
Get It!

Get.Resources!
Quick Start Guide
Release 2.0.1
September 2001

ITEM GET-2.0.1-ENG-01003-00088

Peregrine Systems, Inc.
3611 Valley Centre Drive
San Diego, CA 92130
www.peregrine.com


S Y S T E M S[®]
The Infrastructure Management Company™

© Copyright 2001 Peregrine Systems, Inc. or its subsidiaries
All Rights Reserved.

Information contained in this document is proprietary to Peregrine Systems, Incorporated, and may be used or disclosed only with written permission from Peregrine Systems, Inc. This book, or any part thereof, may not be reproduced without the prior written permission of Peregrine Systems, Inc. This document refers to numerous products by their trade names. In most, if not all, cases these designations are claimed as Trademarks or Registered Trademarks by their respective companies.

Peregrine Systems®, **ServiceCenter®**, and **AssetCenter®** are registered trademarks of
Peregrine Systems, Inc. or its subsidiaries.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). This product also contains software developed by the following companies or individuals: Sun Microsystems, Inc., Jean-Marc Lugin, Netscape Communications Corporation, and Original Reusable Objects, Inc.

This document and the related software described in this manual are supplied under license or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement. The information in this document is subject to change without notice and does not represent a commitment on the part of Peregrine Systems, Inc. Contact Peregrine Systems, Inc., Customer Support to verify the date of the latest version of this document.

The names of companies and individuals used in the sample database and in examples in the manuals are fictitious and are intended to illustrate the use of the software. Any resemblance to actual companies or individuals, whether past or present, is purely coincidental.

This edition applies to version 2.0.1 of the licensed program

Get.It!

Contents



Introduction

About this Guide	v
Related Documentation	vi
Organization of the Guide	vi
Conventions Used in this Guide.....	vii

Part 1 Get.Resources! with AssetCenter

Configuring Get.Resources! with AssetCenter

Overview of Get.Resources! Architecture	1-2
Configuring AssetCenter	1-2
Configuring Get.Resources! to use the AssetCenter Demo Database	1-2
Configuring Get.Resources! to use an Existing AssetCenter Database	1-2
Importing the Scripts.....	1-3
Authorizing Access	1-7
File Attachments	1-7
Archway.ini Parameters.....	1-8

Logging in to Get.Resources!	1-9
Get.Resources! Features	1-11
Selecting the Contents of your Portal	1-13

Administration and Security

Get.Resources! Security	2-1
User Registration and Authentication	2-1
User Rights	2-1
Setting User Rights with AssetCenter	2-2
Sample Profiles	2-3
Get.Resources! Administration	2-5
Parameters	2-5
AssetCenter Modification	2-5
Personalization Access Rights	2-8
Personalization Hierarchy	2-8
User Personalization Rights	2-9
Personalization Database	2-9

Data Reference

AssetCenter Catalog	3-1
Certification	3-1
Calculated Field: cf_Description	3-2
AssetCenter Workflows	3-3
Bundle Ordering Workflow	3-4
Routing Request Workflow	3-4
Request Approval Workflow	3-5
Automatic PO Generation Workflow	3-6
Request Status Workflow	3-7

Part 2 Get.Resources! with ServiceCenter

Configuring Get.Resources! with ServiceCenter

Overview of Get.Resources! Architecture	4-1
Configuring Get.Resources!	4-1
Defining ServiceCenter as the Back-end System	4-2
Loading the ServiceCenter files—ServiceCenter 3.x	4-2
Defining User Rights	4-5
File Attachments	4-7
Archway.ini Parameters.....	4-8
Logging in to Get.Resources!	4-8
Get.Resources! Features	4-10
Selecting the Contents of your Portal	4-12

Administration and Security

Get.Resources! Security	5-1
User Registration and Authentication.....	5-1
User Rights	5-2
Setting User Rights with ServiceCenter.....	5-2
Get.Resources! Administration	5-2
Parameters.....	5-2
Enabling Approval Rights.....	5-3
Personalization Access Rights.....	5-3
Personalization Hierarchy.....	5-3
User Personalization Rights	5-4
Personalization Database.....	5-5

Introduction



Get.Resources! from Peregrine Systems® provides a way for businesses to make the procurement of resources available to users through a corporate intranet Web browser.

A sample AssetCenter database is included with Get.Resources!. If you are using AssetCenter for your back-end system, you will have access to the AssetCenter product catalog for ordering purposes. Get.Resources! processes the initial request through an approval cycle, showing an animated workflow display, and on through the procedures of ordering and receiving the items.

If you are using ServiceCenter for your back-end system, you have access to the Request Management catalog in ServiceCenter. ServiceCenter 3.0 SP 2 and later, and ServiceCenter 4.x are supported with Get.Resources! 2.0.1. Refer to Chapter 4, "Configuring Get.Resources! with ServiceCenter" for configuration instructions.

About this Guide

This guide is designed to be used by administrators configuring and maintaining the Get.Resources! weblication.

To use this guide effectively, you should have a working knowledge of the following:

- XML and ECMAScript (or JScript/JavaScript)
- Operating guides, reference manuals, and other documentation for your PC hardware and operating system
- JRun documentation located in the ...JRun\docs directory
- The back-end system you will be using with Get.Resources!

Related Documentation

This guide should be used in conjunction with the *Get.It! Quick Start Guide*. For tailoring information and instructions for using Get.It! Studio, refer to *Get.It! Studio Introduction* and the *Get.It! Tailoring Guide*.

Organization of the Guide

The following table illustrates the parts of the guide that you need to look in to find the information you need.

To Find This	Look Here
Background information; how to use this manual.	Introduction.
How to configure Get.Resources! for the AssetCenter database; how to log in to Get.Resources!; customizing your portal.	Chapter 1, "Configuring Get.Resources! with AssetCenter."
User authentication; setting Personalization access rights.	Chapter 2, "Administration and Security."
Information about the AssetCenter catalog; AssetCenter workflows.	Chapter 3, "Data Reference."
How to set up Get.Resources! with ServiceCenter as the back-end system; Get.Resources! features.	Chapter 4, "Configuring Get.Resources! with ServiceCenter."
User authentication with ServiceCenter; setting Personalization access rights.	Chapter 5, "Administration and Security."

Conventions Used in this Guide

Screen shots in this guide are included as examples only. Get.Resources! screens are shown using the Classic stylesheet.

The following documentation conventions are used in this guide:

Object	Font	Example
Button	New Century Schoolbook Bold	Click Next
File name	Arial Bold	The login.asp file
Sample script or XML code	Courier New	<pre>var msgTicket = new Message("Problem"); ... msgTicket.set("_event", "epmc");</pre> <p>The ellipsis (...) is used to indicate that portions of a script have been omitted because they are not needed for the current topic. Samples of code are not entire files, but they are representative of the information being discussed in a particular section.</p>
Book title	New Century Schoolbook Italic	Refer to the <i>Get.It! Quick Start Guide</i> ..
Menu option	New Century Schoolbook Bold	Select Start>Program Files .



Part 1 Get.Resources! with AssetCenter

Chapter 1

Configuring Get.Resources! with AssetCenter

Get.Resources! is installed during the installation of Get.It!. Instructions are included in Chapter 2 of the *Get.It! Quick Start Guide*. This chapter leads you through the following procedures:

- Configuring Get.Resources! to use the AssetCenter demo database.
- Configuring Get.Resources! to use an existing AssetCenter database.
- Logging in to Get.Resources!.

A summary of Get.Resources! features and a workflow diagram are also included.

Although an AssetCenter demo database is included in the Get.It! installation, AssetCenter itself is installed separately from Get.It!. Choose either a full AssetCenter installation, or a custom installation, selecting the “AssetCenter API” option. Refer to your AssetCenter documentation for instructions.

Overview of Get.Resources! Architecture

Refer to the *Get.It! Quick Start Guide* for detailed information about the Get.It! architecture.

Get.Resources! connects to the AssetCenter database through a special adapter, the ACAdapter. As requests are opened and processed through Get.Resources!, data is stored and updated in the AssetCenter database.

Users are authenticated using AssetCenter Profiles, with special user rights provided for use with Get.Resources!. For more information about user authentication, see Chapter 2, "Administration and Security". The *Get.It! Quick Start Guide* also contains information about user authentication.

Configuring AssetCenter

Although Get.Resources! is installed with all the proper connectivity to interface with AssetCenter, you will need to configure Get.Resources! to connect to the database you want to use, either the demo database provided with Get.It! or your existing AssetCenter database.

Configuring Get.Resources! to use the AssetCenter Demo Database

The following steps must be completed to enable Get.Resources! to use the AssetCenter demo database supplied with Get.It!:

- In AssetCenter, declare a new connection to the demo database. This configures a User DSN only.
- Set up a System DSN. Chapter 6 of the *Get.It! Quick Start Guide* contains instructions for setting up a DSN in the section about the JDBC adapter.
- Test the connection by logging in to the database from AssetCenter.
- Use the Get.It! Admin module to point Get.It! to the database and define a user name and password. Refer to Chapter 4 of the *Get.It! Quick Start Guide* for information about the Admin module.

Configuring Get.Resources! to use an Existing AssetCenter Database

Configuring Get.Resources! for an existing AssetCenter database includes the following steps:

- Use AssetCenter to connect to the database you want to use.
- Import new scripts supplied with the Get.It! installation.

- Use the Get.It! Admin module to point Get.It! to the database and define a user name and password.

Importing the Scripts

Complete the following steps to import the new Get.Resources! information into your existing database.

1. Log on to AssetCenter.
2. Select **File>Import**.

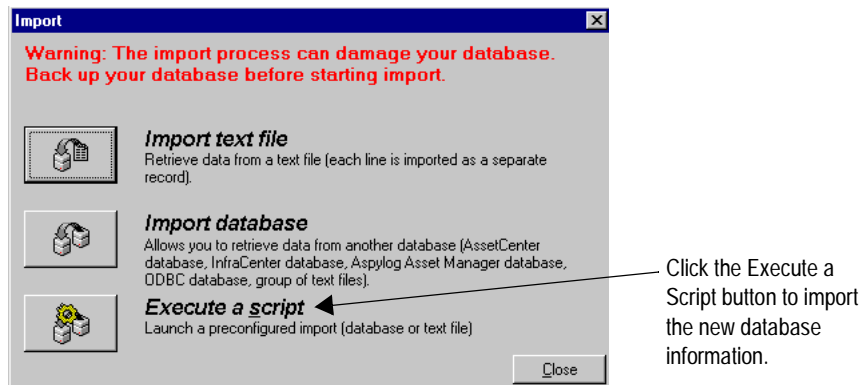


Fig. 1.1 AssetCenter—Import option

3. Click **Execute a script**.

A window titled Execute Import Script is displayed.

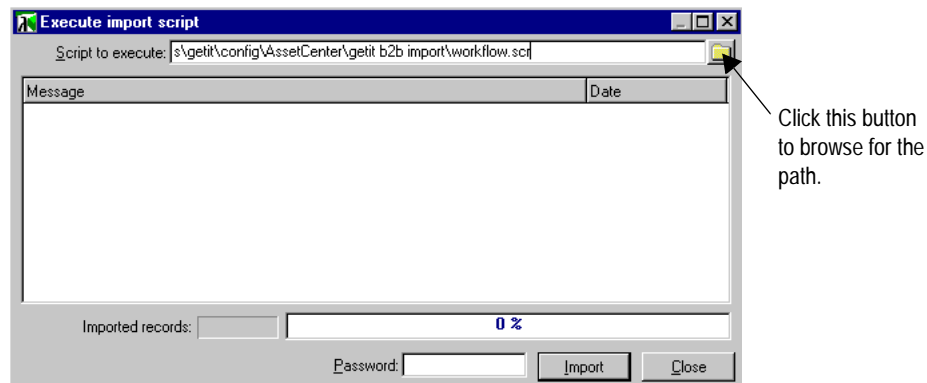


Fig. 1.2 Entering the script to execute

4. There are multiple files you need to import to update the existing file. Click the button to browse for the following folder:
 ... \getit\packages\GetResources 2.0.1\resources\config\ac. (If you installed Get.Resources! into a different directory, you will have a different path.)
5. Within this folder you will see two folders called getit import and getit b2b import. The scripts you need to import first are two files in the getit import folder and the same two files in the getit b2b import. All scripts have the .scr file extension.

