text{server\_name}>\). Or enter \(<\text{server\_name}>.<\text{application\_name}>\) to display a subset of applications on a specific server.

For example, if you entered the following and clicked Button:

```
polycarp.MusicManager Application
```

the Application List field is redisplayed with only the entry points from the MusicManager application on the server polycarp (as shown in Figure 9-11).
Home page navigation aids

This section describes additional functionality that allows users to navigate their way back to their home page. You can use two different methods to give users easy access back to the home page:

- "BMC Remedy User—adding a home page form action field to a form."
- “Creating run process workflow that returns to the home page” on page 330

Web clients and BMC Remedy User behave differently when you click the home page button. The Web will always open the home page in a new window. In BMC Remedy User, if a home page window is already open when the button is clicked, that window will be brought to the front.

BMC Remedy User—adding a home page form action field to a form

The easiest way to navigate users back to the home page in BMC Remedy User is by adding a home page form action button to any form. Forms displayed in web clients include the home button as a required form action field in the toolbar. (For more information about form action fields, see the Installing and Administering BMC Remedy Mid Tier guide.)

To add a home page form action field to a form

1. Open a form in BMC Remedy Administrator.
2. Choose Form > Form Action Fields.
3. From the list of Form Actions, select Home Page.
4. Position the Home button as needed.
5. Save your changes.

This button functions as a handy navigation button you can easily add to any form.
Creating run process workflow that returns to the home page

Adding a Run Process command to workflow provides a great deal more flexibility than simply adding a home page form action button to a form. For example, you could create an entry point guide that opens a form and lets users perform specific tasks. The last action in the guide could be a Run Process action that returns the user to the home page.

The following procedure illustrates the Run Process workflow by creating your own “home” button.

To create run process workflow that returns to the home page

1. Add a button field to your form and name it Button.
2. Save your changes to the form.
3. Create an active link with the following Basic conditions:
   - Execute On condition: Button
4. Create a Run Process If action with the following Run Process syntax:
   
   PERFORM ACTION: GO-HOME

5. Save your active link.

This Run Process command allows workflow to simulate the “Home” form action button.

For more information about workflow, see the Workflow Objects guide.

Configuring home page preferences

The following sections describe configuration settings for using a homepage. This information is important to understanding which form is displayed as a home page.

Tip: Both BMC Remedy User and web clients use a “hierarchy” of values to resolve which home page form to open on what server. As a rule of thumb, the particular settings specified in the AR System User Preference form always override the global server and configuration settings.
Understanding how a home page appears in BMC Remedy User

When you log in to AR System, BMC Remedy User uses the following sequence to determine which home page form is displayed. Suppose that the AR System User Preference form is configured to use the server polycarp and HelpDesk as your home page form.

**Step 1** If you are accessing BMC Remedy User with a preference server, BMC Remedy User checks the server that is defined as a user preference on the Home Page tab.

**Note:** You can define preferences in either the AR System User Preference form or in the Options dialog box of BMC Remedy User.

If you are using a preference server and you configured a home page server, BMC Remedy User uses this setting to search for a home page form. With BMC Remedy User, the user preference is optional. For more information, see “Specifying a server and home page in the AR System User Preference form” on page 335.

If you did not define any user preferences, BMC Remedy User checks for the home page options defined in the Options dialog box (by choosing Tools > Options).

**Step 2** If no options are defined in the Options dialog box, BMC Remedy User checks for the preference server specified in the Login dialog box.

**Step 3** If there is no preference server defined, BMC Remedy User uses the first server in the login list as an alternative.

In this example, suppose the server found polycarp back in step 1. The server then checks polycarp for available home page forms.

**Step 4** If the server locates a home page form defined in your user preferences, BMC Remedy User opens and displays it as your home page.

In this example, the user preference was not the default home page form installed with AR System, but HelpDesk, an alternate home page form. BMC Remedy User would open and display HelpDesk as your home page.
Step 5  If you did not specify a home page in your user preferences, BMC Remedy User uses the home page defined in the Configuration settings in the Server Information dialog box.

For more information, see “Specifying a home page on the server” on page 336.

Step 6  If you did not configure a home page form on your server, BMC Remedy User searches for the first form it can find that has an Application List field.

Step 7  If such a form cannot be found, BMC Remedy User checks the next server in your login list, then repeats step 5 and step 6.

Step 8  If BMC Remedy User checks all the servers in your login list and cannot find any home page form, no form will be opened in BMC Remedy User and a warning is returned (ARWARN 1898).

For BMC Remedy User, the server compiles a list of entry points from all the servers in your login list and displays them in the home page’s Application List field. If two applications have the same name, BMC Remedy User displays their server name as well, as shown in Figure 9-13.

Figure 9-12: Application list field displaying applications with same name

For more information about configuring a home page server, see the Configuring guide.
Understanding how a home page appears in a browser

When you enter the home page URL (http://<host_name>/arsys/home) and then log in to AR System, the mid tier uses the following sequence to determine which home page form will be displayed in a browser. Remember that on the Web, the mid tier can be configured to have multiple preference servers. Suppose that the AR System User Preference form is configured to use the server polycarp and HelpDesk as your home page form.

**Step 1** The mid tier checks the home page server that is defined on the Home Page tab of the AR System User Preference form.

If you defined a home page server in user preferences, the mid tier uses this setting to search for a home page form. For more information, see “Specifying a server and home page in the AR System User Preference form” on page 335.

**Note:** If you do not create an entry in the AR System User Preference form, the mid tier creates a default record for you but it does not populate the fields relevant to the home page settings. These are left blank by default.

**Step 2** If no user preference is defined, the mid tier checks the home page server setting in the Configuration Tool.

If there is no configuration setting, the mid tier displays an error.

In this example, the mid tier found the server polycarp back in step 1. The mid tier then checks for available home page forms.

You define preference servers by clicking Home Page Settings in the Configuration Tool. For more information, see “Specifying a server for the mid tier” on page 336.

**Step 3** If the mid tier locates a home page form defined in your user preferences, the web client displays it as your home page.

In this example, the user preference was not the default home page form installed with AR System, but HelpDesk, an alternate home page form. The browser would open and display HelpDesk as your home page.

**Step 4** If you did not specify a home page in your user preferences, the mid tier searches for a home page form on the server defined in step 1 and step 2.
When a home page is found and opened, the mid tier has compiled a list of entry points from all the servers listed in the AR Server Settings in the Configuration Tool and displays them in the home page's Application List field. If two applications have the same name, the mid tier displays their server name as well, as shown in Figure 9-13.

Figure 9-13: Application list field displaying applications with same name (Web)

If such a form cannot be found, the mid tier displays an error and no form will be opened in the browser.

**Note:** The entry point labels that appear in the application list field are also determined by the form views. You must set the correct view label to expose, for example, a web view of the form's entry points. Otherwise, you might define an entry point that is not displayed properly in your web view. The system displays views based first upon the view label and then the locale. For more information, see the discussion on how views are selected and displayed in the “Preference settings for view selection” on page 301.

For more information about configuring home page servers, see the Configuring guide.
Specifying a server and home page in the AR System User Preference form

Although a default home page form comes installed with the AR System, you can define a different home page. Your user preferences override both the global server settings and mid tier configuration options.

These user preferences allow you to define a customized home page form for particular groups of users (for example, Marketing and Engineering). But you can also specify one home page for each user. For example, people working primarily in Marketing would see one home page. But if they communicate frequently with Engineering, they could have access to engineering applications, and the home page they see could be appropriately customized.

Note: With BMC Remedy User, user preferences are optional. With web clients, if you do not set any user preferences, you will not return any default value from your server or form. However, you do have a server defined in the mid tier configuration, as well as a setting for the home page form for the server.

To specify a server and home page in the AR System User Preference form

1. Open the AR System User Preference form in BMC Remedy User.

   Note: You can also define these preference settings through the Options dialog box. Choose Tools > Options, then click the Home Page tab. Or, you can modify the user preference record on the Web.

2. Click the Home Page tab.

   Here you define how you want users to interact with the home page. You now can restrict all AR System access to the home page. You also can allow users to access applications, forms, and guides through the Object List.

3. Specify the server the home page form resides on.

4. Specify the default home page form.
5 Select how you want users to interact with the home page.
   For example, selecting Disable in the Object List field hides the menu item to
   access the Object List. For backward compatibility, the Object List familiar to
   those who know BMC Remedy User is still available by choosing File > Open
   > Object List. You can also choose to show or hide this dialog box through
   these user preference settings.

6 Click Save.

Specifying a home page on the server

This global setting lets administrators easily specify which form to display as
home page, if this server is specified for its home page. If a different server is
designated, for example, in user preferences, this setting will be ignored.

This setting allows you to define a home page form for particular groups of
users, and customize the content and look of the form. For example, you can
add a character field that displays the applications and servers in the
Application List field.

To specify a home page in the Server Information window

1 Open the Server Information window.
2 Click the Configuration tab.
3 From the Default Home Page field, select the home page that you want to use
   system-wide.
   The default setting is the default installed home page form. If you designate
   a different home page, make sure it includes an Application List field. For
   more information, see “Overview of entry points in applications” on
   page 319.
4 Click OK.

Specifying a server for the mid tier

If you are using a home page with a browser, this global setting lets
administrators configure the mid tier with the AR System server that
contains the home page form.
Configuring home page preferences

To specify a server for the mid tier

1. Open the BMC Remedy Mid Tier Configuration Tool from the following URL:
   
   http://<host_name>/arsys/shared/config/config.jsp

2. Click the General Settings link.

3. From the Homepage Server menu, select the AR System server that you want to run against the mid tier.

4. If you use the subset reserved field (ID 1576) in your workflow and you use a combination of fully qualified domain names and relative host names, you must add all the variations of server names to AR Server Settings.

   For example, add the server names arserver, arserver.staff.acme.com, arserver.acme.com, and arserver.acme2.com.

   For more information about the subset field, see “Viewing a subset of entry points” on page 326.

5. Save your changes.

Suppressing automatic opening of home pages in BMC Remedy User

You can suppress the home page automatically opening in BMC Remedy User in two different ways:

- Using the AR System User Preference form
- Using the Options dialog box

If you use the Options dialog box, these changes are automatically saved in an entry in the AR System User Preference form.

To suppress home pages through user preferences

1. Log in to BMC Remedy User.
   
   Observe how the home page automatically opens.

2. Open the AR System User Preference form.

3. Perform a search for the preference entry of a login name.

4. Click the Home Page tab.

Configuring home page preferences  ➤  337
5 In the **Open Home Page Automatically** field, click **No** to disable the home page.
   The default setting is **Yes**.

6 **Save your changes.**
   You must log in again to see your changes take effect.

7 **Log in to BMC Remedy User** once more.
   Observe how the home page no longer automatically opens. The server on which the AR System User Preference form resides must be specified as the preference server when you log in.

► **To suppress home pages through the Options dialog box**

1 **Log in to BMC Remedy User.**
   Observe how the home page automatically opens.

2 **Open the Options dialog box.**

3 **Click the Home Page tab.**

4 In the **Open Home Page Automatically** field, clear the **Open Home Page Automatically** check box.
   The default setting is **Yes**.

5 **Save your changes.**
   You must log in again to see your changes take effect.

6 **Log in to BMC Remedy User** once more.
   Verify that the home page no longer automatically opens.
Packing lists are functional units that contain an administrator-defined grouping of information. For example, if an application contains ten forms, each form and its related workflow can be organized in its own functional unit, or packing list. Each of these units can be added to the application’s packing list. Another application can add these packing lists to reuse functional units from the preceding application. Packing lists offer the administrator a method of grouping and organizing large amounts of information to make development simpler. This eases the transition from development to production.

This section explains how to create and use packing lists for your applications. The following topics are provided:

- Using packing lists (page 340)
- Creating packing lists (page 340)
- Working in the packing list window (page 341)
- Working with packing lists in BMC Remedy Administrator (page 346)
- Saving packing lists in XML (page 349)
Using packing lists

Administrators can use packing lists within workspaces. Normally, the administrator works within the context of all the objects on the server. By grouping relevant objects in a packing list, the administrator can choose to work within the context of a defined packing list. In workspace mode, only objects belonging to the selected packing list are visible. (For more information about workspace mode, see the Getting Started guide.) New objects automatically become members of the packing list, and objects can be removed from the packing list without being deleted. Objects can belong to multiple packing lists, so they can be visible in more than one workspace. For more information, see “Viewing a packing list in a workspaces” on page 346.

Administrators can construct a packing list to create and apply external utilities, such as installation utilities or external object browsers. In addition, administrators can use packing lists to gather information about specified objects and manipulate those objects.

Packing lists can contain other packing lists. You can also store packing lists through Source Code Control integration. For more information about source control, see the Integrating with Plug-ins and Third-Party Products guide.

If you use the Distributed Server Option (DSO), you cannot track DSO mappings that have been renamed. Packing lists can track only server objects with database IDs. DSO mappings are not server objects, so they are not tracked by their database ID, but by their actual name. Therefore, if you change a DSO mapping’s name and then open a packing list that contained that mapping, the DSO mapping will be missing from the packing list.

Creating packing lists

Use the following procedure to create a packing list.

To create packing lists

1. In BMC Remedy Administrator, select a server to administer.
3. Select Packing List, and click OK.

The Create Packing List window appears.
4 Specify the application basic criteria. (See “Specifying packing list basics” on page 342.)

5 Specify the application permissions. (See “Defining packing list permissions” on page 345.)

6 Specify application subadministrator permissions. (See “Defining subadministrator permissions for packing lists” on page 345.)

7 Define change history. (See the Getting Started guide.)

8 Define help text. (See “Creating help text for packing lists” on page 345 and the Getting Started guide.)

9 Save your changes.

Working in the packing list window

Use the Packing List window (shown in the following figure) to create and modify packing lists. This section describes how to work with the tabs in this window.

Figure 10-1: Packing List window
You can open multiple windows to create or modify the packing lists that you have permission to administer.

Use the following tabs in the Packing List window to define parameters:

- **Basic**: Defines the basic properties of the packing list and the list of system objects needed to create a packing list.
- **Permissions**: Defines which access control groups can access the packing list.
- **Subadministrator Permissions**: Defines which access control groups have subadministrator permissions for the packing list.
- **Change History**: Records the owner of a packing list, 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 for the packing list. In most cases, this help text is a description of the packing list, what it does, and how it is used.

### Specifying packing list basics

The following section describes how to fill in the Basic tab of the Packing List window in Figure 10-1 on page 341.

#### To define packing list basics

1. **Create a packing list or open the packing list with which you want to work.**

   The Available Objects list displays all AR System objects defined on the server.

2. **Enter or update a name in the Name field.**

   Packing list names must be unique on each AR System server. Although there are no naming conventions, create names that provide meaningful descriptions of the packing list. Names can be a maximum of 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name.

   Names are shared across packing lists, active link guides, filter guides, web services, and applications, so each name must be unique.
3 Enter a label in the Label field.
Although there are no naming conventions, create labels that provide meaningful descriptions of the packing list. Labels can be a maximum of 255 bytes, including spaces.

4 Enter a description of the packing list in the Description field (optional).
You can enter a maximum of 2000 bytes.

5 Move items from the Available Objects list to the Packing List Objects list in any of the following ways:
- Select the available objects and click Add to add objects to the Packing List Objects list.
- Select the object type and click Add to add all objects of a specific type to the Packing List Objects list.
- Use the context menu to Add objects.
- Drag and drop objects between Available Objects and Packing List Objects list.
- Right-click on an object, and choose Add from the context menu.
- Use the Add Directly Related button (or right-click option) to limit the scope of server objects when selecting shared workflow to add to the packing list. This option defines new rules for each workflow object, establishing parameters that restrict the objects that will be related to include only the associations defined by the new rules (see table that follows) for each type of workflow.
- Use the Add Contents button (or right-click option) to move only the contents of container objects (for example, active link or filter guides, packing lists or applications) instead of all the workflow associated with the forms in the container. The Add Contents button becomes enabled only after you select a specific container object. This button works with all containers except web services.
- Use the Add All Related button (or right-click option) to move an object and its related objects. To drag and drop the objects instead, hold down the SHIFT key while dragging.
Each of the object definitions defined in the following table are included in the packing list as follows when you use the Add Related - Restrict button.

<table>
<thead>
<tr>
<th>For:</th>
<th>Packing list includes:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forms</td>
<td>All related menus, active links, filters, escalations, active&lt;br&gt;link guides, filter&lt;br&gt;guides, web services, and distributed mapping definitions. In addition, menus from the&lt;br&gt;Change Field action of the active links will be included.&lt;br&gt;Any other forms referenced in workflow actions, guides, and menus will not be associated as related objects.</td>
</tr>
<tr>
<td>Join forms</td>
<td>All forms that were used to create these join forms as well as their related items (as defined in the operations included in the Forms above).</td>
</tr>
<tr>
<td>Active links</td>
<td>All menus that are referenced in the Change Field actions, and all guides that are referenced in the Call Guide action. The guides should see the same form to which the active link refers. The active links that are referenced in the guide also fall within the same scope; therefore, the associated objects of those active links will be included. This cycle continues until it reaches a form.</td>
</tr>
<tr>
<td>Filters</td>
<td>All filter guides referenced in a Call Guide action and all DSO mapping definitions referenced in a DSO action. Filters that are referenced in the guide also fall within the same scope; therefore, the associated objects of those filters will be included. The guides should see the same form to which the filter refers. Escalations do not have any of the above actions, so there will be no associations for escalations.</td>
</tr>
<tr>
<td>Active link guides</td>
<td>All active links referenced in the guide as well as all associated objects for those active links.</td>
</tr>
<tr>
<td>Filter guides</td>
<td>All filters referenced in the guide as well as all associated objects for those filters.</td>
</tr>
<tr>
<td>Applications</td>
<td>All associated forms and the list of related objects associated with those forms.</td>
</tr>
<tr>
<td>Web Services</td>
<td>Included as an independent object.</td>
</tr>
<tr>
<td>Menus</td>
<td>No related items included.</td>
</tr>
<tr>
<td>Flashboards</td>
<td>Included as an independent object.</td>
</tr>
</tbody>
</table>
6 Use any of the methods in step 5 and the Remove button to move the selected objects from the Packing List Objects list to the Available Objects list.

7 Save your changes.

You can now open the packing list as a workspace or save it in XML. For more information about workspaces, see “Viewing a packing list in a workspaces” on page 346.

**Defining packing list permissions**

Use the Permissions tab to determine which access control groups can display the application in BMC Remedy User.

**Defining subadministrator permissions for packing lists**

Use the Subadministrator Permissions tab to define subadministrator permissions for access control groups.

**Building and using packing list change history**

AR System automatically records the owner of a packing list, the user who last modified the packing list, and the date of the modification. To display or add to this information, click the Change History tab in the Packing List window.

**Creating help text for packing lists**

To create help text for a packing list, click the Help Text tab in the Packing List window. In most cases, the help text that you enter is a description of the packing list, what it does, and how it is used. Only administrators and subadministrators can view and edit packing list help text.

For more information subadministration, permissions, creating help text, and on how to build and use change history, see the Getting Started guide.
Working with packing lists in BMC Remedy Administrator

You can perform various tasks with packing lists in BMC Remedy Administrator, as explained in the following sections.

Viewing a packing list in a workspaces

A workspace allows you to limit the objects displayed in a Server window to those objects that are associated with a particular packing list. When you create new objects in the context of the workspace, the objects are added to the packing list.

Workspaces are an effective way for you to view and modify the objects in the packing list. In the Packing List Properties window, you can only add or remove objects in the packing list.

Note: In AR System versions prior to 6.0, you could open applications in a workspace. The Application window—introduced in version 6.0—supersedes this functionality. Unlike Application windows, application workspaces did not let you modify properties for the current application, and you could not open more than one workspace. For more information about applications, see Chapter 1, “Defining applications.”

To view packing list objects in a workspace, see the following procedure.

To display packing list objects in a workspace

1. In the Server window, select a server.
   Make sure that the All Forms option is selected in the By Form dialog box; otherwise, the menu option will be disabled. (See Figure 10-1 on page 341.)

2. Choose View > By Workspace.
   The By Workspace dialog box appears.
Select the Enable Workspace check box to activate the list of available packing lists.

Select a packing list from the list, and click OK.

In the Server window, the object category reflects the options selected in the By Workspace dialog box.

To modify the contents of a packing list used in the workspace, choose Edit > Workspace.

Editing the workspace is useful for adding multiple objects to the workspace all at once.

Modify the objects in the packing list as needed.

Save your changes.

Changes to the packing list appear in the workspace.

To resume viewing objects outside of a workspace, close the Modify <Object> dialog box, choose View > By Workspace, and clear the Enable Workspace check box.
Including packing lists in a deployable application

You can include packing lists in deployable applications using the Packing Lists tab in Application Properties. You can include a packing list in only one deployable application at a time. For more information about deployable applications, see Chapter 1, “Defining applications.”

To specify packing lists to include in an application

1. Open an Application window by double-clicking the name of an application in the Server window.
2. Choose Application > Properties.

![Figure 10-3: Specifying application packing lists](image)

3. Click the Packing Lists tab. The list of packing lists that do not belong to other deployable applications appears.
4. Use the Add and Remove buttons to move packing lists to the appropriate Packing Lists list.
5. Choose File > Save Application.
Dragging and dropping packing lists between windows

You can drag and drop packing lists between windows in BMC Remedy Administrator.

**WARNING:** When you drag and drop objects into a deployable application, AR System removes all explicit group permissions from the objects. You must manually apply role permissions to packing lists and forms, and to every object associated with each form, including fields, and any active links and active link guides for which the form is the reference form.

Likewise, removing objects from a deployable application removes all role permissions from the objects. For more information about roles, see “Roles in AR System” on page 53.

Note the following tips:

- You can drag and drop packing lists between a Server window and a deployable Application window, or between deployable Application windows.
- Dragging and dropping a packing list from a Server window to a deployable Application window includes the packing list in the application, but does not include any of the objects inside the packing list.
- Dragging and dropping a packing list between deployable Application windows performs a move, and includes the packing list in the application. It does not move any of the objects inside the packing list.
- Dragging and dropping a packing list from a deployable Application window to a Server window removes the packing list from the application. It does not remove any of the objects inside the packing list.

Saving packing lists in XML

You can save packing lists in XML format on the administrator’s local machine or on the network. When you save a packing list externally, the AR System server converts its format to XML. In XML format, the group of objects in a packing list contain important header information for easy parsing. The header consists of the packing list’s name, server, and object property information, as the following example shows.
Chapter 10—Defining packing lists

BMC Remedy Action Request System 7.0

```xml
<?xml version="1.0" encoding="windows-1252"?>
<!- - Document Type Definition: Not Found -->
<packinglist name="SavePackingListXMLTest" label="SavePackingListXMLTest" version="1.0">
<description/>
<properties vendor="" product="" version="" copyright="" guid=""/>
<source-control version="" locked-by="" last-modified-by="" location="" timestamp="0"></source-control>
<form-list>
<form name="PackingListTest" server="POLYCARP.eng.remedy.com">
<properties vendor="" product="" version="" copyright="" guid=""/>
<source-control version="" locked-by="" last-modified-by="" location="" timestamp="0"></source-control>
</form>
</form-list>
<active-links-list>
<filters-list>
<filters name="PackingListTestAL" server="POLYCARP.eng.remedy.com">
<properties vendor="" product="" version="" copyright="" guid=""/>
<source-control version="" locked-by="" last-modified-by="" location="" timestamp="0"></source-control>
</filters>
</filters-list>
<escalations-list>
<escalations name="PackingListTestFilter" server="POLYCARP.eng.remedy.com">
</escalations>
</escalations-list>
<active-link-guides-list>
<filter-guides-list>
<applications-list>
</applications-list>
</applications-list>
</filter-guides-list>
</active-link-guides-list>
</packing-lists-list>
</menus-list>
</groups-list>
</distributed-mappings-list>
</distributed-mappings-list>
</distributed-pools-list>
</distributed-pools-list>
</web-services-list>
</web-services-list>
</list>
</list>
</packinglist>
```
To save a packing list in XML format

1. Create and save a packing list as explained in “Creating packing lists” on page 340.

2. In the Packing List window, choose Packing List > Generate XML to open the Save Packing List dialog box.

3. Specify where you want the packing list saved.

4. Type a name for the file in the File Name field.

5. Click Save.

   The AR System server stores all selected definitions in a single file. If you specify an existing file name, a dialog box appears and prompts you to replace the previous version.
This section describes the AR System core fields and their uses. The following topics are provided:

- Core fields (page 354)
- Short Description field (page 358)
- Working with the Request ID field (page 359)
AR System **core fields** are a set of fields that every regular form must have. You can include these fields in other forms; if present, the fields follow the same rules and have the same meanings. The commonality gained by such a convention is useful for conceptual consistency, sharing definitions, and exchanging and merging databases.

Additional limits are placed on the core fields, including the fact that some fields are required, others are maintained by the system, and others have fixed or maximum sizes.

Core fields generally appear on every regular form to make sure that all forms share a common set of concepts. AR System automatically includes **core** fields on all regular forms. Because display-only forms and joins do not directly store data in the database, core fields are not required for these types of forms. Core fields are also not required for view and vendor forms because they map to external data sources, which might not have these fields.

Core fields help provide consistency when merging and sharing data. Core fields significantly aid in the construction of solutions based on AR System. You cannot delete core fields from regular forms—although you can modify their appearance by altering labels, adding or changing menus, altering the display type, altering their location, or hiding them from view.
The following table lists the AR System core fields.

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Request ID</td>
<td>A unique identification value for each request in the system. Groups that have neither Change nor View access to the Request ID field will not have access to any other form information, regardless of the permission settings of the other fields. Data Type: Character Length: 5–15 For join forms, there is no limit to the number of layers of joins that AR System supports, so the Request ID field of a join form will contain more than 15 characters. See “Joining three or more forms” on page 114 for more information. <strong>Note:</strong> Do not change the QBE Match setting to Equal for the Request ID field. Because AR System adds a prefix and a series of zeros to Request IDs before it begins a search, users will not be able to run valid QBE searches against Request ID numbers if you set the QBE Match setting to Equal or Leading.</td>
</tr>
<tr>
<td>2</td>
<td>Submitter</td>
<td>The name of the AR System user who was logged in and submitted the request. This field is tied to the Submitter group when defining row-level security. For more information about row-level security, see “Controlling access by using implicit groups—Row-level security” on page 67. Submitter is a required field. Data Type: Character Length: 30</td>
</tr>
<tr>
<td>3</td>
<td>Create Date</td>
<td>The date and time at which the request was created in the system. The AR System server sets this field, and it cannot be modified. Data Type: Timestamp</td>
</tr>
<tr>
<td>ID</td>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>----</td>
<td>------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>4</td>
<td>Assigned To</td>
<td>The user who has been assigned responsibility for the request. This field is tied to the Assignee group when defining row-level security. For more information, see “Controlling access by using implicit groups—Row-level security” on page 67. Data Type: Character Length: 30</td>
</tr>
<tr>
<td>5</td>
<td>Last Modified By</td>
<td>The name of the user who last altered the request. AR System sets this field to the login name of the user who last changed the request. It cannot be modified. Data Type: Character Length: 30</td>
</tr>
<tr>
<td>6</td>
<td>Modified Date</td>
<td>The date the field was last modified. The AR System server sets this field to the time the last change to this request was made. It cannot be modified. Data Type: Timestamp</td>
</tr>
<tr>
<td>7</td>
<td>Status</td>
<td>Indicates the current state of the request. Users have control over this field. It must have a value at all times; there must be a default value in the event that the user does not specify a value when the request is created. The actual names and values of the status field can be customized. Status is a required field. Data Type: Selection</td>
</tr>
<tr>
<td>8</td>
<td>Short Description</td>
<td>A brief description of the request. A size limit forces the submitter to be concise. Short Description is a required field. Data Type: Character Maximum Length: 254</td>
</tr>
<tr>
<td>15</td>
<td>Status History</td>
<td>The user who last made a change, and the time the change was made to each of the states identified by the Status field. AR System sets and maintains this field, and it cannot be modified. Data Type: Character</td>
</tr>
</tbody>
</table>
Core field characteristics

The following core fields have special characteristics that you should consider when defining a new form.

Request ID field

AR System uses the Request ID field (Field ID 1) to provide a unique identification value for each request entering the system. It is created and maintained by the system; however, it is possible to attach a string to the value for descriptive purposes.

To improve the usability of the Request ID field, you can add a prefix to the field to make it more descriptive to your users. For example, in a distributed server environment where you are transferring tickets from Los Angeles to Chicago, you could have the system add the prefix LA to tickets generated from the server in Los Angeles, and CHI for tickets from Chicago. For more information about changing the Request ID field length, see the Database Reference guide.

The Request ID field is fundamental to access control in AR System. Without access to this field, users have no access to the request, even if they belong to groups with access to other fields on the form. Groups that have neither Change nor View access to the Request ID field will not have access to any other form information, regardless of the permission settings of the other fields. For more information, see “Using the Request ID field with implicit groups” on page 69.

Submitter field

The Submitter field (Field ID 2) defines which user created a request. Users who create a request are automatically assigned membership to the Submitter group. For more information about the role of the Submitter group in AR System, including the ability to change the contents of this field, see the Form and Application Objects guide.
Short Description field

The **Short Description** field (Field ID 8) provides a common place for users to summarize a request. By default, the Short Description field contents appear in the results pane of a form whenever a search is performed.

You might want to include a menu of possible problem and request types for the Short Description field to make request submissions easier and reporting more efficient. For more information, see Chapter 7, “Defining menus.”

Status field

The **Status** field (Field ID 7) enables you to track the different states a request moves through in its life cycle. The meaning of each individual state helps define the workflow process and you can define any number of states. In addition to keeping track of each state of a request, AR System keeps additional information with the Status field called status history. Status history includes the user name of the person who last changed the state of the request and the date and time that the change occurred.

Define states carefully. The Status field is the key field that represents the problem resolution process. The states must capture the important steps in the process, although not all states might be used during the life cycle of a single request. A good process is often represented by four or five states.

It is difficult to modify the Status field choices after users have begun to use the form, because the data for a selection field is stored in the database as an integer that relates to the order of the choices. For more information about selection fields, see “Selection fields” on page 151.

Assigned To field

The **Assigned To** field (Field ID 4) enables ownership of each request to be tracked. If requests are designed to pass ownership from one user to another, create workflow that uses the Assigned To field. Users who are assigned ownership to a request are automatically assigned membership in the Assignee group. For more information about the role of the Assignee group in AR System, see “Reserved groups in AR System” on page 50.
Working with the Request ID field

Every form defined in AR System contains a set of core fields. The Request ID field is one of those core fields and has a unique field ID of 1. You can change the label of this field to something other than Request ID, but the field ID will always remain 1.

The Request ID field contains a character string that holds a unique index for each request. The form of this string is an optional prefix, which can consist of any alphanumeric characters, followed by a 0-padded numeral (for example, `HD0000000000001`). The length of the Request ID field must be either 1 or between 5 and 15 characters, inclusive. Specifying a length of 1 causes leading zeroes to be stripped from the value in the Request ID. The prefix can be as long as the total length of the field less five characters.

When new requests are submitted, AR System generates a new ID for the request by appending the next available ID to the prefix, if a prefix is specified. Then, AR System increments the next available ID in preparation for the next request to be submitted.

The Request ID field contains a unique number sequence. Create other fields to contain information that is specific to your site instead of using the Request ID field. Overloading the Request ID field with other information can restrict your ability to control this data and limits the flexibility of searches on the data.

Sometimes, it is necessary to work with the Request ID value or the Request ID field in the database. The following section contains these topics:

- “Changing the next available ID for new requests” on page 360.
- “Changing the Request ID field length or prefix” on page 362.
- “Preserving existing Request ID field values” on page 363.
- “Changing existing Request ID field values to a new format” on page 363.
- “Updating the Request ID field in other AR System tables” on page 373.
Changing the next available ID for new requests

You can change the next available ID when creating new AR System requests. This ID is used to automatically generate the unique index number that is attached to each AR System request. Under some conditions, you might need to reset the next available ID. For example, you might need to establish different ranges for a similar form on two different servers, or you might need to reserve a range of numbers for later use.

Note: Do not change the next available ID to a number lower than the greatest existing ID. The Request ID field value must be unique within AR System, and resetting the ID to a lower number could conflict with existing Request ID field values. If you try to submit a request with an existing ID, AR System will return an error and prevent the request from being submitted until the conflict is resolved.

If you must change the next available ID, make the change when the system is not in use to avoid conflicts with users who are submitting new requests.

Changing the next available ID for SQL databases

To change the next available ID for a form in an SQL database

1. Stop AR System server.
2. Using any front-end tool that allows direct access to an SQL database, log in as a user with write access to the AR System tables.
3. Connect to the AR System table area.
4. Find the Request ID field for the form you want to modify.
5. Update the next available ID.
6. Restart the AR System server.

Example SQL database procedures

The following sections are examples of how to change the next available ID for DB2 Universal, Informix, Oracle, and Microsoft SQL and Sybase databases. In these examples, the next available ID for a form named ZZZ is changed from the current value of 1291 to a new value of 25000.
DB2 Universal database example
>open DB2 command center
Connect to AR System.
>select name, nextId from arschema where name = 'ZZZ';
NAME    NEXTID
--- -------
ZZZ    1291
1 row(s) retrieved.
>update arschema set nextId = 25000 where name = 'ZZZ';
1 row(s) updated.

Informix database example
% dbaccess ...
>database ARSystem;
Database selected.
>select name, nextId from arschema where name = 'ZZZ';
name    nextId
ZZZ    1291
1 row(s) retrieved.
>update arschema set nextId = 25000 where name = 'ZZZ';
1 row(s) updated.
><Control-C>

Oracle database example
% sqlplus
Enter user-name: ARAdmin
Enter password: <password> (AR#Admin# by default.)
SQL>select name, nextId from ARAdmin.arschema where name = 'ZZZ';
NAME    NEXTID
--- -------
ZZZ    1291
1 row updated.
SQL>Commit;
commit complete
SQL>exit

Microsoft SQL Server and Sybase database example
% isql -Usa
Password: <password>
1>use ARSystem
2>go
1>select name, nextId from arschema where name = 'ZZZ'
2>go
name    nextId
ZZZ    1291
1 row affected
1>update arschema set nextId = 25000 where name = 'ZZZ'
2>go
Changing the Request ID field length or prefix

After using a form for a while, you might need to change the prefix or length of the Request ID field, the key field in a form. Often, this change can be made and existing requests can retain the format used previously. However, you might need to convert existing Request ID field values to match the new prefix or length. This section offers background information and procedures to help you make changes to the Request ID.

To change the length of the request ID field

1. Log in to BMC Remedy Administrator as a user with administrator access.
2. Open the form you want to alter.
3. Double-click the Request ID field.
   The Field Properties dialog box appears.
4. Click the Database tab and specify the desired length in the Input Length field.

   **Note:** The length of the Request ID field must be either 1 or between 5 and 15 characters, inclusive. If you specify 1, leading zeroes are stripped from the value the Request ID field. If you specify a prefix for the Request ID field, the field must be at least five characters greater than the prefix.

5. Save the changes to the form.

To change the prefix of the request ID field

1. Log in to BMC Remedy Administrator as a user with administrator access.
2. Open the form you want to alter.
3. Double-click the Request ID field.
   The Field Properties dialog box appears.
4. Click the Attributes tab.
5. Specify the desired prefix in the Default Value field.
Note: The Request ID field must be between 5 and 15 characters in length. If you specify a prefix for the Request ID field, the field length must be at least five characters greater than the prefix.

6 Save the changes to the form.

Preserving existing Request ID field values

You might want to preserve the existing Request ID field values of your AR System requests for the following reasons:

- **Backward compatibility** — You might have cross-references that see requests by the Request ID field value.
- **History** — The Request ID field values were created with the old format, and there is no need for change.
- **Design** — The design of your AR System calls for periodic change to the Request ID field. For example, you might use the current year as a prefix for the Request ID field.
- **No data** — No requests have been submitted, so there are no Request ID fields to be converted.

Changing existing Request ID field values to a new format

You might want to change the values of existing Request ID fields for your AR System requests for any of the following reasons:

- **Consistency** — All the Request ID field values for a form follow the same format. If the format changes, all the requests change to match the format.
- **Design** — The design of your AR System has changed, and this design references the new format of the Request ID field. This is usually a change of the length of the field from a default setting of 15 to something shorter, and you need to eliminate the extra leading zeros.