From the getit import folder, import the following in the order listed:

- **feat.scr**
- **calcflds.scr**

After importing the above two scripts, import the following two scripts in the order listed from the getit b2b import folder. These must be imported even if you are not using B2B functionality:

- **calcflds.scr**
- **feat.scr**

Finally, from the getit import folder, import the following scripts in the order listed:

- **profile.scr**
- **workflow.scr**
- **empldept.scr**

The following table describes the function of each script.

Script	Description
feat.scr	Features for table amEmployee; contains DelegatedApproverId and DelegatedApprExpiration scripts. These features delegate request approvals to a person given by DelegatedApproverId, until the date specified by DelegatedApprExpiration. Features for table amProduct include ManufacturerURL, a text-based feature containing the manufacturer's URL. Others include Architecture, Processor, ProcSpeed, RAM, and HDCapacity, used by the Description calculated field to enhance the product description, none of which are used directly by Get.It!
calcflds.scr	Calculated field Description for the Product table. Gives a technical description for the product.

Script	Description
profile.scr	<p>Imports Get.It! user rights to control access to the screens.</p> <p>These include:</p> <ul style="list-style-type: none"> getit.requester (access to the request) getit.advancedrequester getit.approver getit.receiver getit.pcardadmin getit.pcarduser getit.pcardbuyer <p>The final user right, getit.receiver, imports three predefined user profiles.</p> <p>These include:</p> <ul style="list-style-type: none"> getit.admin = getit.admin + getit.requester + getit.advancedrequester + getit.approver + getit.receiver + getit.pcardadmin + getit.pcarduser + getit.buyer getit.default = getit.requester getit.full = getit.requester + getit.advancedrequester + getit.approver + getit.receiver + getit.pcardadmin + getit.pcarduser + getit.buyer getit.buyer = getit.advancedrequester + getit.buyer

Script	Description
workflow.scr	<p>Imports necessary workflows to make Get.Resources! work out-of-the-box. Listed in order of Reference, Name, and Description.</p> <p>BUNDLPO; Bundle Ordering; Workflow necessary to work with the bundles</p> <p>REQROUTE; Routing Request; Sample workflow that manages the full qualification of an off catalog request before it enters the approval cycle</p> <p>REQAPPR_GR; RequestApproval; Sample approval workflow. It is important to note that only the workflow with the REQAPPR_GR reference will be displayed in Get.It!.</p> <p>REQSTATUS; Request Status; Workflow that, on fulfillment of the request, changes its status to "Received" for better Get.Resources! visibility.</p> <p>The REQSTATUS script also imports three employee workgroups. These include:</p> <p>IT: Technical approvers group (any employee in this group can give a technical approval for a request, provided one has the getit.approver user right).</p> <p>Finance: Financial approvers group (any employee in this group can give a financial approval for a request, provided one has the getit.approver user right).</p> <p>Getit: Employees group that will qualify the off catalog request.</p> <p>Note: The off catalog request qualification must be performed in AssetCenter. There is no module for this task in Get.Resources!</p>
empldept.scr	<p>This script is optional and is considered demonstration data. It imports one employee per profile, as defined in the following.</p> <p>Valentine: Michael Valentine is a getit.admin.</p> <p>Hartke: Richard Hartke is a getit.full.</p> <p>Tossi: Michaela Tossi is a getit.default.</p>

6. Click on the script you want to import, and then click **Open**.
The path for the script you have chosen will be displayed in the Execute a Script window.
7. Click **Import**.
8. Repeat this process until you have imported all of the scripts.
When the import is complete, you are ready to use the new information in the database. You do not need to restart AssetCenter.

9. Log on to AssetCenter and go to **Tools >Workflow >Workflow schemes**.
The Get.It! imports included the "Request approval" workflow. Another workflow, "Purchase request validation", may conflict with this new Get.It! approval workflow. If the "Purchase request validation" workflow is present in your AssetCenter system, it must be disabled as follows:
 - a. In the list of Workflow schemes, select "Purchase request validation".
 - b. In the Validity section of the General tab, clear the Start field. In the End field enter a date in the past.
 - c. Click **Modify** to save your changes.
10. In AssetCenter, verify that the User Profiles have been updated with the new Get.Resources! options. The profiles should now include getit.default, getit.full and getit.admin.

Authorizing Access

If you have installed AssetCenter for the first time, log on to AssetCenter and enter your AssetCenter authorization code.

Note: If you do not enter an authorization code, Get.Resources! will not be able to access AssetCenter. The DLL connection fails if an authorization code has not already been given for AssetCenter.

If Get.Resources! and AssetCenter are installed on different servers, use the Get.It! Admin module to update the **archway.ini** file with the correct AssetCenter connection information. See the *Get.It! Quick Start Guide* for information on using the Admin module.

File Attachments

Get.Resources! provides functionality that allows users to attach files (for example, a Microsoft Excel or Word file) to Get.Resources! requests to provide additional information.

During the installation of Get.It!, an attachments virtual directory was created in the presentation directory. However, as a security safeguard to your system, it is recommended that you change the permissions on the attachments virtual directory.

The procedure below uses Internet Information Services (IIS) as an example.

1. Open IIS.
2. In the getit directory, right-click on the attachments directory, and then choose **Properties**.

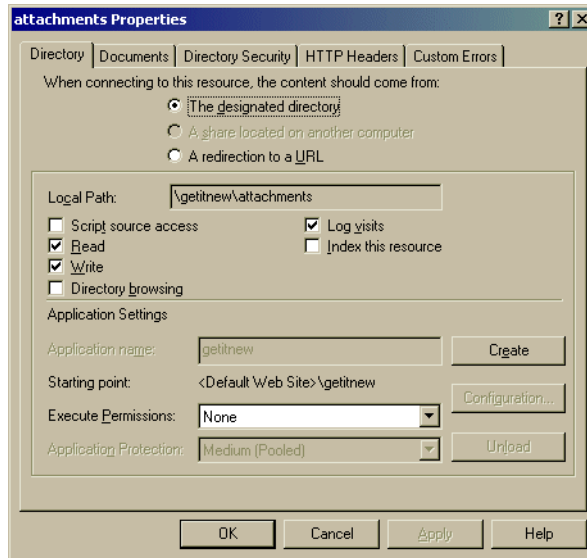


Fig. 1.3 Attachments Properties

3. Set the permissions to *Read* and *Write*.
4. Click **OK**.

Archway.ini Parameters

The File Attachment functionality uses defaults that can be overridden by the following parameters in the **archway.ini** file. These parameters allow you to set the paths for the file attachments and to establish a maximum allowable size for the attachment.

Parameter	Description
attachvirpath	Sets the Web directory reference from which files will be drawn for viewing. The default is /getit/ attachments/ if not set in the archway.ini file.
attachserverpath	Sets the physical path on the Web server where the files will be stored. The default is <getit home>\presentation\attachments\ if not set in the archway.ini file.
attachmaxsize	Establishes the maximum size in KB for an attached file. The default is 2048 KB (2 MB).

Logging in to Get.Resources!

There are three sample users included with Get.Resources!: Richard Hartke, Michaela Tossi, and Karen Buyer. Each of these sample users has a profile set up in the AssetCenter demo database to illustrate various user access rights to Get.Resources!. You can also view the tables in AssetCenter to which each of these user rights have access from the Profile tab for each user. Refer to your AssetCenter documentation for instructions for viewing user rights and access to tables.

The following tables show the user profiles and the user rights included in each profile.

Hartke login

Profile	User Rights
getit.full	getit.requester
	getit.advancedrequester
	getit.approver
	getit.receiver
	getit.b2badmin
	getit.shopdirect<VENDOR>
	getit.pcardadmin
	getit.pcarduser
	getit.buyer

Tossi login

Profile	User Rights
getit.default	getit.requester

Buyer login

Profile	User Rights
getit.buyer	getit.advancedrequester
	getit.buyer

The following procedure uses the Hartke login as an example.

To log in to Get.Resources!:

1. Open a Web browser and type localhost\getit\login.jsp.

The login screen is displayed.

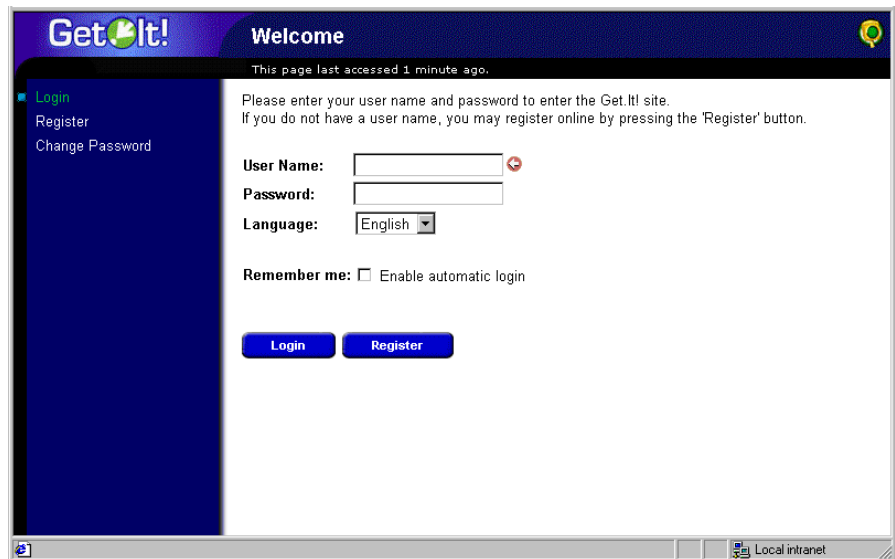


Fig. 1.4 Get.Resources! login screen

2. Type Hartke in the User Name field. The arrow to the right of the field indicates that it is a required field. There is no password required for either of the sample logins.

3. Click **Login**, or press ENTER.

The Get.It! portal is displayed.

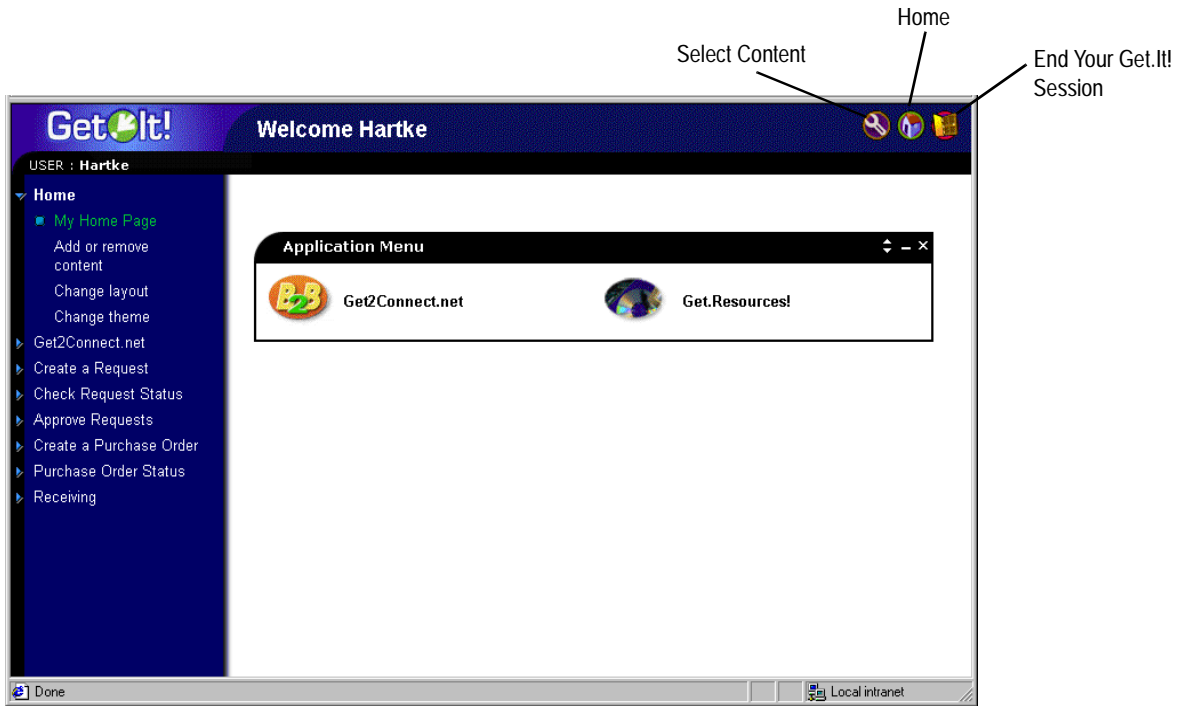


Fig. 1.5 Get.It! portal

The menu on the sidebar includes all of the activities available to the logged-in user. As you navigate through Get.Resources!, the sidebar menu remains visible. To return to the portal, click the **Home** icon at any time.

Get.Resources! Features

Get.Resources! provides a comprehensive interface for processing a request for a product through an approval process, opening a purchase order, to the time the product is received. From the time a request is opened, the status of the request can be followed through animated workflows that are dynamically updated as the request is moved through the approval process.

Get.Resources! is role-based, providing sample users to illustrate this process. Three user logins are provided:

- Hartke, with requesting, approving, and receiving privileges.
- Tossi, an end-user with requesting capabilities only.

- Buyer, with purchasing capabilities only.

Figure 1.6 shows the Get.Resources! workflow.

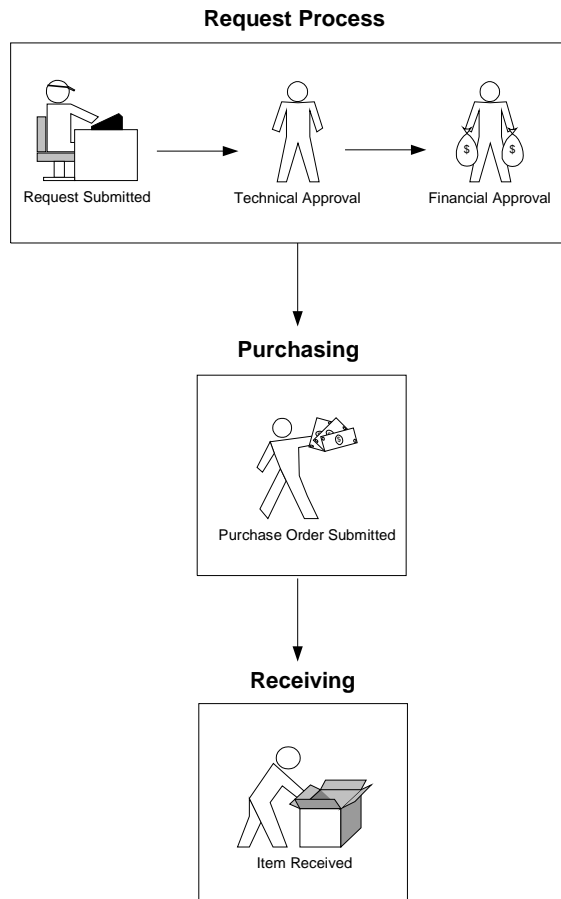


Fig. 1.6 Get.Resources! workflow

When a user starts the process of opening a request, a shopping cart is provided in which totals are dynamically updated as new items are added. Catalog items are displayed according to the category chosen by the user and as defined in the AssetCenter catalog.

Figure 1.7 shows a typical catalog screen in Get.Resources!. The sidebar displays all of the catalog options available, as well as the shopping cart.

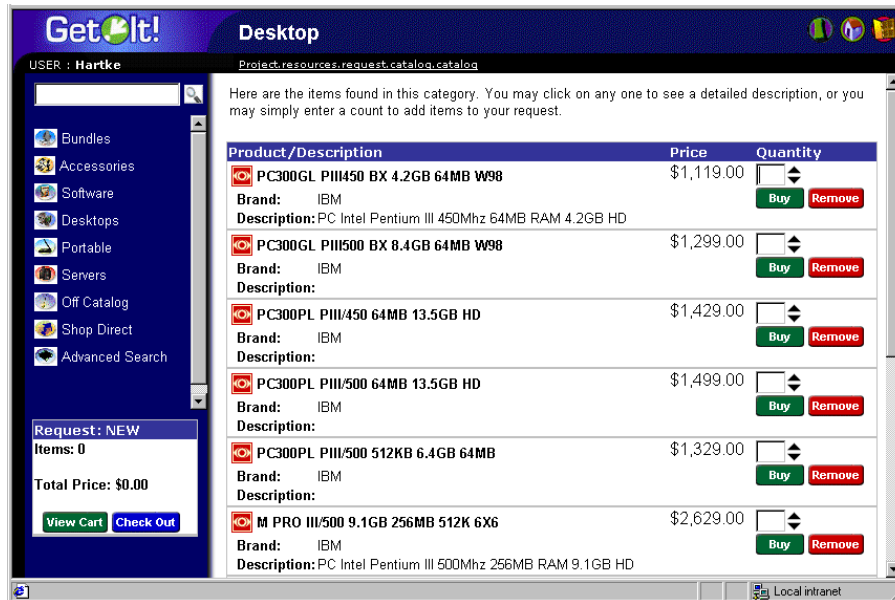


Fig. 1.7 Catalog entries for the Desktop category

You can bookmark a Get.Resources! page as you would any other Web page. With the page you want to bookmark displayed, right-click in the frame you want to return to and select **Add to Favorites**.

You can also send a coworker an HTTP link to a page—for example, someone processing a request might want someone else to look at the request for approval. The link would be something like:

http://webserver/getit/e_service_status_whichone.jsp?Id=REQ000040

If the user is logged in to Get.Resources!, clicking on the link will take them to the designated page. If the user is not logged in, the login screen will be displayed, and after the user has logged in the requested page will be displayed.

Selecting the Contents of your Portal

The portal can be modified using the Select Content feature.

To use Select Content:

1. With the portal displayed, click the **Select Content** icon (wrench).

The Select Content page is displayed.

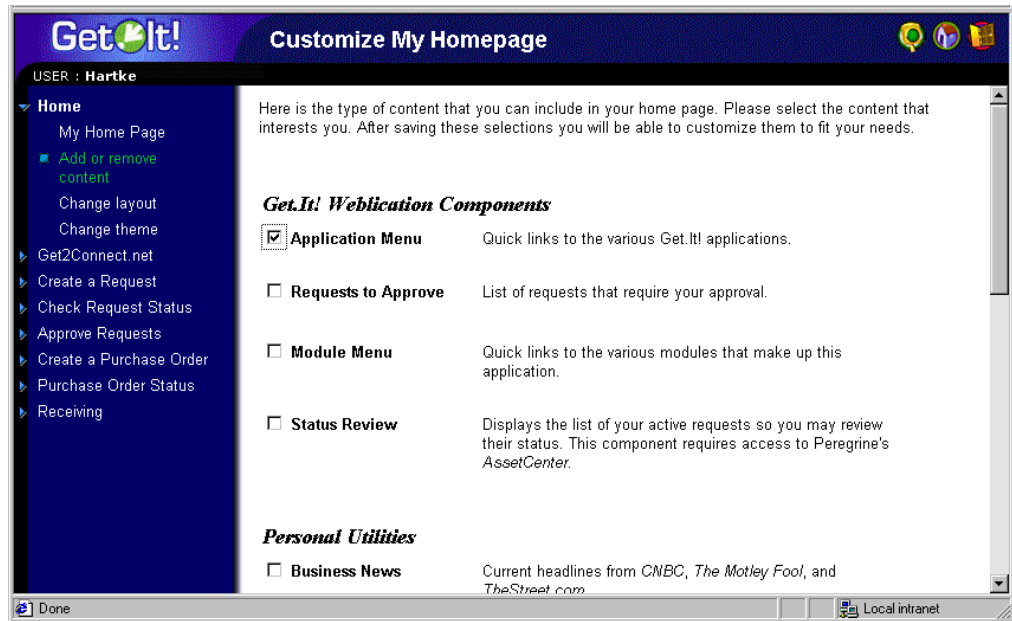


Fig. 1.8 *Select Content*

By default, the Application Menu is selected.

2. Click the check box for the items you want to be displayed on your portal.

3. Scroll to the bottom of the page, and then click **Save**.

You are returned to the portal and the items you have selected are displayed.

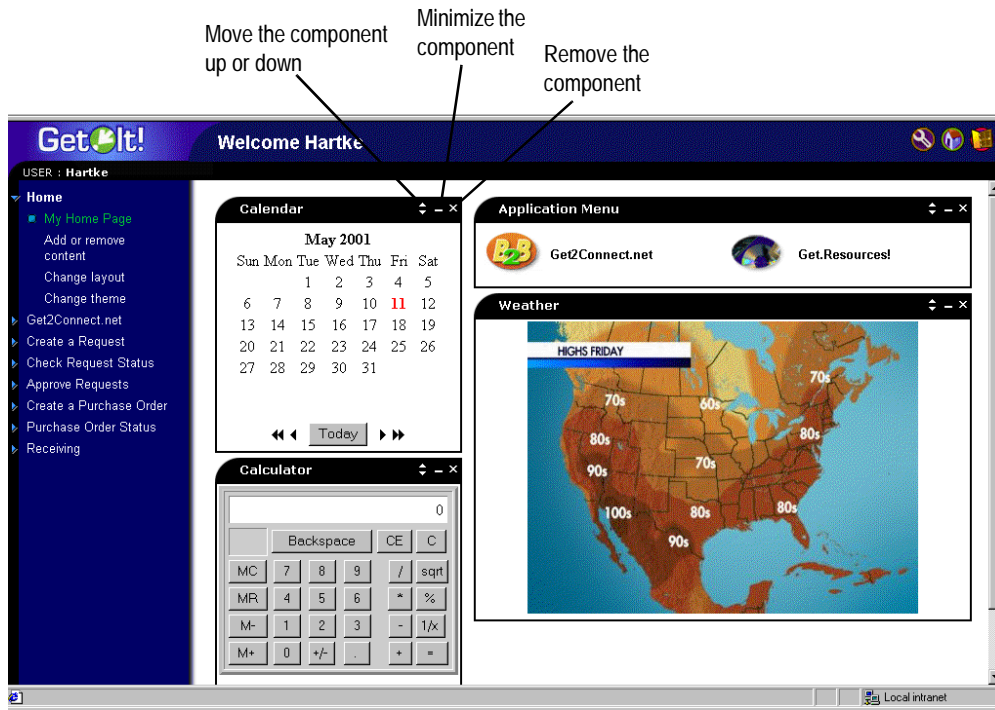


Fig. 1.9 Portal with Calendar, Calculator, and Weather added

Each of the portal components can be moved, minimized, or removed from the portal.

- Click the **triangle** button to move the component up or down on the page.
- Click the **minus** button to minimize the component.
- Click the **x** button to remove the component.

In Figure 3-1.10 on page 3-16, the Calculator component has been moved up and minimized.