This section explains two methods of updating existing Request ID field values:

- “Using an AR Export (.arx) file” on page 364.
- “Using SQL commands to shorten the Request ID field” on page 366.
After implementing one of the strategies in this section, read “Updating status history tables” on page 373 and “Updating attachment tables” on page 373.

**WARNING:** Back up your database before performing the actions described in this section to make sure your original data is saved if there is a failure during the update.

### Using an AR Export (.arx) file

You can edit AR Export (.arx) files regardless of the database underlying your AR System. You can use the .arx strategy or a different strategy that bypasses AR System to operate directly in the database.

1. In BMC User, open the form you want to change.
2. Choose Tools > Reporting.
   - The Report dialog box appears.
   - The Properties - << New Style >> dialog box opens.
4. Click Add All to add all the fields to the report style.
5. Select the Request ID field under Selected Fields and move it to the top of the list.
6. Choose Report > Export To > File and use .arx format to save all the data for the form to a file.
7. Close the Report dialog box.
8. Edit the file to change the format of the Request ID field. See “Editing the .arx file” on page 365.
9. In BMC User, delete all requests in the form.
11. Open BMC Remedy Import.
Choose File > Open Import File.
The Open Import File dialog box appears.

Select the file you edited, and click Open.

Choose File > Open Form.
The Open Form dialog box appears.

Select the form you want to change and click OK.

Click Add All.

Choose File > Preferences.
The Preferences dialog box appears.

Click the Data tab. Disable fields’ pattern matching and make required fields optional during import by selecting the check boxes.

Click the Duplicate Request ID tab, and select the Reject Duplicate Record option.

Click OK to close the Preferences dialog box.

Choose Import to start the import process.

**Editing the .arx file**

After the first few header lines, the remaining lines in the .arx file have the following format:

```
DATA "000000000000001" "<other_data>" 1 "<other_data>"
```

where `<other_data>` is data from the form.

The Request ID field always follows the keyword `DATA`. In this example, the Request ID field has no prefix and is 15 characters in length. Use a text editor, such as WordPad, to convert the format of the Request ID field.

The following procedures show how you can shorten a Request ID field with a length of 15 characters to 10 characters and add a prefix of ABC.
To edit the .arx file in Windows

1. Open the .arx file in a text editor that has a Find/Replace command with a feature for matching case (for example, WordPad).

2. Use the Find/Replace command to search for DATA "00000000.

   Note: This command contains 8 zeros. Five of these represent the difference between the original length of 15 characters and the new length of 10 characters. The other 3 zeros represent the spaces to be replaced by ABC.

3. Use the match case feature.

4. Replace all instances of DATA "00000000 with DATA "ABC.

5. Save the changes to the file.

To edit the .arx file in UNIX

1. Open the file in a text editor.

2. Type the following command:

   g /^DATA "00000000/s//DATA "ABC/

3. Save and close the file.

Using SQL commands to shorten the Request ID field

Only administrators running AR System with an SQL database can update existing request ID field values by directly accessing the SQL database. The syntax for direct access is different for each SQL database that AR System supports. These commands are described in the examples in this section.

To use the methods described in this section, you must be familiar with basic commands in the SQL command interface. SQL commands bypass AR System completely. If you bypass AR System, verify that all data is valid when you are finished.

Tip: Create a practice table in your database and practice the commands you will issue to make sure that you are issuing the correct commands. Make sure you back up your database or all the relevant tables.
Note: When you change the length of the Request ID field in a database table, all related database tables, such as status history tables (H Tables), and Attachment tables (B Tables and BC Tables) must also be updated.

Important: Stop AR System before you attempt any database modifications.

Finding the name of the table

Before you can shorten the request ID field value, you must find the table holding the form being changed. To find the table name, you must know the schema ID and, in some cases, the field ID of the field to be changed.

To find the table name, follow these basic steps:

**Step 1** Find the correct schema ID for your form.

**Step 2** Find the correct field ID for your field, if necessary.

**Step 3** Construct the name of the table using the schema ID and field ID you found in the previous steps.

---

**To find the correct schema ID for your form**

- Perform the following query:

```sql
Select SchemaId, name from arschema order by 2
```

This query returns a list of schema IDs and associated form names.

---

**To find the correct field ID (after you know the schema ID)**

- Perform the following query. This example assumes that the schema ID is 43:

```sql
Select FieldId, FieldName from field where SchemaId = 43
```

This query returns a list of field IDs and associated field names.
To construct the name of a table from schema ID and field ID

Use the schema ID, field ID, and information in the following table to construct your table name.

**Table A-1: AR System table name constructs**

<table>
<thead>
<tr>
<th>Table name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>T&lt;schema_ID&gt;</td>
<td>A table that contains the data in your form. A table named T43 indicates that 43 is the schema ID.</td>
</tr>
<tr>
<td>T&lt;schema_ID&gt;C&lt;field_ID&gt;</td>
<td>(Oracle only) Used for backward compatibility with forms created using AR System versions prior to 4.5. It is a table that contains long text and diary data. For example, a table named T43C536870924 indicates that 43 is the schema ID and 536870924 is the field ID. In many cases, there will be more than one long text or diary field on the form.</td>
</tr>
<tr>
<td>H&lt;schema_ID&gt;</td>
<td>A table that contains the Status History information for your form. A table named H43 indicates that 43 is the schema ID. See the Database Reference guide for information about the Status History table.</td>
</tr>
<tr>
<td>B&lt;schema_ID&gt;</td>
<td>A table that contains a list of all the attachments and related information for each record in your form. A table named B43 indicates that 43 is the schema ID. See the Database Reference guide for information about attachment tables.</td>
</tr>
<tr>
<td>B&lt;schema_ID&gt;C&lt;field_ID&gt;</td>
<td>A table that contains the actual Binary objects for attachment fields in your form. A table named B43C536870924 indicates that 43 is the schema ID and 536870924 is the field ID. In this example, the field ID for the attachment field is 536870924. In some cases, there will be more than one attachment field on the form.</td>
</tr>
</tbody>
</table>

**Note:** For T<schema_ID> and B<schema_ID> tables, the request ID column of the table is always named C1. For the H<schema_ID> tables, T<schema_ID>C<field_ID> tables, and B<schema_ID>C<field_ID> tables, the Entry ID column is equivalent to the C1 column.
Changing existing Request ID field value format when the Request ID has a prefix

The following examples assume that the table is named T43, that the prefix is HD, and that the field size (including the prefix) will be 8 characters. The 6 represents the number of characters to keep, starting from the right side of C1. C1 is originally 15 characters long. Make sure that the number of characters in your prefix plus the second parameter in the RIGHT function is equal to the new size of the C1 field.

**DB2 database examples**

To add a prefix to the T<schema_ID> table, use the following syntax:

```
update T43 set C1 = 'HD' || RIGHT(C1, 6)
```

To add a prefix to the B<schema_ID> table, use the following syntax:

```
update B43 set C1 = 'HD' || RIGHT(C1, 6)
```

For the H<schema_ID> table, use the following syntax:

```
update H43 set entryId = 'HD' || RIGHT(entryId, 6)
```

For the B<schema_ID>C<field_ID> tables, use the following syntax:

```
update B43C536870924 set entryId = 'HD' || RIGHT(entryId, 6)
```

**Informix database examples**

In the following examples, the request ID is being shortened from 15 to 8 characters. The prefix HD is concatenated to the last 6 characters in the string, consisting of positions 10 through 15.

Note that for Informix databases, you must log in as the root user.

To add a prefix to the T<schema_ID> table, use the following syntax:

```
update T43 set C1 = 'HD'||C1[10,15]
```

To add a prefix to the B<schema_ID> table, use the following syntax:

```
update B43 set C1 = 'HD'||C1[10,15]
```

For the H<schema_ID> table, use the following syntax:

```
update H43 set entryId = 'HD'||entryId[10,15]
```

For the B<schema_ID>C<field_ID> tables, use the following syntax:

```
update B43C536870924 set entryId = 'HD'||entryId[10,15]
```

**Note:** In the functions C1[10,15] and entryId[10,15], the 10 represents the starting position of the characters to keep and 15 represents the ending position.

---

Form and Application Objects
To add a prefix to the T<schema_ID> table, use the following syntax:
update T43 set C1 = 'HD'||substr(C1,10,6);

To add a prefix to the B<schema_ID> table, use the following syntax:
update B43 set C1 = 'HD'||substr(C1,10,6);

For the H<schema_ID> table, use the following syntax:
update H43 set entryId = 'HD'||substr(entryId,10,6);

For the B<schema_ID>C<field_ID> tables, use the following syntax:
update B43C536870924 set entryId = 'HD'||substr(entryId,10,6);

For the T<schema_ID>C<field_ID> tables, use the following syntax:
update T43C536870924 set entryId = 'HD'||substr(entryId,10,6);

Note: In the functions substr(C1,10,6) and substr(entryId,10,6), the 10 represents the starting position of the characters to keep and the 6 is the number of characters to keep.

To add a prefix to the T<schema_ID> table, use the following syntax:
update T43 set C1 = "HD"+ RIGHT(C1, 6)

To add a prefix to the B<schema_ID> table, use the following syntax:
update B43 set C1 = "HD" + RIGHT(C1, 6)

For the H<schema_ID> table, use the following syntax:
update H43 set entryId = "HD" + RIGHT(entryId, 6)

For the B<schema_ID>C<field_ID> tables, use the following syntax:
update B43C536870924 set entryId = "HD" + RIGHT(entryId, 6)

Changing existing Request ID field value format when the Request ID does not have a prefix

The following examples assume that the table is named T43 and that the field size will be 8 characters. The 8 represents the number of characters to keep, starting from the right side of C1. C1 is originally 15 characters long. Make sure that the number of characters in the second parameter in the RIGHT function is equal to the new size of the C1 field and that the sum of the two numeric values in the SUBSTR function is 16 (1 greater than the original length of C1).
To add a prefix to the \texttt{T<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update T43 set C1 = RIGHT(C1, 8)
\end{verbatim}

To add a prefix to the \texttt{B<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update B43 set C1 = RIGHT(C1, 8)
\end{verbatim}

For the \texttt{H<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update H43 set entryId = RIGHT(entryId, 8)
\end{verbatim}

For the \texttt{B<schema\_ID>C<field\_ID>} tables, use the following syntax:
\begin{verbatim}
update B43C536870924 set entryId = RIGHT(entryId, 8)
\end{verbatim}

For Informix databases, you must log in as the root user.
In the following examples, the Request ID is being shortened from 15 to 8 characters. The request ID will consist of the last 8 characters in the string, consisting of positions 8 through 15.

To add a prefix to the \texttt{T<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update T43 set C1 = C1[8,15]
\end{verbatim}

To add a prefix to the \texttt{B<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update B43 set C1 = C1[8,15]
\end{verbatim}

For the \texttt{H<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update H43 set entryId = entryId[8,15]
\end{verbatim}

For the \texttt{B<schema\_ID>C<field\_ID>} tables, use the following syntax:
\begin{verbatim}
update B43C536870924 set entryId = entryId[8,15]
\end{verbatim}

\textbf{Note:} In the functions \texttt{C1[8,15]} and \texttt{entryId[8,15]}, the 8 represents the starting position of the characters to keep and 15 represents the ending position.

To add a prefix to the \texttt{T<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update T43 set C1 = substr(C1,8,8);
\end{verbatim}

To add a prefix to the \texttt{B<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update B43 set C1 = substr(C1,8,8);
\end{verbatim}

For the \texttt{H<schema\_ID>} table, use the following syntax:
\begin{verbatim}
update H43 set entryId = substr(entryId,8,8);
\end{verbatim}

For the \texttt{B<schema\_ID>C<field\_ID>} tables, use the following syntax:
update B43C536870924 set entryId = substr(entryId,8,8);

For the $T<$schema_ID>$C<$field_ID>$ tables, use the following syntax:
update T43C536870924 set entryId = substr(entryId,8,8);

**Note:** In the functions `substr(C1,8,8)` and `substr(entryId,8,8)`, the first 8 represents the starting position of the characters to keep, and the second 8 is the number of characters to keep.

**Microsoft SQL Server and Sybase database examples**

To add a prefix to the $T<$schema_ID>$ table, use the following syntax:
update T43 set C1 = RIGHT(C1, 8)

To add a prefix to the $B<$schema_ID>$ table, use the following syntax:
update B43 set C1 = RIGHT(C1, 8)

For the $H<$schema_ID>$ table, use the following syntax:
update H43 set entryId = RIGHT(entryId, 8)

For the $B<$schema_ID>$C<$field_ID>$ tables, use the following syntax:
update B43C536870924 set entryId = RIGHT (entryId, 8)

**Using SQL commands to lengthen the Request ID field value**

The format for all the supported databases is the same for lengthening the Request ID field format as with shortening the Request ID field format. See “Using SQL commands to shorten the Request ID field” on page 366 for hints about how to run the SQL interface, how to find the name of the table to be changed, and how to exit the SQL interface.

**Note:** The maximum length allowed for the Request ID field is 15 bytes.

In the following example, the length of the field is restored to 15 characters from the current 10 characters by adding 5 leading zeros to the existing value of the Request ID field and assigning the resulting 15-character string to the Request ID field. When you have determined the name of the table ($T43$ in the example), issue the one of the following commands at the prompt:

- **DB2**
  
  % update T43 set C1 = '00000' || C1

- **Informix**
  
  % update T43 set C1 = '00000' || C1
Working with the Request ID field

- Oracle

  \% update T43 set C1 = '00000' || C1

- Sybase and Microsoft SQL Server

  \% update T43 set C1 = '00000' + C1

If you want to add a prefix, specify the prefix as part of the string to be added. For example, if you want to expand to 15 characters and add a prefix of ABC, use 'ABC00' instead of '00000' in the preceding example.

Updating the Request ID field in other AR System tables

When you change the Request ID in a main data table, you must also consider whether you need to make a similar change to the status history table and attachment tables.

Updating status history tables

Status History information is stored in a separate table. This table uses the Request ID field as the link to the main table. Accordingly, you use the same procedure to change the Request ID field values in the status history table as you do in other tables.

To update the status history table, use the commands described in the previous examples, substituting H43 for T43 and entryId for C1. For more information, see “AR System table name constructs” on page 368.

For more information about the status history table, see the Database Reference guide.

Updating attachment tables

Attachment information is stored in two tables, the Attachment Details table and the Attachment Data table. The Attachment Details Table holds attachment characteristics, such as the name and size of the attachment, and the Attachment Data table holds the actual attachment. These tables also use the Request ID field as the link to the main table. Accordingly, you use the same procedure to change the Request ID field values in the status history table as you do in other tables.
The Attachment Details table is named with a B followed by the schema ID (for example, B3). The Attachment Data table is named with a B followed by the schema ID, followed by C, followed by the attachment field ID. For example, the Attachment Data table might be called B7C536870920, where 7 is the schema ID, and 536870920 is the attachment field ID.

The column holding the Request ID in the Attachment Details table is named C1, and in the Attachment Data table, it is named entryId. To update the Request ID field in the attachment tables, use the commands described in the previous examples, substituting the appropriate table name, and using C1 or entryId for the Request ID. For more information, see “AR System table name constructs” on page 368.

For more information about attachment tables, see the Database Reference guide.
AR System **reserved fields** are special-purpose data fields. Some of these fields are used in the User or Group forms. Others are used for assignee group access, the Distributed Server Option, web applications, or localization. You can use the reserved fields in your forms. When these fields are used in your forms, they retain the special meaning and use as defined in this section.

AR System contains fields that are reserved for system use. If you create fields with these IDs, certain actions will automatically take place. This section lists the ranges of reserved fields and a description of the fields.

The following topics are provided:

- Reserved field ranges (page 376)
- Reserved IDs used as placeholders in definitions (page 377)
- Reserved fields in access control (page 378)
- Localization reserved field (page 383)
- DSO reserved fields (page 383)
- Form action reserved fields (page 386)
Reserved field ranges

AR System has special ranges of field IDs. Numbers 1–99 are reserved for core fields. Numbers 100–536870911 are reserved for registered fields. The following table displays reserved field ranges that forms and applications use in AR System.

<table>
<thead>
<tr>
<th>ID Range</th>
<th>Type of Fields, Forms, or Applications That Use the Field IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–99</td>
<td>Core fields</td>
</tr>
<tr>
<td>101–149</td>
<td>Access control fields</td>
</tr>
<tr>
<td>150–159</td>
<td>AR System Message Catalog</td>
</tr>
<tr>
<td>200–399</td>
<td>Distributed Server Option forms</td>
</tr>
<tr>
<td>450–469</td>
<td>Audit fields</td>
</tr>
<tr>
<td>600–699</td>
<td>Business Time forms</td>
</tr>
<tr>
<td>700–750</td>
<td>Alert forms</td>
</tr>
<tr>
<td>800–810</td>
<td>Server Events form</td>
</tr>
<tr>
<td>820–880</td>
<td>Business Time forms</td>
</tr>
<tr>
<td>900–999</td>
<td>Server Statistics form</td>
</tr>
<tr>
<td>1000–1100</td>
<td>Form Action fields</td>
</tr>
<tr>
<td>1101–1399</td>
<td>Application Statistics forms</td>
</tr>
<tr>
<td>1500–1525</td>
<td>Currency forms</td>
</tr>
<tr>
<td>1700–1799</td>
<td>Roles form</td>
</tr>
<tr>
<td>2000–2299</td>
<td>Application States</td>
</tr>
<tr>
<td>2300–2699</td>
<td>Business Time forms</td>
</tr>
<tr>
<td>10000–14999</td>
<td>Approval Server Option</td>
</tr>
<tr>
<td>15000–15999</td>
<td>Enterprise Integration Engine</td>
</tr>
<tr>
<td>17000–17399</td>
<td>Reporting forms</td>
</tr>
<tr>
<td>20000–39999</td>
<td>Preference forms</td>
</tr>
<tr>
<td>40000–40499</td>
<td>Flashboards</td>
</tr>
<tr>
<td>60000–60999</td>
<td>Dynamic group fields</td>
</tr>
</tbody>
</table>
Reserved IDs used as placeholders in definitions

Many workflow definitions reference field IDs. In several places, a special reserved ID is used to request a special operation. These fields are described in the following table.

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 97 | Set to Defaults | Used in Open Windows action to indicate that fields on the opened form should be set to their default settings.  
Data Type: Not applicable |
| 98 | LikeID | Used in Push Fields and Set Fields actions to indicate mapping of like IDs.  
Data Type: Character |
| 99 | Weight | Number between 1 and 100 indicating the quality of a match. The closer to 100, the better the match.  
Data Type: Integer |
| 1576 | AppSubset | Used to display a subset of entry points. This field is hidden on the default Home Page form installed with the AR System. Create this field as necessary on your own Home Page form.  
Data Type: Character |
Reserved fields in access control

The User, Group, and Roles forms contain the reserved fields shown in the following table. These fields are involved in access control.

The table lists the form on which the field is defined, but you can add these fields to any form by creating the field and specifying its reserved ID. See Chapter 2, “Defining access control,” for more information about access control.

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Form</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>Login Name</td>
<td>User</td>
<td>The name the user enters in the User Name field of the Login dialog box when logging in to the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character                                                                                                         Length: 30</td>
</tr>
<tr>
<td>102</td>
<td>Password</td>
<td>User</td>
<td>The password that the user enters when logging in to the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>When a user enters information into this field, the text appears as asterisks (*).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If you used an AR System version prior to 6.0 to create workflow involving a Password field (ID 102), the workflow might not function in AR System versions 6.0 and later. Version 6.0 included enhanced encryption and tighter security controls.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>To work around this issue, use the Application-Confirm-Password $PROCESS$ command. For more information about this command, see the Workflow Objects guide.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character                                                                                                         Length: 30</td>
</tr>
<tr>
<td>103</td>
<td>Email Address</td>
<td>User</td>
<td>The email address of the user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character                                                                                                         Maximum Length: 255</td>
</tr>
</tbody>
</table>
### Form and Application Objects

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Form</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>104</td>
<td>Group List</td>
<td>User</td>
<td>The list of access control groups to which the user belongs. Group names are separated by spaces. Although you make entries to the Group list by using the alias (name for a group), the group IDs are stored as integer values separated by semicolons.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Maximum Length: 4000</td>
</tr>
<tr>
<td>105</td>
<td>Group Name</td>
<td>Group</td>
<td>The alias by which the access control group is known. This is the name used in the Group list field of the User form and in the Group Permissions list of each form field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Length: 30</td>
</tr>
<tr>
<td>106</td>
<td>Group ID</td>
<td>Group</td>
<td>The ID of the group named in the Group name field. This ID should be greater than 10 for groups that you create.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Range: 0–100 (can be expanded)</td>
</tr>
<tr>
<td>107</td>
<td>Group Type</td>
<td>Group</td>
<td>The maximum permission type intended for the group named in the Group name field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Selection (None, View, Change)</td>
</tr>
<tr>
<td>108</td>
<td>Default Notification</td>
<td>User</td>
<td>The notification method used if the user specifies the default mechanism.</td>
</tr>
<tr>
<td></td>
<td>Mechanism</td>
<td></td>
<td>Data Type: Selection (None, Alert, Email)</td>
</tr>
<tr>
<td>109</td>
<td>License Type</td>
<td>User</td>
<td>The type of license that the user has.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Selection (Read, Fixed, Floating)</td>
</tr>
<tr>
<td>ID</td>
<td>Field Name</td>
<td>Form</td>
<td>Description</td>
</tr>
<tr>
<td>-----</td>
<td>-------------------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>110</td>
<td>Full Text License Type</td>
<td>User</td>
<td>The type of full text search license that the user has.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Selection (None, Fixed, Floating)</td>
</tr>
<tr>
<td>112</td>
<td>Assignee Group</td>
<td>Any</td>
<td>The groups or users assigned responsibility for the request. This field is</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>tied to the Assignee Group group when defining row-level security. For more</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>information about this type of security, see “Controlling access by using</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>implicit groups—Row-level security” on page 67.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character Maxum Length: 255</td>
</tr>
<tr>
<td>115</td>
<td>Write License Pool</td>
<td>User</td>
<td>The license pool from which floating write licenses are taken.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character Maxum Length: 30</td>
</tr>
<tr>
<td>116</td>
<td>FTS License Pool</td>
<td>User</td>
<td>The license pool from which floating full text search licenses are taken.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character Maxum Length: 30</td>
</tr>
<tr>
<td>117</td>
<td>Authentication Login Name</td>
<td>User</td>
<td>This field is used for external authentication under certain conditions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>For more information, see the Configuring guide.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character Maxum Length: 30</td>
</tr>
<tr>
<td>118</td>
<td>Authentication String</td>
<td>User</td>
<td>This field is used for external authentication under certain conditions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>For more information, see the Configuring guide.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Data Type: Character Maxum Length: 255</td>
</tr>
</tbody>
</table>
### Form and Application Objects

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Form</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>119</td>
<td>Computed Group List</td>
<td>User</td>
<td>After a search, displays the computed groups the user is associated with. Data Type: Character Maximum Length: 255</td>
</tr>
<tr>
<td>120</td>
<td>Group Category</td>
<td>Group</td>
<td>The group category, such as Regular, Dynamic, or Computed. Data Type: Selection</td>
</tr>
<tr>
<td>121</td>
<td>Computed Group Definition</td>
<td>Group</td>
<td>Boolean statement that defines a computed group. For information about computed groups, see “Groups you create—Regular, computed, and dynamic” on page 52. Data Type: Character</td>
</tr>
<tr>
<td>122</td>
<td>Application License</td>
<td>User</td>
<td>For users of licensed applications, the name of the application and the type of license. For more information about licensing applications, see the Integrating with Plug-ins and Third-Party Products guide. Data Type: Character</td>
</tr>
<tr>
<td>179</td>
<td>Unique Identifier</td>
<td>User and Group</td>
<td>This field is used internally by applications installed on top of AR System. It replaces field ID 490000000 (Instance ID) in version 6.x. See your product documentation for more information. Data Type: Character Maximum Length: 38</td>
</tr>
<tr>
<td>1700</td>
<td>Application Name</td>
<td>Roles</td>
<td>Name of the application for which the role is defined. Data Type: Character Maximum Length: 250</td>
</tr>
<tr>
<td>1701</td>
<td>Role Name</td>
<td>Roles</td>
<td>Name by which the role is known. Data Type: Character Maximum Length: 255</td>
</tr>
</tbody>
</table>
### Reserved Fields

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Form</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1702</td>
<td>Role ID</td>
<td>Roles</td>
<td>Integer ID that is the recognized identity of the role. The ID must be a negative number, such as -10001. Data Type: Integer</td>
</tr>
<tr>
<td>2001</td>
<td>Test</td>
<td>Roles</td>
<td>The regular or computed group to which you want to map the role for the Test application state. Data Type: Character Maximum Length: 255</td>
</tr>
<tr>
<td>2002</td>
<td>Production</td>
<td>Roles</td>
<td>The regular or computed group to which you want to map the role for the Production application state. Data Type: Character Maximum Length: 255</td>
</tr>
<tr>
<td>60000-60999</td>
<td>Dynamic groups</td>
<td>Any</td>
<td>The roles, groups, or users assigned responsibility for the request. This field is tied to a dynamic group when defining row-level security. For more information about this type of security, see “Controlling access by using implicit groups—Row-level security” on page 67. Data Type: Character Maximum Length: 255</td>
</tr>
<tr>
<td>4900000</td>
<td>Instance ID</td>
<td>User and Group</td>
<td>Replaced by field ID 179 (Unique Identifier) in version 6.x. This field is used internally by applications installed on top of AR System. See your product documentation for more information. Data Type: Character Maximum Length: 38</td>
</tr>
<tr>
<td>4900001</td>
<td>Object ID</td>
<td>User and Group</td>
<td>This field is used internally by applications installed on top of AR System. See your product documentation for more information. Data Type: Character Maximum Length: 38</td>
</tr>
</tbody>
</table>

---

382 ▶ Appendix B—Reserved fields
Localization reserved field

The following reserved field is used for localization.

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>Locale</td>
<td>Add this field to a form to localize search menus. The system uses this field to search for requests matching the user’s locale. For more information, see Appendix H, “Localizing AR System applications.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Character</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length: 255</td>
</tr>
</tbody>
</table>

DSO reserved fields

The Distributed Server Option can add one or more of the reserved fields shown in the following table.

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>300</td>
<td>To Mapping</td>
<td>The name of the mapping to use when transferring the request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Character</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length: 254</td>
</tr>
<tr>
<td>301</td>
<td>Transfer Status</td>
<td>The status of a distributed server transfer operation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (Success, Retry, Failure, Timeout, Canceled)</td>
</tr>
<tr>
<td>302</td>
<td>Update Status</td>
<td>The status of a distributed server update operation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (Success, Waiting, Retry, Failure, Timeout, Canceled)</td>
</tr>
<tr>
<td>303</td>
<td>Master Flag</td>
<td>A flag indicating whether the request is the master.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (No, Yes)</td>
</tr>
<tr>
<td>304</td>
<td>Current Form</td>
<td>The form in which the master copy of the request resides.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Character</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length: 254</td>
</tr>
<tr>
<td>ID</td>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----</td>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>305</td>
<td>Current Server</td>
<td>The server on which the form with the master copy of the request resides. Data Type: Character Maximum Length: 64</td>
</tr>
<tr>
<td>306</td>
<td>From Mapping</td>
<td>The name of the mapping used to transfer this request. Data Type: Character Length: 254</td>
</tr>
<tr>
<td>307</td>
<td>From Entry ID</td>
<td>The ID of the request from which this copy was transferred. Data Type: Character Length: 15</td>
</tr>
<tr>
<td>308</td>
<td>To Entry ID</td>
<td>The ID of the transferred request. Data Type: Character Length: 15</td>
</tr>
<tr>
<td>309</td>
<td>Mapping History</td>
<td>Transfer history information—the date and time of transfer, source request ID, source form, source server, and the name of the specific mapping used (created at transfer time). Data Type: Character Length: Unlimited</td>
</tr>
<tr>
<td>310</td>
<td>From Form</td>
<td>The form from which this request was transferred. Data Type: Character Length: 254</td>
</tr>
<tr>
<td>311</td>
<td>From Server</td>
<td>The server from which this request was transferred. Data Type: Character Length: 64</td>
</tr>
<tr>
<td>312</td>
<td>To Form</td>
<td>The form to which the request should be transferred. Data Type: Character Length: 254</td>
</tr>
<tr>
<td>313</td>
<td>To Server</td>
<td>The server to which the request should be transferred. Data Type: Character Length: 64</td>
</tr>
</tbody>
</table>
### Form and Application Objects

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>314</td>
<td>When to Update</td>
<td>The frequency with which to update the original request if a transferred copy is updated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (Immediately, Hourly, Daily, On Return, No Update)</td>
</tr>
<tr>
<td>315</td>
<td>Transfer Mode</td>
<td>The type of transfer to perform.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (Data Only, Data+Ownership, Independent Copy, Copy+Delete)</td>
</tr>
<tr>
<td>316</td>
<td>Duplicate Entry ID Action</td>
<td>The action to take if you transfer a request and a request with the same ID already exists in the form listed in the To Form field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (Error, Overwrite, Create New)</td>
</tr>
<tr>
<td>317</td>
<td>Max Time to Retry</td>
<td>The maximum time (in seconds) that the system should retry a distributed operation before canceling the operation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Integer</td>
</tr>
<tr>
<td>318</td>
<td>From Pool</td>
<td>DSO pool that was used to transfer the entry.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Character</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Length: 254</td>
</tr>
<tr>
<td>319</td>
<td>Pattern Match</td>
<td>Flag indicating whether to enforce field pattern matching when the entry is transferred or merged.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (No, Yes)</td>
</tr>
<tr>
<td>320</td>
<td>Required Fields</td>
<td>Flag indicating whether to enforce required fields. If a required field is empty, this enables you to transfer an entry with a NULL value.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data Type: Selection (No, Yes)</td>
</tr>
</tbody>
</table>
Form action reserved fields

The following fields are used in web applications to help users perform actions. For more information about form action fields, see the Installing and Administering BMC Remedy Mid Tier guide.

<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| 706  | Alert List | Displays the alert list.  
Data Type: Table |
| 1001 | Submit     | Performs a save operation in New mode to create a record.  
Data Type: Button |
| 1002 | Search     | Performs a search operation.  
Data Type: Button |
| 1003 | Modify     | Performs a save operation in Modify mode to save a record.  
Data Type: Button |
| 1004 | Modify All | Performs a save operation in Modify mode to save all selected records.  
Data Type: Button |
| 1005 | Query Bar  | Contains the contents of the Advanced Search Bar.  
Data Type: Character  
Length: Unlimited |
| 1006 | Clear      | Clears all data in fields on the screen.  
Data Type: Button |
| 1007 | Set to Default | Sets the form to the default settings.  
Data Type: Button |
| 1008 | Help       | Opens help for the form.  
Data Type: Button |
| 1009 | New Search | Changes the form to search mode.  
Data Type: Button |
| 1010 | New Request | Changes the form to create mode.  
Data Type: Button |
<table>
<thead>
<tr>
<th>ID</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1011</td>
<td>Show Status History</td>
<td>Displays the progress that has been made on an AR System request. Data Type: Button</td>
</tr>
<tr>
<td>1012</td>
<td>Home</td>
<td>Displays the form you have configured as your Home Page. Data Type: Button</td>
</tr>
<tr>
<td>1020</td>
<td>Results List</td>
<td>Displays results of a search. Data Type: Button</td>
</tr>
</tbody>
</table>
The AR System server requires certain system-defined forms that serve to support baseline AR System functionality such as user preferences, reporting, and access control. These system forms are automatically installed with AR System. Most cannot be deleted, and if deleted, are restored with a server restart.

- Special system forms for AR System (page 390)
## Special system forms for AR System

The following table describes forms loaded automatically during installation of the AR System server. Those forms that have web views are saved with the locale of `en_US`. If you need a web view of the form in another locale, open the web view of the form on a machine set to the locale you require, and save it.

<table>
<thead>
<tr>
<th>Form Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Group Form</strong></td>
<td>Used to create access control groups to which you grant or deny access to AR System objects. See “Creating groups” on page 78 for instructions on how to use this form.</td>
</tr>
<tr>
<td><strong>User Form</strong></td>
<td>Used to define users, their characteristics, and their access rights within AR System. See the Configuring guide for instructions on how to use this form.</td>
</tr>
<tr>
<td><strong>Roles</strong></td>
<td>Defines roles for each deployable application, and maps the roles to explicit groups on the server. You will map roles to groups for each application development state, such as Test or Production. See “Creating and mapping roles” on page 83 for instructions on how to use this form.</td>
</tr>
<tr>
<td><strong>Home Page</strong></td>
<td>Default form installed automatically with the AR System, used as a convenient starting point for administrators and site managers to display entry points.</td>
</tr>
<tr>
<td><strong>Preference Forms:</strong></td>
<td>Store user preferences centrally, providing “roaming profiles” for any AR System user. These forms are loaded when they are selected in the Select Action Request System Components dialog box during installation of the AR System server. Users can access these forms in BMC Remedy User or in a browser to view and set their preferences. In BMC Remedy User and BMC Remedy Administrator, users can also set the options in these forms by choosing Tools &gt; Options and File &gt; Preferences, respectively. See the Configuring guide for information about how to use these forms.</td>
</tr>
<tr>
<td><strong>AR System Searches Preference</strong></td>
<td>Stores searches that users can create and save for a form. Each search is an entry in this form. For more information, see the Installing and Administering BMC Remedy Mid Tier guide.</td>
</tr>
</tbody>
</table>
## Form Name | Description
--- | ---
**AR System Administration Console and other forms** | Allows you to view and modify the AR System server information about the web. This is the same information as you see in the Server Information window in BMC Remedy Administrator. For more information, see the Configuring guide.

### Reporting forms:

- **Report Form** | Links reports to forms on the same AR System server that hosts the Report Form, and provides the structures needed for granting permissions to run a report for specified groups. Administrators and individual users can submit entries to this form. See the Installing and Administering BMC Remedy Mid Tier guide for instructions on how to use this form.

- **ReportCreator Form** | Provides the interface to create and maintain AR System native report definition files. This form is a vendor form using an ARDBC plug-in. The data is actually stored in the Report form as attachments. See the Installing and Administering BMC Remedy Mid Tier guide for instructions on how to use this form. For more information about ARDBC, see the C API Reference guide.

- **ReportType Form** | Specifies how each type of report (for example, Crystal or user-defined types) is created, edited, and run. Generally, only administrators can submit or modify entries in this form, but user sessions must be able to view the entries. See the Installing and Administering BMC Remedy Mid Tier guide for instructions on how to use this form.

- **ReportSelection Form** | Used in workflow to prompt users to select a report to run. This form has no entries. See the Installing and Administering BMC Remedy Mid Tier guide for instructions on how to use this form.

**AR System Message Catalog Form** | Enables administrators to provide localized versions of error messages, help text, menus, and other text strings displayed to users in applications that are customized by locale. The use of this form can be enabled or disabled. See Appendix H, “Localizing AR System applications,” for instructions on how to use this form.
### BMC Remedy Alert forms:
- **Alert Events Form**: Contains alerts that are sent to users. If a notify action of a filter or escalation sends an alert, the alert text and reference is stored in this form. See the Configuring guide for instructions on how to use this form.
- **Alert List Form**: An Alert List form is a web view with an alert list field already created. You can add this form to your web-based applications for viewing lists of alerts on the Web.

### Server Statistics Form
Enables the server to automatically store server statistics. These statistics can then be graphically displayed by client programs such as Flashboards and used to analyze server performance. See the Optimizing and Troubleshooting guide and the C API Reference guide for instructions on how to use this form.

<table>
<thead>
<tr>
<th>Form Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server Events Form</strong></td>
<td>Contains a record of internal events for a particular server. Event types that can be recorded include server structure changes, user and group changes, and server setting changes. Options for recording server events are set in BMC Remedy Administrator. See the Configuring guide for instructions on how to use this form.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>BMC Remedy Alert forms:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Alert Events Form</strong></td>
</tr>
<tr>
<td><strong>Alert List Form</strong></td>
</tr>
</tbody>
</table>

| **inetorgperson** | An example form based upon an industry standard directory service user schema. Each field on the form references an attribute defined by the inetorgperson object class. |

### Application forms:
- **Application Pending**: Used for queuing processes and requests. This form works with the Dispatcher thread, which routes requests to the appropriate queues. The Dispatcher wakes up the process that the Application Pending request indicates requires execution. For more information about the Dispatcher thread, see the Concepts guide.