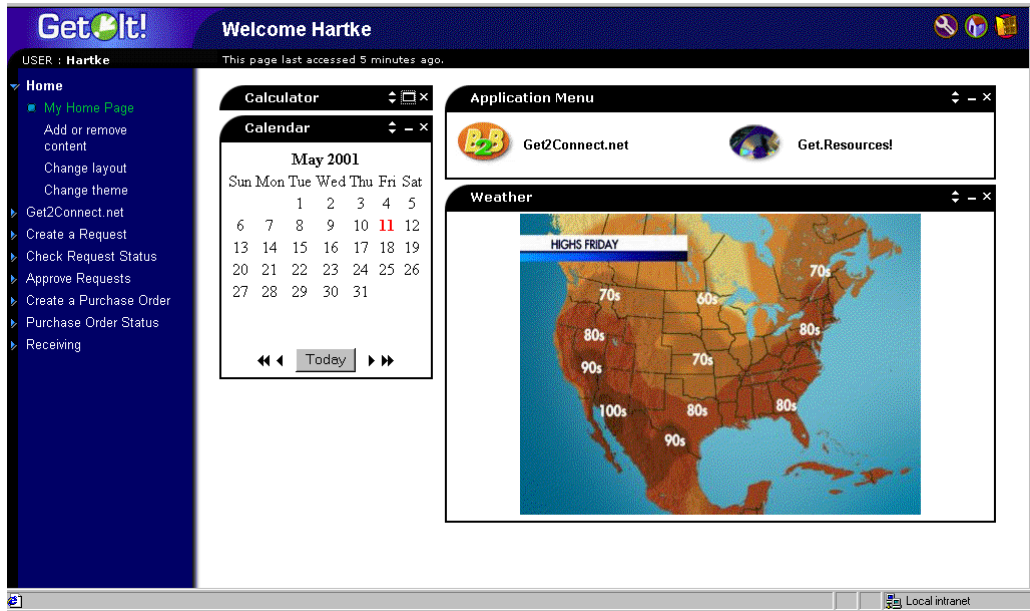


Fig. 1.10 Calculator component moved and minimized



Chapter 2

Administration and Security

This chapter includes information about administrative tasks and security issues specific to Get.Resources!. It is recommended that you refer to the *Get.It! Quick Start Guide* for information about general Get.It! administrative tasks in conjunction with reading this guide.

Get.Resources! Security

With Get.Resources!, users can register themselves and log on using any name currently registered in the AssetCenter or ServiceCenter database.

To access AssetCenter, each user must have a Profile record established. To access Get.Resources!, specific Get.It! user rights must be added to the user's AssetCenter Profile or ServiceCenter Operator record.

User Registration and Authentication

When a user registers in Get.Resources! for the first time, an **amEmpIDept** record is created in AssetCenter. The user name and password are stored in the **amDempIDept** table and a default Profile is assigned, *getit.default*.

Each time the user attempts to log on to Get.Resources!, the user name and password are validated against AssetCenter. If the entered user ID and password are not valid, the user will be prompted to enter valid data. If the log on information is correct, Get.Resources! will retrieve the access rights for the user and log the user on to Get.Resources!.

User Rights

AssetCenter utilizes user rights to identify which tables an operator can modify within the system. User rights are grouped together in Profiles. Individual Employee records can have a single Profile associated to them.

Get.Resources! examines the user rights of the logged-on user and provides access to the appropriate portions of the webapplication. AssetCenter continues to restrict table access based on the detail of any given user rights record.

Setting User Rights with AssetCenter

Get.Resources! uses an administration profile to connect to AssetCenter through the AssetCenter adapter. The parameters **acadmin** and **acadminpassword** can be set by using the Admin module Settings page. This allows Get.Resources! to connect to AssetCenter. User rights that have been defined in AssetCenter are then used to determine which tables in AssetCenter a logged-on user will be able to access and, therefore, which parts of Get.Resources! the user will see.

Get.Resources! users must have a profile that has one or more of the following user rights keywords as appropriate:

User rights keywords	Access rights
getit.advancedrequester	Enables access to advanced request form features in Get.Resources!, including the ability to split request lines and assign request line items to different end users. This is useful for requesters who typically request items for a group of people.
getit.requester	Enables access to create requests in Get.Resources!
getit.approver	Enables access to approve requests.
getit.receiver	Enables access to receive items associated with a request.
getit.b2badmin	Enables access to administer ShopDirect B2B connectivity.
getit.shopdirect	Enables access to perform ShopDirect sessions at supported vendor Web sites.
getit.shopdirect.[VENDOR]	Enables access to a specific vendor Web site.
getit.buyer	Enables user to create and change the status of purchase orders.
getit.pcardadmin	Enables users to create new Pcards and manage the rights for the Pcards they created.
getit.pcarduser	Gives the user access to Pcards on the request and purchase order screens. These fields are not displayed if this access is not granted.

The demo database provided with Get.Resources! assigns a number of access capabilities to the user rights used by the weblications. However, administrators can change these rights as necessary for users who need access to a standard AssetCenter client.

For instance, the `getit.requester` user right as shipped contains access to many AssetCenter tables. Figure 2.1 shows how these rights can be revoked to limit access to certain Get.Resources! functionality. Refer to your AssetCenter documentation for more information.

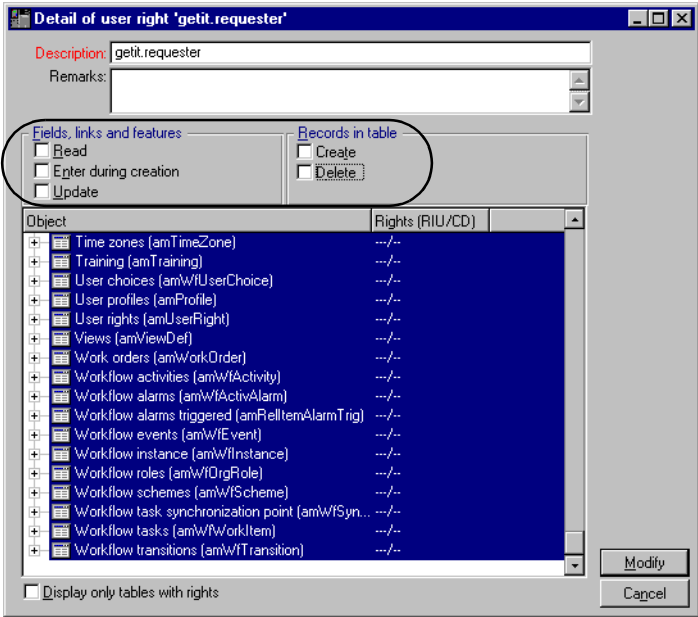


Fig. 2.1 User rights revoked

Sample Profiles

Get.Resources! provides several pre-set sample profiles which combine user rights to give access to various modules. You can create profiles in AssetCenter with any combination of rights you want.

The following sample profiles are included with Get.Resources!:

Profile	Module you can Access
getit.admin	Administration, B2B Administration, Resources, Status, Approval, Receiving, Shop Direct, Pcard
getit.default	Resources, Status
getit.full	Resources, Status, Approval, Receiving, B2B Administration, ShopDirect, Pcard
getit.buyer	Purchasing, Pcard, Resources
getit.requester	Resources, Status, Pcard. This profile provides the minimum rights needed to complete a request.

When users first register, they are assigned *getit.default* authority. You can update the AssetCenter Employee records of those users who will need full or administration access as shown in Figure 2.2.

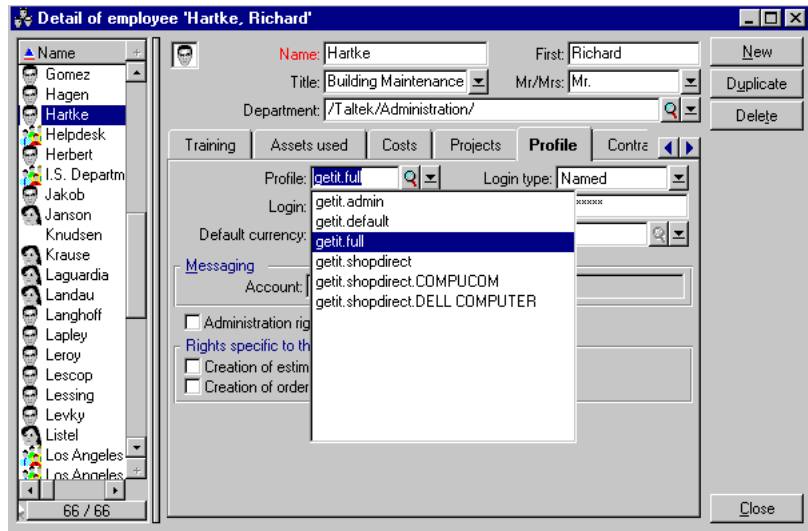


Fig. 2.2 AssetCenter profile options

If you want to revoke user rights, use the procedure shown in Figure 2.1 in the previous section. If you want to add rights, it is better to create a new user profile in AssetCenter with the rights you want to grant to that profile.

Get.Resources! Administration

This section includes administrative tasks specific to Get.Resources!.

Parameters

Refer to the *Get.It! Quick Start Guide* for a complete list of parameters.

Get.Resources! has two parameters that are specific to the webplication:

- Request Backend—establishes the back-end system being used with Get.Resources!. The default is *ac* (AssetCenter).
- B2B Server—the URL for the Peregrine B2B Server. The default is <https://www.getmarketaccess.net>.

These parameters are located on the Get-Resources tab of the Settings page in the Get.It! Admin module.

AssetCenter Modification

If you will be using the Purchase Order function in Get.Resources!, you will need to change the Mandatory attribute for the Purchase Order Supplier field. This procedure is necessary for all versions of AssetCenter, and will enable users to save Purchase Orders for later completion without naming a Supplier.

To modify the Mandatory attribute:

1. Open the AssetCenter Database Administrator.
2. Open the AssetCenter database you are using.
3. From the drop-down list at the top of the screen, select the Orders (AmPOrder) table. This table name may display as AmPOrder (Orders), depending on how your version of AssetCenter is configured.

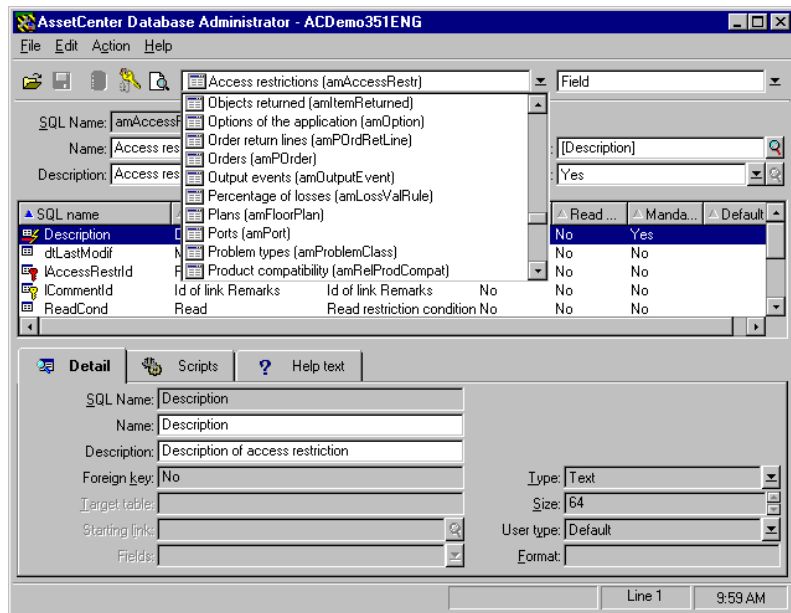


Fig. 2.3 AssetCenter Database Administrator

4. Locate the field named ISuppld and select it.
5. Click the Scripts tab. Change the Mandatory property to (Script).

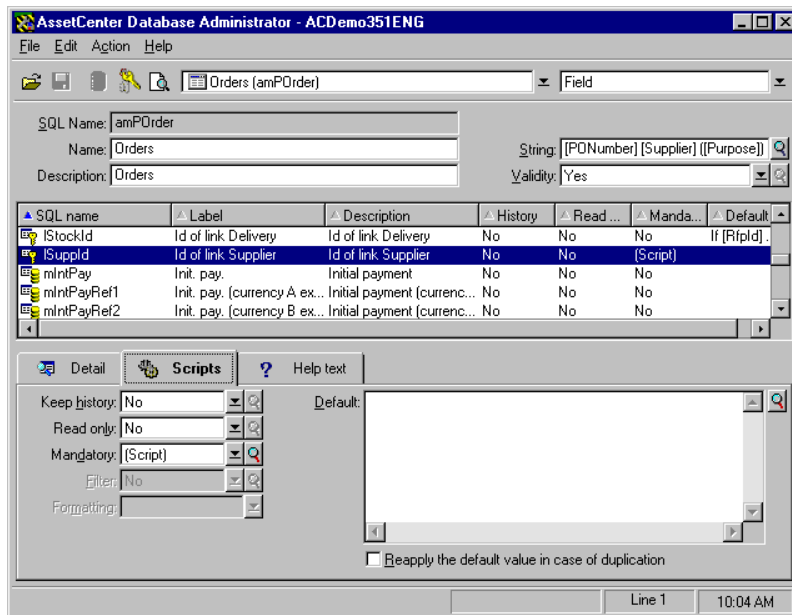


Fig. 2.4 ISuppld field—Scripts tab

6. Click the magnifying glass next to the Mandatory field.
7. In the Basic Script window displayed, type the following in the Function text field:

```
RetVal = [Status]<>"In Preparation"
```

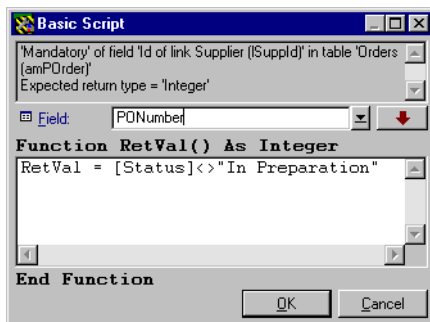


Fig. 2.5 Basic Script window

8. Click **OK**. This closes the Basic Script window.
9. Save the database (**File>Save**).
10. If your Get.It! server is running, go to the Admin Module Control Panel and reset the server.

Personalization Access Rights

The Personalization feature in the Get.It! weblications enables you to tailor the weblication screens without manually changing and compiling code. Instructions for using Personalization are included in the *Get.It! Tailoring Guide*. However, there are several administrative tasks that can allow or deny access to Personalization.

Personalization Hierarchy

Configuration of Personalization is defined in a hierarchy of user access.

Typically, an administrator would use personalization to configure the contents and features of various screens, making the settings standard and default for all users of the system. In many organizations, this is the only type of personalization that may be enabled.

It is also possible to establish specific settings that apply to different groups of users in the organization. For instance, users in the IT department may see a Product screen that has more details than users in other departments.

Finally, it is also possible to allow individual users to further customize their screens in addition to any default customization done at the global and departmental levels.

This personalization hierarchy is defined by an ECMAScript invoked for each user when they log into the system. The job of the script is to return a string showing the hierarchy for each user. For instance, consider a user named Hartke who works in the IT department. The user's hierarchy could be defined as follows:

```
/admin/IT/hartke
```

This tells the Personalization system to provide user Hartke with all personalization settings defined at the admin level, plus any settings defined at the IT level. Finally, any personal settings are stored at the "hartke" user level.

The default implementation is defined in the **personalize.getHierarchy** script. This script defines two levels of Personalization:

- **admin**—this level is given to any user who has *getit.admin* or *getit.personalization.admin* rights.
- **/admin/<user>**—this level is given to all other users. Any settings defined at the /admin level are inherited, and any new changes are stored separately for each user.

This default scheme can be changed by tailoring the **personalize.getHierarchy** script as necessary.

User Personalization Rights

The personalization features rely on specific user rights in order to be enabled. These roles are:

getit.personalization—If defined, the user is classified as a Personalization End User. This is the default setting for all users. These users have access rights as defined by the **personalizeaccess** setting, which can be set to one of the following:

- Disabled – The end user operates with personalization settings defined by the administrator but has no rights to make changes.
- Enabled – The end user inherits default personalization settings, but also has access to add or remove fields to their screens.
- Limited – The end user inherits default personalization settings and has access to make changes. However, the changes are limited by the settings defined by the administrator—schema fields not included by the administrator are not available to the end user. In addition, the user is not allowed to make read-only fields editable.

getit.personalization.admin – If defined, the user is classified as an administrator and is given special rights, including:

- The ability to define whether DocExplorer supports update, create, and delete operations.
- The ability to define settings that are inherited by all users.

Personalization Database

Personalization settings are stored in a database repository and processed at run time when a user logs on to the system. The storage is performed through an adapter target named **weblication**.

To enable personalization, the weblication name must be mapped to a valid adapter in the **archway.ini** adapters section.

For example:

```
adapters=weblication=ACAdapter
```

This maps Personalization storage to the ACAdapter, causing the data to be stored in the AssetCenter database.

The storage of settings is performed using the **webDocExplorer** schema defined in the Get.It! Studio **studio** package. Currently, there are mappings of this schema for AssetCenter and ServiceCenter.

In AssetCenter, the schema is mapped to the amOption table, with an OptSection key of the form “personalize”. If you need to move personalization settings from one database to another, you can export these records from the amOption table and import them into the second system.

In ServiceCenter, the schema is mapped to the giComponentUsers table, with a corkboard.name key of the form “personalize”. If you need to move personalization settings from one database to another, unload these records and load them into the second system.

Chapter 3

Data Reference

AssetCenter Catalog

Get.Resources! uses the AssetCenter product catalog contained within the amProduct table. There are two areas in the catalog which require special configuration:

- Certification field
- Calculated field

Certification

Get.Resources! uses the Certification field to determine the availability of items in the AssetCenter catalog.



Fig. 3.1 Standard catalog options in Get.Resources!

These buttons from the Get.It! menu each drive a database call against the amProduct table. The queries executed are similar to the following for Desktop Computers:

```
SELECT lProdId,Brand,Model,mPrice FROM amProduct WHERE  
(Certification LIKE 'Desktop%')
```

With the exception of the *Bundle* certification, all may be easily changed to meet your business requirements.

The *Bundle* certification is special within the webapplication. *Bundles* are groups of items tied together for a specific purpose. For example, a “Sales Laptop” Bundle may consist of a laptop, PCMCIA NIC, Operating System software, and some applications. This relationship is built within the amProdCompo table, tying together several records from the amProduct table. There is code in place within the **procure.js** script to give special treatment to Bundles.

Calculated Field: cf_Description

A calculated field is used as a descriptive name for records within the catalog. As identified in the Product schema, the field, Description, maps to the field cf_Description, a calculated field.

Figure 3-3.2 on page 3-3 shows a sample of how AssetCenter’s calculated fields may be used within Get.Resources! to ease data presentation. Refer to your AssetCenter documentation for information about calculated fields.

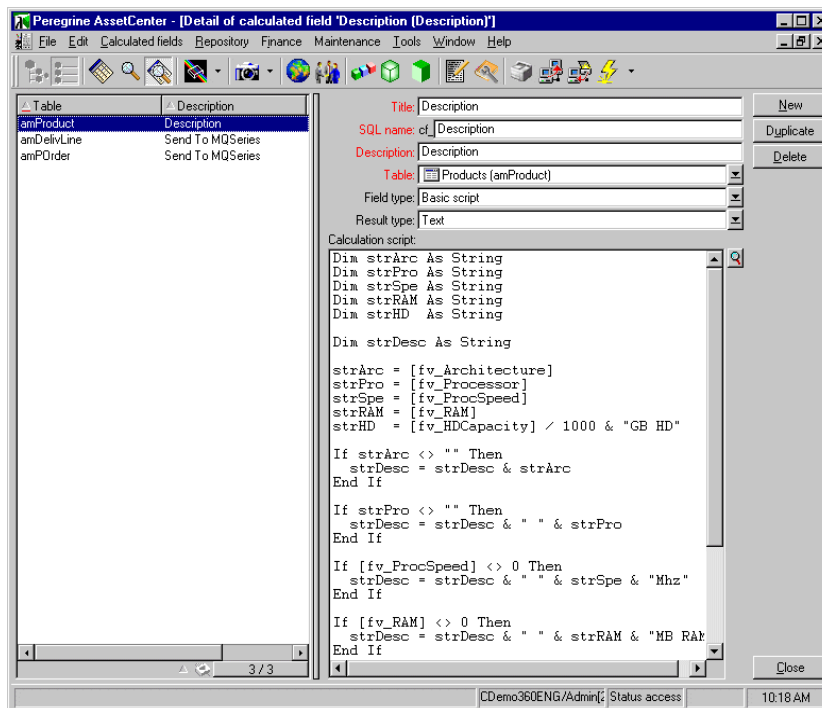


Fig. 3.2 Calculated fields example

AssetCenter Workflows

AssetCenter provides workflows to help you automate and formalize your business procedures. The following AssetCenter workflows are available for use with Get.Resources! and are used by the system in this order:

- Bundle Ordering
- Routing Request
- Request Approval
- Automatic PO Generation
- Request Status

Each of these workflows follows a default process established in AssetCenter for Get.Resources! as shipped. The workflows can be modified to suit your business needs.

Note: It is important that you leave the first and last boxes in a workflow unchanged, since these boxes are linked to the workflows that precede and follow each workflow. Changing these boxes will break the links between the workflows and render them unworkable. Refer to your AssetCenter documentation for information on creating and modifying workflows.

Bundle Ordering Workflow

When a request is submitted, it is checked by the Bundle Ordering workflow to see if a bundle request has been submitted. If true, the workflow then removes the bundle from the purchase order part of the request process. This is done so that the individual items that make up the bundle will be processed through to the purchase order, rather than the bundle itself.



Fig. 3.3 Bundle Ordering Workflow

Routing Request Workflow

The Routing Request workflow is activated when the status of a request is set to *submit*. By default, all requests are processed as OnCatalog request, so the Qualify Request box is not used. The last box in this workflow starts the approval cycle.

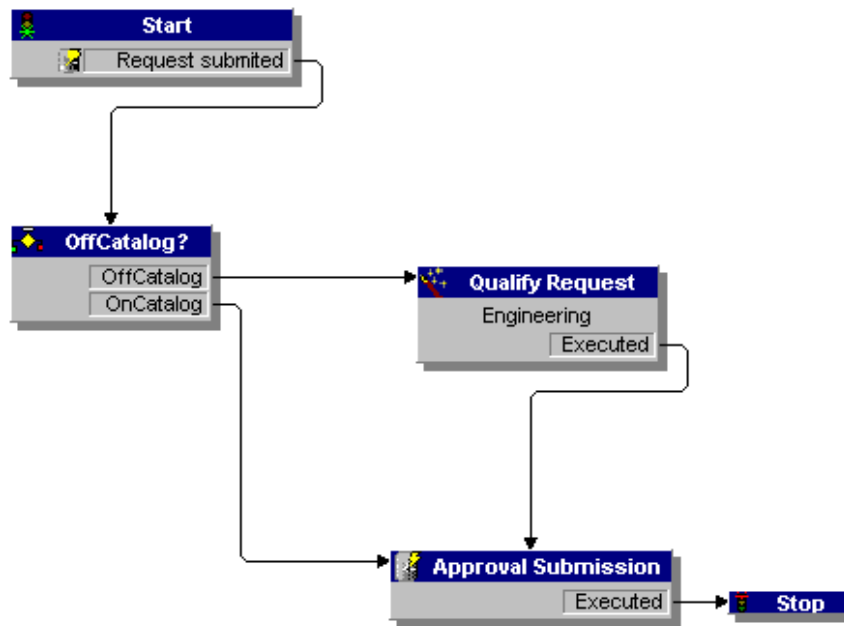


Fig. 3.4 Routing Request Workflow

Request Approval Workflow

The Request Approval workflow shows the approval steps for a request that has been submitted. After the request has gone through the approval process, there are two possible results at the end of this workflow.

- If the request was approved, the approval status and the request status are set to *approved*. The request status determines what is displayed on the screen when the status of a request is viewed.
- If the request was not approved, the approval status and request status are set to *refused*.

If you change this workflow in AssetCenter, make sure you retain the first part of the reference field designation: REQAPPR_GR. Failure to do this will make it impossible for the system to process the workflow.