- **Application Statistics**: Enables you to monitor and analyze the performance of your deployable applications and forms.
  - For deployable applications, logs entry, filter, and escalation statistics for all forms participating in the application statistics.
  - Also logs application licensing statistics.
  - For forms, logs entry, filter, and escalation statistics.
  See the Optimizing and Troubleshooting guide for instructions on how to use this form.
### Form and Application Objects

<table>
<thead>
<tr>
<th>Form Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Statistics Configuration</td>
<td>Verifies logging settings, or to change application and form statistics logging settings. See the Optimizing and Troubleshooting guide for instructions on how to use this form.</td>
</tr>
<tr>
<td>AR System Application State</td>
<td>Defines the development state (such as Test or Production) for a deployable application. You can edit the entries in this form in the user client or create workflow that acts upon this form to change the application's state. Changing the state changes the access permissions to the application (and to objects owned by the application) according to the role-group mappings defined for each state in the Roles form. See “Working with deployable application states” on page 30 for more information about application states.</td>
</tr>
<tr>
<td>Currency forms:</td>
<td></td>
</tr>
<tr>
<td>AR System Currency Codes Form</td>
<td>This form holds the currency types that are available on a server. Each currency code can be activated or inactivated by checking the Active field on the AR System Currency Code form. Activating a currency makes it available to the clients.</td>
</tr>
<tr>
<td>AR System Currency Label Catalog Form</td>
<td>An entry in this form overrides the currency labels that appear in the Currency Pickers (the menus associated with currency fields) of BMC Remedy User and a browser.</td>
</tr>
<tr>
<td>AR System Currency Localized Labels Form</td>
<td>A join form that clients query to retrieve overridden currency labels. There is no interaction with this form.</td>
</tr>
<tr>
<td>AR System Currency Ratios Form</td>
<td>Form that holds the ratios for converting one currency type to another. There will be a ratio for both directions, for example from USD to Euro and from Euro to USD, because these conversion rates are sometimes different. The table can store current conversion rates, as well as historical ones.</td>
</tr>
<tr>
<td>LDAP forms:</td>
<td></td>
</tr>
<tr>
<td>Configuration ARDBC</td>
<td>The ARDBC Configuration form and plug-in is required for the ARDBC and AREA LDAP plug-ins. This form is used to get and set parameters stored in the ar.cfg (or ar.conf) file. For more information about ARDBC, see the Integrating with Plug-ins and Third-Party Products guide as well as the C API Reference guide.</td>
</tr>
</tbody>
</table>
### Forms used for the search database:
- object_search_admin
- object_search_details
- object_search_ref

<table>
<thead>
<tr>
<th>Form Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREA LDAP Configuration Form</td>
<td>Form through which the administrator can view and modify the parameters for the AREA LDAP plug-in. The parameters are used to query the LDAP enabled directory service for authentication purposes and user information.</td>
</tr>
<tr>
<td>ARDBC LDAP Configuration Form</td>
<td>Form through which administrators can view and modify the parameters for the ARDBC LDAP plug-in. The parameters are used to establish connections with LDAP-enabled directory services.</td>
</tr>
<tr>
<td>Data Visualization forms:</td>
<td>Forms used to set up a data visualization module to display graphical data in a field on a form. For more information, see the Integrating with Plug-ins and Third-Party Products guide.</td>
</tr>
<tr>
<td>Business Time forms:</td>
<td>Forms used to define periods of availability and unavailability, workdays, and holidays to calculate business schedules. For more information, see the Configuring guide.</td>
</tr>
<tr>
<td>Business Segment-Entity Association</td>
<td></td>
</tr>
<tr>
<td>Business Time Holidays</td>
<td></td>
</tr>
<tr>
<td>Business Time Segment</td>
<td></td>
</tr>
<tr>
<td>Business Time Shared Entity</td>
<td></td>
</tr>
<tr>
<td>Business Time Workdays</td>
<td></td>
</tr>
<tr>
<td>Flashboards forms:</td>
<td>Forms used to create and add flashboards to a form. For more information, see the Administering BMC Remedy Flashboards guide.</td>
</tr>
<tr>
<td>FB:Alarm Events</td>
<td></td>
</tr>
<tr>
<td>FB:DataSourceVariables</td>
<td></td>
</tr>
<tr>
<td>FB:Variable Attributes</td>
<td></td>
</tr>
<tr>
<td>Forms used for the search database:</td>
<td>Forms used to set up the search database, which enables you to see which workflow is related to fields in a form. For more information, see the Optimizing and Troubleshooting guide.</td>
</tr>
<tr>
<td>Form Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>Assignment Engine forms:</strong></td>
<td></td>
</tr>
<tr>
<td>ASE: Assignment Association Form</td>
<td>Forms used to run the Assignment Engine. For more information, see the Configuring guide.</td>
</tr>
<tr>
<td>ASE: Process Rule Form</td>
<td></td>
</tr>
<tr>
<td>Assignment Engine Administration</td>
<td></td>
</tr>
<tr>
<td>Assignment Forms</td>
<td></td>
</tr>
<tr>
<td>Assignment Processes</td>
<td></td>
</tr>
<tr>
<td>Assignment Rules</td>
<td></td>
</tr>
<tr>
<td>Search Rules</td>
<td></td>
</tr>
<tr>
<td><strong>Email Engine forms:</strong></td>
<td></td>
</tr>
<tr>
<td>AR System Email Mailbox Configuration Form</td>
<td>Used for configuring all AR System mailboxes.</td>
</tr>
<tr>
<td>AR System Email Templates Form</td>
<td>Stores the email templates.</td>
</tr>
<tr>
<td>AR System Email User Instruction Templates Form</td>
<td>Used to create additional “User defined Instructions” based on templates defined in the AR System Email Templates Form.</td>
</tr>
<tr>
<td>AR System Email Error Logs Form</td>
<td>Used for storing status and error information.</td>
</tr>
<tr>
<td>AR System Email Security Form</td>
<td>Used to store configuration information for Security Keys used in conjunction with incoming email.</td>
</tr>
<tr>
<td>AR System Email Messages Form</td>
<td>Used for sending emails through a specified mailbox and as a repository for all incoming emails. Outgoing and incoming messages are both stored in this form.</td>
</tr>
<tr>
<td>AR System Email Attachments Form</td>
<td>Used for storing attachments for an email and for templates. Attachments for incoming messages are also stored in this form. The system associates the attachment with a specific email in the AR System Email Association form.</td>
</tr>
<tr>
<td>AR System Email Instructions Form</td>
<td>Stores the instructions that have been extracted from an incoming email by the parsing engine.</td>
</tr>
<tr>
<td>AR System Email Instruction Parameters Form</td>
<td>Contains information that defines any parameters required by an instruction, defined in the AR System Email Instructions form. This form contains fields that define the parameter type, such as a field or form name, and the associated value as extracted from the incoming email.</td>
</tr>
</tbody>
</table>
### Form Name | Description
--- | ---
AR System Email Association Form | This form is used exclusively to associate either an email message (in the AR System Email Messages form) with one or more attachments (in the AR System Email Attachment form), or to associate a template (in the AR System Email Templates form) with one or more attachments (in the AR System Email Attachment form).
AR System Email Attachment Join Form | This form is used by the table fields located on the AR System Email Messages form and the AR System Email Templates form to display attachment information. It ensures that the workflow for the forms works correctly, enabling you to add, delete, or modify attachments through the forms without having to access the AR System Email Association form.

**DSO forms:**

- Distributed Mapping Form: Defines and maintains parameter and data control values for a specific distributed mapping.
- Distributed Pending Form: Maintains a queue of pending distributed transfers, updates, returns, and deletes.
- Distributed Pool Form: Defines and maintains definitions of specific distributed pools.
For each field, you use a Field Properties window to define properties that determine how a field will look and perform during operations performed in BMC Remedy User. The Field Properties window contains various tabs depending on the type of field you are creating or modifying.

This appendix describes each of the tabs and organizes them in alphabetical order. The following topics are provided:

- Advanced properties (page 398)
- Advanced Display properties (for tables) (page 400)
- Allowable Types properties (currency fields only) (page 403)
- Attach Fields properties (page 405)
- Attributes properties (page 406)
- Change History properties (page 417)
- Color/Font properties (page 417)
- Database properties (page 420)
- Database properties (for attachment fields) (page 427)
- Display properties (page 428)
- Functional Types properties (currency fields only) (page 437)
- Help Text properties (page 438)
- Image properties (page 439)
- Navigation Items properties (page 441)
- Pages properties (page holder fields only) (page 443)
Advanced properties

For view and application list fields, you can use the Advanced tab to specify border and scroll bar parameters. For more information about view fields and application list fields, see “Creating view fields” on page 199 and “Creating application list fields” on page 202.

For data visualization fields, you can define the module type that you want to display (report or Flashboards). For more information about data visualization fields, see “Data visualization fields” on page 166. For information about flashboards, see Administering BMC Remedy Flashboards.
### Field Properties—Advanced tab

**Module Type**
The module type for the data visualization. The options are:
- Flashboard
- Report
- <Modules that you install> (See the Integrating with Plug-ins and Third-Party Products guide for information about installing modules for data visualization fields.)

**Server**
The AR System server that contains the data visualization module.

**Definition Name**
The definition name for the data visualization module.

**Border**
Identifies how borders are shown:
- **Default**—The field border is displayed when only the content of the view field is a URL. The Default option works only with web-based clients. Selecting this option for native-based clients is equivalent to selecting the Show option.
- **Hide**—The field border is not displayed.
- **Show**—The field border is displayed.

**Scroll Bar**
Identifies how scroll bars are shown:
- **Auto**—Show the scroll bars only when the view field content does not fit completely within the field.
- **Show**—Always show scroll bars.
- **Hide**—Always hide scroll bars.
Advanced Display properties (for tables)

The Advanced Display tab allows you to define refresh, chunking, and user selection options for table (list view and tree view), results list, and alert list fields.

The following figure shows the field properties on the Advanced Display tab. The table on the following page provides descriptions of each field property.

Figure D-2: Field Properties—Table Field, Advanced Display tab

![Field Properties - Table Field](image-url)
### Refresh Table Options section

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Row Selection</td>
<td>Describes what will happen the first time the list view table, tree view table, or results list field is displayed. Your choices are:</td>
</tr>
<tr>
<td></td>
<td>- Select First, Fire Workflow— The first row or leaf is selected, and enabled workflow will be executed.</td>
</tr>
<tr>
<td></td>
<td>- Select First, No Workflow— The first row or leaf is selected, and no workflow will be executed.</td>
</tr>
<tr>
<td></td>
<td>- No Selection— No item is selected.</td>
</tr>
<tr>
<td>Refresh Row Selection</td>
<td>Describes what will happen on all subsequent refreshes. Your choices are:</td>
</tr>
<tr>
<td></td>
<td>- Retain Selection, Fire Workflow— The current selection is retained, and enabled workflow will be executed.</td>
</tr>
<tr>
<td></td>
<td>- Retain Selection, No Workflow— The current selection is retained, and no workflow will be executed.</td>
</tr>
<tr>
<td></td>
<td>- Select First, Fire Workflow— The first row or leaf is selected and enabled workflow will be executed.</td>
</tr>
<tr>
<td></td>
<td>- Select First, No Workflow— The first row or leaf is selected and no workflow will be executed.</td>
</tr>
<tr>
<td></td>
<td>- No Selection— The table or results list will display with no selection.</td>
</tr>
</tbody>
</table>
## Data Chunking section

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size of Chunk</td>
<td>For client-side list view table fields and results list fields, defines the size of data chunks (the number of records) displayed.</td>
</tr>
<tr>
<td></td>
<td>- If the value is 0, chunking is disabled.</td>
</tr>
<tr>
<td></td>
<td>- If the value is greater than 0, chunking is enabled, according to the number of records indicated.</td>
</tr>
<tr>
<td></td>
<td>If you enable chunking, the Max Rows setting in the Table Property tab is ignored.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Use the Configuration tab in the Server Information window to set the chunk size of server-side table fields for the entire server. For more information, see the Configuring AR System guide.</td>
</tr>
<tr>
<td>Next Label</td>
<td>When chunking is enabled, specifies the label to display to proceed to the next chunk. This label does not appear if the current chunk is the last chunk.</td>
</tr>
<tr>
<td>Previous Label</td>
<td>When chunking is enabled, specifies the label to display to return to the previous chunk. This label does not appear if the current chunk is the first chunk.</td>
</tr>
</tbody>
</table>

## Other settings

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Selection Not Allowed</td>
<td>Specifies whether a user can select rows. If selection is not allowed:</td>
</tr>
<tr>
<td></td>
<td>- A user cannot:</td>
</tr>
<tr>
<td></td>
<td>- Trigger active links.</td>
</tr>
<tr>
<td></td>
<td>- Delete a row item.</td>
</tr>
<tr>
<td></td>
<td>- A user can:</td>
</tr>
<tr>
<td></td>
<td>- Scroll.</td>
</tr>
<tr>
<td></td>
<td>- Sort.</td>
</tr>
<tr>
<td></td>
<td>- Drill-down.</td>
</tr>
<tr>
<td></td>
<td>- Create a report.</td>
</tr>
<tr>
<td>Single Selection Only</td>
<td>Specifies that a user with selection capability can select only one row at a time.</td>
</tr>
</tbody>
</table>
Note: The other settings are not supported for tree view table fields.

**Allowable Types properties (currency fields only)**

Use the Allowable Types tab to define allowable currency types and precisions for a currency field.

**Figure D-3: Field Properties—Currency Field, Allowable Types tab**

If you defined default allowable currencies, those currencies are listed under the Currency Type column when you create a new currency field. Use the Allowable Types tab if you did not define default currency types, or if you want to change the allowable currency type properties for an individual currency field. For information about default currency types, see “Defining default currency types” on page 176.
To define allowable currency types

1. Select a currency type in the Available list and click Add to add that currency to the Selected list.
   
   If you select the All Allowable Currency Types allowed check box, all currency types are allowable.

2. To optionally change the decimal precision, select and edit the number under the Precision column for each currency type.

3. To delete an allowable currency, select a currency type in the Selected list and click Remove.

4. From the Choose Primary Allowable Type list, select the currency type used when only a decimal value is entered in the currency field.
   
   If none is selected, the first allowable type is used.

5. To set the default allowable currencies in the Server Information window using the Selected currency types in the Allowable Types tab, click Save As Server Default.

   To reset the allowable currency types for the field based on default allowable currencies in the Server Information window, click Reset To Default.
Attach Fields properties

Use the Attach Fields tab to define attachment fields within an attachment pool. If you are creating a form or a view that will be viewed on the web, you can define display labels as described in “To define display labels for attachment pools viewed in a browser” on page 198.

Figure D-4: Field Properties—Attachment Pool, Attach Fields tab

The following table shows the available display labels, their default values, and their descriptions.

<table>
<thead>
<tr>
<th>Field</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add Label</td>
<td>Add</td>
<td>Label on button used for adding an attachment.</td>
</tr>
<tr>
<td>Delete Label</td>
<td>Delete</td>
<td>Label on button used for deleting the selected attachment.</td>
</tr>
<tr>
<td>Display Label</td>
<td>Display</td>
<td>Label on button used for opening the selected attachment on the client.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For web clients, only files with file name extensions that are configured by</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the browser will open. If the browser is not configured for a particular file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>type, a Save dialog box will open.</td>
</tr>
<tr>
<td>Filename Label</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File Size</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
From the Attributes tab, you can specify the kind of information that will be contained in the fields that you create. The settings that appear differ based on the data type of the field.

**Character field attributes**

The following figure shows the Attributes tab for character field properties.

**Figure D-5: Field Properties—Character Field, Attributes tab**
The following table lists the definitions and uses of attributes you can set for the character data type.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>Assigns an administrator default value to the field. The value that you enter will appear in the field whenever users load default values before performing a search or submitting a new request in BMC Remedy User. Default values are helpful for application usability. For example, creating a default value as a prompt in a required field (“Enter your name here”) informs the user what information must be entered into the form. You can use keywords to define a default value. Keyword values that are not likely to change, such as $USER$, expand to a value when defaults are first set. Other keywords (such as $TIME$) expand to a value as late as possible when defaults are loaded. For a list of keywords you can use to define a field’s default value, see the Workflow Objects guide.</td>
</tr>
<tr>
<td>Pattern</td>
<td>Restricts what the user can enter into the field. There are two types of character patterns that you can specify. The first type enables you to enter any character pattern that you want to match. It is similar to that used in the LIKE operator and can include any of the same wildcard characters (see the discussion on operators in the Workflow Objects guide). The second type uses a keyword to specify a style for a character field. You can specify only one keyword for a pattern, and it cannot be combined with a pattern of characters and wildcards. If you select:</td>
</tr>
</tbody>
</table>
|             | $ALNUM$ — The value must be alphabetic characters and digits (and blank space).  
|             | $ALPHA$ — The value must be alphabetic characters (and blank space).  
|             | $DIGIT$ — The value must be digits.  
|             | $LOWER$ — The value can be any character except uppercase letters. This includes special characters, digits, and blank spaces.  
|             | $MENUS$ — The value must match an item defined in the default menu attached to the field. Avoid using $MENUS$ together with a Change Field workflow action that attaches a new menu (with new values) to the field. You cannot use the $MENUS$ keyword for file menus and data dictionary menus.  
|             | $PRINT$ — The value must be printable characters.  
|             | $UPPER$ — The value can be any character except lowercase letters. This includes special characters, digits, and blank spaces.  

**Note:** The way in which keywords are interpreted is language-dependent. For example, $LOWER$ is not valid in Japanese, and $MENUS$ is not valid in a multilingual environment.
In the Default Value field of the Attributes tab, you can enter the value that you want to appear in the field whenever users load default values before performing a search or submitting a new request. The field value can be a static value or one of the following keywords: $TIMES$, $DATES$, and $TIMESTAMP$. For more information about keywords, see the Workflow Objects guide.
Diary field attributes

In the Default Value field of the Attributes tab, you can enter the value that you want to appear whenever users load default values when performing a search or submitting a new request. You can choose a keyword from the menu at the end of the field. (For more information about keywords, see the Workflow Objects guide.)

Currency field attributes

Currency fields store:

- Information entered by the user (a decimal value and an allowable currency type)
- System-generated data (one or more functional currency types and the date on which functional values were converted)
The following table lists the definitions and uses of attributes you can set for the currency data type.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>The minimum decimal value.</td>
</tr>
<tr>
<td>Maximum</td>
<td>The maximum decimal value.</td>
</tr>
<tr>
<td>Default Value</td>
<td>A decimal value and an allowable currency type. The default value overrides any administrator or user defined initial currency type.</td>
</tr>
<tr>
<td>Audit Option</td>
<td>Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>None—Changes to this field are not recorded by any audit processing.</td>
</tr>
<tr>
<td></td>
<td>Audit—Changes to this field trigger audit processing and its new value will be recorded in the audit form or log form, depending on the audit style you specified at the form level.</td>
</tr>
<tr>
<td></td>
<td>Copy—Changes to this field are recorded during an audit, but will not trigger audit processing.</td>
</tr>
</tbody>
</table>

**Integer field attributes**

Integer fields accept integer values between -2147483647 and 2147483647. Specify values in the Minimum and Maximum fields to limit the range for a field. During data submission and modification, the user is limited to entering values that fall within the range that you specify.
In the Default Value field of the Attributes tab, you can enter the field value that you want to appear whenever users load default values when performing a search or submitting a new request.

**Figure D-9: Field Properties—Integer Field, Attributes tab**

Real fields accept and contain floating-point numbers. The attributes for real number fields are defined in the Attributes tab.

**Figure D-10: Field Properties—Real Number Field, Attributes tab**
The following table lists the definitions and uses of attributes you can set for the real number data type.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>The low limit the field value can have during data submission and modification. This setting is required.</td>
</tr>
<tr>
<td>Maximum</td>
<td>The high limit the field value can have during data submission and modification. This setting is required.</td>
</tr>
<tr>
<td>Precision</td>
<td>The number of decimal places displayed in the user’s view. The displayed number is rounded off, but the value stored in the database is not changed.</td>
</tr>
<tr>
<td>Default Value</td>
<td>The value that you want to appear whenever users load default values when performing a search or submitting a new request.</td>
</tr>
<tr>
<td>Audit Option</td>
<td>Select one of the following options:</td>
</tr>
<tr>
<td>None</td>
<td>Changes to this field are not recorded by any audit processing.</td>
</tr>
<tr>
<td>Audit</td>
<td>Changes to this field trigger audit processing and its new value will be recorded in the audit form or log form, depending on the audit style you specified at the form level.</td>
</tr>
<tr>
<td>Copy</td>
<td>Changes to this field are recorded during an audit, but will not trigger audit processing.</td>
</tr>
</tbody>
</table>

For real fields, the representation in the database keeps a maximum of 10 digits worth of data. After 10 digits, the number is rounded, and the succeeding digits are ignored. For example, if 12345.090009 is entered, the value after a submit is 12345.090010. But, if 1234567.090099 is entered, the value after a submit is 1234567.090000. The last three digits are ignored because the rounded answer comes after the 10th position.

**Decimal field attributes**

Decimal fields accept and contain fixed-point decimal numbers. Decimal number field attributes are defined in the Attributes tab.
The following table lists the definitions and uses of attributes you can set for the decimal data type.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>The low limit the field value can have during data submission and modification. This setting is required.</td>
</tr>
<tr>
<td>Maximum</td>
<td>The high limit the field value can have during data submission and modification. This setting is required.</td>
</tr>
<tr>
<td>Precision</td>
<td>The number of decimal places displayed in the user’s view. The default setting is 2 and the maximum value of precision in AR System is 9.</td>
</tr>
<tr>
<td>Default Value</td>
<td>The value that you want to appear whenever users load default values when performing a search or submitting a new request.</td>
</tr>
<tr>
<td>Audit Option</td>
<td>Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td><strong>None</strong>- Changes to this field are not recorded by any audit processing.</td>
</tr>
<tr>
<td></td>
<td><strong>Audit</strong>- Changes to this field trigger audit processing and its new value will be recorded in the audit form or log form, depending on the audit style you specified at the form level.</td>
</tr>
<tr>
<td></td>
<td><strong>Copy</strong>- Changes to this field are recorded during an audit, but will not trigger audit processing.</td>
</tr>
</tbody>
</table>
Selection field attributes

Selection fields provide for a small number of choices, which are defined on the Attributes tab.

Figure D-12: Field Properties— Selection field, Attributes tab

The following table lists the definitions and uses of attributes you can set for the selection data type.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Database value of the selection item.</td>
</tr>
<tr>
<td>Default Value</td>
<td>Default value that will appear when the form is opened.</td>
</tr>
<tr>
<td>Alias Value</td>
<td>Value that appears on the option that users see in their BMC Remedy User or web clients.</td>
</tr>
</tbody>
</table>
The settings that appear in the Attributes tab of the trim properties window vary depending on whether the trim is a line, box, or text.

**Figure D-13: Field Properties—Trim Field, Attributes tab**
The following tables describe available trim field attributes.

### Text section

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Style   | Specifies a style for a text field. You can redefine how a style maps to a specific font. The style choices, default attributes, and uses are listed in discussion on Form Font preferences in the Getting Started guide.  

**Note:** The user can change the font and size of each style with BMC Remedy User Display Options settings. As the administrator, your font preferences should match that of the majority of your users' tools. |
| Justification | Defines where text is positioned relative to the left and right edges of the box. Your choices are:  
- **Left**  
- **Center** (the default)  
- **Right** |
| Alignment | Defines where text is positioned relative to the top and bottom edges of the text box. The choices are:  
- **Top**  
- **Middle** (the default)  
- **Bottom** |

### Line section

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Thickness | For a box or line, specifies a line width of 1 to 9 pixels. The default is 2.  
Thickness cannot be edited if the Depth Effect is Etched. |
| Depth Effect | For a box or line, specifies the appearance of depth. The choices are:  
- **None**  
- **Raised**  
- **Sunken**  
- **Etched** (the default)  
See Figure 4-15 on page 172 for examples of these styles. |
Change History properties

AR System automatically records the owner of a field, the user who last modified the field, and the date of the modification. To display or add to this information, select the Change History tab in the properties window.

For more information about building and using change history, see the Getting Started guide.

Color/Font properties

From the Color/Font tab, you can set the color and font style for a field.

### Line section

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orientation</td>
<td>Specifies the orientation of a line. The choices are:</td>
</tr>
<tr>
<td></td>
<td>Vertical</td>
</tr>
<tr>
<td></td>
<td>Horizontal</td>
</tr>
<tr>
<td>Color</td>
<td>Specifies the color for the line. The color selection is enabled when the Depth Effect is set to None. When you uncheck the Default box, the color palette is enabled, and you can choose a color for the line.</td>
</tr>
</tbody>
</table>

### Background section

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Mode</td>
<td>For text or box fields, specifies whether the background is opaque or transparent. For text fields in web clients, if you choose Opaque, the background will be white when displayed in browsers.</td>
</tr>
<tr>
<td>Color</td>
<td>For text or box fields, specifies the background color. The color selection is enabled when the Display Mode is set to Opaque. When you uncheck the Default box, the color palette is enabled, and you can choose a background color.</td>
</tr>
</tbody>
</table>
Label/Text color

For a data field, the field label is the text that appears to the left of a field or above it. For a button field, the field label consists of the text that appears within the button. Colors are set one field at a time.

To change label or text color

1. Select the Color/Font tab.
2. Clear the Default Label/Text Color check box. The field under Label/Text Color is enabled.
3. Select a color from the palette attached to the field. You can also select Other from the palette to create a custom color.
4. Choose File > Save Form to save your changes.
**URL color**

You also can define the color of the URL when you insert a URL link into a text trim field, as described in “To change the URL color (text trim fields)” on page 215. For information about adding a URL to a text field, see “Adding a URL to a text field” on page 214.

**Fonts**

You can choose a font type for the field label and for the data the user enters into the field. Each font type represents a particular font family, style, and size. You can change the characteristics of each font type in your BMC Remedy Administrator preferences. For more information, see the discussion on Form Font preferences in the Getting Started guide.

In a newly created regular form, the field labels appear in different font styles in the form. By default, each font style identifies a different field behavior. As the administrator, you can override the font properties of the fields, but use caution to avoid confusing users. In addition, users can change the fonts assigned to the font styles using preferences. All fields with the same style will remain consistent.

The various default font styles have the following meanings:

- **Italic text**: Maintained and automatically updated by AR System
- **Bold text**: Requires a value
- **Plain text**: Value is optional
To change the font style

1. Select the Color/Font tab.
2. From the Label field list, select a font style for the field label.
3. From the Data field list, select a font style for data entered into the field, if applicable.
4. Choose File > Save Form.

Database properties

The settings described in the following table appear in the properties window when you select the Database tab. These settings apply to all views.

Note: Attachment fields use different database properties, which are described in “Database properties (for attachment fields)” on page 427.
Figure D-16: Field Properties—Character Field, Database tab
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>

**ID**

Identifies the field internally throughout AR System. Every field in a form must have an integer field ID that is unique in that form. If you leave the ID field empty or set it to zero when you are defining a field, AR System will automatically assign a number from the unrestricted number set. Restrictions on field ID numbers are as follows:

- Numbers 1–99 are reserved for core fields. You cannot assign an ID in this range, unless you are modifying core fields. See Appendix A, “Core fields” for information about these fields.
- Numbers 100–536870912 are reserved. If you use an ID in this range, you will receive a warning. Numbers 1000000–1999999 and 3000000–3999999 are specifically reserved for regular global fields and window-scoped global fields, respectively. For more information about global fields, see page 218.
- Numbers 536870913–2147483647 are administrator-defined. There are no restrictions on assigning numbers in this range. If you choose to assign field IDs instead of letting AR System do it automatically, be aware that view IDs are also drawn from the low end of this range.

Columns in table fields and pages in page fields also have an ID. For purposes of assigning order in workflow, you can assign the ID yourself, or let AR System assign the number for you.

The field ID remains constant even if the database name or display label changes. You cannot modify the field ID after it is applied to the database.

If you are defining fields that serve the same purpose in more than one form, assign identical IDs to the identical fields in the different forms. You can then write workflow once for that field (with minor edits to AR System field definition) and reuse the field in multiple forms. Reusing the ID provides a consistent definition for the field across the forms.
Form and Application Objects

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Identifies the field in the database. Every field in a form must have an alphanumeric field name that is unique in that form. Names can be as many as 80 characters, including spaces. Names can include double-byte characters, but avoid using numbers at the beginning of the name. If you leave the Name field empty, BMC Remedy Administrator generates a name based on the field type and appends a number to the name to make it unique. For example, if there is already a field called Column1, BMC Remedy Administrator names the next field of this type Column2. Do not use the keyword FUNCTION as a field name; otherwise, the system will generate an error, and the underlying database view will not be created. <strong>Note:</strong> If you create a field with a dollar sign ($) or an apostrophe ('') in the database name, you must double the dollar sign or the apostrophe when using the field in workflow in addition to adding the surrounding characters. For example, a field named MyMoney$ would be entered in workflow as $MyMoney$$, while a field named John's Money would be 'John''s Money'. The field name is easier to use than the field ID when creating workflow such as active links and filters. Unlike the field label, the field name is not specific to a view of the form. Do not confuse the field name with the field label that you define in the Display properties tab, especially when creating workflow. To avoid naming conflicts with the database server, do not use a word reserved by the database server software as a field name. Consult your database documentation for a list of reserved words.</td>
</tr>
</tbody>
</table>
### BMC Remedy Action Request System 7.0

**Input Length**

For **attachment fields**, determines the maximum size of the attachment. For more information, see “Database properties (for attachment fields)” on page 427.

For **character fields**, determines the maximum number of bytes the field can contain. You will get the most efficient use of database storage if you set the maximum length of a character field to be less than or equal to 255 bytes (except for Oracle, as explained below). When the field length is this size, the storage is dynamically allocated to the length of the actual field contents, not the input length. If you set the maximum length to more than 255 bytes, storage is allocated in blocks that average between 1K to 2K bytes (depending on the database). A full block is allocated for the first byte. When that block is filled with the field contents, another full block is allocated. Leaving the Input Length field empty or setting it to 0 specifies an unlimited length field.

For Oracle databases, character fields up to 4000 bytes are created as `varchar`, and storage space is allocated dynamically to the length of the actual field contents. Character fields larger than 4000 bytes are created as `clob`, which is managed differently by the database. For better performance, use `varchar` whenever possible. For more information about database structure in AR System, see the Database Reference Guide.

Do not use more space than you need to store the intended field contents. If you allot more space for storage than your system needs, more space is searched during queries.

If the Expand Box display property is set to Default, and if the field length is 70 bytes or more, AR System automatically inserts a text icon to the right of the field that the user can click to open a Text dialog box. This enables you to conserve space on the user’s view of the form by making the field’s Display Length smaller than its Length attribute.

The default upper size limit for character fields is different for each of the following databases:

- For Sybase/MS SQL Server and Informix databases, 2 GB
- For DB2, 10 MB
- For Oracle, 4 GB

For scalability reasons, limit the number of long character fields in a form.

**Input Length (continued)**

**Note:** Use the `ar.conf` or `ar.cfg` files to configure a different upper limit (than the default) for Oracle and MS SQL Server databases using the `Db-Max-Text-Size` configuration option. For more information about AR System configuration files, see the Configuring AR System guide.

You cannot use the Indexing form property to create an index for a long or character field with a database length over 255 bytes. However, if you are licensed for full text search, you can use the Index for FTS database property to create a search index for the field. For more information about FTS, see Configuring guide.

For some databases, you cannot search fields that are over 255 bytes. If you have questions about the capabilities of your database, see your database reference guides, as well as information in the AR System release notes and the Installing AR System guide.

### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Input Length   | For attachment fields, determines the maximum size of the attachment. For more information, see “Database properties (for attachment fields)” on page 427. For character fields, determines the maximum number of bytes the field can contain. You will get the most efficient use of database storage if you set the maximum length of a character field to be less than or equal to 255 bytes (except for Oracle, as explained below). When the field length is this size, the storage is dynamically allocated to the length of the actual field contents, not the input length. If you set the maximum length to more than 255 bytes, storage is allocated in blocks that average between 1K to 2K bytes (depending on the database). A full block is allocated for the first byte. When that block is filled with the field contents, another full block is allocated. Leaving the Input Length field empty or setting it to 0 specifies an unlimited length field. For Oracle databases, character fields up to 4000 bytes are created as `varchar`, and storage space is allocated dynamically to the length of the actual field contents. Character fields larger than 4000 bytes are created as `clob`, which is managed differently by the database. For better performance, use `varchar` whenever possible. For more information about database structure in AR System, see the Database Reference Guide. Do not use more space than you need to store the intended field contents. If you allot more space for storage than your system needs, more space is searched during queries. If the Expand Box display property is set to Default, and if the field length is 70 bytes or more, AR System automatically inserts a text icon to the right of the field that the user can click to open a Text dialog box. This enables you to conserve space on the user’s view of the form by making the field’s Display Length smaller than its Length attribute. The default upper size limit for character fields is different for each of the following databases:
- For Sybase/MS SQL Server and Informix databases, 2 GB
- For DB2, 10 MB
- For Oracle, 4 GB

For scalability reasons, limit the number of long character fields in a form. Use the `ar.conf` or `ar.cfg` files to configure a different upper limit (than the default) for Oracle and MS SQL Server databases using the `Db-Max-Text-Size` configuration option. For more information about AR System configuration files, see the Configuring AR System guide. You cannot use the Indexing form property to create an index for a long or character field with a database length over 255 bytes. However, if you are licensed for full text search, you can use the Index for FTS database property to create a search index for the field. For more information about FTS, see Configuring guide. For some databases, you cannot search fields that are over 255 bytes. If you have questions about the capabilities of your database, see your database reference guides, as well as information in the AR System release notes and the Installing AR System guide. |
### Form and Application Objects

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Type</strong></td>
<td>Indicates the type of information that the field is designed to contain, for example, character or trim. For information about Date, Time, and Date/Time fields, see “Date and time fields” on page 146. For more information about the data types that have attribute properties, see “Attributes properties” on page 406.</td>
</tr>
</tbody>
</table>
| **Entry Mode**| Defines whether the field is:  
- **Required**—Field requires a value (by user input, a default value, or workflow) when the user submits a request. A required field appears with a bold label. A form has at least three required fields: Submitter, Short Description, and Status. Make a field required only if it must be filled in for every new request.  
- **Optional**—Users can enter information in the field or leave it empty. If you have optional fields that must be filled in under certain circumstances, you can create filters and active links that force the user to fill in the field when specified conditions are met.  
- **Display Only**—The field is used as a temporary field. No space is allocated for a display-only field in the database, so a display-only field’s value can never be recorded to the database. For this reason, display-only fields have a value of NULL when a request is retrieved. In all other ways, a display-only field can be used in the same way as any other field (for example, you can reference it in workflow).  
- **System**—The field is populated by AR System. Certain core fields such as Request ID and Last Modified By are system fields. You can select the Display Only option only when you create a field. After you save a form, the following changes occur:  
  - For a required or optional field, the display-only option is no longer available as an entry option.  
  - For a display-only field, the Entry Mode field is disabled and cannot be changed. |
| **Index for FTS** | If you are licensed for full text search, specifies whether to index a character, diary, or attachment field for FTS.  
Global fields (field IDs 1000001–2000000) must be display-only. |
## Appendix D—Field properties

### QBE Match

For character fields, specifies how a match will be determined when a user performs a query-by-example (QBE) in BMC Remedy User. If you select:

- **Anywhere** (the default)—Finds a match if the value entered occurs anywhere in the corresponding field. For example, if the user enters \textit{Bob} in the Submitter field, the search will return all requests submitted by Bobby Jones, Bob Smith, and Jill Bobbington.

- **Leading**—Finds a match only if the value entered occurs at the beginning of the corresponding field. For example, if the user enters \textit{Bob} in the Submitter field, the search will return all requests submitted by Bob Smith and Bobby Jones, but not those submitted by Jill Bobbington.

- **Equal**—Finds a match only if the value entered matches the value in the corresponding field exactly. For example, to find requests submitted by Bob Smith, the user must enter \textit{Bob Smith}, with exact spelling and capitalization, in the Submitter field. However, for some databases (for example, Sybase or MS SQL Server), case-sensitivity depends on the underlying DBMS settings, regardless of what QBE Match you specify here.

You can use the Preferences window to set a default QBE match setting for all new character fields that are not core fields (Request ID, Submitter, Assigned To, Last Modified By, and Short Description). For more information, see the discussion on Form preferences in the Getting Started guide.

A search on a character field with a QBE match type of Anywhere will perform a full table scan of the database, reading every record in a form and ignoring any indexes for the field. Searches on fields where the QBE match type is Leading or Equal are typically faster than searches on fields where the match type is Anywhere, especially if the field is indexed, as described in “Defining indexes” on page 139.

Some relational operators and wildcards work during a query-by-example regardless of the QBE Match setting. This means that users can specify an exact match in a field with a QBE Match setting of Anywhere by using the equal sign (=) relational operator. Users can also use the percent sign (%) wildcard at the beginning of the search string (\textit{%abcd}) to override a QBE Match setting of Leading or Equal. However, using the % wildcard anywhere else in a string (\textit{abcd%}) does not override the Equal setting. Overriding the Leading or Equal QBE Match settings overrides the performance benefits of using those settings.

### Join Information

For fields on join forms, includes read-only join information about the field.

For more information, see “Fields in join forms” on page 220.

### View Information

For fields on view forms, specifies the table and column names.

For more information about view forms, see the Integarting with Plug-ins and Third-Party Products guide.
For each attachment field within an attachment pool, you set database properties. The Database tab enables you to define the:

- Attachment field name.
- Maximum size (input length) of the attachment.
- Entry Mode. You can select Optional or Display Only. For more information, see “Entry Mode” on page 425.

**Figure D-17: Field Properties—Attachment Field, Database tab**
Display properties

From the Display tab, you can define how fields are displayed in BMC Remedy Administrator and BMC Remedy User.

Figure D-18: Field Properties—Character Field, Display tab
The option settings in the following table appear in the properties window when you select the Display tab. These settings apply only to the specific form views in which they are set. For example, in one view a field might be visible, while in another view, the field might be hidden or located somewhere else in the form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Identifies a field in the currently selected view of the form. You can enter a name with as many as 80 characters that describes the meaning and purpose of the field. The label can include spaces and double-byte characters. Avoid using spaces at the beginning of field labels; spaces added at the beginning of field labels will not appear in some browsers. If you leave this field empty, the field appears on the screen with no label. Users with Customize permissions can define labels in their personalized views in BMC Remedy User. For tree view table fields, if the Label field has text, it becomes the root of the tree. The field label text need not be unique. However, if duplicate field labels exist in a form, AR System will issue a warning message every time you apply changes to that form (unless you disable the BMC Remedy Administrator preferences for duplicate blank and nonblank field label warnings, as described in the Getting Started guide). You can use single quotation marks in field labels; however, when performing searches, users will need to enter two single quotation marks when specifying the quotation mark in the label. This is because field labels that contain special characters must be enclosed in single quotation marks in searches, and a single quotation mark in the label would otherwise be interpreted as the end of the field label.</td>
</tr>
<tr>
<td>Location</td>
<td>Defines where a data field’s label will appear in relation to the field: Top (above the field) or Left (the default).</td>
</tr>
<tr>
<td>Justify</td>
<td>Defines where a label is positioned relative to the left and right edges of the region available to the label. Your choices are Left (the default), Center, and Right.</td>
</tr>
<tr>
<td>Align</td>
<td>Aligns labels to the top, center, or bottom of the region available for the label. The default is Top for labels located to the left of the field. The default is Bottom for labels located above the field. For more information, see “Using the alignment field to align form fields” on page 292. For text fields with one row, when the Align property is set to Top, labels appear in the center in BMC Remedy User and on the web. For more information, see “Rows” on page 430.</td>
</tr>
<tr>
<td>X</td>
<td>Defines the horizontal position of the left edge of the field in the form. The X and Y settings are relative to the top left corner of the form. For example, X=0 and Y=0 is the top left corner of the screen.</td>
</tr>
</tbody>
</table>
**Appendix D—Field properties**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>Defines the vertical position of the left edge of the field in the form. The X and Y settings are relative to the top left corner of the form. For example, X=0 and Y=0 is the top left corner of the screen. If you have a series of data fields on the same line, these fields must have the same Y coordinate for tabbing between fields to work from left-to-right, top-to-bottom, unless you define a tab order of fields in the form. For more information, see “Setting the tab order for the fields in a form” on page 297.</td>
</tr>
<tr>
<td>Field Width</td>
<td>Defines the width of the field in pixels.</td>
</tr>
<tr>
<td>Field Height</td>
<td>Defines the height of the field in pixels. This setting applies to data, button, line trim, table, and page holder fields.</td>
</tr>
<tr>
<td>Data Length</td>
<td>Defines the width of the field’s data entry region in pixels. Display length often differs from the setting of the maximum length allowed for data entered in the field (see “Input Length” on page 424). If a user enters more characters than can be displayed, the text will scroll off the end of the field, provided the internal field length can accommodate the input.</td>
</tr>
<tr>
<td>Rows</td>
<td>Defines the number of rows displayed in a field. For selection fields displayed as radio buttons, defines the number of rows used for the selection choices. A setting of 1 produces a single horizontal row. A setting of 2 divides the choices evenly into two horizontal rows, and so on. For text fields, defines the number of rows of text that are displayed.</td>
</tr>
<tr>
<td>Hidden</td>
<td>Specifies whether the field is visible in the user’s view. If the check box is selected, the field cannot be seen in the view, although users with Customize permissions can make a hidden field visible. A hidden field remains in the database and can be accessed by workflow. You can create active links to hide and unhide fields dynamically. Hidden columns in tree view table fields are ignored.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| If Hidden, No Space   | In legacy relative web views, defines the field as taking up no space when hidden. Select this property if your application includes overlapped or hidden fields to make sure the correct alignment of surrounding fields. The If Hidden, No Space option is not available in Standard (Recommended) and Web - Alternate (Fixed) views. If you select Maintain Field Layout when creating the web view and select the If Hidden, No Space option, you must also modify the HTML source code for the view as follows:  
- In the source code, locate the HTML table and table cells associated with each field pagelet.  
- Identify pagelets by the field ID, and locate the associated `<TABLE>` and `<TD>` tags.  
- Remove the `width` tag from the HTML tables containing hidden fields. For example, change `<TABLE height=128 width=910 border=0> to `<TABLE height=128 border=0>`.  
- Remove the `width` tag from each HTML table cell that contains a hidden field. For example, change `<TD width=455>` to `<TD>`.
  
  **Note:** This property does not make the field hidden. To hide the field, select the Hidden property or use a Change Field action. |
| Disable Change Flag   | For data and attachment fields, defines whether the change flag (or “dirty bit”) is affected by the field you are creating. When the check box is selected, the field will not affect the change field status of the form. This can be helpful when you have calculations that use hidden fields.  
  
  **Note:** Any field not in the user’s current view does not affect the change flag, even if the Disable Change Flag check box is cleared. You can associate a `SET-CHANGE-FLAG` Run Process action with a field not in the view to set the change flag as needed. For more information, see “Including and excluding fields from form views” on page 299 and the discussion on special run processes and `$PROCESS$` in the Workflow Objects guide.  
  
  For more information, see `GET-CHANGE-FLAG` and `SET-CHANGE-FLAG` in the Workflow Objects guide. |
| Custom CSS Style      | Specifies a custom CSS style for the field. For more information about CSS styles in AR System, see the Installing and Administering BMC Remedy Mid Tier guide. |
| Display As Text       | For data fields, displays the contents of the field as plain text, without a field border or background. You might want to set an initial value for the field as described in “Database properties” on page 420. |
### Appendix D—Field properties

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Read/Write, Read Only, Disabled | For data fields, defines how users initially access a field in the current view.  
  - **Read/Write** (the default)—Users can read, access (for example, copy), and edit field information. (This is the default.)  
  - **Read Only**—Users can read or access field information, but cannot edit it.  
  - **Disable**—Users can read field information, but cannot access or edit it. |
| Expand Box | For character, diary, date/time, date, and currency fields, determines whether the expand box icon appears next to the field. You might want to hide the expand box if you select the Display As Text check box. The options are:  
  - **Default**—Hides the expand box for character fields if the field length is less than 70. Displays the expand box for diary, date, and date/time regardless of field length.  
  - **Hide**—Always hides the expand box.  
  - **Show**—Always displays the expand box. |
| Enabled, Disabled | For trim or control fields (buttons and form menu items), controls how the field is initially displayed. The options are:  
  - **Enabled**—Results in an active field. (This is the default.)  
  - **Disabled**—Results in an inactive (grayed out) field. |
| Table Drill-Down | For table fields, determines whether the source request can be displayed. When Table Drill-Down is enabled, to open the source request of a row in modify mode, users can double-click the row in the table in BMC Remedy User or web clients. For tree views, the user must double-click on a leaf.  
  See “Table Labels properties” on page 449 for more information. |
| Refresh on Entry Change | For table fields, refreshes data when the request in which the table is embedded appears. Enable this mode if users must see the contents of the field whenever they access a request. To reduce performance impact, limit the use of this feature because each refresh causes a database search.  
  If the Refresh on Entry Change check box is not selected, the user can refresh data manually in BMC Remedy User by clicking the table field or by right-clicking in the table field and choosing Refresh Table. On the web, users can click the Refresh button, if the administrator supplies this button. See “Table Labels properties” on page 449 for more information.  
  You can refresh a table field using workflow by selecting the Refresh Table Field check box when defining a Change Field active link action. |
| Display NULL Values As | For tree view table fields, specifies what will appear if a node is a NULL value. You can enter a string of a maximum of 255 bytes or 80 characters into this field. If you leave the field empty, it shows [No Value], which is the default.  
  For more information about how NULL values are treated in tree view fields, see the note on page 155. |
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto Fit Columns</td>
<td>For list view tables, results lists, and alert lists, specifies that column widths should be scaled proportionally to the width of the table, results list, or alert list field. This ensures that no horizontal scrolling is necessary to see all columns. For tree views, this property is ignored. If Auto Fit Columns is not checked, the column widths are not scaled, and a horizontal scroll bar might appear so that the user can access all columns. See also the Column Width property on page 436. Unlike Windows, column size on the web is not exact. The web client will try to fit the columns into the requested table width size area. If the width of your column title and the data in the column is narrow, and the column width size defined in BMC Remedy Administrator is wide, the web client might shrink the column width to accommodate other columns so that all the columns can be seen.</td>
</tr>
<tr>
<td>Fixed Headers</td>
<td>For list view tables, results lists, and alert lists in web clients only, prevents the table header from disappearing when scrolling down a table. In some browsers, editable drop-down list fields within a table might not appear correctly with Fixed Headers enabled. For tree views, this property is ignored.</td>
</tr>
<tr>
<td>Data Type</td>
<td>Indicates the field type. This property also appears on the Database tab. The data type property is read-only. See Chapter 4, “Types of fields,” for more information.</td>
</tr>
</tbody>
</table>
**BMC Remedy Action Request System 7.0**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Type</td>
<td>Defines how field information is presented in a form. Each type of field has different Display Type options, as described in the following sections:</td>
</tr>
<tr>
<td><strong>Character fields</strong></td>
<td></td>
</tr>
<tr>
<td>Drop-Down List</td>
<td>Users can select a value from a drop-down list, but cannot type directly in the field. Values available in the list are from a menu you attach to the character field. The drop-down list option is only available if the character field has a menu attached to it. This option is different from the drop-down list for a selection field although they look similar.</td>
</tr>
<tr>
<td>Edit</td>
<td>Users can type values directly into the field.</td>
</tr>
<tr>
<td>Edit Masked</td>
<td>Users can type values directly into the field, but the value is displayed as asterisks. This display type, however, offers no special security. For example, a value you enter is not stored in the database as an encrypted value. Edit masked values are exported, imported and transmitted from client to server in clear text.</td>
</tr>
<tr>
<td><strong>Diary fields</strong></td>
<td></td>
</tr>
<tr>
<td>Edit</td>
<td>Users can type values directly into the field.</td>
</tr>
<tr>
<td><strong>Date/time fields</strong></td>
<td></td>
</tr>
<tr>
<td>Date and Time</td>
<td>Users can click a calendar that enters date and time values directly into the field. Users can also directly edit date/time fields.</td>
</tr>
<tr>
<td>Date Only</td>
<td>Users can click a calendar that enters date values directly into the field. Users can also directly edit date/time fields.</td>
</tr>
<tr>
<td>Time Only</td>
<td>Users can click increment and decrement arrows (▼ and ▲) to enter a time value directly into the field. Users can also directly edit date/time fields.</td>
</tr>
<tr>
<td><strong>Date fields</strong></td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Users can click a calendar that enters date values directly into the field. Users can also directly edit date/time fields.</td>
</tr>
<tr>
<td><strong>Time fields</strong></td>
<td></td>
</tr>
<tr>
<td>Time Only</td>
<td>Users can click increment and decrement arrows (▼ and ▲) to enter a time value directly into the field. Users can also directly edit date/time fields.</td>
</tr>
<tr>
<td><strong>Currency fields</strong></td>
<td></td>
</tr>
<tr>
<td>Edit</td>
<td>Users can type values directly into the field.</td>
</tr>
<tr>
<td><strong>Integer fields</strong></td>
<td></td>
</tr>
<tr>
<td>Edit</td>
<td>Users can type values directly into the field.</td>
</tr>
<tr>
<td>Numeric Spinner</td>
<td>For BMC Remedy User clients and Internet Explorer browsers only. Users can click increment and decrement arrows (▼ and ▲) to increase or decrease the number in a field. Users can also directly edit numeric fields.</td>
</tr>
<tr>
<td><strong>Real number fields</strong></td>
<td></td>
</tr>
<tr>
<td>Edit</td>
<td>Users can type values directly into the field.</td>
</tr>
<tr>
<td><strong>Decimal number fields</strong></td>
<td></td>
</tr>
<tr>
<td>Edit</td>
<td>Users can type values directly into the field.</td>
</tr>
</tbody>
</table>
Form and Application Objects

Display Type (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Selection fields</strong></td>
<td></td>
</tr>
<tr>
<td>▪ <strong>Drop-Down List</strong> — Users can select from a list of choices.</td>
<td></td>
</tr>
<tr>
<td>▪ <strong>Radio Button</strong> — Users can select from a visible set of choices.</td>
<td></td>
</tr>
<tr>
<td>▪ <strong>Check box</strong> — Users can select only one choice.</td>
<td></td>
</tr>
</tbody>
</table>

**For table fields (list view, tree view, results list, and alert list)**

- Table, Tree, Results List, or Alert List, respectively.

**Columns in a list view or tree view table fields**

- **Editable** — Users can change the value in cells in a column. Editing cells in a table does not affect data in the supporting form. On refresh or sort, changes the user makes are lost, and data displayed in the table will be the data from the parent form. Also, editing a table does not affect the modify flag of the form, nor does it affect row colors.

  The Editable option is ignored for columns in tree view table fields.

- **Read Only** — Users cannot change the field value.

- **Read Only-HTML** — Users cannot change the field value.

  In web clients, data in the cell is displayed as HTML. For example, if a cell contains `<b>my cell</b>`, it will be displayed as **my cell**, but in BMC Remedy User, it will be displayed as `<b>my cell</b>`.

  The Read Only-HTML option is ignored for columns in tree view table fields.

If the column references a display-only field, you can specify the initial value of the column by entering text in the Default Value field. See the Default Value property that follows. For more information about display-only fields, see “Entry Mode” on page 425.

**Columns in results list and alert list table fields**

- **Read Only** — Users cannot change the field value.

**Attachment fields**

- **List Control** — Allows users to view data as a list of elements (such as attachments).

**Buttons**

- **Button** — Users can click a button to initiate active link actions.

- **URL** — The button looks like a URL, which users can click to initiate active link actions.

  With the URL display type, you can also select a Justify property for the button label. See “Justify” on page 429 for more information.

**Page holders, pages, and attachment pools**

- These fields do not have an associated display type.
### Appendix D—Field properties

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Table Type**      | For list view and tree view tables only, defines the format of the table. The options are:  
  - List  
  - Tree  
  You can change the table format at any time.  
  If a tree is displayed in a pre-7.0 BMC Remedy Administrator or BMC Remedy User tool,  
  the format will revert to a table.                                                                 |
| **Column Width**    | For columns in list view, results list, and alert list table fields, defines the width of the column in pixels. See also Auto Fit Columns property on page 433.  
  For tree view columns, this property is ignored.                                                                                          |
| **Wrap Text**       | For columns in list view, results list, and alert list table fields, defines that multiple lines in table column fields are automatically wrapped  
  or that carriage returns are allowed in their row data.  
  For tree view columns, this property is ignored.                                                                                   |
| **Default Value**   | For columns in table fields that reference display-only fields, sets the initial value for cells in the column, up to 255 characters.  
  If the column has a display type of Editable or Read Only, its initial value can be either text or the value from another column in the same table.  
  If the column has a display type of Read Only-HTML, its initial value can be a combination of column references and text.  
  If Default Value is a column reference, such as $Column2$, the value in the display-only column is set to the corresponding value in that column.  
  If the column reference is not valid, the reference will be displayed as text.                                                            |
| **Attachment Pool List** | For attachment fields, copies an attachment field to a second attachment pool when multiple pools are defined on a form.                      |
| **Text**            | For text fields (trim), defines the text that will appear in the trim of a form. For view fields, defines the initial display value for the field.  
  For more information, see “Trim fields” on page 172.                                                                                       |
| **Insert URL Link** | For text fields (trim), enables you to define the URL that is linked to the text defined in the Text field of the properties window.  
  After you highlight a portion of the text you have entered in the Text field, click this button to turn the highlighted text into the label of a URL, which the user can click to link to the specified destination.  
  The destination can be any of the following URLs: file://, ftp://, gopher://, http://, or mailto:. By default, the URL will appear in blue text.  
  For more information about inserting URLs into a form, see the Installing and Administering BMC Remedy Mid Tier guide. |

436
Use the Functional Types tab to define functional currency types and precisions for a currency field.

**Figure D-19: Functional Types tab—Currency field**
If you defined default functional currencies, those currencies are listed under the Currency Type column when you create a new currency field. Use the Functional Types tab if you did not define default currency types, or if you want to change the functional currency type properties for an individual currency field. For information about default currency types, see “Defining default currency types” on page 176.

To define functional currency types

1. Select a currency type in the Available list and click Add to add that currency to the Selected list.

2. To optionally change the decimal precision, select and edit the number under the Precision column for each currency type.

3. To delete a functional currency, select a currency type in the Selected list and click Remove.

4. To set the default functional currencies in the Server Information window using the Selected currency types in the Functional Type, click Save As Server Default.

To reset functional currency types for the field based on default functional currencies in the Server Information window, click Reset To Default.

Help Text properties

From the Help Text tab, you can create or modify the help text for a field. In most cases, the help text that you enter is a description of the data field and how it is used.

In BMC Remedy User, when the field has input focus, users can view this help text in the prompt bar (if visible) or by right-clicking on a field and choosing What’s This? Help, as shown in the following figure.
When the administrator has not defined help, AR System automatically generates field help that appears in the prompt bar of BMC Remedy User when a field has input focus. For more information about creating help text, see the Getting Started guide.

**Image properties**

The Image tab enables you to add an image to a button.
When creating graphics in a form, use images that employ a "web-safe" palette, which is a palette of 256 colors.

The following table describes the available image properties.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display as Flat Image</td>
<td>Enables you to display the button without its default three-dimensional border. This option is helpful when you want to use images as trim.</td>
</tr>
<tr>
<td></td>
<td>If you select this option and your button performs an action when clicked, make sure that the button looks like an object that users should click. For example, you can include a label for the button.</td>
</tr>
<tr>
<td>Image Position</td>
<td>Specifies where the image is positioned relative to the label. The choices are:</td>
</tr>
<tr>
<td>Relative To Label</td>
<td>- Center (default)</td>
</tr>
<tr>
<td></td>
<td>- Left</td>
</tr>
<tr>
<td></td>
<td>- Right</td>
</tr>
<tr>
<td></td>
<td>- Top</td>
</tr>
<tr>
<td></td>
<td>- Bottom</td>
</tr>
<tr>
<td></td>
<td>When the image is centered, the label will not be visible inside the button.</td>
</tr>
</tbody>
</table>

Figure D-21: Field Properties—Button, Image tab
Form and Application Objects

The functionality to gray out disabled buttons is part of the button feature; you need not use different bitmaps to represent the different button “states” (for example, normal, grayed, depressed, reversed).

There are no restrictions on the size of the bitmaps, either in dimension in the form or in disk space. However, all forms are stored in memory, and forms with large images will cause a performance decline. Furthermore, having images contained in multiple views causes additional copies of the same images to be stored in memory. On the web, large images will increase the download time. For best performance, use images with small file sizes.

### Navigation Items properties

From the Navigation Items tab, you can add the menus and menu items to your horizontal and vertical navigation fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scale Image To Fit</td>
<td>Scales the image to fit the size of the button. If this option is not selected, the image might be cropped, and you might need to resize the button frame.</td>
</tr>
<tr>
<td>Maintain Aspect Ratio</td>
<td>Maintains the proportions of the image when the image is resized to fit the size of the button. This option is enabled when you select Scale Image To Fit.</td>
</tr>
<tr>
<td>Change Image</td>
<td>Enables you to browse to and choose an image for the button. You can choose files that are in .jpg, .jpeg, .bmp, and .dib formats. The image you choose appears in the Image Preview region.</td>
</tr>
<tr>
<td>Save Image To File</td>
<td>Enables you to save the image to a different file or folder.</td>
</tr>
<tr>
<td>Clear Image</td>
<td>Deletes the image within the button.</td>
</tr>
</tbody>
</table>

The functionality to gray out disabled buttons is part of the button feature; you need not use different bitmaps to represent the different button “states” (for example, normal, grayed, depressed, reversed).

There are no restrictions on the size of the bitmaps, either in dimension in the form or in disk space. However, all forms are stored in memory, and forms with large images will cause a performance decline. Furthermore, having images contained in multiple views causes additional copies of the same images to be stored in memory. On the web, large images will increase the download time. For best performance, use images with small file sizes.
The following table describes the buttons and field on the Navigation Items tab.

<table>
<thead>
<tr>
<th>Button or Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Navigation Items</td>
<td>Enables you to create and edit items on your navigation field. For information about creating menus and menu items, see “To add a menu to a navigation field” on page 206.</td>
</tr>
<tr>
<td>Attach Orphaned Navigation Items</td>
<td>Enables you to attach deleted menu bar items and items from navigation fields that were deleted and are no longer attached to anything.</td>
</tr>
</tbody>
</table>
| Fire Workflow Again on Selected Item    | Defines whether workflow is fired again. The options are:  
  - Do not fire workflow (the default)  
  - Fire workflow  
  If you select Fire workflow, the workflow will fire even if the navigation item was already selected. |
Pages properties (page holder fields only)

From the Pages tab, you can set the order of the tabbed pages and create any new tabbed pages that you want to appear in the page holder.

**Figure D-23: Field Properties—Page Holder, Pages tab**

The Available Pages field lists the name (followed by the label) of each existing page field.

**Note:** Use the Field Properties window for each page to edit the names and labels of the three pages, for example, Page(Page). To avoid this step, remove the pages, and use the Create a New Page to name and label each page.

**To set properties for the page holder**

1. Open the properties window for the page holder.
2. Select the Pages tab.
3. Determine the number of pages you want available in the view by selecting the check box to the left of each page.
   
   If there is only one form view and you uncheck the page, the page is deleted and any fields on the page will be removed from the view.

**Note:** Multiple rows of tabs are not supported in forms viewed in a browser.
4 Click Create a New Page to add a new page to the page holder field. You will be prompted for a label and a name. The label you enter appears on the tab (if tabs are displayed). The name identifies the field in the database, and it must be unique. The name also enables you to identify a different tab label for different views.

5 To delete a page, select it and click Delete Page.

6 Use the Move arrows to change the order of pages in the view.

7 To display the page holder field without a border and without page tabs in web clients, select the Tableless Borderless Page Holder check box. You must create workflow to allow users to navigate from one page to another. See “Workflow considerations for page fields” on page 171 for more information.

8 Choose File > Save Form to save your changes.

Permissions properties

Use the Permissions tab to define permissions for a field. Field permissions define which access control groups receive View or Change access for fields. For more information, see the detailed discussion in “Defining permissions for individual or multiple AR System objects” on page 90.

Related Workflow properties

The workflow objects related to the field are listed in the Field Properties window when you select the Related Workflow tab.
The Related Workflow tab will not appear if any of the following statements are true:

- The search database reference form does not exist. (You can create it by choosing Tools > Set Up Search Database.)
- The search database was never synchronized in the current server. (You can synchronize the database by choosing Tools > Sync Search Database.)
- The search database is currently running.

For more information about setting up the search database, see the Optimizing and Troubleshooting AR System guide.
Results Color properties

The Results Color tab enables you to set the colors for list view, results list, and alert list table fields after a search (or refresh). For example, you can color all New requests red and all Assigned requests green.

Tree views do not use Results Color properties.

Figure D-25: Field Properties—Table Field, Results Color tab

The following table describes the available results color properties.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection Field</td>
<td>Defines the selection field that contains the items to which you will be adding color in the results list. You can define results color for only one selection field per form.</td>
</tr>
<tr>
<td>Value</td>
<td>Defines the item to which you are adding color. You can add color to each item in the selection list by selecting an item and a color, selecting another item and color, and so on.</td>
</tr>
<tr>
<td>Color</td>
<td>Defines the color that will be added to the selected value (selection field item). You can select Default Color, or you can select Custom Color and choose a color from the color palette.</td>
</tr>
</tbody>
</table>
Shared Fields properties (page fields only)

Use the Shared Fields tab to display the same fields in all of the pages in a page holder in a view.

When you select the check box to the left of each field, that field will be shared and visible in the same position on each page. Fields that are not selected appear only on one page.

Figure D-26: Field Properties—Page Holder, Shared Fields tab

If you share an opaque trim box across page fields, there is no logical way to make sure the box is behind all fields on each page. To make sure that users can view the fields on each page field, make sure such boxes are transparent.

In BMC Remedy User, fields that are shared across page fields are hidden if all the page fields in the page holder are hidden. However, if a page holder is shared across all pages of a hidden outer page holder, some of the pages in the inner page holder might be visible. The contents of these pages are not visible.

On the web, fields that are shared across page fields (including nested page holders and page fields) are visible even though all the page fields are hidden. This is because shared fields are children of the page holder and independent of page display properties. Use workflow to hide these shared fields or to hide the page holder.
Sort properties (table fields)

From the Sort/Levels tab, you can define the order in which requests appear in a list view, tree view, results list, or alert list table field. (For tree views, you are defining the visible levels of the tree.) For example, you can sort requests first by department and then by products within a department.

**Note:** You can sort on columns that reference display-only fields in BMC Remedy User, but not in web clients.

The sort/levels properties (and the Max Rows property) determine:

- The sequence in which rows of data appear when the field is refreshed (that is, the primary sort column, the following sort columns, and columns that do not participate in the sort).
- The default sort direction (per column) that users see before they click a column title to re-sort data. (List view, results list, and alert list fields only)
- The visible tree structure. (Tree views only)

![Field Properties— Table Field, Sort tab](image)

For more information, see “To define the sort order for table fields” on page 187.
Table Labels properties

From the Table Labels tab, you can define hyperlinks, button labels, and informational strings displayed in table, results list, and alert list fields. You can define strings for functions that are supported in both BMC Remedy User and web clients, such as Select All, Delete, Refresh, and Report. You cannot define strings for functions that are not supported on web clients, such as Refresh All in Table, Print Table, and Clear Table.

Tree views do not use Table Labels properties.

Figure D-28: Field Properties—Table Field, Table Labels tab
The following table describes the field properties in the Table Labels tab.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Not Loaded String (Table fields only)</td>
<td>The message that appears when a table is initially displayed.</td>
</tr>
</tbody>
</table>
| Number of Entries Returned String          | The message that appears when data is loaded into the table or results list field. In forms viewed in a browser, the message is displayed in the table or results list header. In BMC Remedy User (where there is no table header), the message appears in the status bar. You can pass the following four parameters to this string: 
  {0} starting row number
  {1} ending row number
  {2} total number of rows returned
  {3} maximum number of rows that could have been returned
  If chunking is not enabled, this field will default to “{2} entries returned - {3} entries matched.”
  If chunking is enabled, this field will default to “Showing {0} - {1} of {3}.” |
<p>| Auto Refresh String (Alert list fields only) | The message that appears when the Alert Refresh Interval in the Web tab of the AR System User Preference form is set to a value greater than 0. This is an informational message only. The presence of this string does not enable or disable auto refresh. For more information about user preferences, see the Configuring AR System guide. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row Header/Web Drill-Down Column</td>
<td>This field is not supported for forms using version 6.3 and later versions of the mid tier. Table fields on the web behave like they do in BMC Remedy User. If a user double-clicks on a row, the selected record will appear in the new window. For tables viewed on a pre-6.3 mid tier, the following information applies: If Table Drill-Down is enabled in the Display tab, the Row Header/Web Drill-Down Column field specifies which column is used as the drill-down column on the web. The value in each cell in this column becomes a hyperlink to the corresponding request in the supporting form. For alert list and table fields, details are displayed in another form in the same browser window. For results list fields, details are displayed in the same form. If Table Drill-Down is enabled and no Web Drill-Down Column is specified, the first defined visible column in the table becomes the drill-down column by default. If the Display Type property for this first column is Editable and the first column is the drill-down column, drill-down will not work. For information, see “Display properties” on page 428. If the drill-down column is hidden by workflow, you cannot drill-down. This is true even if drill-down column was selected by system default—the system will not designate another drill-down column. For more information, see the discussion on the Change Field in the Workflow Objects guide. If a Set Fields action changes the value of a drill-down cell, the drill-down ability of that cell will be lost (the cell value is no longer a link).</td>
</tr>
<tr>
<td>Select Column Label</td>
<td>The column header that will appear above the selection column when in Accessibility mode (Internet Explorer browsers only). For all other browsers not in Accessibility mode, this string will be ignored at runtime.</td>
</tr>
</tbody>
</table>
### Footer Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select All</td>
<td>The label of the hyperlink to select all rows.</td>
</tr>
<tr>
<td>Deselect All</td>
<td>The label of the hyperlink to deselect all rows.</td>
</tr>
<tr>
<td>Refresh Button</td>
<td>The label for the refresh button. In version 6.3 and later versions of the mid tier, the Refresh button will appear above the table to the right (instead of in the footer as in previous versions).</td>
</tr>
<tr>
<td>Preferences</td>
<td>When a value is entered in this field, a right-mouse menu is added to the table field in BMC Remedy User, and a button is added to the table header in the web. The value entered becomes the label for the menu or the button. The default value is “Preferences.” To disable table column preferences, clear the Preferences field. When this field does not have a value, no preference menu or button appears in the table field, and users cannot set preferences for the table.</td>
</tr>
<tr>
<td>Report Button</td>
<td>The label for the report button.</td>
</tr>
<tr>
<td>(Table and results list fields only)</td>
<td></td>
</tr>
<tr>
<td>Delete Button</td>
<td>The label for the delete button.</td>
</tr>
<tr>
<td>(Results list and alert list fields only)</td>
<td></td>
</tr>
<tr>
<td>Read Button</td>
<td>The label for the button to mark an alert as read.</td>
</tr>
<tr>
<td>(Alert list fields only)</td>
<td></td>
</tr>
<tr>
<td>Mark as Unread Button</td>
<td>The label for the button to mark an alert as unread.</td>
</tr>
<tr>
<td>(Alert list fields only)</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** The Delete Button, Read Button, and Mark as Unread Button fields are only available for the web. BMC Remedy User expects you to open the Alert Events form or use the View Alerts option.
Table/Tree Property properties

The Table/Tree Property tab enables you to define properties about the source form, fields, rows, and columns.

**Figure D-29: Field Properties—Table Field, Table Property tab**

These properties are described in the following table.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>List view and tree view table fields only. Defines the server that contains the form used in the table field.</td>
</tr>
<tr>
<td>Form</td>
<td>List view and tree view table fields only. Defines the supporting form that is used in the table field.</td>
</tr>
<tr>
<td>Restricted List</td>
<td>List view and tree view table fields only. Defines if the table field is restricted only to the forms in a deployable application.</td>
</tr>
<tr>
<td>Fields On Form</td>
<td>Specifies data fields from the supporting form, and display-only fields from the current form. The Field Type column indicates whether the field is a data or display-only field.</td>
</tr>
</tbody>
</table>
### Field As Column

Enables you to specify which fields from the supporting form will appear in the table field as columns. For a tree view, these column fields are used as levels within a tree structure. They are all available for workflow, but they might not be visible as levels in the tree. The order of this list cannot be changed in this field. (The visibility and level order are set in the Sort/Levels tab.)

The names listed beneath Column Title represent the field names that you want to appear in the table field. The names listed beneath Field represent the field names used in the supporting form.

The up and down arrows enable you to arrange the order in which the fields will appear in the table field. (For tree views, it is not necessary to change column order. The order is designated on the Sort/Levels tab.)

You can add as many as 128 columns to a table.

You can add character fields with a data length of 0 or more than 255, but these fields will not appear on the Sort/Levels tab.

### Max. Rows

Specifies the maximum number of rows that will appear. If you enable chunking, this setting is ignored.

If users select Refresh or Refresh Table, the maximum number of rows that the administrator defines is displayed. In BMC Remedy User, if users select Refresh All in Table, the maximum number of rows that the administrator defines is ignored. (On web clients, there is no Refresh All function. For more information, see “Table Labels properties” on page 449.)

**Note:** For results list fields on the web, there is no default maximum for the number of items returned. If you do not specify a value greater than zero for Max. Rows, and if the Limit Number of Items Returned user preference is not selected, then the results list field will return unlimited items during a search.

### Qualification

In a list view, tree view, or alert list table field, enables you to define the search criteria to determine which requests appear. You can use fields from the current form, fields from the selected supporting form, or a keyword in the qualification statement. The way qualifications function in table fields is similar to how they are used with the Set Fields action. For information about building qualifications, see the Workflow Objects guide.

The character limit for strings in a qualification is 4K.

### Sample Server Name (Advanced)

In a list view or tree view table field, enables you to define the server and forms names dynamically. Based on values that workflow or users enter into certain fields, you can dynamically change which server’s and form’s entries will be displayed. For more information, see “Defining advanced table field functionality” on page 191.
Views properties

From the Views tab, you can specify the form views in which a field will appear. Even if you remove a field from all views, it is still usable in workflow. The Views tab is present in the properties window only if you have defined multiple views for a form, as described in “Creating form views” on page 256.

Figure D-30: Field Properties—Character Field, Views tab

When you select the Views tab, you will see every view defined for the form with the exception of the current form view. To remove a field from the current form view, you must use the Fields in View dialog box described in “Including and excluding fields from form views” on page 299.

If you have defined multiple views for your form, the new field is added to the view of the current form. The new field is also added to all other views of the form depending on how your preferences are set (as described in the Form Preferences discussion in the Getting Started guide).

A field can be hidden in one form view and displayed in a different form view. Some field properties affect all form views while other field properties do not. For more information about form views, see Chapter 8, “Creating and managing form views” and Installing and Administering BMC Remedy Mid Tier guide.
To select the form views that display a field

1. Open the form with which you want to work.
2. Double-click the field to open the properties window.
3. Select the Views tab.
   The views defined for the form appear in the Not In Views list and the In Views list.
4. From the In Views and Not In Views lists, specify the views in which the field will and will not appear.
5. Choose File > Save Form to save your changes.
Data archiving enables you to make archive copies of data on a specified form.

The following topics are provided:

- Understanding data archiving (page 458)
- Configuring data archiving for a form (page 459)
- Saving the main form and deleting archive forms (page 462)
- Properties of archive forms (page 464)
- Configuring data archiving for a server (page 466)
Understanding data archiving

The archive feature of AR System provides a convenient way to store data periodically on the main form; this reduces the amount of data accessed during searches on the main form thus improving system performance. Data archiving works with data related to a form; it applies to all types of forms except display-only forms. The main form is the form on which archive is set (data is read from this form) and the archive form is the form to which data is copied.

The archive form is a regular type of form with some special behaviors. It must be on the same server as the main form. You can designate an existing form as the archive form, and AR System will use that form if it meets the property requirements of an archive form. See “Properties of archive forms” on page 464 for details.

Note: The AR System will automatically create the archive form for you if it does not exist. You do not have to create the archive form manually.

The following archive types are available:

- Copy to Archive and Delete from Source— The data from the main form is copied to the archive form and deleted from the main form.
- Copy to Archive— The data from the main form is copied to the archive form; the data remains on the main form. This is the only archive type that is available for vendor forms.
- Delete from Source— The data is deleted from the main form; no archive form is involved.
- None— Select this option to delete the archive settings for the form. The form will not be archived.