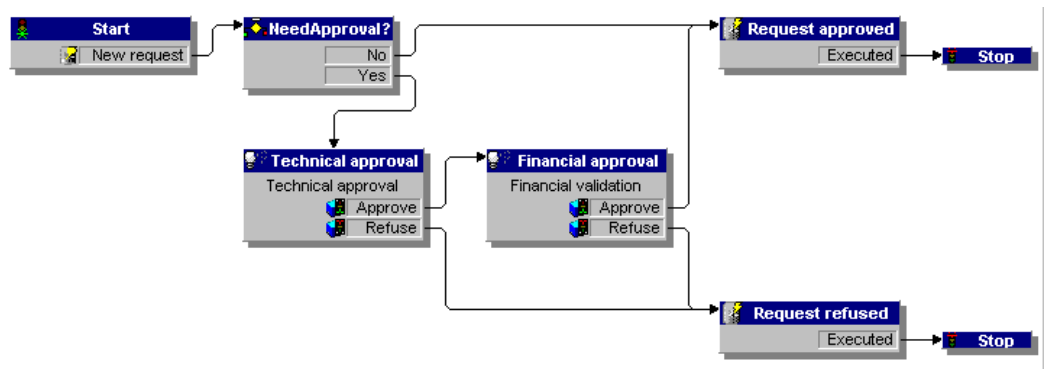


Fig. 3.5 Request Approval Workflow

Automatic PO Generation Workflow

When the approval status of a request is changed to *approved*, the Automatic PO Generation workflow is activated. One purchase order per request is automatically created. You can change the process to create multiple POs by vendor or to have the system bundle several requests into one PO.

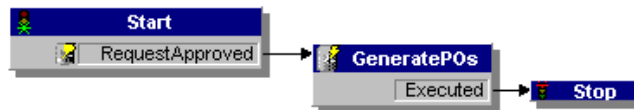


Fig. 3.6 Automatic PO Generation Workflow

Request Status Workflow

After ordered items have been received, the Request Status workflow changes the status of the request to *received*. When the status of a request is viewed, the status is now shown as *received*.

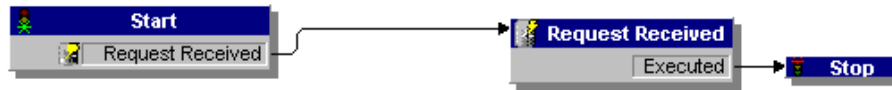


Fig. 3.7 Request Status Workflow



Part 2 Get.Resources! with ServiceCenter

Chapter 4

Configuring Get.Resources! with ServiceCenter

Get.Resources! can be configured to use Peregrine's ServiceCenter as your back-end system. Get.Resources! accesses the catalog in the Request Management module of ServiceCenter. Requests opened in Get.Resources! are stored as quotes in the ServiceCenter database.

Overview of Get.Resources! Architecture

Refer to the *Get.It! Quick Start Guide* for detailed information about the Get.It! architecture.

Get.Resources! connects to the ServiceCenter database through a special adapter, the SCAdapter. As requests are opened and processed through Get.Resources!, data is stored and updated in the ServiceCenter database.

Users are authenticated using ServiceCenter Operator records, with special capability words provided for use with Get.Resources!. For more information about user authentication, see Chapter 2, "Administration and Security." The *Get.It! Quick Start Guide* also contains general information about user authentication.

Configuring Get.Resources!

There are three steps to configuring Get.Resources! to use ServiceCenter:

- Define ServiceCenter as the back-end system to be used with Get.Resources!.
- Load the ServiceCenter files into Get.It!. This is necessary only if you are using a version of ServiceCenter prior to ServiceCenter 4.0.
- Define user rights with capability words in the user Operator records. User rights can also be set globally on the Admin Settings form in Get.It!.

Defining ServiceCenter as the Back-end System

1. Log in to Get.It! as an administrator.
2. On the Admin Module Settings form, click the Get-Resources tab.
3. In the Request Backend field, type `sc`, and then click Save.

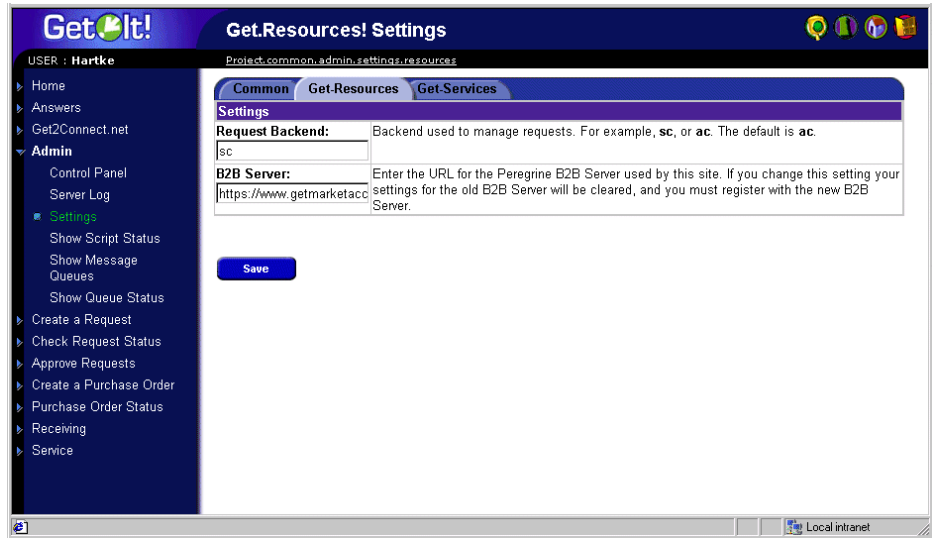


Fig. 4.1 Admin Settings—Get-Resources tab

4. On the Control Panel form, click **Reset Server**.

Loading the ServiceCenter files—ServiceCenter 3.x

Important: This procedure is for versions of ServiceCenter prior to version 4.0 only. If you are using ServiceCenter 4.0 or later with Get.Resources!, these files are not needed.

The following procedure includes the process for loading the `scgr.unl` file. Repeat steps 4 through 6 to load the `scgrdata.unl` file.

1. Verify that the ServiceCenter server is running.
2. Open a full (system) client and log in to the system as an administrator.

3. From the Toolkit tab, select **Database Manager**.
4. From the **Options** menu, select **Import/Load**.
5. Enter the full file name and path where you installed the Get.Resources! files. The default location is:

C:\getit\packages\GetResources 2.01\resources\config\sc3\scgr.unl

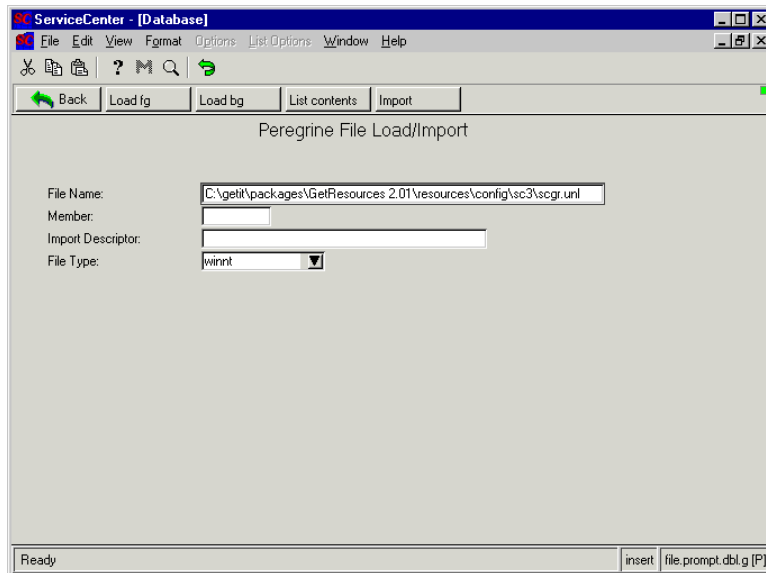


Fig. 4.2 Loading the files into ServiceCenter

6. Click **Load fg**.
You are returned to the Database Manager form. The following message is displayed at the bottom of the form:
C:\getit\packages\GetResources 2.01\resources\config\sc3\scgr.unl loaded.
7. Repeat steps 4-6 to load the **scgrdata.unl** file.
8. Click **Back** to return to the home menu.
9. Do one of the following:
 - If you are able to stop and start the ServiceCenter service without impacting other users, log out of the client and then stop and start the ServiceCenter service.
 - If you are using one of multiple clients pointing to one ServiceCenter server and cannot stop and start the ServiceCenter service, do the following:
 - a. Click **Command**, to open a command prompt.

- b. Type `*aapm.global.initer`, and then press ENTER.
- c. Type `*aapm.server.initer`, and then press ENTER.
- d. Click **Back**, and log out of the client.

Defining User Rights

You can define user rights so that all employees have the same rights, or limit certain user rights to specific employees.

Individual user rights are defined in the user's Operator record in ServiceCenter; global user rights are defined on the Get.It! Admin module Settings page.

To define user rights in ServiceCenter:

1. Log in to ServiceCenter as an administrator and establish the following capability words as needed in the Operator records for the users you want to have access to the various tasks in Get.Resources!
 - Requester—`getit.requester`, OCML, OCMQ. These capability words give access to line item and quote functionality in ServiceCenter Request Management, as well as requester capability in Get.Resources!.
 - Advanced Requester—all of the Requester capability words, plus `getit.advancedrequester`. The Advanced Requester can split line items on a request and assign them to different employees for processing.
 - Approver—`getit.approver`. The Approver is able to view and approve requests and create orders.

You will also need to add the Approver to an approval group using the Request Management module in ServiceCenter. This can be done by accessing an existing user's profile with the approver rights you want to use, changing the name of the user, and adding the new user. Refer to your ServiceCenter documentation for instructions.

2. Save the Operator records, and log out of ServiceCenter.

To define user rights for a group:

1. Open the Admin module in Get.It!.
2. On the Settings page, add the rights you want to use to the Capabilities field. For example, to give all users the ability to open requests, add `getit.requester` to `sc`, as shown in Figure 4.3.

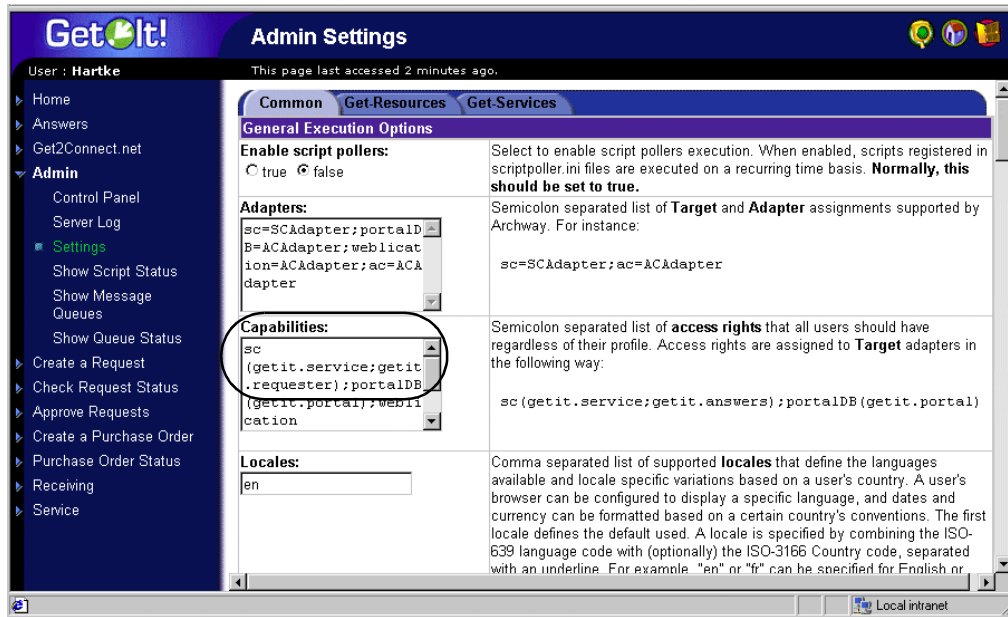


Fig. 4.3 Admin Settings—Capabilities field

3. Scroll to the bottom of the form and click **Save**.
4. When the Control Panel is displayed, click **Reset Server**.

File Attachments

Get.Resources! provides functionality that allows users to attach files (for example, a Microsoft Excel or Word file) to Get.Resources! requests to provide additional information.

During the installation of Get.It!, an attachments virtual directory was created in the presentation directory. However, as a security safeguard to your system, it is recommended that you change the permissions on the attachments virtual directory.

The procedure below uses Internet Information Services (IIS) as an example.