Archive Type None and the Enable option settings

In the Archive tab of the Form Properties dialog box, the Archive Type selection of None results in a break in the connection between the archive form and the main form. All archive information is lost, and the archive form is treated as a regular form. If you clear the Enable checkbox, the archiving information is still preserved and archiving can be resumed when you select the Enable checkbox.
Qualifications
You can select the data to be archived based on a qualification; if you do not specify a qualification, all the data in the form is archived. Consider the effect on performance when using this option.

Archive process
The AR System server has a special user called AR_ARCHIVER to perform data; if you run a filter log file, you can see this entry in the log file. The AR System server also reserves an internal thread for archiving.

Licensing
When you license an application and license the main form, the archive form will be licensed also.

Deleting archive fields from the main form
When deleting multiple fields from the main form, AR System will attempt to delete those fields from the archive form as well. If any of those fields contain data, then none of the fields will get deleted from the archive form. However, if the fields are deleted one by one from the main form, then the fields that do not contain data will be deleted from the archive form.

Configuring data archiving for a form
You can use the data archiving feature to back up and delete data from forms.

To configure data archiving for a form
1. In BMC Remedy Administrator, open the form to configure for data archiving.
2. Choose Form > Form Properties.
3. Click the Archive tab.
4 Select one of the following archiving types.

- **Copy to Archive and Delete from Source**—Select this option to copy data in the form to an archive form on the same server. The system will copy all data in the form if the Qualification field is left blank, or it will copy only entries fulfilling the criteria specified in a qualification. The data that is copied to the archive form is then deleted from the main form.

- **Copy to Archive**—Select this option to copy to an archive form on the same server. The system will copy all data in the form if the Qualification field is left blank, or it will copy only entries fulfilling the criteria specified in a qualification. The data that is copied to the archive form is retained on the main form.

- **Delete from Source**—Select this option to delete data in the main form. The system will delete all data in the form if the Qualification field is left blank, or it will delete only those entries fulfilling the criteria specified in a qualification.

- **None**—Deletes the archive settings for the form. The form will not be archived.
5 Select the Enable option to enable archiving of the form.

6 If you selected Copy to Archive and Delete from Source or Copy to Archive type, enter a name for the archive form. For example, if you are archiving data on the Application Statistics form, you might name the archive form Application Statistics - Archive.

If the form that you entered does not exist, it is automatically created by AR System with the properties listed in “Properties of archive forms” on page 464. If the form exists, it must have these required properties of an archive form.

7 If you selected Copy to Archive and Delete from Source or Copy to Archive type, you can select the No Attachments and No Diary Fields options to exclude this data from the archive form. These options can be selected to save space in the archive form if the main form has large attachments or a large amount of data in diary fields.

If you select Copy to Archive and Delete from Source, and select the No Attachments and No Diary Fields options, the data in the attachments and diary fields are deleted from the main form and are not copied to the archive form.

8 Click Set Time to set the day and time to perform data archiving for the specified form.

You cannot set the interval time between each archiving process for less than 1 hour.

9 Enter a qualification if you want to specify a limited amount of data on the form to archive. For example, to archive statistics older than thirty days in the Application Statistics form, enter the following qualification:

‘Time Stamp’ < ($TIMESTAMP$ - 2592000)

Only the selected data will be archived.

10 Click OK.

11 Choose File > Save Form.

12 Open your archive form in BMC Remedy Administrator, and adjust the field layout to suit your requirements.
When the AR System creates a new archive form, it does not place the fields in the view. You must add them to the view. See “Properties of archive forms” on page 464 for more information about these fields.

A warning is displayed the first time you open the form informing you that the AR System will upgrade the layout information so that it looks the same on all clients.

**Figure E-2: Archive form upgrade warning**

After data has been entered into your main form, archiving of the data according to your qualification criteria will take place at the time specified. You can open your archive form, and view archived data with BMC Remedy User.

**Important:** When a view form, join form, or vendor form is archived, the archive form is created as a regular form containing core fields that are not in the source form. You must supply default values for the required Submitter and Short Description core fields in the archive form.

---

### Saving the main form and deleting archive forms

The following sections provide information relating to saving a main form using the Save As function and to deleting an archive form.

**Saving the main form to a different form name**

If you save a main form in BMC Remedy Administrator using the Save As function, the archived settings are not saved to your new form.

**Deleting an archive form**

You can delete an archive form using BMC Remedy Administrator or by using an API call.
Deleting an archive form using BMC Remedy Administrator

You can delete the archive form in BMC Remedy Administrator.

**To delete an archive form**

1. Select the form in the server window
2. From the menu, choose Edit > Delete Form(s).
   You will see a confirmation message.

   **Figure E-3: Archive form deletion confirmation**

   ![Archive form deletion confirmation]

3. Click Yes.
   Another confirmation message reminds you that the form being deleted is an archive form.

   **Figure E-4: Archive form deletion warning**

   ![Archive form deletion warning]

4. Click Yes to delete the archive form and all the data associated with it.
   When an archive form is deleted, archive settings are cleared for the main form. The Archive Type field is reset to None in the Archive tab of the Form Properties dialog box.

Saving the main form and deleting archive forms 463
Deleting an archive form using an API call

To delete an archive form regardless of whether the main form has archiving turned off or on, or when deleting an archive form that is part of a Lock block, use the `AR_SCHEMA_SHADOW_DELETE` option with the `ARDeleteSchema` API call.

If the archive form has data, use the `ARDeleteSchema` API call with both `AR_SCHEMA_DATA_DELETE` and `AR_SCHEMA_SHADOW_DELETE` options. This will delete the archive form with data.

Properties of archive forms

The following sections describe properties of archive forms.

General properties

Each archive form has the following properties:

- The form type is Regular.
- Two extra fields, Original Request ID (ID 450) and Original Create Date (ID 451), are included. These fields contain the Request ID and Create Date from the main form. The Create Date of the archive form can be used as the archive date. The remaining core fields on the archive form contain the same values as the main form.
- Workflow from the main form is not attached to the archive form when it is created. You can add workflow to an archive form, but workflow cannot modify data in an archive form.
- Any changes to the definitions in the main form (such as adding or deleting a field) are applied to the archive form also. You can also change the form and view properties of the archive form.

Note: Make sure that the main form in BMC Remedy Administrator is closed before deleting the archive form, otherwise your archive form might be regenerated and the archive settings might not be cleared. (If your main form does remain open, close it without saving after deleting the archive form.)
Data fields (character, currency, date, date/time, diary, integer, real, decimal, drop-down, radio, check box, attachment, attachment pools, page, and page holder) cannot be modified or added to an archive form. All other field types, such as trim or table, can be added or modified.

The data fields in the main and archive form have identical field limits. The permission on archive forms will always be read access.

Data can be exported from an archive form and data can be imported into an archive form, but existing entries in an archive form cannot be overwritten.

During import, if only the main form is present, archiving will be disabled for that form, and a warning is returned. When archive is enabled for that form, AR System checks to see if the archive form is present. If no form is found, AR System creates the archive form. If the archive form is found, it will be used only if it is a valid archive form.

If the main form belongs to a deployable application, the archive form also belongs to the same application.

If the main form is part of a Lock block, the archive form is part of the same Lock block.

If the main form is made “licensable,” then the archive form is also made licensable.

The archive form cannot be audited or further archived.

**Archive forms created by the AR System**

Archive forms created by the AR System have the following properties in addition to those listed in “General properties” on page 464:

- Any uniqueness constraints (indexes) that exist on the main form are removed from the archive form. You can add indexes to the archive form.

- When archive forms are created, the entry points are cleared. The administrator can add entry points.

- All other form properties are copied over from the main form to the archive form.
Existing forms specified as archive forms

If you specify an existing form to be an archive form, it must follow the properties defined in “General properties” on page 464. Otherwise, you will see an error message.

The Distributed Server Option and archive forms

If a form used in a Distributed Server Operation (DSO) is archived, the distributed fields will be copied to the archive form but will not be updated. You cannot add distributed fields to an archive form.

Configuring data archiving for a server

The data archiving feature is enabled by default on an AR System server. To disable archiving for all forms on a server, select the Disable Archive option on the Configuration tab of the Server Information dialog box. If multiple AR System servers use a single database, you can disable archive on all the servers except one if you want archiving to take place on only one server. If the server is a member of a server group, configure the server group in the AR System Server Group Operation Ranking form to make sure that only one server performs the archiving operation. See the Configuring guide for more information about how to configure server groups.

Errors will be logged in to the arerror.log file; because there is no API there will be no entry in the API log file.

Server events and logging

To create an entry for archiving in the Server Events form, select the Archive option on the Server Events tab of the Server Information dialog box.
If you select the Archive check box, every archive event is logged into the Server Events form.
Appendix E—Archiving data

Figure E-6: Server Events form

The entries are as follows:

Event Type: (14) AR_SVR_EVENT_ARCHIVE_DONE.

Event Cause:

One of the following is entered in the field:

- (1) AR_SVR_EVENT_ARCHIVE_FORM (Copy to archive only).
- (2) AR_SVR_EVENT_ARCHIVE_DELETE (Delete from source only).
- (3) AR_SVR_EVENT_ARCHIVE_FORM_DELETE (Copy to archive and delete from source).

Event Date: Date and time of the end of the archive operation.

Event Details 1: Source form name.

Event Details 2:

- Copy to archive and Copy to archive and delete from source
  \(<\text{number of records transferred}> : <\text{Total number of entries attempted}>\)

- Delete from source
  \(<\text{number of records deleted}> : <\text{Total number of entries attempted}>\)

Event Details 3: Destination form name.
Appendix F: Audit

This section contains excerpts from the AR System documentation set that contains new information related to the new audit feature.

This section contains the following topics:

- Understanding auditing (page 470)
- About audit styles (page 470)
- Configuring auditing (page 476)
- Auditing view and vendor forms (page 482)
- Considerations for join forms (page 482)
- Fields Changed flag fields (page 484)
- The Distributed Server Option and audit forms (page 485)
- Audit processing and filters (page 485)
Understanding auditing

The AR System audit feature lets you keep track of changes to data in a form. Any create, update, merge, or delete actions for data in a main form can be recorded in an audit form or log form. Regular, view, join, and vendor forms can be audited. Display-only forms cannot be audited.

AR System will automatically create the audit form or log form if it does not exist. You do not have to create these forms manually.

Using the audit feature requires configuration at the form level and at the field level. At the form level, you enable auditing for a form, specify an audit style, and specify the name of the form that will contain the audited data. If the audit form does not exist, AR System will create it. At the field level, you specify whether a field should be audited, copied, or audited and copied.

Audit indicates that a change to this field triggers audit processing. Copy indicates that the field value will be copied to the audit form. Audit fields that have not changed will not be copied. Audit and copy indicates that the field will trigger audit if the field is changed, but if it is not changed, it will still be copied (that is, behave like a copy field).

You can selectively audit entries by providing an audit qualification. Even if the values of audit fields have changed, if the audit qualification fails, then the audit does not occur.

About audit styles

When you configure a main form for auditing, you specify whether to perform a form-style audit or a log-style audit.

Since AR System updates the audit forms for both styles, a special user named AR_AUDITOR has been created to perform the audits. You will see this name in the Last Modified By field for all audits.
Form and Application Objects

Form-style audits

A form-style audit records data from the main form into an audit form. The audit form is a regular form that serves as the destination for data audited in the main form. The audit form resides on the same server as the main form and contains the same fields as the main form. In addition, the audit form contains several audit-specific fields. See “Audit form fields” on page 471.

When you configure a main form for a form-style audit, you specify a name for the audit form. When a main form is audited, data from the main-form fields configured for auditing is copied to corresponding fields on the audit form. If there are fields in the main form not configured for auditing, the corresponding fields on the audit form are left blank.

Audit form fields

The audit form is a regular form (except when auditing a join form) that serves as the destination for data audited in the main form. The data fields in the main form and the audit form have identical field permissions and field limits. On the audit form, however, field permissions and field limits cannot be modified.

Important: When deleting multiple fields from the main form, AR System will attempt to delete those fields from the audit form as well. If any of those fields contain data, then none of the fields will get deleted from the audit form.

However, if the fields are deleted one by one from the main form, then the fields that do not contain data will be deleted from the audit form.
Each audit form contains the following audit-specific fields. Special fields will not be in the view. You must add them to the view.

### Audit form field name | Description
--- | ---
**Action** | Indicates the actions that triggered the audit. The options are:
- 2 — Set
- 4 — Create
- 8 — Delete
- 16 — Merge

**Fields Changed** | Contains the database names of the audit fields that changed. The syntax for the list is as follows:
<br>;<database_name>; ;<database_name>; ;<database_name>;

**User** | Indicates the user that modified the entry in the main form.

**Original Request ID** | The Request ID of the entry being audited.

**Audit Date** | The date and time when the audit occurred.

**Last Modified By** | Indicates who created the entry in the audit form last. (AR_AUDITOR will always be the user who creates the entries.)

**Audit Join Key** | Used for join form auditing. AR System maintains this field.

AR System will not create these fields as part of any view, so you must add them to the view to use them.
Audit form characteristics

Audit forms have the following characteristics:

- When an audit form is created, the workflow from the main form is not copied over to the audit form. You can add workflow to an audit form, but workflow cannot modify data in an audit form.

- Limit information of the fields must be the same as the corresponding fields in the main form. The permission on the Audit forms will be read access.

- Any changes to the definitions in the main form (such as adding or deleting a data field) are applied to the audit form also.

- You can change the form and view properties of the audit form.

- Data fields (character, currency, date, date/time, diary, integer, real, decimal, drop-down, radio, check box, attachment, attachment pools, page, and page holder) cannot be modified or added to an audit form. All other field types, such as trim, or table, can be added or modified.

- Data can be exported from an audit form and data can be imported into an audit form, but existing entries in an audit form cannot be overwritten.

- Only an administrator can delete entries.

- While importing main forms that are audited, if the main form is audited Copy style and the shadow form is not found then audit for the main form is disabled and a warning (Form not Found. ARError 303) is returned.

- During an import in place, if the main form has fields added or deleted, those fields will also be added to the audit form.

- If the main form belongs to a deployable application, the audit form also belongs to the same application.

- If the main form is part of a Lock block, the audit form is part of the same Lock block. If a field in the main form is audited after locking the form, the corresponding flag field is not created. (For more information, see “Fields Changed flag fields” on page 484.)

- If the main form is made “licensable,” then the audit form is also made licensable.

- An audit form cannot be further audited, but it can be archived.

- Archive forms cannot be audited.
If an audit form is created by the AR System, it has the following additional properties:

- Any uniqueness constraints (indexes) that exist on the main form are removed from the audit form. You can add indexes to the audit form.
- When audit forms are created, the entry points are cleared. The administrator can add entry points.
- All other form properties are copied over from the main form to the audit form.
- The audit form by itself cannot be imported. Either the main form by itself or both the main and audit forms can be imported.
- When the audit form is created for the first time, all fields (including non-data fields) are created. After that, if non-data fields are added to the main form, they are not added to the audit form.

Deleting an audit form using an API call

To delete an audit form regardless of whether the main form has auditing turned off or on, or when deleting an audit form that is part of a Lock block, use the `AR_SCHEMA_SHADOW_DELETE` option with the `ARDeleteSchema` API call.

If the audit form has data, use the `ARDeleteSchema` API call with both `AR_SCHEMA_DATA_DELETE` and `AR_SCHEMA_SHADOW_DELETE` options. This will delete the audit form with data.

Log-style audits

A log-style audit records data from the main form into a log form. The log form is a regular form that serves as the destination for data audited in the main form. If the log form does not exist, it is created by the AR System server. The log form resides on the same server as the main form. Unlike the audit form, the log form contains only audit-specific fields. See “Log form fields” on page 475. When you configure a main form for a log-style audit, you specify a name for the log form. When a main form is audited, AR System copies values from the main form to a text field in the log form.

**Important:** A log-style audit form can contain data from multiple main forms.
Log form fields

Unlike the audit form used for form-style auditing, the log form does not contain the same fields as the main form. Instead, the log form contains the following fields in addition to the core fields:

<table>
<thead>
<tr>
<th>Log form field name</th>
<th>Description</th>
</tr>
</thead>
</table>
| Action              | Indicates the actions that triggered the audit. The options are:  
  - 2 — Set  
  - 4 — Create  
  - 8 — Delete  
  - 16 — Merge |
| Fields Changed      | Contains the database names of the audit fields that changed. The syntax for the list is as follows:  
  ; <database_name>; <database_name>; <database_name>; |
| User                | Indicates the user that modified the entry in the main form. |
| Original Request ID | The Request ID of the form being audited. |
| GUID                | This field will be set if the main form contains a field with ID 179. |
| Log                 | A list of field-value pairs that represent the changes to the main form. The syntax is as follows:  
  <fieldname1>:<fieldvalue1>  
  <fieldname2>:<fieldvalue2> |
| Form name           | The name of the main form. |
| Log Key 1           | Fields that help in searching of log entries. See the following section for more information. |
| Log Key 2           | |
| Log Key 3           | |
| Audit Date          | The date and time when the audit occurred. |
| Last Modified By    | Indicates who created the entry in the audit form last. (AR_AUDITOR will always be the user who creates the entries.) |
Log keys

When a log form is created, it contains special character fields named Log Key 1, Log Key 2, and Log Key 3. These fields can help in searching of log entries.

In BMC Remedy Administrator, a field can be set to any of these Log Key fields. During an audit, the value of this field will go to the key that was selected.

For a form, only three keys are available, and no two fields can be set to the same log key. Also, log keys cannot be set for diary and attachment fields.

Log form characteristics

Log forms have the following characteristics:

- Has a fixed set of fields as mentioned in the previous section.
- Does not belong to any application when it is created.
- Does not belong to any lock block when it is created.
- While importing main forms that are audited, if the main form is audited Log style and the Log form is not found then audit for the main form is turned off and a warning (Form not Found. ARError 303) is returned.

Configuring auditing

To configure auditing, you configure a form as the source of audited data and specify which fields on the main form should be audited.

Configuring a form for auditing

The following procedure describes how to enable auditing for a form.

To enable auditing for a form

1. In BMC Remedy Administrator, open a form.
2. Choose Form > Form Properties to open the Form Properties dialog box.
3. Click the Audit tab.
In the Audit Style field, choose the type of audit you want to perform.

- **None**: No auditing is performed.
- **Form**: A snapshot of the audited form is saved to an audit form you specify. Only the audit and copy fields in the audit form will contain values.
- **Log**: Whenever a form is saved after an audit field or set of fields changes values, an entry is created in the log form you specify.

5 Select the Enable check box.

You can clear the Enable check box to disable audit functionality temporarily. Clearing the Enable check box leaves any other audit configuration values you might have specified intact.

6 If you specified a form audit, specify an audit form name in the Audit Form field.

If you specified a log audit specify a log form name in the Log Form field.

The audit or log form you specify is created when you save the main form.
Optionally, specify a qualification. The incoming entry will be audited only if it satisfies this qualification.

Click OK.

In the Audit tab of the Form Properties dialog box for the main form specifies the Audit form’s name.

The audit form’s Audit tab specifies the Audit From Form that specifies the main form.

In case of log style, the log form’s Audit tab will contain an Audit From Ref Count field, which indicates the number of forms using the log form.

### Specifying fields to be audited

On a form configured for auditing, you specify which fields should be audit fields, which should be copy fields, and which should be audit and copy. Both audit and copy field values are copied to the audit or log form, but only changes to audit fields trigger audit processing.
Audit fields are copied only if their value changes. For copy fields, either the value in the current transaction is taken (if present) or the value is taken from the database. If creating an entry and no value is entered for a copy field, then nothing is copied. Fields specified as audit and copy will trigger an audit and be copied if changed. If not changed, they will behave like copy fields.

**Note:** System fields, including Create Date, Last Modified by, cannot be audited.

▶ **To specify a single field to be audited**

1. In BMC Remedy Administrator, open a form for which auditing has been enabled.
   
   See “Configuring a form for auditing” on page 476 for more information.

2. Double-click the field you want to configure for auditing.
   
   The Field Properties dialog box appears.

3. Click the Attribute tab.
In the Audit Option menu, select one of the following options:

- **None**—Changes to this field are not recorded by any audit processing.
- **Audit**—Changes to this field trigger audit processing and its new value will be recorded in the audit form or log form, depending on the audit style you specified at the form level. If the value does not change, its value will not be recorded.
- **Copy**—Either the database value or the value in the current transaction of present will be recorded during an audit, but will not trigger audit processing.
- **Audit and Copy**—Changes to this field trigger audit processing. If the value has not changed, then the value from the database will be recorded (similar to the behavior of the Copy option).

If the form is audited in log style, then this tab will instead contain an Audit Log Key menu with the following options:

- **Key 1**—The value of this field will appear in the Log Key 1 field in the log form.
- **Key 2**—The value of this field will appear in the Log Key 2 field in the log form.
- **Key 3**—The value of this field will appear in the Log Key 3 field in the log form.

5 Close the Field Properties dialog box and save the form.

**To specify multiple fields to be audited**

1 In BMC Remedy Administrator, open a form for which auditing has been enabled.
   See “Configuring a form for auditing” on page 476 for more information.
2 Select the fields you want to configure for auditing.
3 Choose Forms > Multiple Field Properties. The Set Multiple Field Properties dialog box appears.

4 From the Audit Option menu, select one of the following options:

- **Unchanged** - Selected by default. Use this option if you are setting other field properties and you do not want to change the Audit Option value.
- **None** - Changes to this field are not recorded by any audit processing.
- **Audit** - Changes to this field trigger audit processing and its new value will be recorded in the audit form or log form, depending on the audit style you specified at the form level.
- **Copy** - Either the database value or the value in the current transaction of present will be recorded during an audit, but will not trigger audit processing.
- **Audit and Copy** - Changes to this field trigger audit processing. If the value has not changed, then the value from the database will be recorded (similar to the behavior of the Copy option).

**Note:** The value you choose from the Audit Option menu will apply to all of the selected fields.

5. Click OK.
6. Save your main form.

**Auditing view and vendor forms**

When vendor or view forms have form-style auditing, the audit form created is a regular form. You might have to provide default values for the Short Description and Submitter fields.

**Considerations for join forms**

Both form-style and log-style auditing are available on join forms. An audit of a join form is triggered if the join form contains audit fields and the audit qualification (if present) is TRUE.

**Form style**

In the case of a form-style audit, the join forms' underlying forms must also be configured for form-style audit and must be enabled. AR System creates the join forms' audit form as a join form of the underlying forms' audit forms and use the Audit Join Key fields in the join criteria, as shown in Figure F-3.
After AR System creates the audit join form, you can modify the join criteria for the audit form if you want to add more qualifications.

Figure F-4 illustrates how join-form audits work in join forms. If Join Form 2 satisfies the join audit criteria, then an audit occurs for Forms A, B, and C (irrespective of A, B, and C’s audit qualification), and audit records will be visible by way of Audit Join Form 2.

If Join Form 2 fails the join audit criteria but Join Form 1 satisfies the audit criteria, then an audit will occur for Forms A and B, and audit records will be visible by way of Audit Join Form 1, but not Audit Join Form 2. If Form C has audit enabled, then Form C will be audited, and Audit Form C will have entries, but audit data cannot be viewed from Audit Join Form 2.

In summary, for the first audited join form that passes the join audit criteria, AR System will generate a unique GUID and will use this GUID to update the Audit Join Key fields in this join form’s underlying audit forms. Since the audit join form has a join criteria based on the Audit Join Key, the audit join form will display only data entered or modified in the corresponding audited join form. If the base forms are modified directly, then these base forms will be audited, but the audit join form will not display the modifications because the value of the Audit Join Key fields will be empty.
Log style

In the case of a log-style audit, a regular form is created and contains the special log-style audit fields.

**Important:** Data entered in the join form is copied to the log form regardless of whether any of that data is pushed to the underlying base forms. This means that the data captured in a log-style audit form might not reflect the content of the main form or its underlying base forms.

Fields Changed flag fields

For a form-style audit, a special Fields Changed field is created, and this field contains the database name. Using this field to search for entries can be inefficient. Using this field to search for audited fields in the audit join form can also be difficult.
To solve this issue, for every audit field in the form, AR System will create a flag field. These fields are not in any view, and they are created with the name in the format, for example, \texttt{F_<field\_ID>_C}, where \texttt{<field\_ID>} is the field ID of the audited field. (If a join form is audited, then the \texttt{<field\_ID>} is the field ID of the audit field in the join form.)

After the Fields Changed flag field is created, it remains until the main field is deleted. (If a field in the base form is not audited or is a copy field, then the Fields Changed flag field is not created.)

\textbf{Note:} If a field in the base form is set as audit after the audit join form is created, the flag field is created in the base's audit form, but not in the audit join form.

When auditing is triggered, and if the Audit field changes value, the corresponding Fields Changed flag field will contain a “1” to indicate that the field changed; otherwise, it will remain empty.

\textbf{The Distributed Server Option and audit forms}

DSO will work on audit and log forms, but the Transfer and Update flags will not be updated.

\textbf{Audit processing and filters}

Both audit forms and main forms can have filters. Auditing occurs at the end of Filter Phase 2 for the main form that is audited. For example, if Form A has Set Fields and Push Fields filter actions, and Form A has Audit enabled, then the audit will occur after the Set Fields and Push Fields actions have been executed.

If an error occurs in the transaction (including errors while auditing), then the entire transaction is rolled back.

For more information about filter phases, see the Workflow Objects guide.
Importing and exporting object definitions and applications

This section describes the administrative tasks involved in managing AR System objects (for example, for backup purposes). Managing objects of a particular server involves exporting definitions from one server and importing them to another server or into a source control system that has been integrated with the AR System server. A definition is the description of the structure in which the objects (forms, menus, filters, escalations, active links, applications, packing lists, and guides) in AR System are organized, identified, and manipulated in the AR System server.

The following topics are provided:

- AR System object definitions (page 488)
- Exporting and importing definitions (page 490)
- Exporting and importing deployable applications (page 500)
- Exporting and importing data with deployable applications (page 504)
- Exporting and importing view definitions (page 510)
- Locking objects (page 511)
Note: Exporting and importing definition (.def or .xml) files is not the same task as importing data (.arx, .asc, .csv, or .xml) file records. You must use BMC Remedy User to export the data to a file, as described in the help for BMC Remedy User and in the Configuring guide. To import data, you must use BMC Remedy Import, which is also described in the Configuring guide.

AR System object definitions

You now can use two different methods for exporting and importing object definitions:

- You can move AR System objects from one server to another by exporting the object definitions to a file, and then importing the definitions from that file to a server on the same machine, or a server on another machine. When exported, objects can be locked to prevent them from being modified or viewed. For more information, see “Exporting and importing definitions” on page 490.

  Object definitions are the descriptions of the structures featured in an application or AR System solution. They contain no user data or entries. You can export object definitions by type (for example, all menus), or you can export all object definitions that pertain to a server.

- You can export and import deployable applications from one server to another. Exporting and importing a deployable application automatically includes all the forms, workflow, menus, web services, packing lists, localized messages, reports, application roles, support files, and the form data (if specified) that are referenced by the application. For more information, see “Exporting and importing deployable applications” on page 500.

  To learn about exporting and importing data, see the Configuring guide.

For information about creating reports for the web and exporting data, see the Installing and Administering BMC Remedy Mid Tier guide.

When you export object definitions to a file, you choose a file type—either AR System definition or AR System XML.
The AR System definition (*.def) file type

The AR System definition file format is the default format for exporting object definitions. It is a proprietary format for storing the definitions of AR System structures. You can include several applications within the same .def file.

The AR System XML (*.xml) file type

Choosing the AR System XML format as the file format for exported objects produces an XML document that is comparable to the AR System definition file format. It is designed to follow the syntax of the XML specification 1.0. Specifically, every AR System object type will have an associated structure definition in XML, which is specified by the XML Schema Definition (*.xsd) file. The *.xsd files reside on the AR System server and are used to validate the AR System object definitions as valid XML.

Exported objects in XML format comprise an XML document, which might also be referred to as an instance of a particular XML schema definition for that object. If the XML schema definitions are loaded into an XML editor, someone who is knowledgeable about AR System objects and XML can edit the XML document.

The XML schema definitions are designed to be similar to the definitions in the *.def files. You can only include one application inside an *.xml file. See the data structure information in the C API Reference guide for more information about the XML Schema definitions of AR System objects.
Exporting and importing definitions

Use the following BMC Remedy Administrator tools, accessed through the Tools menu, to import and export definitions:

- **The Export Definitions tool**—Exports structure definitions to a file. If you are licensed for the Distributed Server Option, you can also export distributed mapping and pool definitions. In addition, you can create mail template files with the Export Mail Templates tool.
  
  You can also export definitions into source control.

- **The Import Definitions tool**—Imports structure definitions to a server. If you are licensed for the Distributed Server Option, you can also import distributed mapping and pool definitions.
  
  You can also import definitions from source control.

Exporting object definitions

Exporting definitions to a file is the first step in moving object definitions from one server to another. You can export one type or multiple types of objects at a time.

**Note:** BMC Remedy Administrator also includes a quick shortcut method of exporting object definitions when you are in the Server Information window. Right-click a single object or a group of objects, then choose Export to File from the context menu that appears. You can include related objects as needed. This method is especially helpful for sorting objects by date for easy selection and modification.

**To export object definitions to definition files**

1. Select a server to administer.
2. Choose Tools > Export Definitions > To Definition File.

The Export Definitions dialog box appears.
The dialog box lists the available structure types. A plus (+) next to an object type means that at least one item of that type exists.

All object definitions from the specified server are displayed in the Objects on <server> list. To see a more detailed list of available objects, click each object icon.

3 Move the objects from the Objects on <server> list to the Objects to Export list in any of the following ways:

- Select available objects, and then click Add to add objects to the Objects to Export list.
- Select the object type, and then click Add to add all objects under the selected type to the Objects to Export list.
- Right-click on an object, and choose Add from the context menu.
- Click Add All to add all the objects to the Objects to Export list.
- Drag and drop objects between the Objects on <server> and Objects to Export lists.
- Double-click an object or an object type to move it to the other list.

Exporting and importing definitions ▶ 491
As needed, use the Remove or Remove All buttons (or right-click option) to move the selected objects from the Objects to Export list to the Objects on <server> list.

Use the Add Directly Related button (or right-click option) to limit the scope of server objects when exporting shared workflow to a definition file. This option defines new rules for each workflow object, establishing parameters that restrict the objects that will be related to include only the associations defined by the new rules (see the table that follows) for each type of workflow.

Each of the object definitions defined in this table are exported as follows when you use the Add Directly Related button.

<table>
<thead>
<tr>
<th>For:</th>
<th>Export operation includes:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forms</td>
<td>All related menus, active links, filters, escalations, active link guides, filter guides, web services, and distributed mapping definitions. In addition, menus from the Change Field action of the active links will be included. Any other forms referenced in workflow actions, guides, and menus will not be associated as related objects.</td>
</tr>
<tr>
<td>Join forms</td>
<td>All forms that were used to create these join forms as well as their related items (as defined in the operations included in the Forms above).</td>
</tr>
<tr>
<td>Active links</td>
<td>All menus that are referenced in the Change Field actions, and all guides that are referenced in the Call Guide action. The guides should see the same form to which the active link refers. The active links that are referenced in the guide also fall within the same scope; therefore, the associated objects of those active links will be included. This cycle continues until it reaches a form.</td>
</tr>
<tr>
<td>Filters</td>
<td>All filter guides referenced in a Call Guide action and all DSO mapping definitions referenced in a DSO action. Filters that are referenced in the guide also fall within the same scope; therefore, the associated objects of those filters will be included. The guides should see the same form to which the filter refers. Escalations do not have any of the above actions, so there will be no associations for escalations.</td>
</tr>
<tr>
<td>Active link guides</td>
<td>All active links referenced in the guide as well as all associated objects for those active links.</td>
</tr>
</tbody>
</table>
Use the Add Contents button (or right-click option) to export only the contents of container objects (for example, active link or filter guides, packing lists or applications) instead of all the workflow associated with the forms in the container.

The Add Contents button becomes enabled only after you select a specific container object.

This button works with all containers except web services.

Use the Add All Related button (or right-click option) to move an object and any objects that are related to it. (If you are using the drag-and-drop method, press the SHIFT key while dragging to move related objects.)

If you do not click the Add All Related button, only the selected objects move to the Objects to Export list.

Each of the object definitions defined in this table are exported as follows when you use the Add All Related button.

<table>
<thead>
<tr>
<th>For:</th>
<th>Export operation includes:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter guides</td>
<td>All filters referenced in the guide as well as all associated objects for those filters.</td>
</tr>
<tr>
<td>Applications</td>
<td>All associated forms and the list of related objects associated with those forms.</td>
</tr>
<tr>
<td>Packing lists</td>
<td>All contents of the packing list and all related objects for those contents.</td>
</tr>
<tr>
<td>Menus</td>
<td>No related items exported.</td>
</tr>
<tr>
<td>Web Services</td>
<td>Exports as an independent object.</td>
</tr>
<tr>
<td>Flashboards</td>
<td>Exports as an independent object.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>For:</th>
<th>Export operation includes:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forms</td>
<td>All related menus, active links, filters, escalations, active link guides, filter guides, web services, and distributed mapping definitions. Forms associated with distributed mapping definitions are exported with all related forms and workflow on the current server.</td>
</tr>
<tr>
<td>Join forms</td>
<td>All related forms and their form-related items</td>
</tr>
<tr>
<td>Filters, active links, and escalations</td>
<td>All related forms and their form-related items</td>
</tr>
</tbody>
</table>
Appendix G—Importing and exporting object definitions and applications

BMC Remedy Action Request System 7.0

1. Choose File ➤ Export ➤ Export Definitions...

2. Select the objects you want to export:
   - Guides
   - Applications
   - Packing lists
   - Menus
   - Web Services
   - Flashboards

3. Enter the path and file name for the export file.

4. Select or clear the Export as Server-Independent check box to indicate whether to remove references to the current server in your export definitions.

   - Selected (the default) — Removes all references, thus permitting definitions in the export file to be imported to other servers.
   - Cleared — Embeds the name of the source server into the export file so that definitions can only be imported to this server.

   This option exists for backward compatibility for older servers. For 4.5 and later servers, this option is obsolete as all definitions are stored independently of server name.

5. If the check box is:
   - Selected — Include the following option in the *ar.conf* (*ar.cfg*) file to make sure that the IP address does not appear in the *def* file:

     ```
     IP-Name: <IP_address>
     ```

     For more information about the *ar.conf* (*ar.cfg*) file, see the Configuring guide.

6. If you are exporting menus and select this check box, include the following option:

   ```
   IP-Name: <IP_address>
   ```

7. Choose File ➤ Export ➤ Export Definitions...

8. Select or clear the Export as Server-Independent check box to indicate whether to remove references to the current server in your export definitions.

   This option exists for backward compatibility for older servers. For 4.5 and later servers, this option is obsolete as all definitions are stored independently of server name.

   If the check box is:
   - Selected (the default) — Removes all references, thus permitting definitions in the export file to be imported to other servers.
   - Cleared — Embeds the name of the source server into the export file so that definitions can only be imported to this server.

   The Server-Independent option keeps any references to other servers intact during the export. References to the server from which you are exporting (the source server) will change to reflect the name of the server to which you are importing the object definition files.
Form and Application Objects

**Note:** Exporting .def files from BMC Remedy Administrator will clean up only known names on the AR System server. For example, if you log in to ServerA, the export will clean up all names found with ServerA. Your computer might not know ServerA and ServerA.domain.com are the same, so it will not clean up the fully qualified name or an alias name.

Also, table fields and active links can point to other servers to gather data. Exporting with server independent will not clean up those names.

9 Select or clear the Lock Exported Objects check box to specify objects you want to protect from unwanted modifications.

For more information, see “Locking objects” on page 511.

10 Click Export.

The Export File dialog box appears.

11 Specify where you want the definitions stored, and then save your changes.

All selected definitions are stored in a single file (with a .def or .xml extension). If you specify an existing file name and location for the definition file, a dialog box appears so that you can select the appropriate option:

- **Overwrite**—Overwrites the object definition of an existing file. This option is useful when you are re-exporting definitions that have changed.

- **Append**—Appends the object definition to an existing file. This option is useful when you are compiling definitions from several different servers in a single location.

When the export is finished, an Export Complete message appears.