1. Open IIS.
2. In the getit directory, right-click on the attachments directory, and then choose **Properties**.

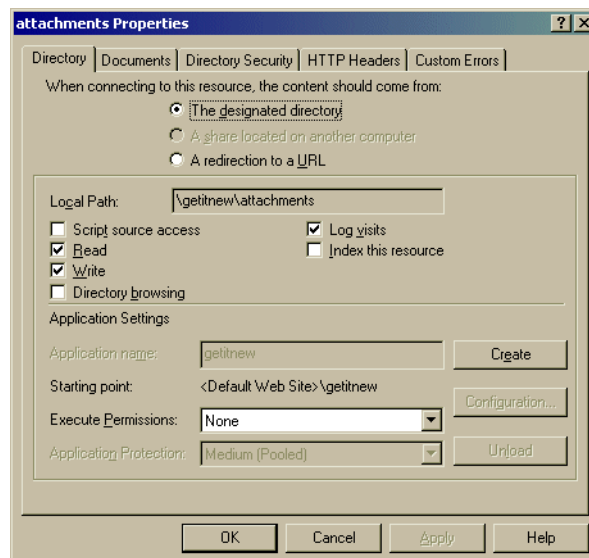


Fig. 4.4 Attachments Properties

3. Set the permissions to *Read* and *Write*.
4. Click **OK**.

Archway.ini Parameters

The File Attachment functionality uses defaults that can be overridden by the following parameters in the **archway.ini** file. These parameters allow you to set the paths for the file attachments and to establish a maximum allowable size for the attachment.

Parameter	Description
attachvirpath	Sets the Web directory reference from which files will be drawn for viewing. The default is /getit/attachments/ if not set in the archway.ini file.
attachserverpath	Sets the physical path on the Web server where the files will be stored. The default is <getit home>\presentation\attachments\ if not set in the archway.ini file.
attachmaxsize	Establishes the maximum size in KB for an attached file. The default is 2048 KB (2 MB).

Logging in to Get.Resources!

There are two sample user Operator records included in the ServiceCenter database for use with Get.Resources!: Richard Hartke and Michaela Tossi. Each of these sample users has capability words defined to give access to Get.It!.

You will need to modify the capability words or give global access to getit.requester to enable access to Get.Resources!. Instructions are included on page 4-4.

To log in to Get.Resources!:

The following procedure uses the Hartke login as an example.

1. Open a Web browser and type localhost\getit\login.jsp.
The login screen is displayed.

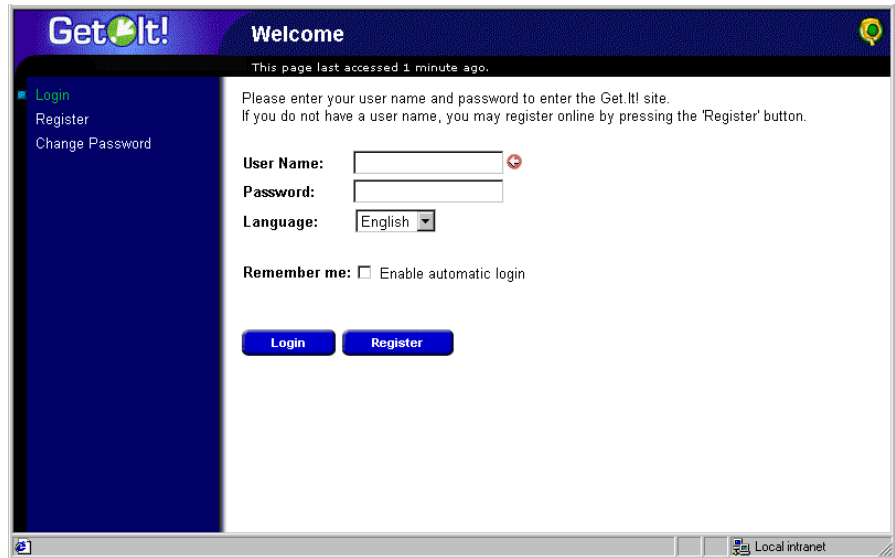


Fig. 4.5 Get.Resources! login screen

2. Type Hartke in the User Name field. The arrow to the right of the field indicates that it is a required field. There is no password required for either of the sample logins.
3. Click **Login**, or press ENTER.
The Get.It! portal is displayed.

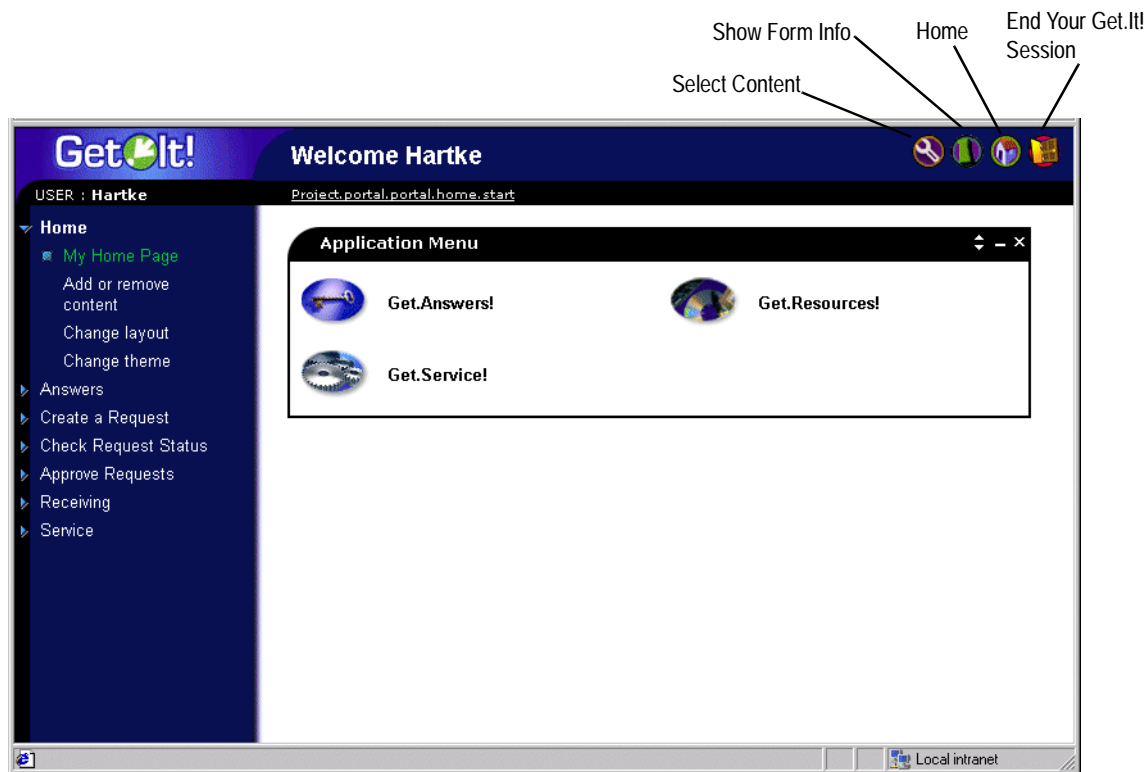


Fig. 4.6 Get.It! portal

The menu on the sidebar includes all of the activities available to the logged-in user. As you navigate through Get.Resources!, the sidebar menu remains visible. To return to the portal, click the **Home** icon at any time.

Get.Resources! Features

Get.Resources! provides a comprehensive interface for processing a request for a product through an approval process. From the time a request is opened, the status of the request can be followed through workflows that are dynamically updated as the request is moved through the approval process.

Get.Resources! is role-based, providing sample users to illustrate this process. Two user logins are provided:

- Hartke, with requesting and approving privileges.

- Tossi, an end-user with requesting capabilities only.

When a user starts the process of opening a request, a shopping cart is provided in which totals are dynamically updated as new items are added. Catalog items are displayed according to the category and subcategory chosen by the user as defined in the ServiceCenter catalog.

Figure 4.7 shows a typical catalog screen in Get.Resources! using the Request Management catalog. The sidebar displays all of the catalog options available, as well as the shopping cart.



Fig. 4.7 Catalog entries for the Furniture subcategory

You can bookmark a Get.Resources! page as you would any other Web page. With the page you want to bookmark displayed, right-click in the frame you want to return to and select **Add to Favorites**.

You can also send a coworker an HTTP link to a page—for example, someone processing a request might want someone else to look at the request for approval. The link would be something like:

`http://webserver/getit/e_service_status_whichone.jsp?Id=REQ000040`

If the user is logged in to Get.Resources!, clicking on the link will take them to the designated page. If the user is not logged in, the login screen will be displayed, and after the user has logged in the requested page will be displayed.

Selecting the Contents of your Portal

The portal can be modified using the Select Content feature.

To use Select Content:

1. With the portal displayed, click the **Select Content** icon (wrench).
The Select Content page is displayed.

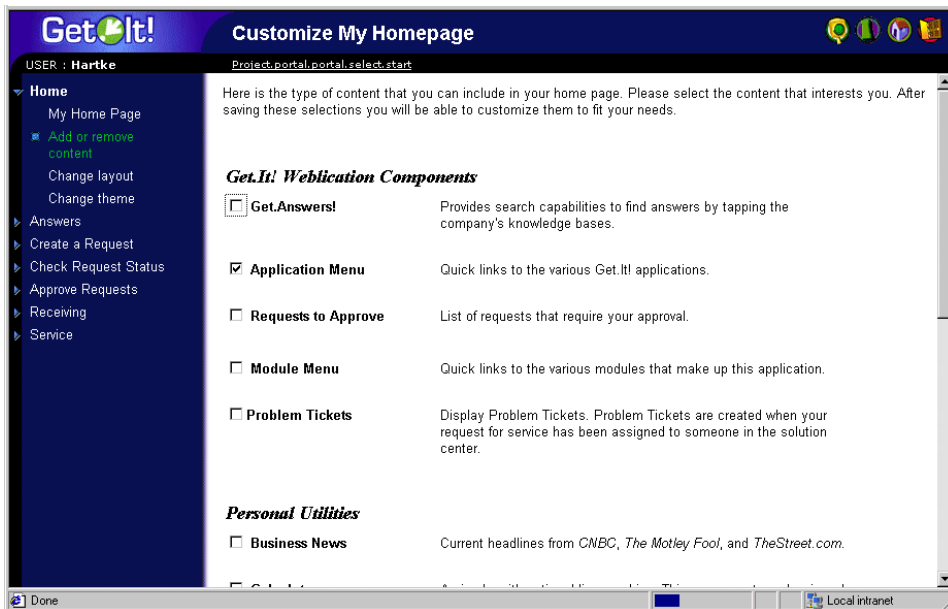


Fig. 4.8 Select Content

By default, the Application Menu is selected.

2. Click the check box for the items you want to be displayed on your portal.
3. Scroll to the bottom of the page, and then click **Save**.

You are returned to the portal and the items you have selected are displayed.

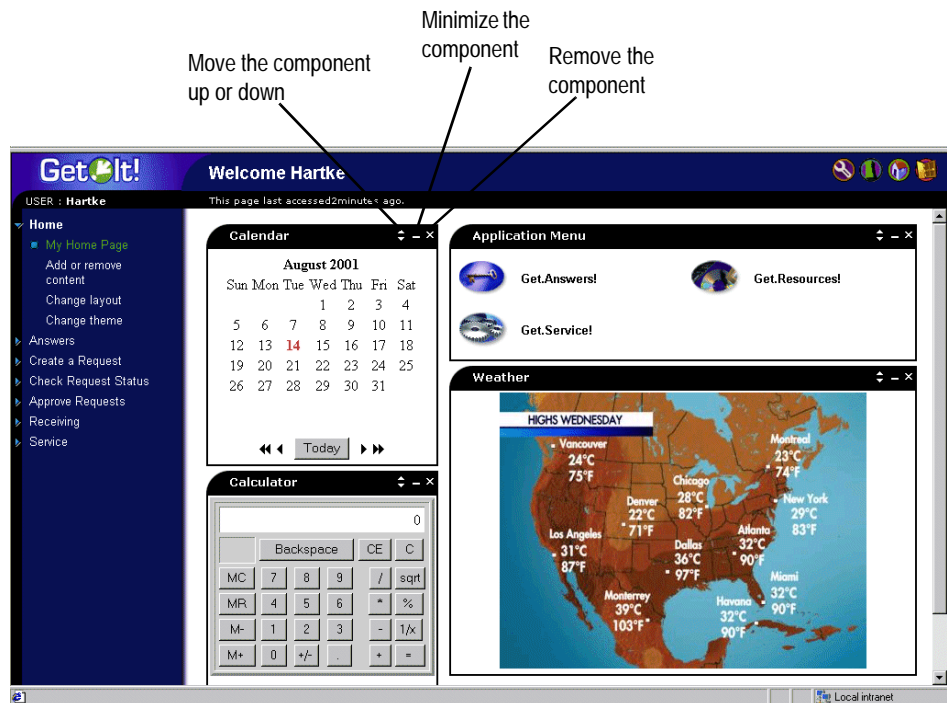


Fig. 4.9 Portal with Calendar, Calculator, and Weather added

Each of the portal components can be moved, minimized, or removed from the portal.