12 Click OK to acknowledge the message.

13 Click Cancel to close the Export Definitions dialog box.
Importing object definitions

The following section describes how to import object definitions to a server. You can overwrite existing forms without deleting existing data or breaking links to workflow and you can import data along with a specified form.

You also can import definitions into the AR System server from a source control system. For example, you could update the current version of your AR System server by using the import option to synchronize the AR System server with the source control database. For information about source control and AR System, see the Integrating with Plug-ins and Third-Party Products guide.

To import object definitions from definition files, you must already have created an export file that contains the object definitions that you want to import by using the procedure described in “To export object definitions to definition files” on page 490.

Import operations involving many definitions can take a long time to complete and require a large amount of database space. Your database must have adequate resources before you begin the operation. Perform large imports during hours when users do not require access to the system.

To import object definitions from definition files

1 Select a server to administer.
2 Choose Tools > Import Definitions > From Definition File.
3 Specify the name of the file (with a .def or .xml extension) that contains the definitions that you want to import, and then click Open.

The Import Definitions dialog box appears.
The dialog box lists the available structure types. A plus (+) next to an object type means that at least one item of that type exists.

All object definitions from the specified server are displayed in the Available Objects list. To see a more detailed list of available objects, click each object icon.

4 Move the objects from the Available Objects list to the Objects to Import list in any of the following ways:
   - Select available objects, and then click Add to add objects to the Objects to Import list.
   - Select the object type, and then click Add to add all objects under the selected type to the Objects to Import list.
   - Right-click on an object, and choose Add from the context menu.
   - Click Add All to add all the objects to the Objects to Import list.
Drag and drop objects between the Available Objects and Objects to Import lists.

Double-click an object or an object type to move it to the other list.

You cannot have two objects with the same name on a single server. If an object in the import file has the same name as an object on the destination server, take one of the following actions before completing the import:

- Remove the object with the same name from the destination server.
- Rename the object with the same name on the destination server.

To rename an object, select it from the Objects to Import list, click again to highlight the name, and type a new name.

**WARNING:** If you rename a form in the Import Definitions dialog box and then import the newly named form to the destination server, the form will not retain the connection it had with its associated workflow on the source server. To maintain the connection between a renamed form and its associated workflow, you must rename the form in BMC Remedy Administrator before you create the object definition file. For more information about renaming forms, see the Getting Started guide. If you rename a menu, active link, filter, or escalation in the Import Definitions dialog box, you must reattach it to the appropriate fields on the destination server.

As needed, use the Remove button to move the selected objects from the Object to Import list to the Available Objects list using any of the methods described in the preceding step.

Click the Check in to Source Control check box to indicate whether to add the object definitions to source control.

If the check box is:

- Cleared (the default) — Imports the object definitions to the destination server.
- Selected — Imports the object definitions to the destination server and also checks the file in to Source Control.
- Disabled — Source code integration is not enabled (in Enforced mode).

For more information about source control, see the Integrating with Plug-ins and Third-Party Products guide.
To overwrite an existing server object with the version from the .def or .xml file, choose from the following options:

- **Click the Replace Objects on the Destination server check box.**
  This option lets you overwrite existing form definitions without deleting data or breaking workflow.

- **Click the Delete Excess Fields from Destination Server check box to delete fields on the server that are not present in the file.**

**WARNING:** If you delete excess fields on the server, the associated data will be deleted and any workflow could be broken.

- **Select a Handle Conflicting Types menu options.**
  This option lets you decide on what path to take if field IDs are the same on the server and in the file, but the data types are different.

<table>
<thead>
<tr>
<th>Display Conflict Error</th>
<th>Check for conflicting data types and report error. Import operation is not performed. (Default)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Take No Action</td>
<td>Leave the field as it is on the server. Data is not overwritten during import operation.</td>
</tr>
<tr>
<td>Replace with New Type</td>
<td>Replace the field with the new type during import operation.</td>
</tr>
</tbody>
</table>

**WARNING:** If you replace the field with the new data type, the original data stored in the field is deleted and the workflow could be broken. For pre-6.0 servers, if you overwrite an existing form, all previous data contained in the form will be deleted or overwritten.

You can mix and match any combination of the Delete Excess Fields and Handle Conflicting Types options as needed.

- **Click Import to import the definitions to the destination server.**
  The new definition names are added to the corresponding categories of the server window. When the import is completed, an Import Complete message appears.

- **Click OK to acknowledge the message.**

- **Click Cancel to close the Import Definitions dialog box.**
Exporting and importing deployable applications

Use the following BMC Remedy Administrator tools, accessed through the Tools menu, to import and export the definitions of deployable applications:

- **The Select Application tool**— Chooses and exports a deployable application definition to a file.

  Exporting a deployable application through the Export Application window is similar to the process for exporting definitions, but is much simpler. For example, when you export a deployable application, all the forms, workflow, menus, web services, packing lists, localized messages, reports, application roles, support files, form data (if specified), and so on, are automatically exported together without the need to select individual objects. For information, see “Exporting deployable applications.”

- **The Import Deployable Application tool**— Imports deployable application definitions to a server.

  Importing a deployable application through the Import Deployable Application window is similar to the process for importing definitions, but is much simpler. For example, when you import a deployable application, all the forms, workflow, menus, web services, packing lists, localized messages, reports, application roles, support files, form data (if specified), and so on, are automatically imported together without the need to select individual objects. For information, see “Importing deployable applications” on page 502.

**Exporting deployable applications**

When you export a deployable application, every object “owned” by the application is exported as well. This includes forms, active links, filters, escalations, active link guides, filter guides, menus, web services, packing lists, localized messages, reports, application roles, and form data (if specified).

**Note:** All menus referenced by the application are exported. A menu is referenced by the application if a form in the application has a field that references the menu or if an active link in the application includes a Change Fields reference to the menu.
In addition, you can export data that is stored in AR System forms. During the export operation, the data export entries in the application object are examined and the appropriate form data is exported into the .def file. For more information, see “Importing deployable applications” on page 502.

When you export an application, the server also exports role information. The server gathers all the roles corresponding to the deployable application and exports them. However, the role mappings itself is not exported from the source server.

To export deployable applications

1. Open the Server Information window.
2. Choose Tools > Export Application.
   The Open Application dialog box appears.
3. Select which deployable application you want to export.

   **Figure G-3: Select application dialog box**

   ![Select Application dialog box](image)

   The Export File dialog box appears.
4. Specify where you want the definitions stored, and then save your changes.
   All selected definitions are stored in a single file (with a .def or .xml extension). If you specify an existing file name and location for the definition file, a dialog box appears so that you can select the appropriate option:
   - **Overwrite**—Overwrites the object definition of an existing file. This option is useful when you are re-exporting definitions that have changed.
Append—Appends the object definition to an existing file. This option is useful when you are compiling several different application definitions in a single location.

Cancel—Quits the export operation.
When the export is finished, an Export Complete message appears.

5 Click OK.

Importing deployable applications

When you import a deployable application, every object owned by the application and any relevant form data are imported.

When the application data is imported, each entry is inserted into the target form with a merge entry operation. Therefore, if an entry already exists, the new entry will be merged with the existing entry.

The application roles are imported into the AR System Role Mapping form.

To import deployable applications

1 Open the Server Information window.
2 Choose Tools > Import Application.
   A file open dialog box appears.
3 Choose a .def or .xml file you want to import and click Open.
   The Import Deployable Application dialog box appears.

Figure G-4: Import Deployed Application dialog box

4 From the Application To Import menu, select an application.
To overwrite an existing server object with the version from the `.def` or `.xml` file, choose from the following options:

a. Click the Replace Objects on the Destination server check box. This option lets you overwrite existing application definitions without deleting data.

b. Click the Delete Excess Fields from Destination Server check box if you want to remove extra fields on the server that are not present in the file.

**WARNING:** If you delete excess fields on the server, associated data will be deleted and any workflow could be broken.

c. Choose from the Handle Conflicting Types menu options. This option lets you decide on what path to take if field IDs are the same on the server and in the file, but the data types are different.

- **Display Conflict Error**—Check for conflicting data types and report error. Import operation is not performed.
- **Take No Action**—Leave the field as it is on the server. Data is not overwritten during import operation.
- **Replace with New Type**—Replace the field with the new type during import operation.

**WARNING:** If you replace the field with the new data type, the original data stored in the field is deleted and the workflow could be broken.

You can mix and match any combination of the Delete Excess Fields and Handle Conflicting Types options as needed.

6. Click Import to import the application definitions to the destination server. The new definition names are added to the corresponding categories of the server window. When the import is finished, an Import Complete message appears.

7. Click OK.

8. Import another application as needed, or click Cancel to close the Import Definitions dialog box.
Exporting and importing data with deployable applications

Deployable applications are designed to be portable, easily exported from one server to another. All the objects owned by the application are included in the application export. In addition, you can also define data to include in the export.

With deployable applications, you can include configuration data that you can export and import in your definition file. This data might include backend data users might need to enable or disable workflow. You use the Data tab in the Application window to create the settings that will be used when you create your definition file.

- When exporting form data, you select from a list of available forms, then specify what configuration data to include by defining an export qualification. These queries become part of the application in the definition file and will be stored as external references. For more information, see “Including form data when exporting definitions” on page 505.

- When importing form data, you specify how to map the field data against the target server and how to resolve possible conflicts. To include form data when you import object definition files, see “Importing form data” on page 507.

You can mix and match how you want to export and import the configuration data with your deployable application. For example, if you select a form and do not define an export qualification nor any import options, all data from the form will be exported and all data from the form will be imported into the target server. Equally, you could define an export qualification but not choose any import options.

**Note:** Role information is automatically included when you export a deployable application, but none of the mappings from roles to groups are exported. If you add a custom state to an application, the custom state is a server-specific configuration setting, not a configuration setting of the application. As a result, any information related to application states on application export is not automatically exported.
Including form data when exporting definitions

The following section describes how to include form data from a deployable application for use when you export definitions. You could bundle form data from one application (such as entries from the Group form) to use with a different application. But you would not need to include the Group form itself in the application.

This section also describes how to handle exported data during an import operation.

To include form data when exporting definitions

1. Open a deployable application.
2. Choose Application > Properties.
   The Modify Application window appears.
3. Click the Data tab.

Figure G-5: Modify Application window—Data tab
4 Use the Add or Add All buttons to move forms to the Selected forms list. These forms contain the data you want to export.

5 In the Export Qualification field, specify a query to narrow the amount of data included when you export your application.

   If you do not specify a query, all the form’s underlying data will be included. Using an unqualified query could create a huge .def file during the export operation.

6 In Import Options, select which fields you want to map against the target server when you import your data.

7 Choose how you want to resolve duplicate request IDs when there are data conflicts during the import operation:

   - **Generate New ID for All Records**: New request IDs are assigned to all requests in the data file, whether or not any IDs are duplicates.
   - **Reject Duplicate Records**: Entries are imported using their existing IDs. If an ID is already in use, an error is generated. (Default)
   - **Generate New ID for Duplicate Records**: Entries are imported using their existing IDs. If a record with the same ID already exists in the database, a new ID is generated for the imported record with the duplicate ID.
   - **Replace Old Record with New Record**: Entries are imported using their existing IDs. If a duplicate ID exists, the entire database record is overwritten with the record being imported. You must map the required core fields with this option. If required core fields are not mapped, the server will reject the records.
   - **Update Old Record with New Record’s Data**: Entries are imported using their existing IDs. If a duplicate ID exists, only the fields being imported are replaced, merging the record. This setting also automatically makes all required fields that are not core fields optional.

8 Save your changes.

9 Use the procedure described in “Exporting object definitions” on page 490 to create a definition file.

   The form data will be exported along with the object definition files.
Importing form data

The following section describes how to import data for specified forms into a server from a definition file. Along with the list of server objects, you can select a form data object in the Import Definitions window. You can also import configuration data along with a specified form. This feature allows you to import data definitions from your exported application definition file, so that you can restore a form and its data to its original exported condition.

To import form data from definition files, you must already have created an export file that contains the object definitions and the form data that you want to import by using the procedures described in “To export object definitions to definition files” on page 490 and “To include form data when exporting definitions” on page 505.

Import operations involving much data can take a long time to complete and require a large amount of database space. Your database must have adequate resources before you begin the operation. Perform large imports during hours when users do not require access to the system.

To import data with deployable applications

1. Select a server to administer.
2. Choose Tools > Import Definitions > From Definition File.
3. Specify the name of the file (with a .def or .xml extension) that contains the form data that you want to include during the import operation, and then click Open.

The Import Definitions dialog box appears.
Figure G-6: Import Definitions dialog box (with available form data)

4 Click the Form Data object from the Available Objects list to the Objects to Import list.

A list of forms for which exported data is available in the definition file is displayed. As shown in Figure G-6, you could select from three available forms.

5 Move the form data to the Objects to Import list.

These form data objects are now added to the list of objects to be imported.

6 As needed, use the Remove button to move form data from the Object to Import list to the Available Objects list using any of the methods described in the preceding step.
7 Click the Check in to Source Control check box to indicate whether to add the object definitions to source control.

If the check box is:

- Cleared (the default)—Imports the object definitions to the destination server.
- Selected—Imports the object definitions to the destination server and also checks the file in to Source Control.
- Disabled—Source code integration is not enabled (in Enforced mode).

For more information about source control, see the Integrating with Plug-ins and Third-Party Products guide.

8 To overwrite existing form data with the version from the .def or .xml file, choose from the following options:

a Click the Replace Objects on the Destination server check box.

This option lets you overwrite existing data without deletion.

b Click the Delete Excess Fields from Destination Server check box to delete fields on the server that are not present in the file.

c Click the Delete Excess Views from Destination Server check box to delete views on the server that are not present in the file.

**WARNING:** If you delete excess fields or views on the destination server, the associated data will be deleted and workflow could be broken.

d Select a Handle Conflicting Types menu options.

This option lets you decide on what path to take if field IDs are the same on the server and in the file, but the data types are different.

<table>
<thead>
<tr>
<th>Display Conflict Error</th>
<th>Check for conflicting data types and report error. Import operation is not performed. (Default)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Take No Action</td>
<td>Leave the field as it is on the server. Data is not overwritten during import operation.</td>
</tr>
<tr>
<td>Replace with New Type</td>
<td>Replace the field with the new type during import operation.</td>
</tr>
</tbody>
</table>
WARNING: If you replace the field with the new data type, the original data stored in the field is deleted and the workflow could be broken.

You can mix and match any combination of the Delete Excess Fields and Handle Conflicting Types options.

9 Click Import to import the form data to the destination server.
   When the import is finished, an Import Complete message appears.

10 Click OK.

11 Click Cancel to close the Import Definitions dialog box.

Exporting and importing view definitions

Exporting and importing view definitions use procedures similar to that of object definitions. The differences are the initial menu option you select and some extra fields in the dialog box.

To export view definitions, follow the same steps outlined in “To export object definitions to definition files” on page 490, but choose:

   Tools > Export Definitions > To View Definition File

instead of:

   Tools > Export Definitions > To Definition File

The Export View Definitions dialog box appears.
The View Label, View Type, and View Locale fields list the details about the selected view. If several views have the same label, make sure that you import the correct view locale.

To import view definitions, follow the same steps outlined in “To import object definitions from definition files” on page 496, but choose:

Tools > Import Definitions > From View Definition File

instead of:

Tools > Import Definitions > From Definition File

**Locking objects**

Object locking allows application developers to protect AR System server objects. Locking objects prevents the modification and optionally the viewing of server objects that are neither intended nor designed to be customized. AR System developers who resell their applications will find this feature especially helpful. (For more information, see “Levels of object locking” on page 513.)
To create a locked object, you use the export definition mechanism, not the export deployable application mechanism. For more information, see “Locking objects using export operation” on page 515.

**WARNING:** After you lock an object, it cannot be unlocked, even if you know its lock key. Keep unlocked, backup copies of your objects.

In general, it is best to lock objects when you are ready to release your application. This practice prevents customers—but not fellow developers—from modifying or viewing the workflow, as needed. However, distributing locked objects will prevent other developers from modifying or viewing the locked objects.

**Modifying locked objects**

If you want to modify a locked object, you must have the following items ready so that you can use the Import in Place mechanism to replace the locked object:

- Unlocked copy of the object so that you can modify it.
- Lock key to recreate the locked object.

First you modify the unlocked object and export it with the same lock key that was previously used for that locked object. This newly locked object can then overwrite a corresponding existing locked object on a server when you use the Import in Place mechanism. For information about importing definitions in place, see “Importing locked objects” on page 517.

**Note:** All objects that are locked with the same key, on a given server, become “one block” of locked objects on that server. You can not remove any object from a lock block without deleting the entire lock block. You can overwrite any locked object with the procedure above, but you can not remove any objects at a future time.
Levels of object locking

You can apply different levels of locking to objects:

- **Read-Only**—Prevents users from modifying server-side workflow objects (filters, filter guides, and escalations), but allows them to view the details of the objects.

- **Hidden**—More restrictive lock that prevents users from viewing details of locked server-side workflow objects. In the Server window, you can see the name, execution order, and so on, of the workflow object. But you cannot open the server object (for example, the Modify Filter window) to view Run If qualifications, If or Else actions, change history, or help text.

  Hiding server-side workflow (filters, filter guides, and escalations) is intended to protect the intellectual property of those objects. Hidden locks (hiding the object details) only applies to filters or escalations.

Table G-1 describes important tasks you can perform with locked objects and how these apply to lock levels.

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Read-Only</th>
<th>Hidden</th>
</tr>
</thead>
<tbody>
<tr>
<td>View details of locked objects?</td>
<td>Yes—Can open the object details window. However, most of the details are not alterable.</td>
<td>No—Can only view limited details (name or execution order) of workflow objects in the object lists in the Server window. Viewing the full object details is not allowed.</td>
</tr>
<tr>
<td>Modify locked forms?</td>
<td>No—Can edit display, permissions, and VUI information. On a locked form, you cannot create new fields nor delete existing ones.</td>
<td>No</td>
</tr>
<tr>
<td>Modify workflow objects?</td>
<td>No—Can add forms and edit permissions.</td>
<td>No</td>
</tr>
<tr>
<td>Add forms to locked workflow object?</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Hidden locks apply only to server-side objects, for example, filters and escalations. Hidden locks do not yet apply to client-side objects.
### Table G-1: Object tasks and lock levels

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Read-Only</th>
<th>Hidden</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remove forms from locked workflow object?</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Copy locked objects?</td>
<td>Yes—Can perform Save As operation.</td>
<td>No—Enforced on API level.</td>
</tr>
<tr>
<td>Lock all server objects?</td>
<td>Yes</td>
<td>No—Only workflow objects that run on server: filters, filter guides, and escalations.</td>
</tr>
<tr>
<td>Increase restrictiveness?</td>
<td>Yes (to Hidden)</td>
<td>No</td>
</tr>
<tr>
<td>Decrease restrictiveness?</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Rename locked objects?</td>
<td>No—Enforced on API level.</td>
<td>No—Enforced on API level.</td>
</tr>
<tr>
<td>Delete locked objects?</td>
<td>Yes, but only as a block of objects. Deleting one object that belongs to locked group deletes the entire group. Deleting a locked form that is part of a join deletes the join form.</td>
<td>Yes, but only as a block of objects. Deleting one object that belongs to locked group deletes the entire group. Deleting a locked form that is part of a join deletes the join form.</td>
</tr>
<tr>
<td>Export locked objects?</td>
<td>No—Enforced on API level.</td>
<td>No—Enforced on API level.</td>
</tr>
<tr>
<td>Import locked objects?</td>
<td>Yes, if and only if the lock key of the objects to be imported matches the lock key for those objects on a server, then the server object will be replaced during an import in place operation. Otherwise, a locked object cannot be overwritten. If the objects are locked with a key that does not currently exist on the server, then the objects will be imported.</td>
<td>Yes, if and only if the lock key of the objects to be imported matches the lock key for those objects on a server, then the server object will be replaced during an import in place operation. Otherwise, a locked object cannot be overwritten. If the objects are locked with a key that does not currently exist on the server, then the objects will be imported.</td>
</tr>
<tr>
<td>Lock an object that is already locked?</td>
<td>Yes, if you use the same key to increase restrictiveness to Hidden. If you use a different key, you can lock the object but the existing key will be kept.</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Can applications include locked and unlocked objects?</td>
<td>Yes—Customers can customize the unlocked objects.</td>
<td>Yes—Customers can customize the unlocked objects.</td>
</tr>
</tbody>
</table>
Locking objects

You use the following export procedure to lock objects.

**WARNING:** Lock forms and workflow with different keys. As a general rule, use one key to lock multiple forms; use a different key to lock multiple pieces of workflow. If you use the same key to lock both forms and workflow, you might encounter problems when migrating locked objects.

### To lock exported objects

1. Open the Export Definitions dialog box.
   
   For information, see “Exporting object definitions” on page 490.

2. Select only the objects you want to lock with a specific key.

3. Click the Lock Exported Objects check box.
   
   The Object Lock dialog box appears.

---

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Read-Only</th>
<th>Hidden</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can locked objects that are imported be viewed in BMC Remedy Administrator?</td>
<td>Yes—You can edit display and view information. You cannot add or remove fields.</td>
<td>No—Details of locked objects are not visible. However, their names do appear in the lists of filters and escalations in BMC Remedy Administrator.</td>
</tr>
<tr>
<td>Can a pre-6.x Administrator Tool view data from locked server object?</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Logging?</td>
<td>Does not affect client-side workflow logging.</td>
<td>Only name and server are seen in filter and escalation logs.</td>
</tr>
</tbody>
</table>

---

Form and Application Objects

Table G-1: Object tasks and lock levels

---

Locking objects using export operation
4 Enter and verify the lock key.

You must enter a valid lock key consisting of alphanumeric and punctuation characters, for example, ?123#4% or eugene vinsky. The lock key can be up to 27 characters in length. You cannot use double-byte characters. Objects with the same lock key are encrypted as a group in the definition file. You can also apply different lock keys to different groups of objects, and both groups can reside within the same definition file.

5 Select a Hidden or Read-Only lock type.

For a detailed list of the capabilities of Hidden and Read-Only locks, see Table G-1 on page 513.

6 Click OK.

7 (Optional) During the export operation, append the locked objects into the same definition file.

During the export operation, you can append the locked objects to any existing definition file, or you can create a new definition file. The locked objects are encrypted in the definition file, so that changes to the locked objects are prevented in this format.
Importing locked objects

Use the following procedure to import locked objects. Remember that you use the import definition mechanism to import locked objects, not the import deployable application mechanism. Instead of importing an entire deployed application, you are actually only importing the definitions of the locked server objects.

For detailed information about importing definitions, see “Importing object definitions” on page 496.

To import locked objects

1. Select a server to administer.
2. Choose Tools > Import Definitions > From Definition File.
3. Specify the name of the file (with a .def extension) that contains the definitions that you want to import, and then click Open.

The Import Definitions dialog box appears.
4 Move the objects from the Available Objects list to the Objects to Import list.

5 To overwrite an existing server object with the version from the .def file, perform the following tasks:
   a Select whether to replace objects on the destination server.
   b Select whether to delete excess fields from the destination server. If the import operation deletes fields, you might suffer data loss or broken workflow.

   **WARNING:** If you choose to delete excess fields on the server, then associated data will be deleted and any workflow could be broken.

   c Select how to handle conflicting field types.
**WARNING:** If you replace the field with the new data type, the original data stored in the field is deleted and the workflow could be broken.

You can mix and match any combination of the Delete Excess Fields and Handle Conflicting Types options as needed.

6 Click Import to import the definitions to the destination server. When the import is finished, an Import Complete message appears.

Table G-1 contains a complete listing of Read-Only and Hidden functionality, to help you when you are importing locked objects. For example, you cannot import and overwrite locked objects on the server if the lock key does not match. You can overwrite locked objects with a more restrictive lock, for example, increasing the lock level from Read-Only to Hidden; however, you cannot decrease restrictiveness, from Hidden to Read-Only. When objects are imported into the server, all objects with the same lock are considered one locked block on the server, even if you created them at different times.

Finally, you should remember the following points:

- After objects are imported into the server, all objects with the same lock key are considered one locked block on the server, even if you create or import them at different times.
- During import, if one object from the lock block is selected, all the objects belonging to that block will be imported as well.
- Deletion of any locked object on a server will result in all of the objects that use the same lock key to be deleted as well.

7 Click OK to acknowledge the message.

8 Click Cancel to close the Import Definitions dialog box.

9 View the imported objects in the Server Information window, as shown in Figure G-10.
Figure G-10: Object details in Server Information window

This example shows a filter with a hidden lock that was imported into the server cordova. Objects with hidden locks are not completely invisible. Administrators can still see certain details, for example, the firing order of the filter, the primary form it is attached to, and so on. The Server Information window also displays the object's lock state: None, Read-Only, or Hidden.

If you try to access objects with locks, you see the following behavior in the Server Information window:

- If you try to open objects with a Read-Only lock, the object opens but you cannot modify their execution conditions, their Run If qualification, their If or Else actions, and so on.
- If you try to open objects with a Hidden lock, you receive an AR System warning, No information is returned for the hidden locked object. (ARWARN 8838)
Tip: On a Read-Only locked form, you cannot create new fields nor delete existing ones. This creates problems if later you want to add form action fields to a locked form. When you create the form, add all the form action fields to one or more views (these views can even be regular native views) and mark them as Hidden if showing them in that particular view is inappropriate. Then export the form as locked. The end result for the user that imports this form is that all the necessary form action fields are already present in the form; the user is not required to add the form action fields except for choosing which of these fields they want (or do not want) on a specific view by choosing Form > Current View > Fields in View.
You can localize components of AR System, allowing users to move seamlessly between locale-specific views by simply changing their locale option and logging in.

This section summarizes the areas of AR System related to the task of creating localized AR System applications. The following topics are provided:

- Distinguishing between data and display languages (page 524)
- Localizing AR System forms and applications (page 526)
- Selecting languages during AR System installation (page 531)
- Creating a localized view of a form (page 533)
- Localizing the user interface of a form view (page 535)
- Localizing message components of a form view (page 539)
- Localizing menus (page 547)
- Localizing currency type descriptions (page 549)
- Localizing the mid tier (page 550)
- Settings and procedures for the localized environment (page 551)
- Defining the ARDATE, ARDATEONLY, and ARTIMEONLY environment variables (page 557)
Distinguishing between data and display languages

The distinction between the data language and the display language in the AR System is crucial for localization. If all of your users will work in a single language, then you do not need to worry about the difference between the data language and the display language; they are the same.

If you have users in different locales (for example, users in England, France, and Germany), then you must pick a single data language in which users will enter, search, and modify the data. Typically, this language is the one that is the most common across all of your user locales. There can be only one data language for a single AR System installation, and it is chosen at the time of installation. This requirement guarantees that users in France can find help desk tickets created in Germany or England because there are common data to work against.

On the other hand, the AR System provides many ways to localize the forms that users use to operate on the data, and those form views can use multiple display languages. Each user can choose the display language (assuming they are installed) that is closest to their native language. Accordingly, the French users will see form views, error messages, and online help in their native language. Applications built upon the AR System can support as many display languages as you have users, and each user can choose their own preferred display language. With BMC Remedy applications like the ITSM suite, you can add display languages after installation. For example, if an office is opened in Holland, you could add a Dutch language pack to the application.

**Note:** For general messages, menus, and buttons (such as OK and Cancel), the browser uses the language that was installed on it. For example, if you localize your forms in German, but the user is using a browser with English installed, the user might still see some messages, menus, and buttons in English.
Data language

Think of the data language as the one language that is common to your entire community. The data language is the language of the data that is stored in the database. There can only be one set of data per application installed even if users accessing this data are located in different countries. The data language becomes the language in which the data is displayed to users. For example, if you want your users to see data in Japanese, you must choose the Japanese data language.

The data language specified is provided during installation and you cannot change it. You can choose only one data language.

Display language

The display language is the language used in the user interface (views). You should think of the display language as the language you choose during installation of the clients.

During installation of AR System clients, you can choose more than one display or “view” language for your users to use, but the display language must be compatible with the underlying data language. For example, you can install Japanese and English in any combination (data or display), but you cannot install a display language of French with a Japanese data language.

To localize a form, you change the field labels to a specific language. For a list of the application components you can localize, see “Localizing AR System forms and applications” on page 526.

To preview the same form in different languages, you create different views of the form. You can set a locale view property for each of these views. The system can then determine the locale a particular BMC Remedy User client is opening the form and thus open the right view.
Localizing AR System forms and applications

Any AR System component that a user interacts with can be localized, including:

- User interface
- Error messages
- Workflow messages
- Banners
- Container labels and descriptions
- Help text
- Layout
- Icons
- Reports
- Number and date formats

Getting started

Localization begins with decisions made during installation. AR System has the option of installing multiple language forms and DLLs on to a single client machine, enabling users to move seamlessly between languages by changing the preferred locale setting.

You can enable a localized environment with a single option in BMC Remedy Administrator as described in “Localize Server option” on page 551. Working in an enabled localized environment adds some extra processing time, but the benefit of managing a multi-language version of AR System on a single client platform exceeds the minor performance overhead.

Structure of the localized environment

Administrators define the structure of the localized environment, but users decide the language in which they want to view and process data. Users set the locale preference that drives the selection process when looking for localized AR System components. Users define their locale by the language they speak, and the country where that language is spoken. AR System components are localized by associating them with a specific locale.
As the architect of a localized environment, you can create locale-specific views of forms to accommodate the preferred locale of your users. Users will generally log in specifying their `<language_country>` preference, but, when defining the locale for a form view (or any AR System component), administrators can define only the `<language>` portion of the variable. Users can log in with their `<language_country>` preference, for example, `fr_CA` (French Canadian), but only a single form view is needed with a `fr` locale specified to accommodate all French-speaking countries.

A localized view of a form is accessed through a browser by specifying the preferred locale in a URL (see the information about opening forms in a browser in the installing and administering BMC Remedy Mid Tier guide).

AR System is designed with a fallback lookup mechanism when searching for a view to display. This enables AR System to always find a view to open, even when the user does not specify a preferred locale. For more information about the fallback lookup mechanism, see “Creating a localized view of a form” on page 533. For more information about how a view is selected, see “Preference settings for view selection” on page 301.

**Localizing form views**

After a form view is defined for a locale, the following components of the form view are localized:

- **User interface components**— Components that users see, such as:
  - Field labels
  - Selection field values
  - Banner names

  These values are localized by manually entering the localized values individually through the View Properties dialog box, or by using an automated procedure. (For information, see “Localizing view components through Export/Import” on page 535.)

- **Message components**— Underlying AR System messages and text, such as:
  - System messages
  - Active link messages
  - Filter messages
Help text linked to active link or filter workflow triggers
- Character menus
- Form and field help
- Descriptions and help text embedded within applications and guides
- Labels for applications and guides
- External files linked to applications or reporting

Automatically generated forms—These views are automatically saved with the locale of en_US. If you need a view of the form in another locale, open the view of the form on a machine set to the locale you require, and save it.

When the Localize Server field is checked (see “Localize Server option” on page 551), the retrieval of messages is redirected from the default system loaded DLL to the AR System Message Catalog form. The AR System Message Catalog form enables administrators to localize or customize messages, and is populated using manual procedures or through automation.

An example of a customized environment would be as follows: An administrator can take a single application and tailor the message components to users’ needs, leaving the original application definition untouched. AR System identifies and retrieves the localized or customized messages from the AR System Message Catalog form. Messages for other AR System system objects that were not customized are safely retrieved from the system’s default location because no entry for them exists in the AR System Message Catalog form.

Reporting

You can create localized reports by specifying a locale for a Report form entry, and attaching a localized report definition file.

Locale-specific reports created using the Run Macro report active link action with releases prior to AR System 5.0 can be converted to equivalent active links (see “Backwards compatibility—Run Macro report actions” on page 553). Converting the run macro report action embeds the location of the localized report within the active link.
Tasks for localizing AR System forms and applications

To provide a complete localized experience for users, you must create localized views and applications and then apply the appropriate locale settings. This process consists of the following six tasks:

**Step 1** Installing the available language DLLs

**Step 2** Creating a localized view of a form

**Step 3** Localizing the user interface components of a view

**Step 4** Localizing messages, character menus, and reports

**Step 5** Localizing menus

**Step 6** Applying the appropriate locale settings

The following checklist describes each task and refers you to the appropriate place in the documentation to complete the localization process.

<table>
<thead>
<tr>
<th>Done</th>
<th>Task</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Install the available language DLLs.</td>
<td>See “Selecting languages during AR System installation” on page 531.</td>
</tr>
<tr>
<td></td>
<td>Create a localized view of a form.</td>
<td>See “Creating a localized view of a form” on page 533.</td>
</tr>
<tr>
<td></td>
<td>✔️ Localize the user interface components of a view.</td>
<td>You can edit each field manually or localize automatically through importing and exporting. For manual procedures, see “Localizing messages manually” on page 541. For the automated procedure, see “Localizing view components through Export/Import” on page 535.</td>
</tr>
<tr>
<td></td>
<td>✔️ View components targeted for translation include: field labels, request aliases, and selection field values.</td>
<td></td>
</tr>
</tbody>
</table>
## Localizing BMC Remedy Action Request System 7.0 Applications

### Localize Messages, Including System Error Messages
- **Reference:** This process is accomplished by creating entries in the AR System Message Catalog form. You can populate the AR System Message Catalog automatically using the `ARTEXT` utility and BMC Remedy Import. Or, you can enter the information manually into the AR System Message Catalog form.
- **For the automated procedure using the `ARTEXT` utility and BMC Remedy Import, see** "Localizing messages automatically" on page 540.
- **For manual procedures, see** “Localizing messages manually” on page 541.

### Localize Menus
- **Reference:** See “Localizing menus” on page 547.

### Set the Localize Server Option in BMC Remedy Administrator
- **Reference:** This setting indicates to the server that workflow messages are being retrieved from the AR System Message Catalog form, rather than using default AR System messages.
- **See “Localize Server option” on page 551.**

### Verify Localized Views in BMC Remedy User, and Adjust the View Size, as Appropriate
- **Reference:** Adjusting the size of a view allows for label size variation between languages, making it possible to fit field labels within the borders of a view.
- **To adjust the size of a view in BMC Remedy Administrator, see** “Adjusting view size in BMC Remedy Administrator” on page 553.

<table>
<thead>
<tr>
<th>Done</th>
<th>Task</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Localize messages, including system error messages.</td>
<td>This process is accomplished by creating entries in the AR System Message Catalog form. You can populate the AR System Message Catalog automatically using the <code>ARTEXT</code> utility and BMC Remedy Import. Or, you can enter the information manually into the AR System Message Catalog form. For the automated procedure using the <code>ARTEXT</code> utility and BMC Remedy Import, see &quot;Localizing messages automatically&quot; on page 540. For manual procedures, see “Localizing messages manually” on page 541.</td>
</tr>
<tr>
<td></td>
<td>Localize menus.</td>
<td>See “Localizing menus” on page 547.</td>
</tr>
<tr>
<td></td>
<td>Set the Localize Server option in BMC Remedy Administrator.</td>
<td>This setting indicates to the server that workflow messages are being retrieved from the AR System Message Catalog form, rather than using default AR System messages. See “Localize Server option” on page 551.</td>
</tr>
<tr>
<td></td>
<td>Verify localized views in BMC Remedy User, and adjust the view size, as appropriate.</td>
<td>Adjusting the size of a view allows for label size variation between languages, making it possible to fit field labels within the borders of a view. To adjust the size of a view in BMC Remedy Administrator, see “Adjusting view size in BMC Remedy Administrator” on page 553.</td>
</tr>
<tr>
<td></td>
<td>Set User Preferences for: Display locale, Date/time formats, Time zone</td>
<td>See “BMC Remedy User preferences settings” on page 554.</td>
</tr>
</tbody>
</table>
Advanced tasks

You can also perform the following localization tasks:

- Set up email templates to accommodate the locale of the view created. See “Exporting email templates in different locales” on page 554.
- Export localized form views to another server. See “Exporting a single view” on page 553.
- Localize report files created using macros. See “Backwards compatibility—Run Macro report actions” on page 553.
- Access a localized view of a form in a browser. See “Accessing a localized view of a form in a web client” on page 556.

Selecting languages during AR System installation

AR System 7.0 supports data input and manipulation in the following languages and character sets:

- Western Europe: Danish, Dutch, English, Finnish, French, German, Icelandic, Italian, Norwegian, Portuguese, Spanish, and Swedish (windows-1252)
- Central Europe: Albanian, Croatian, Czech, Hungarian, Polish, Romanian, Serbian, Slovak, and Slovenian (windows-1250)
- Baltic: Estonian, Greenlandic, Lappish, Latvian, and Lithuanian (windows-1257)
- Traditional Chinese (BIG5)
- Simplified Chinese (GB2312)
- Japanese (Shift-Jis and EUC)
- Korean (EUC-KR)

You can select a single language, or select multiple languages during installation of AR System server and clients. To add languages not selected during the initial install, the installation for each client must be run again. English serves as the default language on all platforms, and users cannot unselect it during installation.
During installation of BMC Remedy User and BMC Remedy Alert, the language resource file, which holds AR System interface translations, is stored within the `resdlls` directory found at the following default location:

```
C:\Program Files\AR System\<client_sub_directory>\resdlls
```

In the `resdll` directory, each language installed is assigned a folder, identified by a unique four-digit code:

- German (0007)
- English (0009)
- Spanish (000a)
- French (000c)
- Italian (0010)
- Japanese (0011)
- Chinese (Simplified) (0804)
- Korean (0012)

During installation of the AR System server, the language resource file, which holds system and error messages, is stored within the AR System server directory found at the following default location:

```
C:\Program Files\AR System\<server_directory>
```

In the default server directory, each language installed has its own resource file and follows the format: `arcatalog_<language_code>`.

The `<language_code>` follows the three-letter ISO 639 standard. The following website has a complete listing of the ISO 639 3-letter language codes:

```
http://www.w3.org/WAI/ER/IG/ert/iso639.htm
```

Care must be taken when selecting the language options during installation of each AR System client. AR System does not support the use of clients connected to an AR System server in a different character set, for example, connecting a Japanese client to an English server and the reverse.

To view the system messages in the same language as your client, you need to select your client language in the Components dialog box of the server installation (only valid for Japanese, French, German, English, Italian, or Spanish). If the client is set to a locale that is not provided, but that is supported, you can add system messages to the AR System Message Catalog form for that locale, otherwise you receive system messages in the server’s locale.

Installation of AR System server loads user-selected translations of the Group, User, User Preferences, and Reporting forms for the languages supported by AR System. Administrators create additional language-specific forms and applications using procedures in this section.
The ability to load multiple languages onto a single client machine enables users to move seamlessly between languages by simply changing the preferred locale setting. It also enables multiple users to share a single machine by selecting a unique locale for each user.

Creating a localized view of a form

For each form you want to localize, you will need to create a localized view of that form. You will use an existing view as the base for creating a localized view. For information about creating form views, see Chapter 8, “Creating and managing form views.”

The Locale field on the Manage Views dialog box associates a language and country dialect with a view in the following format: `<language_country>`. Select only the language to include all variations of a language. For example, if a user has `es_AR` (Spanish_Argentina) selected as a preference, but there is no view created for `es_AR`, then the view created with locale `es` is displayed. So creating a view with locale `es` will cover all Spanish users, regardless of whether the user’s preferred locale is `es_AR` or `es_ES`.

Figure H-1: Manage Views dialog box—locale selection

The label name for the localized view must be the same as the label name for the view that is used as the base when creating a localized view. This label must then be selected in the Default View list. The label name selected in the Default View list provides the selection base when AR System searches for the preferred locale among views. When a user opens a form, the view selected for display is determined by the criteria described in “Searching for a view” on page 534.
Tip: You can use a utility, ARLABEL, which is automatically installed in the same directory as the AR System server. This utility assists with localizing labels on a view. ARLABEL allows you to extract all of the text that can appear on a view screen into a CSV file: labels, button text, URL links, text trim, menu items, selection field choices, and so on. After the translated names are supplied for each item, the file can be imported and the new labels loaded into an appropriate language view (created if needed or updated in place if already present). For more information about using ARLABEL, go to http://supportweb.remedy.com.

Searching for a view

AR System uses the following fallback lookup mechanism when selecting a Windows view for the user:

- If a user indicates a locale as a preference in the Options dialog box in BMC Remedy User, then the view that matches the selected locale is used. It is first matched by language and country, and then by just language if the country has not been specified.
- If no locale preference is set, then a view with a blank locale, or the locale of the user’s operating system, is used.
- If no locale preference is set and there is no view with a blank locale, then an available view for the selected form is displayed.

AR System uses the following fallback lookup mechanism when selecting a web view for the user:

- The mid tier first uses the locale defined in the AR System User Preference form.
- If there are no user preferences, the system selects the appropriate view for the user’s browser.
- If the view with the user’s preferred locale is not found, the system check the OS of the user.

If there is no exact match during any of these stages, the fallback mechanism finds the closest possible locale to the one requested. The resulting view is then displayed.

For more information about how a view is selected for display, see “Preference settings for view selection” on page 301.

After a locale-specific view is created, it is ready to localize.
Localizing the user interface of a form view

After a view is associated with a specific locale, you can then select that view localize the components that users see, including:

- **Field labels**
- **Request aliases**— Banner names
- **Selection fields**— Values linked to lists or radio buttons

Localizing view components through Export/Import

You can export view components to a definition file, localize them, and then import them back into the form from where they were exported.

1. Open the file `<server_directory>\CONF\ar.cfg` if the server is on Windows, or `ar.conf` on UNIX or Linux, and add the line `XML-VUI-Export-Default-Is-Localized: T`

   To maximize the server’s performance, set `XML-VUI-Export-Default-Is-Localized:` to `False (F)` if you do not intend to localize the views you export.

2. Restart the server.

3. In BMC Remedy Administrator, select a server to administer.

4. Export view components to a file by choosing `Tools > Export Definitions > To View Definition File` menu.

   The export utility allows you to select multiple views for export, but the output will be stored in a single file. You should export one single language view at a time, giving each output a unique name.

   When saving the export file, select XML for the format.

   For more information about export formats, see Appendix G, “Importing and exporting object definitions and applications.”

5. In the XML editor of your choice, translate the text located between the `<l10nCharacterValue>` and `</l10nCharacterValue>` tags.

   When modifying the XML file, be sure to keep the basic syntax and file layout intact.
6 To import the translated view definitions back into the form from where they were exported:
   a Choose Tools > Import Definitions > From View Definition File in BMC Remedy Administrator.
   b Select the Import in Place check box to replace the original view with the localized version.

For more information about exporting and importing view definition files, see “Exporting and importing definitions” on page 490.

7 Verify the layout and field alignments, and make adjustments as appropriate (see “Adjusting view size in BMC Remedy Administrator” on page 553).

Localizing view components manually

Although selecting every field on a view and entering the localized label can be time-consuming, localizing views manually is useful if a specific component needs editing. These sections will help you locate field labels, request aliases, and selection fields that simply need a quick fix.

Field labels

Field labels are localized by entering the customized text in the Label field on the Display tab in the Field Properties dialog box. For more information about the Display tab properties, see “Display properties” on page 428.
Localizing the user interface of a form view

Request aliases

The Aliases and Labels tab in the View Properties dialog box allows you to define alias names and labels to be used for views. The fields on this tab are localized by entering the appropriate text into the Singular, Plural, Short Singular, Short Plural, Entry Point New, and Entry Point Search fields. Only the Singular edit box is available for web-only views.

For more information about the Aliases and Labels tab, see “To define view properties in the Aliases and Labels tab” on page 268.
Selection fields

You can modify the values for selection fields in the Attributes tab in the Field Properties dialog box. The following procedure describes how to localize lists and radio buttons.

To localize display values for selection fields

1. In BMC Remedy Administrator, open the form you want to localize.
2. Choose Form > Manage Views, and select a view.
3. Click on the Display button to open the selected view.
4. Double-click a list or radio button field on the view.
5. Click the Attributes tab in the Field Properties dialog box.
6. Click the entry you want to modify in the Selection Value list, and enter the localized text in the Alias Value edit box. Click the Modify button for each entry that you modify.
7. Close the Field Properties dialog box.
8 Close the Manage Views dialog box.
9 Choose File > Save Form to save the form.

Figure H-4: Field properties dialog box—attributes tab

You can use AR System Message Catalog form to override system default messages with equivalent localized or customized entries. This override is indicated to the server with an entry in the Server Information dialog box in BMC Remedy Administrator (see “Localize Server option” on page 551).

When logged in to multiple servers, AR System first retrieves localized messages from the preference server. If no preference server is available, it uses the first localized server found in the user's server list.

If a localized version for any message type is not found in the AR System Message Catalog form, AR System loads the system default message.

Messages associated with any AR System object can be loaded into the AR System Message Catalog form automatically with BMC Remedy Import, which uses as input a data file created from the ARTEXT utility (a tool that is automatically installed in the same directory as the AR System server).
You can also enter or modify messages manually by opening the AR System Message Catalog form in BMC Remedy User. The following section details the procedures for populating the AR System Message Catalog form with localized messages. Types of messages that can be localized include:

- System Message
- Active Link Message
- Filter Message
- Active Link Help Text
- Form Help Text
- Field Help Text
- Container Description (includes applications and guides)
- Container Label (includes applications and guides)
- Container Help (includes applications and guides)
- List Menu Definition (character menu)
- External Report
- Application Help
- Application About
- Application Help Index

**Localizing messages automatically**

The ART TEXT utility extracts messages from forms and applications workflow and inserts them into .arx, .csv, or .xml file formats. Once in a data file, the messages can be translated and then imported into the AR System Message Catalog form using BMC Remedy Import.

Messages in the AR System Message Catalog form can also be exported using BMC Remedy User, and the resulting .arx file can be localized.

**WARNING**: For localization, do not export messages using the .csv file format. Doing so removes carriage returns that might be required (for example, in character menus). Export using the .arx file format and then edit the resulting file using a text editor that does not add formatting information to the file.
ARTEXT is automatically installed with your AR System server and resides in the server’s installation directory. For information about using ARTEXT, see the artext.txt file that is included with the utility.

For information about using BMC Remedy Import, see the Configuring guide.

Localizing messages manually

You can enter messages manually, which can help you verify the automated process. Using the procedures that follow, you can also edit errors within the AR System Message Catalog form without having to run the automated procedure again.

Figure H-5: AR System Message Catalog form

Fields on the AR System Message Catalog form are relevant to some message types, but not all.
To enter messages manually into the AR System Message Catalog

1. Open the AR System Message Catalog form.
2. Enter or select the appropriate data for the following fields:
   - **Message Type** — The type of message you are localizing. Select the message type from the list. Only the number value associated with the selected message is entered in the Message Type field.
   - **Message Identifier** — The object whose messages you are localizing. Type in the name of the object.
   - **Locale** — The locale for the message. If you are storing localized messages for multiple languages, and the AR System Message Catalog form is enabled (see “Localize Server option” on page 551), then AR System will compare the user’s preference setting for locale and attempt to match an appropriate message from the catalog, when requested. Type in the locale following the format: `<language_country>`.
     For a list of standard choices for this field, open the Manage Views dialog box in BMC Remedy Administrator (see Figure H-1 on page 533). It is recommended that only the language portion be entered, allowing for all country variations of a language. For example, an entry of `fr` would include all country variations of French.
   - **Status** — The status of localized messages to be retrieved. An Active status enables a message for retrieval. Select Inactive if you do not want a particular message accessed when a server is set as localized.
   - **Field ID or Msg Num** — The field identifier (Field ID) is found in the Database tab in the Field Properties dialog box. The message number (Msg Num) for an active link or filter message, is found in the If Action tab in the Active Link dialog box.
   - **Return Type** — Identifies whether text or file is returned when the message is called.
The following table notes the entry requirements that uniquely identify each message type.

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Entry Description</th>
</tr>
</thead>
</table>
| System message                     | 1 Select **System Message** from the Message Type list.  
2 Enter the error number in the Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| Active link message                 | 1 Select **Active Link Message** from the Message Type list.  
2 Enter the name of the active link in the Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Enter the message number of the active link.  
5 Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| Active link help text               | 1 Select **Active Link Help Text** from the Message Type list.  
2 Enter the name of the active link in the Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| List menu definition (character menu) | 1 Select **List Menu Definition** from the Message Type list.  
2 Enter the character menu name in the Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Select **Message Text** for Return Type and enter the localized text in the **Message Text** field.  
The format for the **Message Text** field the same as a file menu. (For more information, see “Defining file menus” on page 238.) Each line in the file contains a definition of a menu entry in the following format:  
*label*\vert*value*  

Localizing message components of a form view ▲ 543
### Message Type Entry Description

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Entry Description</th>
</tr>
</thead>
</table>
| Filter message     | 1. Select **Filter Message** from the Message Type list.  
2. Enter the name of the filter in **Message Identifier** field.  
3. Enter the locale in the **Locale** field.  
4. Enter the message number in the **Field ID or Msg Num** field.  
5. Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| Form help text     | 1. Select **Form Help Text** from the Message Type list.  
2. Enter the name of the form you are localizing in the **Message Identifier** field.  
3. Enter the locale in the **Locale** field.  
4. Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| Field help text    | 1. Select **Field Help Text** from the Message Type list.  
2. Enter the name of the form you are localizing in the **Message Identifier** field.  
3. Enter the locale in the **Locale** field.  
4. Enter the field identifier in the **Field ID or Msg Num** field.  
5. Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| Container description | 1. Select **Container Description** from the Message Type list.  
2. Enter the name of the container in the **Message Identifier** field.  
3. Enter the locale in the **Locale** field.  
4. Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
| Container label    | 1. Select **Container Label** from the Message Type list.  
2. Enter the name of the container in the **Message Identifier** field.  
3. Enter the locale in the **Locale** field.  
4. Select **Message Text** for Return Type and enter the localized text in the **Message Text** field. |
### Form and Application Objects

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Entry Description</th>
</tr>
</thead>
</table>
| **Container help text** | 1. Select **Container Help** from the Message Type list.  
2. Enter the name of the Container in the Message Identifier field.  
3. Enter the locale in the Locale field.  
4. Select Message Text for Return Type and enter the localized text in the Message Text field.                                               |
| **External report**   | 1. Select **External Report** from the Message Type list.  
2. Enter the name of the active link that the External Report is linked to in the Message Identifier field.  
3. Enter the locale in the Locale field.  
4. Enter the File ID for the report in the Field ID or Msg Num field.  
Support files are saved according to object name and File ID. The File ID differentiates between multiple support files when there is more than one file or report associated with a single active link.  
You can identify how many reports or files are associated with an active link, and then enter the appropriate number; for example 1, 2, and so on.  
Another solution to finding the File ID is to export the active link and review the .def file.  
5. Select Binary Attachment for Return Type and attach the localized report in the Binary Attachment field. |
| **Application help**   | 1. Select **Application Help** from the Message Type list.  
2. Enter the name of the application in the Message Identifier field.  
3. Enter the locale in the Locale field.  
4. Select Binary Attachment for Return Type and attach a help file (.hlp) in the Binary Attachment field.  
Application Help refers to the file attached in the Help Text tab in the Application dialog box. This message type is indicated by selecting the External Help File radio button in the Help Text tab. |
<table>
<thead>
<tr>
<th>Message Type</th>
<th>Entry Description</th>
</tr>
</thead>
</table>
| Application about               | 1 Select ***Application About*** from the Message Type list.  
2 Enter the name of the application in the Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Select Binary Attachment for Return Type and attach an image file (.bmp, .jpeg, .jpg, .dib) in the Binary Attachment field. |
| Application help index          | 1 Select ***Application Help Index*** from the Message Type list.  
2 Enter the name of the application in the Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Select Binary Attachment for Return Type and attach an index file (.cnt) in the Binary Attachment field.  
  The help index (.cnt) file is the contents file used in conjunction with the application help (.hlp) file. Both will be used together only if the locale is matched between them; otherwise, only the help file is used. |
| Flashboards system message      | 1 Select ***Flashboards Message*** from the Message Type list.  
2 Enter the name of the flashboard in Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Enter the message number in the Field ID or Msg Num field.  
5 Select Message Text for Return Type and enter the localized text in the Message Text field. |
| Flashboards label               | 1 Select ***Flashboards Label*** from the Message Type list.  
2 Enter the name of the container in Message Identifier field.  
3 Enter the locale in the Locale field.  
4 Select Message Text for Return Type and enter the localized text in the Message Text field. |
Localizing menus

A menu is a server object that contains items that the user selects. The items in a menu can be defined within a character menu, or are retrieved from a file menu. Although there are different types of menus in AR System, only character, file, and search type menus can be localized.

Localizing character menus

Character menus can be localized all at one time using Export/Import or the ARTEXT utility (see “Localizing view components through Export/Import” on page 535 and “Localizing messages automatically” on page 540). You can also localize character menus manually by modifying the values in the Menu Definition tab in BMC Remedy Administrator. The following figure indicates the Value field in the Menu Definition dialog box where you enter the localized text for a selected menu item.

Figure H-6: Modify Menu dialog box—character menus

For more information, see “Defining character menus” on page 236.
File menus require a separate procedure described in the section that follows.

**Localizing file menus**

*To localize a file menu*

1. Open the file indicated in the path on the Menu Definition tab and localize the entries in a text editor.

2. Save a localized version of the text file in the format: `[filename].[file extension].[language_country]`. AR System searches for the appropriate version of a file menu according to what locale is set as the user preference.

**Localizing search menus**

Creating localized menus enables automatic searches on forms based on the user’s locale.

*To create a localized search menu*

1. Open the AR System User Preference form
   For each Windows user who is going to use the localized search menu:
   a. Click the Locale tab.
   b. Enter the locale for that user in the User Locale field.
   c. Save the form.
   For each web user who is going to use the localized search menu:
   a. Open the Configuration Tool, and select the General Settings link.
   b. Make sure that the preference server names for the relevant users is set in the Preference Server (s) field.

2. Create the form that will contain the search menu.

3. Add a special field with field ID 160 (the Locale field) to the form.

4. Create a Search menu on this form.

   The server automatically appends a query to the search statement, in which the special field value equals the user’s locale.
Localizing currency type descriptions

You can localize the descriptions associated with currency types used in currency fields. When you localize the currency descriptions, you associate them with a particular locale. These descriptions appear when the user clicks the menu next to the currency field to display currency types. The localized version displayed depends on the user preference locale and the installed components selected during the Server installation.

AR System includes currency descriptions in French, German, Italian, Spanish, Japanese, Korean, and Simplified Chinese. Use the following procedure to create currency descriptions for other languages.

**Note:** You will use the AR System Currency Label Catalog form to localize currency descriptions. The AR System Currency Localized Labels form is used internally by the system, and requires no modification.

**To localize currency descriptions**

1. In BMC Remedy User, open the AR System Currency Label Catalog form.
2. In the Currency Code field, select a currency code from the list.
3. In the Locale field, enter the locale for which you are creating a localized currency label.
   - For example, enter fr for all French languages. Enter fr_CA for French Canadian.
4. Enter the localized string in the Localized Currency Label field.
5. Click Save to save the request.
Localizing the mid tier

In AR System 6.3 and later releases, the mid tier and BMC Remedy User pick up localized values, such as error messages, titles, menu labels, table page headers, application lists, workflow messages, and so on, from the server where the form is stored.

To display localized components on the mid tier, make sure that:

- Customization of mid tier messages is done in the mid tier property files, for example, `LocalizedMessages_en.properties`. You can find these `.properties` files for the localized languages that BMC Remedy supports in the `<mid_tier_installation>/WEB-INF/classes` directory.
- The Localize Server check box is selected for the catalog server in the Server Information window of BMC Remedy Administrator.
- An AR System Message Catalog form is present on that server and contains the localized values you want to access. (For more information, see “Localize Server option” on page 551.)

Login and logout pages

In AR System 6.3 and later releases, only one `login.jsp` and `logout.jsp` file exists in the mid tier shared directory for all supported locales. There are now single login and logout pages, which are localized based upon the user's locale when it is loaded using the `LocalizedMessages_<locale>.properties` files. For example, if your browser language is set to `ja`, you will receive a Japanese login page.

Creating localized login and logout pages

To localize the `login.jsp` and `logout.jsp` files for languages that BMC Remedy does not support (for example, Vietnamese), you must create new `.properties` and new `.js` files for your locale.

1. Make copies of the `LocalizedMessages_<locale>.properties` file (found in the `<mid_tier_installation>/WEB-INF/classes` directory) and `LocalizedMessages.js` file (found in the `<mid_tier_installation>/resources/standard/javascript` directory).
2. Translate the strings in these files for your specific locale.
3 Store the new versions of the files in their appropriate directory.
   For example, if you wanted to localize the mid tier into Vietnamese, put the
   localizedMessages_KL.properties file into the
   <mid_tier_installation>/WEB-INF/classes directory.
4 Set the locale in the AR System User Preference form.
5 Make sure your browser and OS are set to the correct locale.
   The next time you access a form in the browser, the login page will display
   the localized version. Based upon the translations of the strings in the new
   .properties and .js files, the mid tier automatically opens the login.jsp
   and logout.jsp pages in the correct language.

Settings and procedures for the localized environment

Previous sections described procedures for localizing form view
components. This section describes procedures used to finalize or fine-tune
a localized environment. Some tasks in this section are required, such as the
“Localize Server option” procedure, and appear in the “Tasks for localizing
AR System forms and applications” on page 529. Other tasks might be
appropriate for specific environments.

BMC Remedy Administrator settings and procedures

Settings and tools described in this section apply only to BMC Remedy
Administrator.

Localize Server option

You must select the Localize Server option in BMC Remedy Administrator to
enable the AR System Message Catalog form. If the AR System Message
Catalog form has been populated with locale-specific or customized
messages, you must select this option for those messages to be retrieved.

To indicate to the server that messages are retrieved from the AR System
Message Catalog form, rather than from the system default server, perform
the following steps.
To set the Localize Server option

1. In BMC Remedy Administrator, choose File > Server Information.
2. Click the Advanced tab.
3. Select the Localize Server check box.
4. Click Apply to enable the change.
5. Click OK to close the Server Information dialog box.

For more information about this setting, see the Configuring guide.

The following figure displays the Advanced tab in the Server Information dialog box with the Localize Server check box indicated. The Catalog Form Name field is read-only.

Figure H-7: Server Information dialog box—Advanced tab

Check to enable message retrieval from the AR System Message Catalog form.
Adjusting view size in BMC Remedy Administrator

Localizing view labels can sometimes alter the display of a view, positioning some fields where a user cannot see them. Modifying the view size allows a view to adapt to label changes, whatever their size or shape.

To adjust the size of a view

1. With a view open, modify the size of the window frame using the resizing handles at the edge of the view. The size in pixels is displayed in the lower-left corner of the view.
2. Choose File > Save.

The view will be displayed in the size that it is saved in when opened in BMC Remedy Administrator or BMC Remedy User.

Backwards compatibility— Run Macro report actions

You can make older language-specific reports created using Run Macro report actions available to users by using any of the following methods:

- Converting the Run Macro report action to an equivalent active link
- Attaching the active link to a workflow trigger, such as a button field, and placing it on a form
- Creating an entry in the AR System Message Catalog form

For details on the AR System Message Catalog form entry required for localized reports embedded in an active link, see “External report” on page 545.

Exporting a single view

The following procedure describes how to export a single view from one server to another. This feature is useful if many views are defined for a single form, each with a different locale. You can export a view to any server, and then use BMC Remedy Administrator to import the view definition into the server where they are currently logged in.

To export a single view

1. In BMC Remedy Administrator, select a server to administer.
2. Choose Tools > Export Definitions > To View Definition File.
For complete instructions about the options available in the Export View Definitions window, see “Exporting object definitions” on page 490.

Exporting email templates in different locales

A mail template is a form used for submitting requests to the server through email. The Export Mail Templates dialog box displays all views that exist for a selected form. The localized view can be selected to create a localized email template. Mail templates for locale-specific views should only be created after the view has been completely localized.

For information about how to set up mail templates, see the Administering BMC Remedy Email Engine guide.

BMC Remedy User preferences settings

Settings described in this section finalize the task of localizing AR System for a locale-specific environment. Users or administrators can set BMC Remedy User preferences.

Users logging in to web clients, or who want to have the same settings and customizations available when logging in to multiple machines, must use centralized preferences, and log in with a preference server. For more information about centralized user preferences, see the Configuring guide. For more information about preferences on the web, see the Installing and Administering BMC Remedy Mid Tier guide.

Choose Tools > Options in BMC Remedy User to display the Options dialog box used for setting user preferences. Select the Locale tab, which is used to set:

- Display locale
- Date/time styles and formats
- Time zone
- Currency

A null selection for any of the fields on the Locale tab defaults to the setting of the user’s operating system, browser, or web server, depending on whether a locale preference is set.

Selecting which character set to use for entering data in AR System depends on the operating system. You should review the manual of your OS platform for instructions.
Setting the display locale

Selecting a locale allows users to make entries following the prescribed format for their native country and dialect. The AR System server uses this setting to identify and return localized information.

The Display Locale selection follows ISO standards in the following format: `<language_country>`. At login, the server will search for form views that match the locale selected in the Locale tab.

Setting the Date/Time style and time zone

AR System stores all Date/Time field values as integers relative to 00:00:00 GMT, January 1, 1970. AR System displays the Date/Time value relative to the time zone of the client viewing the field. See the Getting Started guide for more information.

AR System calendars follow the Gregorian format and display the month and day of the week according to the selected locale.
The Long and Short options for Date/Time Style are based on the following settings:

- In Windows, the date and time display format is based on the Regional Setting Properties Control Panel. If the AR System server is running under a different account name or using the default user configuration and you are unable to change the regional properties, you can set the ARDATE, ARDATEONLY, or ARTIMEONLY environment variable.

- In UNIX, the date and time display format is based on the ARDATE, ARDATEONLY, or ARTIMEONLY environment variable for UNIX. If you do not use ARDATE, the display format is the default format for the language setting, with the time zone determined by the TZ environment variable.

For more information about short and long date/time formats, see the Configuring guide.

Accessing a localized view of a form in a web client

In AR System 6.3 and later releases, accessing a localized version of a form view has been greatly simplified. Users no longer need to specify the locale of the view in a URL. Instead, views are selected by the mid tier on the basis of the fallback lookup mechanism described in “Searching for a view” on page 534.

If you installed the mid tier in the default location, the basic URL to open a web view is:

http://<host>/arsys/forms/<server_name>/<form_name>

For detailed information about accessing a form view in a browser, see the Installing and Administering BMC Remedy Mid Tier guide.
Defining the ARDATE, ARDATEONLY, and ARTIMEONLY environment variables

The ARDATE, ARDATEONLY, and ARTIMEONLY environment variables are used to customize the time and date settings for the AR System server, DSO server, and Approval Server. The format you specify using these variables is used by the AR System during processing of date and time fields.

For example, to set ARDATE, use the following syntax:

```
% setenv ARDATE "<your_format>"
```

When to use ARDATE, ARDATEONLY, and ARTIMEONLY

The ARDATE, ARDATEONLY, and ARTIMEONLY environment variables are useful when you need to override system settings for the AR System server to use.

Customizing Date/Time formats

Field descriptors

You can create any date formats by combining field descriptors. Field descriptors represent different components of dates and times. Available field descriptors might differ, depending on your UNIX machine type. By default, the AR System uses the format "%x %X".

Table H-1 lists the UNIX field descriptors that you can use with ARDATE, ARDATEONLY, and ARTIMEONLY.

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>%</td>
<td>Same as %</td>
</tr>
<tr>
<td>%a</td>
<td>Day of week using locale’s abbreviated weekday names</td>
</tr>
<tr>
<td>%A</td>
<td>Day of week using locale’s full weekday names</td>
</tr>
<tr>
<td>%b or %h</td>
<td>Month using locale’s abbreviated month names</td>
</tr>
<tr>
<td>%B</td>
<td>Month using locale’s full month names</td>
</tr>
<tr>
<td>%d</td>
<td>Day of month (01–31)</td>
</tr>
<tr>
<td>%D</td>
<td>Date as %m/%d/%y</td>
</tr>
</tbody>
</table>
The format you specify can contain any characters (including the field descriptors in Table H-1) in any combination that you choose. So, if you want to put a comma in your date, include the comma in the appropriate place in the format.

Table H-2 shows examples of ARDATE, ARDATEONLY, and ARTIMEONLY settings.

**Table H-1: Field descriptors**

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>%e</td>
<td>Day of month (1–31; single digits are preceded by a blank)</td>
</tr>
<tr>
<td>%H</td>
<td>Hour (00–23)</td>
</tr>
<tr>
<td>%I</td>
<td>Hour (00–12)</td>
</tr>
<tr>
<td>%k</td>
<td>Hour (0–23; single digits preceded by a blank)—Solaris only</td>
</tr>
<tr>
<td>%m</td>
<td>Month number (01–12)</td>
</tr>
<tr>
<td>%M</td>
<td>Minute (00–59)</td>
</tr>
<tr>
<td>%p</td>
<td>Locale’s equivalent of a.m. or p.m., whichever is appropriate</td>
</tr>
<tr>
<td>%r</td>
<td>Time as %I : %M : %S %p</td>
</tr>
<tr>
<td>%R</td>
<td>Time as %H : %M</td>
</tr>
<tr>
<td>%S</td>
<td>Seconds (00–59)</td>
</tr>
<tr>
<td>%T</td>
<td>Time as %H : %M : %S</td>
</tr>
<tr>
<td>%w</td>
<td>Day of week (Sunday is day 0)</td>
</tr>
<tr>
<td>%x</td>
<td>Date, using locale’s date format</td>
</tr>
<tr>
<td>%X</td>
<td>Time, using locale’s time format</td>
</tr>
<tr>
<td>%y</td>
<td>Year within century (00–99)</td>
</tr>
<tr>
<td>%Y</td>
<td>Year, including century (for example, 2004)</td>
</tr>
</tbody>
</table>

**Table H-2: Settings**

<table>
<thead>
<tr>
<th>Output</th>
<th>ARDATE Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/31/03</td>
<td>setenv ARDATE &quot; %m/%d/%y&quot;</td>
</tr>
<tr>
<td>31 January 2004</td>
<td>setenv ARDATE &quot; %d %B %Y %X&quot;</td>
</tr>
<tr>
<td>January 31 2004</td>
<td>setenv ARDATEONLY &quot; %m/%d/%Y&quot;</td>
</tr>
<tr>
<td>16:30:00</td>
<td>setenv ARTIMEONLY &quot; %X&quot;</td>
</tr>
</tbody>
</table>
Defining the ARDATE, ARDATEONLY, and ARTIMEONLY environment variables

**Note:** The wording of the second example might differ, according to the LANG environment variable or your system’s default language.

For Windows servers, use the time and date formats listed in the Regional Options dialog box to set the environmental variables under your System settings.

**Table H-3: Sample windows settings**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARDATE</td>
<td>M/ d/ yyyy h: mm: ss</td>
</tr>
<tr>
<td>ARDATEONLY</td>
<td>M/ d/ yyyy</td>
</tr>
<tr>
<td>ARTIMEONLY</td>
<td>h: mm: ss tt</td>
</tr>
</tbody>
</table>

**Errors in formatting ARDATE, ARDATEONLY, and ARTIMEONLY**

If you set up an incorrectly formatted ARDATE, ARDATEONLY, and ARTIMEONLY environment variable, the AR System parses the correctly formatted parts and returns the invalid parts as a literal string in the date/time.

See Table H-1 on page 557 for a list of valid field descriptors.

It is possible to write an ARDATE, ARDATEONLY, and ARTIMEONLY environment variable that is correctly formatted but does not make sense. For example, avoid repeating a valid descriptor excessively (e.g., %B %B %B %B %B). If a format such as this is chosen, dates will be formatted correctly on output, but the system will not be able to interpret any date that you specify as input.
Symbols

- .def file type (exported objects) 489
- .xml file type (exported objects) 489

A

About box
  in applications 40

access control
  See also access control groups
  active links 65, 90
  additive 56
  applications 25, 57, 90
  fields 60, 90
  forms 57, 90
  Group form 78
  groups 48
  guides 57, 90
  illustrated 77
  implicit groups 67
  objects, AR System 90
  overview 48
  requests 66, 69
  reserved fields 378
  roles 53
  Roles form 83
  row-level security 67, 380
  users 53

access control groups
  See also groups; permissions; access control
  Assignee group 68, 69
  Assignee Group group 68, 69, 380
  access control groups (continued)
    computed groups 52, 80
    dynamic groups 68
    explicit groups 49
    implicit groups 49, 67
    requests at group level 69
    requests at user level 69
    Submitter group 68, 69

access points
  creating 33
  described 32
  using 34

active link actions
  Close Window 106
  Commit Changes 106
  Open Window 106

active links
  access control 65, 90
  menu items 208
  permissions 65, 91
  workflow issues with entry point guides 318

adding
  fields to join forms 221
  form views and 225
  groups 81
  items (menus) 408
  menu items 236
  menus 236
  roles 85

additive access control 56

administrator default for fields 407
BMC Remedy Action Request System 7.0

Administrator group 50
advanced
data fields 62
search menus 243
table fields 191
advanced display properties 400
advanced tasks in localization 531
aggregates. See table fields statistics
Alert Events form 392
alert list fields
auto refresh 450
creating 182
drill-down on web 451
labels 449
overview 153, 157
permissions 62
results color 446
sort order 448
workflow and 161
Alert List form 392
aliases
group 79
request 268
web 265
aligning
fields 284, 292, 296, 429
grid and 285
trim 416
Allow Any User to Submit setting 63
allowable types properties 403
anywhere QBE match 426
appearance of applications 35
appending export files 495, 502
appending. See adding
application list fields
AppSubset field 377
creating 202, 204
entry point labels 270
Home Page form 167
overview 166
using style sheets with 323
workflow 167
workflow and 167
application mode 38, 40
Application Name field 84, 381
Application Owner property 29
application permissions
individual applications 90
multiple applications 91
subadministrator 92
Application Statistics Configuration form 393
Application Statistics form 392
Application window
creating objects in 34
saving contents of 28
using 20
applications
See also Application Owner property;
application permissions; Application
window; deployable applications; local
applications; web applications
About box 40
access control 25, 57, 90
access points 32, 37
appearance 35
application mode 38, 40
change history 38
converting local to deployable 45
creating 23, 24
customizing title bar 40
data properties 37
deleting 45
deployable 21
deployable, exporting and importing 500
described 20
developing 23
entry points 309, 319
entry points, creating 320
forms properties 27, 36
general properties 35
help 42, 43
icons 40
including objects in 26
including packing lists in 348
label property 36
local 20
localizing 526
names 35
opening in BMC Remedy User 38
ownership of objects 29
### Form and Application Objects

<table>
<thead>
<tr>
<th>Applications (continued)</th>
</tr>
</thead>
<tbody>
<tr>
<td>permissions 25</td>
</tr>
<tr>
<td>primary form 41</td>
</tr>
<tr>
<td>primary view 41</td>
</tr>
<tr>
<td>properties, defining 35</td>
</tr>
<tr>
<td>run with form window maximized 40</td>
</tr>
<tr>
<td>saving 28</td>
</tr>
<tr>
<td>shortcuts to 44</td>
</tr>
<tr>
<td>show only forms in 41</td>
</tr>
<tr>
<td>states 22, 30</td>
</tr>
<tr>
<td>statistics properties 37</td>
</tr>
<tr>
<td>support files 37</td>
</tr>
<tr>
<td>web alias 36</td>
</tr>
<tr>
<td>workspaces 346</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AppSubset field (entry points) 377</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR Export file 364-366</td>
</tr>
<tr>
<td>AR System Application State form 393</td>
</tr>
<tr>
<td>AR System Message Catalog, localizing 539</td>
</tr>
<tr>
<td>AR System User Central File form 390</td>
</tr>
<tr>
<td>AR System User Preference form 390</td>
</tr>
<tr>
<td>AR Task shortcut 39, 44</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Archive forms</th>
</tr>
</thead>
<tbody>
<tr>
<td>See also archiving</td>
</tr>
<tr>
<td>created by AR System 465</td>
</tr>
<tr>
<td>deleting 462</td>
</tr>
<tr>
<td>and DSO 466</td>
</tr>
<tr>
<td>properties 131, 464</td>
</tr>
<tr>
<td>saving the main form 462</td>
</tr>
<tr>
<td>special considerations for view forms 462</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Archiving</th>
</tr>
</thead>
<tbody>
<tr>
<td>configuring data archiving for a form 459</td>
</tr>
<tr>
<td>configuring data archiving for a server 466</td>
</tr>
<tr>
<td>logging 466</td>
</tr>
<tr>
<td>overview 458</td>
</tr>
<tr>
<td>server events 466</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ARDATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>dates 557</td>
</tr>
<tr>
<td>environment variable 557</td>
</tr>
<tr>
<td>errors in formatting 559</td>
</tr>
<tr>
<td>field descriptors 557</td>
</tr>
<tr>
<td>time 557</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ARDATEONLY environment variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>defining 557</td>
</tr>
<tr>
<td>errors in formatting 559</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>arInvoke 204</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARLABEL, using 534</td>
</tr>
<tr>
<td>ARTEXT 539, 540</td>
</tr>
<tr>
<td>ARTIMEONLY environment variable</td>
</tr>
<tr>
<td>defining 557</td>
</tr>
<tr>
<td>errors in formatting 559</td>
</tr>
<tr>
<td>Assigned To core field 356, 358</td>
</tr>
<tr>
<td>Assignee group</td>
</tr>
<tr>
<td>access 50</td>
</tr>
<tr>
<td>access control groups 69</td>
</tr>
<tr>
<td>reserved group 50</td>
</tr>
<tr>
<td>Assignee Group field 69</td>
</tr>
<tr>
<td>Assignee Group group</td>
</tr>
<tr>
<td>access 68</td>
</tr>
<tr>
<td>access control 69</td>
</tr>
<tr>
<td>access control to requests 72, 76</td>
</tr>
<tr>
<td>reserved group 51</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attachment fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>creating 196</td>
</tr>
<tr>
<td>database properties 427</td>
</tr>
<tr>
<td>deleting 196, 226</td>
</tr>
<tr>
<td>size considerations 198</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attachment pools</th>
</tr>
</thead>
<tbody>
<tr>
<td>creating 194</td>
</tr>
<tr>
<td>creating attachment fields in 196</td>
</tr>
<tr>
<td>deleting attachment fields in 198</td>
</tr>
<tr>
<td>list 436</td>
</tr>
<tr>
<td>overview 147, 161</td>
</tr>
<tr>
<td>permissions 63</td>
</tr>
<tr>
<td>properties 405</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attachment tables</th>
</tr>
</thead>
<tbody>
<tr>
<td>data 373</td>
</tr>
<tr>
<td>details 373</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attachments</th>
</tr>
</thead>
<tbody>
<tr>
<td>copying requests and 163</td>
</tr>
<tr>
<td>maximum file size 424</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attributes properties 406</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Auditing</th>
</tr>
</thead>
<tbody>
<tr>
<td>configuring 476</td>
</tr>
<tr>
<td>filters and processing 485</td>
</tr>
<tr>
<td>form-style 471</td>
</tr>
<tr>
<td>log-style 474</td>
</tr>
<tr>
<td>overview 470</td>
</tr>
</tbody>
</table>

| Authentication Login Name field, User form 380 |
| Authentication String field, User form 380 |
auto fit column 433
automating localization 535, 540

B
background
  color 267
  mode (trim) 417
backward compatibility
  localizing macros 553
banner, hiding 267
blank form view 257
BMC Remedy Administrator
  localized environment settings 551
  Preference form 390
BMC Remedy User
  entry points appearing in Object List 308
  home pages, creating 321
  how home page appears 331
  localizing preference settings 554
  Object List 309
  suppressing home pages 337
bold field labels 419
bottom (align) 284
box. See trim
bring to front (align) 285
browsers. See web clients
button fields
  creating 207
  disabling 441
  displaying as URLs 435
  image properties 439
  overview 169

C
cascading style sheets
  arsystem.css 324
  BMC Remedy User, using with 324
  browser limitations 325
  mid tier installation directory 323
  web clients, using with 325
Change Field action
  attachment pools and 164
  currency fields and 149
change flag 431
change history
change history (continued)
  applications 38
  fields 417
  forms 132
  menus 253
  packing lists 345
Change permissions
  fields 60
  groups 79
changes to data, auditing 470
character fields
  attributes 406
  default size limit 424
  overview 144
character menus
  adding to fields 408
  creating 233
  described 236
  localizing 547
chunking data (table field) 402
client file location (menus) 239
clients, using home pages with web 311
client-side table fields
  chunks returned, defined 402
  creating 182
  overview 153
Close Window action 106
color
  background on Windows views 267
  properties 417
  results list 276, 277, 446
  table fields 446
  view properties 276
columns
  properties (table field) 436
  width 433
  wrap text in table fields 436
Commit Changes action 106
compatibility, backward with macros 553
components of home pages 309
Computed Group Definition field 80
computed groups
  examples 80
  overview 52
configuring
   auditing 476
   data archiving for a form 459
   data archiving for a server 466
   home page preferences 330
connect refresh mode 233
control panels 104
converting local applications to deployable applications 45
copying
   fields 224
   forms 102
   menus 235
   requests 163
core fields
   Assigned To 356, 358
   Create Date 355
database properties 422
deleting 226
form views with none 257
Last Modified By 356
Modified Date 356
overview 354
Request ID 355, 357
Short Description 356, 358
special characteristics 357
Status 356, 358
Status-History 356
Submitter 355, 357
Create Date core field 355
creating
   applications 23, 24
   attachment fields 196
   attachment pools 194
   computed groups 80
custom application states 32
dialog boxes 105
display-only forms 104
fields 173
fields in view 225
form entry points 312
forms 99, 100
groups 81
home pages (BMC Remedy User) 321
creating (continued)
   home pages (overview) 321
   home pages (web clients) 322
   join forms 120
   localized templates 554
   localized views 533
   locked objects 515
   menus 232
   packing lists 340
   page fields 444
   roles 85
   sample workflow to view subset of entry points 327
   shortcuts 44
   views 256
criteria for join forms 110, 122, 129, 130
currency fields
   attributes 409
   creating 176
   initial currency type 437
   overview 147
   properties 403, 437
currency types, localizing 549
Current Form field 383
Current Server field 384
custom About box 40
custom title bar icon (applications) 40
Customize group 50
customizing
   form layout 50
   results list fields 135
D
data
   archiving 459
deployable applications, exporting with 504
deployable applications, importing with 502
form, exporting and importing with deployable applications 504
importing with deployable applications 509
languages (localization) 524
data chunking (table field) 402
data dictionary menus 233, 251
BMC Remedy Action Request System 7.0

data fields
  creating 174
  overview 144
data languages (localization) 525
data visualization fields
  overview 166
  properties 398
database
  core fields, properties of 422
  field properties 420
  names of fields 423
  security 251
databases
  attachment tables 373
  Request ID and 360
  status history table 373
date and time fields
  attributes 408
  date fields 146
date formats
  ARDATE 557
  combining field descriptors (UNIX) 557
  customizing for localization 557
  localizing 555
  Windows localizing 559
date/time fields 146
DB2 database (diary field limit) 145
decimal number fields
  attributes 412
  overview 151
Default Notification Mechanism field 379
defaults
  field value 407
  fields 436
  label color 418
  permissions 86
  URL color 215
definitions
  exporting 490
  exporting and importing views 510
  file types 489
  importing 496
  overview of object 488
deleting
  applications 45
  archive form 462
  attachment fields 198, 226
  excess fields from destination servers 499, 509
  fields 225
  fields in join forms 222
  forms 103
  groups 82
  join form criteria 130
  menu items 236
  menus 235
  pages (field) 444
  roles 86
  views of forms 263
deployable applications
  See also applications
  access points 32
  advantages of 21
  conflicting types, handling 503
  converting from local applications 45
  data exporting 504
  deleting excess fields from destination server 503
  described 21
  exporting and importing 500
  exporting and importing form data 504
  features 22
  forms in 27
  group permissions warning 26, 45, 349
  importing 502
  including forms with group permissions warning 26, 349
  ownership of objects 29
  replacing objects on destination server 503
  roles 53
  states 22, 30
  depth effect of trim 416
description (predefined searches) 272
designing forms 99
developing applications 23
dialog boxes, display-only forms as 105
diary fields
  attributes 409
  overview 144
Form and Application Objects

<table>
<thead>
<tr>
<th>Entry Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>active link workflow and guides 318</td>
</tr>
<tr>
<td>administrators, using 305</td>
</tr>
<tr>
<td>appearance in applications 319</td>
</tr>
<tr>
<td>application designers, using 305</td>
</tr>
<tr>
<td>application, creating 320</td>
</tr>
<tr>
<td>AppSubset fields 377</td>
</tr>
<tr>
<td>custom view labels 314</td>
</tr>
<tr>
<td>end-users, using 306</td>
</tr>
<tr>
<td>form, creating 312</td>
</tr>
<tr>
<td>forms, defining in 133</td>
</tr>
<tr>
<td>guides, using 307</td>
</tr>
<tr>
<td>limiting number 305</td>
</tr>
<tr>
<td>new 133</td>
</tr>
<tr>
<td>overview 304</td>
</tr>
<tr>
<td>search 133</td>
</tr>
<tr>
<td>subset 167</td>
</tr>
<tr>
<td>subset, creating 327</td>
</tr>
<tr>
<td>using 305</td>
</tr>
<tr>
<td>views, selecting 313</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Entry Points (HTML)</th>
</tr>
</thead>
<tbody>
<tr>
<td>in HTML 204</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARDATE variable 557</td>
</tr>
<tr>
<td>ARDATEONLY variable 557</td>
</tr>
<tr>
<td>ARTIMEONLY variable 557</td>
</tr>
<tr>
<td>localizing 551</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dynamic Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>search menus, defining 243</td>
</tr>
<tr>
<td>table fields, defining 191</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dynamic Group Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>example 76</td>
</tr>
<tr>
<td>overview 49, 52</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dynamic Menus</th>
</tr>
</thead>
<tbody>
<tr>
<td>233, 240</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dynamic Web Views</th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>editable columns, tables 435</td>
</tr>
<tr>
<td>editing text 145, 424</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Email Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>reserved field 378</td>
</tr>
<tr>
<td>email, creating localized templates 554</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enabling</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields 432</td>
</tr>
<tr>
<td>workspaces 347</td>
</tr>
<tr>
<td>entry mode 425</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Equal QBE Match</th>
</tr>
</thead>
<tbody>
<tr>
<td>426</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>formatting date and time environment</td>
</tr>
<tr>
<td>variables (localizing) 559</td>
</tr>
<tr>
<td>RPC and memory allocation 199</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Example Workflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>to view subset of entry points 327</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Expanding</th>
</tr>
</thead>
<tbody>
<tr>
<td>control spacing 291</td>
</tr>
<tr>
<td>distance between fields in form 285</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Exporting</th>
</tr>
</thead>
<tbody>
<tr>
<td>add all related 493</td>
</tr>
<tr>
<td>add contents 343, 493</td>
</tr>
<tr>
<td>append definition file 495, 502</td>
</tr>
<tr>
<td>definitions 490</td>
</tr>
<tr>
<td>deployable applications 500</td>
</tr>
<tr>
<td>email templates 554</td>
</tr>
<tr>
<td>files, appending 495, 502</td>
</tr>
</tbody>
</table>
exporting (continued)
   form data with deployable applications 505
   localized view components 535
   lock objects 495
   locked objects 515
   object definitions 490
   overwrite definition file 495, 501
   server-independent 494
   shortcut 490
   view definitions 510
   views 553

EXTERNAL () operator 160
external help files 43

F
field access
   Assignee access control 68
   Assignee Group access control 68
   dynamic group access control 68
   restricting contents 407
   Submitter group access 68
field descriptors
   ARDATE 557
   date formats 557
field ID 422
field labels
   definition 429
   localizing 535, 536
   styles 419
field patterns 406
field permissions
   alert list 62
   attachment pool 63
   individual fields 90
   multiple fields 91
   overview 60
   page 62
   results list 62
   table 62
field properties
   advanced 398
   advanced display 400
   allowable types 403
   attach fields 405
   attributes 406

field properties (continued)
   color 417
   database 420
   display 428
   font 417
   functional types 437
   help text 438
   image 439
   navigation items 441
   pages 443
   permissions 444
   related workflow 444
   results color 446
   shared fields 447
   sort 448
   table labels 449
   table/tree 453
   view field 398
   views 455
Field Properties dialog box, keeping open 175
field spacing
   overview 279
   specifying even spacing 281
   specifying length 282
field types
   alert list 182
   application list fields. See application list fields
   attachment pools 147, 161, 194
   button 169, 207
   character 144
   client-side table fields 153
   core 354
   currency 147, 176
   data 144, 174
   data visualization 166
   date 146
   date/time 146
   decimal number 151
   determining which to use 174
   diary 144
   global 218
   GUID 220
   horizontal navigation 168
   integer 151
   nonoperational 227
### Field Types (continued)
- Page 170, 209
- Real number 151
- Reserved 375
- Results list 153, 182
- Selection 151
- Server-side table fields 153
- Table 153, 182
- Time 146
- Trim 172, 213
- Vertical navigation 168
- View fields 164, 199

### Fields
See also core fields; field access; field labels; field permissions; field properties; field types; field values, assigning from

#### Adding to View
- Adding to view 225
- Administrator default value 407
- Advanced data 62
- Aligning 284, 292, 296
- Arranging 282
- Change history 417
- Changing existing value 363
- Changing length 362–363
- Copying 224
- Core, database properties of 422
- Creating 173
- Creating attachment fields 196
- Deleting 222, 225
- Deleting attachment fields 198
- Descriptors, combining (UNIX) 557
- Displaying in a view 456
- Excess fields from destination server, deleting 499, 509
- Finding 228
- Form action 386
- Form with none 257
- Global 218
- Grid alignment 287
- Group form 79
- Help text in 438
- Hidden 430, 431
- In form view 297
- Including/excluding in form views 299
- Input length 424

### Fields (continued)
- Join forms and 111, 220, 222, 426
- Localizing 535, 536
- Localizing selection 538
- Location position 429
- Maximum/minimum values 412, 413
- Modifying 223
- Moving 282, 290
- On form (table fields) 453
- Precision 412, 413
- Preserving existing values 363
- Request ID and 359
- Reserved 422
- Resizing 290, 296
- Returned in results list pane 135
- Roles form 84
- Setting multiple properties 293
- Spacing between 291
- Stacked horizontally in joins 126
- Stacked vertically in joins 125
- States 358
- Tabbing order 430

#### Fields Changed Field
- 484

#### Fields in View Dialog Box
- 299

#### Fifteen-minute Interval Refresh Mode
- 233

#### File Menus
- 233, 238

#### File Sizes for Attachment Fields
- 198

#### Files
- .def 489
- .xml 489
- Export, appending 495, 502
- Location (menus) 239
- Login.jsp, localizing 550
- Logout.jsp, localizing 550
- Menu 240

#### Filters, Execution Order
- 118

#### Finding a Field
- 228

#### Flashboards, Reserved Group
- 51

#### Floating Licenses Field in Group Form
- 80

#### Fonts
- Changing style 420
- Properties 417

#### Form Action Fields
- Add before locking forms 521
- Adding to form 329
form action fields (continued)
described 386
home pages 314
form entry points, creating 312
form permissions
access control 57, 90
individual forms 90
multiple forms 91
subadministrator 92
form types
hidden 58, 93
join forms 120
regular 96
vendor 98
view 98
form views
appearance 266
creating 256
defining searches 270
deleting 263
menu access 273
modifying 261
naming 262
properties 263
request aliases 268
results list color 276, 277, 446
format of menu files 239
formats, ARDATE errors 559
forms
See also form permissions; form types; form views; join forms; special forms
See also special forms
blank form 257
copying 102
core fields 354
creating 99
customizing layout 50
data, exporting and importing with deployable applications 504
data, importing with deployable applications 507
data, including when exporting definitions 505
deleting 103
designing 99
forms (continued)
display-only 98
entry points, defining 133
Group form 78
help text in 132
Home Page 308
home page form action field, adding to 329
including in applications 26
including/excluding fields in views 299
join 96
localizing 526, 533
modifying 101
multiple form views 256
primary 41
properties 131
properties of archive forms 464
regular 96
renaming 102, 498
Request ID and 359
Roles form 83
tabbing ID and 430
types 96
User Preferences 335
vendor forms 132
view forms 98, 133
views 256
views, localizing 527
Forms properties, of applications 36
form-style audits 471
From Entry ID field 384
From Form field 384
From Mapping field 384
From Pool field 385
From Server field 384
FTS License Pool field, User form 380
Full Text License Type reserved field 380
functional types properties 437
G
general properties of applications 35
global fields 218
grid
adjusting size 289
aligning fields 287
grid (continued)
   controlling cell movement 289
   size 286
Group Category field 80
Group form
   access control 78, 390
   fields 79
   using 78
   warning 102
Group ID field 79, 379
Group List field 379
Group Name field 79, 379
group permissions 87
Group Type field 79, 379
group types
   Administrator 50
   Assignee 50
   AssigneeGroup 51
   Customize 50
   Flashboards 51
   Public 50
   Subadministrator 50
   Submitter 50
groups
   access control 48
   computed 52
   creating 81
   deleting 82
   dynamic 49, 52
   explicit 49
   implicit 49
   modifying 82
   permissions 79
   regular 52
   reserved by AR System 50
See also access control groups
guest users 63, 89
GUID fields 220
guide permissions
   access control 57
   individual guides 90
   multiple guides 91
guides
   See also guide actions; guide permissions
   active link workflow and entry points 318
   entry point guides 306
H
height, field 430
help
   applications 42, 43
   external files 43
   forms 132
   packing lists 345
help text properties 438
hidden
   fields 430, 431
   forms 58, 93
   pane banner 267
hidden lock type 516
history, status 358
Home Page form
   default installed 308
   modifying 321
   overview 304
   special AR System form 390
home pages
   application list fields 167
   AppSubset field 377
   BMC Remedy User, creating 321
   components 309
   creating 321
   form action fields 329
   form action reserved field ID 387
   how BMC Remedy User appears 331
   how web clients appear 333
   mid tier, specifying server 336
   modifying 321
   navigation aids 329
   Object List, instead of 309
   overview 304
   preferences, configuring 330
   Run Process return 330
   server, user preferences with 335
   suppressing in BMC Remedy User 337
   URL for web client 311
home pages (continued)
  User Preferences form, specifying server and home page 335
  using 308
  web clients, creating 322
horizontal expansion (fields) 291
horizontal field spacing 280
IBM DB2 database and Request ID 360
ID, field 422
image properties 439
images
  on buttons 441
  views and 267
implicit groups
  overview 49
  Request ID field 69
  row-level access 67
import in place
  data with deployable applications 509
  definitions 499
  deployable applications 503
importing
  conflicting types, handling 499, 509
  definitions 490
  deployable applications 500
  excess fields from destination servers, deleting 499, 509
  form data with deployable applications 507
  localized view components 535
  locked objects 517
  object definitions 496
  replacing objects on destination server 499, 509
  source control 498, 509
  view definitions 510
  warning if forms renamed 498
indexes
  defining 139
  form properties 131
individual permissions 86
inetorgperson form 392
Informix database (diary field limit) 145
Informix database and Request ID 360
inherit field help text 121
Initial Currency Type 437
inner join forms 112
input length 424
insert URL link 436
installing languages 531
Instance ID field 382
integer fields 151, 410
Internet Explorer and style sheet attributes 325
interval refresh mode (15-minute interval) 233
italic field labels 419
join criteria 122
join forms
  adding fields to 221
  auditing and 482
  creating 120
  criteria 110, 122
  deleting fields from 222
  fields 126
  fields in 111, 125, 220, 222, 426
  filter execution order 118
  information 426
  Join Wizard 120
  modifying 128
  outer joins 112
  overview 96
  primary form 110, 128
  qualification bar and 123
  removing fields from 222
  Request ID field in 119
  restricted list 121
  secondary forms 110, 128
  self-joins 116
  transaction control 119
  type 129
  understanding 110
  workflow, using in 117
Join Wizard 120
jsp file 265
justifying
  field alignment 429
  text trim 416
Form and Application Objects

K
keeping window open 175
keywords 407

L
labeling table fields 449
labels
align 285
applications and 36
changing color 418
field 429
languages, selecting during install 531
Last Modified By core field 356
layout
forms 282
users changing 268
views 268
Layout menu 284
leading QBE match 426
left (align) 284
length
field 430
field input 424
levels of locked objects 513
License Type reserved field 379
licenses
floating 80
write 63
Like ID field 377
limitations with cascading style sheets in browsers 325
limits (diary fields) 145
line. See trim
links, adding to form 214, 436
list view table fields 154
local applications
See also applications
advantages of 20
converting to deployable applications 45
described 20
locale
field, using search menus with localized 548
setting display 555
view, system searching to display 534
Locale reserved field 383
localizing
advanced tasks 531
BMC Remedy Administrator settings 551
BMC Remedy User preference settings 554
currency types 549
data languages 524
date and time formats 555
display languages 524
e-mail templates 554
environment settings 551
exporting views 535
field labels 536
forms and 533
getting started 526
importing views 535
languages supported 531
languages, selecting during install 531
login pages 550
menus 547
messages 539
mid tier 550
overview 526
process 529
reports 528
request aliases 537
search menus, using Locale field 548
selection fields 538
settings of clients 551
sizes of views 553
user interface of form views 534
view components, manually 536
view to display, system searching for 534
views in web clients 556
location (field position) 429
lock key 516
lock types 513
locked objects
creating 515
exported 515
exporting 495
hidden 513
importing 517
key 516
levels 513
modifying 512
locked objects (continued)
  overview 511
  read-only 513
  types 513
  warning 512
logging archives 466
Login Name field 378
login page
  customizing 550
  login.jsp, localizing 550
logout page, customizing 550
log-style audits 474
Long Group Name field 79

M
mail, creating localized templates 554
maintain aspect ratio 441
Mapping History field 384
Master Flag field 383
match values 426
Max Rows field 185
Max Time to Retry field 385
memory allocation errors 199
menu items, adding 408
menu types
  character 233, 236, 408
  data dictionary 233, 251
dynamic 233, 240
  file 233, 238
  search 240
  SQL 233, 246
menus
See also menu bar; menu items; menu types
  accessing 273
  attaching to field 232
  change history 253
copying 235
creating 232
deleting 235
dynamic fields in search, setting 243
file format 239
localizing 547
modifying 234
qualification for search menu 242
refresh mode 233
menus (continued)
  save as 235
  search 240, 243
Message Catalog form, Remedy 391
messages
  automatically localizing 540
  localizing 539
  manually localizing 541
method to combine views 125
Microsoft
  browser limitations in cascading style sheets 325
  localizing date and time formats in Windows 559
Microsoft SQL Server database
  Request ID and 360
mid tier
  home page in web client 333
  localizing 550
  specifying home page server 336
  style sheets in installation directory 323
Modified Date core field 356
modifying
  fields 223
  form views 261
  forms 101
  groups 82
  Home Page form 321
  home pages 321
  join forms 128
  locked objects 512
  menus 234
  roles 85
moving fields 290
Mozilla and style sheet attributes 325
MSSQL database (diary field limit) 145
multiple
  fields, setting 293
  form views 256
  group names 79
Form and Application Objects

N
name
  field 423
  menu 231
  of menu for character field 408
naming
  applications 35
  packing lists 342
  predefined searches 272
  views of forms 262
navigation aids for home pages 329
navigation fields
  creating 168
  workflow and 169
navigation items properties 441
next available Request ID, changing 360
nonoperational fields 227
numeric text fields 434

O
Object ID field 382
Object List
  entry points appearing in BMC Remedy User 308
  home page, replacing with 309
  web clients, using in 311
objects
  applications and 20
  definitions 488
  displaying by workspace 346
  exporting definitions 490
  importing definitions 496
  including in applications 26
  ownership by applications 29
on 15-minute interval refresh mode 233
on connect refresh mode 233
on open refresh mode 233
open refresh mode 233
Open Window action and dialog boxes 106
operators, EXTERNAL() 160
optional entry mode 425
optional fields 419, 425
Oracle database and Request ID 360
Oracle database, diary field limit 145
orientation (trim) 417
outer join forms 112
overwrite (export) 495, 501
ownership of objects (deployable applications) 29

P
packing lists
  adding objects to workspaces 346
  basics 342
  change history 345
  creating 340
  defining 339
  help, BMC Remedy Administrator 345
  including in applications 348
  naming 342
  overview 339
  permissions 345
  saving in XML 349
  subadministrator permissions 345
  using 340
  window, working in 341
  workspaces 346
page fields
  creating 209
  overview 170
  page holders 443
  permissions 62
  shared field properties 447
  workflow and 171
page holder properties 443
palette, field 174
pane banner visibility 267
Password field 378
Pattern Match field 385
pattern, character 407
performance tips
  archiving data 459
  bitmaps on forms, size of 441
  display-only fields, using for calculations 425
  frequent menu retrieval 233
  indexed fields, using in join criterion 111
  QBE match 426
  refresh table field on entry change 432
  SQL 250

Index ▶ 575
permission types
  Administrator 50
  Assignee 50
  Assignee Group 51
  Assignee group 50
  Customize 50
  Flashboards 51
  Public 50
  subadministrator 53, 92, 132
  Subadministrator group 50
  Submitter 50
  user 82
permissions
  See also access control groups; permission
types; roles
  active links 65
  applications 25
  default 86
  fields 444
  form properties 131
  forms 90
  group 87
  individual 86
  individual objects 90
  multiple objects 91
  packing lists 345
plain field labels 419
position
  field label 429
  fields 282
  pound sign 240
  precision (fields) 412, 413
  predefined searches 270
  preference forms 390
preferences
  forms 390
  home pages, configuring 330
  views 301
prefixes
  lengthening 372
  shortening 369
primary forms
  in applications 41
  in joins 110, 128
primary view in applications 41
printing, disabling 274
Production state field 85, 382
prompt bar 268
properties
  field 397
  field, in joins 220
  form views 263
  forms 131
  join forms 128
properties, archive forms 464
Public group 50
Push Fields action
  attachment pools and 164
  currency fields and 149
Q
QBE match 426
qualifications
  join forms and 123
  predefined searches 272
  search menus 242
  table fields 454
R
R/D (results/details) 268
Read Only columns, tables 435
Read Only field property 432
Read Only-HTML columns, tables 435
read/write field 432
read-only lock type 516
real number fields
  attributes 411
  overview 151
refresh mode (menus) 233
refreshing
  alert list fields 450
  table fields 432
regular forms 96
regular global fields 219
regular groups 52
related workflow 444
Remedy Message Catalog form 391
removing. See deleting
renaming
  forms 102, 498
  views 262
replacing objects on destination server 499, 509
report forms
  Report 391
  ReportCreator 391
  ReportSelection 391
  ReportType 391
reporting
  disabling 274
  localized 528
  run macro actions 553
request aliases
  defining 268
  localizing 537
Request ID field
  changing length or prefix 362–363
  changing value 363
  described 355, 357
  implicit groups and 69
  join forms and 119
  number 360
  preserving existing values 363
  working with 359
Request ID, changing next available 360
requests
  access control 66, 72, 76
  Assignee access 68
  Assignee Group access 68
  copying 163
  dynamic group access 68
  row-level access 72
  row-level access with dynamic groups 76
  sort order 131, 138
  Submitter access 68
  users submitting 63
required entry mode 425
required fields 419, 425
Required Fields field 385
reserved fields
  access control 378
  database properties and 422
  described 375
  DSO 383
reserved fields (continued)
  form action 386
  Locale 383
  placeholders in definitions 377
  ranges 376
reserved groups 50
resizing fields 290, 296
restricted list
  join forms 121
  table fields 453
Restricted List check box 34
results list fields
  creating 182
  drill-down on web 451
  form properties and 131
  labels 449
  overview 153, 156
  permissions 62
  results color 446
  sort order 448
  workflow and 161
results list pane
  defining fields for 135
  fields in 135
  separator 137
right (align) 284
Role ID field 84, 382
Role Name field 84, 381
permissions 53
roles
  creating 85
  deleting 86
  described 53
  modifying 85
  See also access control groups
Roles form
  access control 83
  described 390
  fields 84
  using 83
row-level security 67
Rows display property, fields 430
Rows, maximum number of, tables 454
RPC errors in attachments 199
run process action  
  URLs, opening 318  
  workflow to return to home pages 330  
Run Process action and attachment pools 164  
run with form windows maximized  
  (applications) 40  
runmacro action 553

S
save to disk, applications 40  
saving  
  applications 28  
  image to file 441  
scaling image to fit 441  
search menus  
  creating 240  
  described 233  
  localizing 548  
  qualifications 242  
searches, defining for form views 270  
security  
  databases 251  
  direct SQL commands 251  
select all (align) 286  
selection fields  
  localizing 538  
  overview 151  
  table fields 446  
  view properties 276  
selection values (field menus) 153  
self-joins 116  
send to back (align) 286  
separator for results list pane 137  
Server Events form 392, 468  
server events, archiving 466  
Server Statistics form 392  
servers  
  deleting excess fields from destination 499, 509  
  file location (menus) 239  
  home pages, specifying in User Preferences 335  
  independent, exporting definitions 494  
  information, specifying home page 336  
  localizing 551  
servers (continued)  
  mid tier, specifying home page 336  
  replacing object during import 499, 509  
server-side table fields  
  creating 182  
  overview 153  
Set Fields action and attachment pools 149, 164  
Set Tab Order dialog box 297  
Set to Defaults field 377  
settings  
  special submit 63  
  tab order 297  
shared fields 447  
Short Description core field 356, 358  
shortcuts  
  creating 39  
  exporting objects 490  
  to applications 44  
show  
  grid (align) 286  
  only forms in application 41  
Size 402  
sort order  
  requests 131, 138  
  search menus 242  
  tables 448  
source control  
  checking in definitions 498, 509  
  import in place 499, 503, 509  
spacing, adjusting between fields 291  
spacing, fields 279  
special forms  
  See also forms  
  Alert Events 392  
  Alert List 392  
  Application Statistics 392  
  Application Statistics Configuration 393  
  AR System Application State 393  
  AR System Currency Codes 393  
  AR System Currency Label Catalog 393  
  AR System Currency Localized Labels 393  
  AR System Currency Ratios 393
special forms (continued)
  AR System Email Association 396
  AR System Email Attachment Join 396
  AR System Email Attachments 395
  AR System Email Error Logs 395
  AR System Email Instruction Parameters 395
  AR System Email Instructions 395
  AR System Email Mailbox Configuration 395
  AR System Email Messages 395
  AR System Email Security 395
  AR System Email Templates 395
  AR System Email User Instruction Templates 395
  ARDBC LDAP Configuration 394
  AREA LDAP Configuration 394
  Configuration ARDBC 393
  Distributed Mapping 396
  Distributed Pending 396
  Distributed Pool 396
  Group form 390
  Home Page 308, 390
  inetorgperson 392
  overview 390
  preference 390
  Remedy Message Catalog 391
  Report 391
  ReportCreator 391
  ReportSelection 391
  ReportType 391
  Roles 390
  Server Events 392
  Server Statistics 392
  User form 390

SQL
  menus 233, 246
  rules for performance 250

SQL commands
  Request ID field prefix, lengthening 372
  Request ID field prefix, shortening 369

states
  application state property 36
  applications and 30
  custom, creating 32
  deployable applications and 22
  workflow and 31

statistical processing 151
statistics in table fields 161
Status core field 356, 358
status history 358
status history table 373
Status-History core field 356
Subadministrator group
  access 50
  permissions 53
  subadministrator permissions
    applications 92
    forms 92, 132
    packing lists 345
  submitter access 50
  Submitter core field 355, 357
  Submitter group
    access control 68, 69
    reserved group 50
  subset of entry points
    creating 327
    displaying 167
  Swap Forms button (join forms) 129
  Sybase database and Request ID 360
  Sybase/MSSQL database (diary field limit) 145
  system fields 419

T
  tab order 297, 430
  table drill-down 432
  table fields
    advanced functionality, defining 191
    chunking data 402
    client-side 153, 182
    column properties 435
    creating 182
    described 153
    displaying as trees 436
    drill-down on web 451
    dynamic fields, setting 191
    editable columns 435
    fields in 454
    labels 449
    maximum number of rows 454
    number of rows 430
    permissions 62
table fields (continued)
  properties 184, 453
  qualifications 454
  Read Only columns 435
  Read Only-HTML columns 435
  refresh options, tables 401
  refreshing 432
  restricted list 453
  results color 446
  server-side 153, 182
  sort order 448
  statistics 161
  web views and 451
  workflow and 161
  wrap text 436

table results list color 446

table types, database
  attachment details 373
  status history 373

templates, creating localized email 554
Test state field 85, 382

  text
    editing window 145, 424
    fields 434, 436
    specifying a pattern 407
    trim style 416

  thickness of trim 416

time fields 146
time formats
  ARDATE 557
  customizing for localization 557
  localizing 555
  Windows localizing 559

tips
  add form action fields before locking forms 521
  application default permissions, easily setting up 319
  hierarchy of values in opening home pages 330
  limiting number of entry points 305
  order of entry points 314

  title bar icon 267
  title bar, customizing in applications 40
  To Entry ID field 384
  To Form field 384
  To Mapping field 383
  To Server field 384
  toolbars, hiding on forms in browser 267
top (align) 284
tracking changes 470
transactions in join forms 119
Transfer Mode field 385
Transfer Status field 383
tree view table fields 155, 436
trim
  adding to a form 172
  alignment 416
  color 417
  creating 213
  depth effect 416
  display mode 417
  fields 415
  justification 416
  orientation 417
  overview 172
text style 416
  thickness 416
  URLs 172, 214, 419
types of forms 96

U
unique (index) check box 141
UNIX
  date and time environment variables 556, 557
  field descriptors, combining 557
Update Status field 383
URLs
  adding to a form 214, 436
  buttons displayed as 435
  changing color 215
  home pages for web client 311
  trim 172, 419
  web views of forms, opening 556
User form
  access control 390
  warning 102
  user interface of form views, localizing 534
user preferences
  home page and server, specifying 335
  suppressing home pages in BMC Remedy User 337
users
  access control 53
  changing layout 268
  guest 63, 89, 91
  permissions 82
V
values
  color of (results list) 276
  setting initial 165
variables
  ARDATE environment 557
  ARDATEONLY environment 557
  ARTIMEONLY environment 557
vendor forms
  definition 98
  viewing information 132
vertical expansion (fields) 291
vertical field spacing
  by length 281
  evenly 279
view fields
  creating 199
  overview 164
  workflow and 166
view forms
  definition 98
  displaying information 133
view forms, special considerations for archiving 462
view properties 455
viewing entry points, subset of 326
views
  adjusting size 553
  appearance 266
  changing current display 261
  creating 256
  custom labels for entry points 314
  defining searches 270
  deleting 263
  display, system searching to 534
views (continued)
  displaying fields 456
  entry points 313
  exporting 553
  exporting and importing definitions 510
  field permissions 60
  form permissions 79
  form, localizing 527
  including/excluding fields 299
  localized 533
  menu access 273
  message components, localizing 539
  modifying 261
  multiple 256
  naming 262
  preference settings 301
  primary 41
  properties 263
  request aliases 268
  results list color 276, 277, 446
  visible pane banner 267
W
warnings
  deleting excess fields on server while importing definitions 499, 509
  deleting excess fields on server while importing locked objects 518
  deleting excess fields on the server while importing deployable applications 503
  exporting messages using .csv file format 540
  Group form 102
  locked objects cannot be unlocked 512
  locking forms and workflow with different keys 515
  renaming forms during import 498
  replacing fields with new data type while importing definitions 499, 510
  replacing fields with new data type while importing deployable applications 503
  replacing fields with new data type while importing locked objects 519
  User form 102
web alias 36, 265
BMC Remedy Action Request System 7.0

web clients
  browser limitations in cascading style sheets 325
  home pages, creating 322
  how home page appears 333
  localized form views, accessing 556
  Object List, displaying 311
  style sheets, using with 325
web views
  tables 451
  URLs of forms, opening 556
Weight field 377
When to Update field 385
width
  of fields 430
  of results list columns 137
windows
  Application 20, 28
  global fields and 219
Windows. See Microsoft
window-scoped global fields 219
workflow
  See also dynamic fields
  alert list fields and 161
  application list fields 167
  application list fields and 167
  attachment pools and 164
  join forms and 117
  navigation fields and 169
  page fields and 171
  related to fields 444
  Restricted List 34
  results list fields and 161
  Run Process return to home page 330
  subset of entry points, sample 327
  table fields and 161
  view fields and 166
workflow actions
  Change Field 149, 164
  Message 164
  Push Fields 149, 164
  Run Process 164
  Set Fields 149, 164
worksaces
  displaying objects in 346
  enabling 347
  using with packing lists 346
  wrap text in table fields 436
  write field 432
  Write License Pool field 380

X
  X (horizontal position of field) 429
XML
  file type 489
  saving packing lists in 349

Y
  Y (vertical position of field) 430