- Click the **triangle** button to move the component up or down on the page.
- Click the **minus** button to minimize the component.
- Click the **x** button to remove the component.

In Figure 4.10 on page 3-13, the Calculator component has been moved up and minimized.

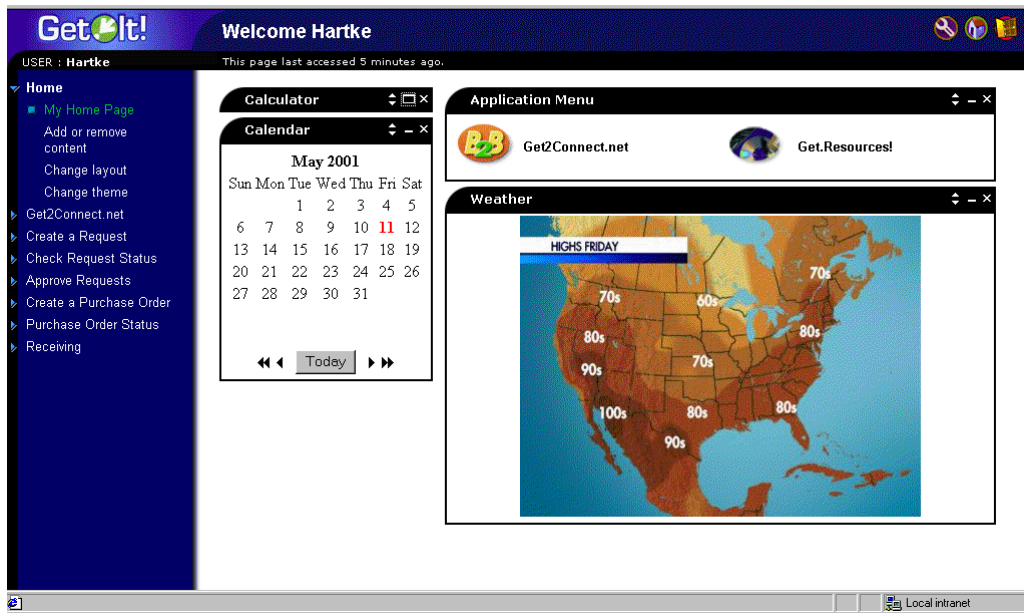


Fig. 4.10 Calculator component moved and minimized

Chapter 5

Administration and Security

This chapter includes information about administrative tasks and security issues specific to Get.Resources! when used with ServiceCenter. It is recommended that you refer to the *Get.It! Quick Start Guide* for information about general Get.It! administrative tasks in conjunction with reading this guide.

Get.Resources! Security

To log in to Get.It!, a user must have an Operator record established in the ServiceCenter database. This can be a record that was created in ServiceCenter prior to login, or the user can self-register in Get.It!. Operator and Contact records are automatically created at the time of registration and a password is e-mailed to the new user.

To access Get.Resources!, specific Get.It! capability words must be included in the user's ServiceCenter Operator record.

User Registration and Authentication

When a user registers in Get.Resources! for the first time, Operator and Contact records are created in ServiceCenter. Default capability words are established in the Operator record to give the user login access to Get.It!.

Each time the user attempts to log in to Get.Resources!, the user name and password are validated against ServiceCenter. If the entered user ID and password are not valid, the user will be prompted to enter valid data. If the login information is correct, Get.It! will retrieve the access rights for the user and log the user in to Get.It!.

User Rights

ServiceCenter utilizes capability words defined in the user's Operator record to identify which tables a user can modify within the system. Get.Resources! examines the capability words of the logged-on user and provides access to the appropriate portions of Get.Resources!.

Setting User Rights with ServiceCenter

Get.Resources! uses an administration profile to connect to ServiceCenter through the ServiceCenter adapter. The parameters **sadmin** and **sadminpassword** can be set by using the Admin module Settings page. This allows Get.Resources! to connect to ServiceCenter. User rights that have been defined with capability words in ServiceCenter are then used to determine which tables in ServiceCenter a logged-on user will be able to access and, therefore, which parts of Get.Resources! the user will see.

Get.Resources! users must have a profile that has one or more of the following user rights keywords as appropriate:

Capability words	Access rights
getit.requester	Enables access to create requests in Get.Resources!
getit.advancedrequester	Enables access to advanced request tasks, for example, assigning line items to multiple users to process.
getit.approver	Enables access to approve requests.
getit.admin	Provides access to the Admin module in Get.It!.

Get.Resources! Administration

This section includes administrative tasks specific to Get.Resources!.

Parameters

Get.Resources! has one parameter that is specific to the weblication:

Refer to the *Get.It! Quick Start Guide* for a complete list of parameters.

- Request Backend—establishes the back-end system being used with Get.Resources!. The default is *ac* (AssetCenter). This must be changed to *sc* to use ServiceCenter as the back-end system.

This parameter can be defined on the Get-Resources tab of the Settings page in the Get.It! Admin module. For information about using the Admin module, refer to Chapter 4 of the *Get.It! Quick Start Guide*.

Enabling Approval Rights

Access to the approval process is granted by adding `getit.approver` to a user's Operator record capability words. The user must also be added to an approval group in the Request Management module in ServiceCenter. Approvals are controlled in ServiceCenter using the ServiceCenter approval process. Information about this process can be found in your ServiceCenter documentation.

Personalization Access Rights

The Personalization feature in the Get.It! weblications enables you to tailor the weblication screens without manually changing and compiling code. Instructions for using Personalization are included in the *Get.It! Tailoring Guide*. However, there are several administrative tasks that can allow or deny access to Personalization.

Personalization Hierarchy

Configuration of Personalization is defined in a hierarchy of user access.

Typically, an administrator would use personalization to configure the contents and features of various screens, making the settings standard and default for all users of the system. In many organizations, this is the only type of personalization that may be enabled.

It is also possible to establish specific settings that apply to different groups of users in the organization. For instance, users in the IT department may see a Product screen that has more details than users in other departments.

Finally, it is also possible to allow individual users to further customize their screens in addition to any default customization done at the global and departmental levels.

This personalization hierarchy is defined by an ECMAScript invoked for each user when they log into the system. The job of the script is to return a string showing the hierarchy for each user. For instance, consider a user named Hartke who works in the IT department. The user's hierarchy could be defined as follows:

```
/admin/IT/hartke
```

This tells the Personalization system to provide user Hartke with all personalization settings defined at the admin level, plus any settings defined at the IT level. Finally, any personal settings are stored at the "hartke" user level.

The default implementation is defined in the **personalize.getHierarchy** script. This script defines two levels of Personalization:

- **admin**—this level is given to any user who has *getit.admin* or *getit.personalization.admin* rights.
- **/admin/<user>**—this level is given to all other users. Any settings defined at the /admin level are inherited, and any new changes are stored separately for each user.

This default scheme can be changed by tailoring the **personalize.getHierarchy** script as necessary.

User Personalization Rights

The personalization features rely on specific user rights in order to be enabled. These roles are:

getit.personalization—If defined, the user is classified as a Personalization End User. This is the default setting for all users. These users have access rights as defined by the **personalizeaccess** setting, which can be set to one of the following:

- **Disabled** – The end user operates with personalization settings defined by the administrator but has no rights to make changes.
- **Enabled** – The end user inherits default personalization settings, but also has access to add or remove fields to their screens.
- **Limited** – The end user inherits default personalization settings and has access to make changes. However, the changes are limited by the settings defined by the administrator—schema fields not included by the administrator are not available to the end user. In addition, the user is not allowed to make read-only fields editable.

getit.personalization.admin – If defined, the user is classified as an administrator and is given special rights, including:

- The ability to define whether DocExplorer supports update, create, and delete operations.

- The ability to define settings that are inherited by all users.

Personalization Database

Personalization settings are stored in a database repository and processed at run time when a user logs on to the system. The storage is performed through an adapter target named **weblication**.

To enable personalization, the weblication name must be mapped to a valid adapter in the **archway.ini** adapters section.

For example:

```
adapters=weblication=ACAdapter
```

This maps Personalization storage to the ACAdapter, causing the data to be stored in the AssetCenter database.

The storage of settings is performed using the **webDocExplorer** schema defined in the Get.It! Studio **studio** package. Currently, there are mappings of this schema for AssetCenter and ServiceCenter.

In AssetCenter, the schema is mapped to the amOption table, with an OptSection key of the form “personalize”. If you need to move personalization settings from one database to another, you can export these records from the amOption table and import them into the second system.

In ServiceCenter, the schema is mapped to the giComponentUsers table, with a corkboard.name key of the form “personalize”. If you need to move personalization settings from one database to another, unload these records and load them into the second system.



Index

A

- access rights, Personalization 2-8, 5-3
- administration, ServiceCenter 5-2
- AssetCenter
 - calculated field 3-2
 - catalog 3-1
 - Certification field 3-1
 - sample users 4-10
 - server 1-7
 - workflows 3-4
- authentication, user 2-1, 5-1

B

- back-end system, ServiceCenter 4-2
- bookmarking a page 1-13, 4-11
- Buyer profile 1-10

C

- calculated field 3-2
- capability words 4-5, 5-2
- catalog, AssetCenter 3-1
- Certification field in AssetCenter 3-1
- configuring Get.Resources! with ServiceCenter 4-1

F

- file attachment
 - archway.ini parameters 1-8, 4-8
 - virtual directory 1-7, 4-7

G

- getit.admin 2-4

H

- Hartke profile 1-9

L

- logging on 1-9, 4-8

M

- Mandatory attribute, modifying 2-5

O

- Operator records in ServiceCenter 4-5

P

- Personalization
 - access rights 2-8, 5-3
 - database 2-9, 5-5
 - hierarchy 2-8, 5-3
 - user rights 2-9, 5-4
- portal
 - Select Content 1-13, 4-12
- profiles
 - Buyer 1-10
 - Hartke 1-9
 - Tossi 1-9
- profiles, sample 2-3
- purchase order Mandatory attribute 2-5

R

- registration, ServiceCenter 5-1
- registration, user 2-1
- requestbackend parameter with ServiceCenter 5-3

S

- sample profiles 2-3
- sample users, AssetCenter 1-11, 4-10
- security
 - sample user profiles 2-3
 - user registration and authentication 2-1
 - user rights
 - AssetCenter 2-2
 - user rights, ServiceCenter 5-2
- security, ServiceCenter
 - user registration and authentication 5-1
- Select Content, using 1-13, 4-12
- server
 - AssetCenter 1-7
- ServiceCenter
 - administration 5-2
 - defining as back-end system 4-2, 5-3
 - defining user rights 4-5
 - loading files 4-2
 - security 5-1
- ServiceCenter and Get.Resources! 4-1

T

- Tossi profile 1-9

U

- user registration and authentication 2-1
- user registration and authentication, ServiceCenter 5-1
- user rights
 - Personalization 2-9, 5-4
 - ServiceCenter 4-5, 5-2
- user rights, AssetCenter 2-2

V

- virtual directory, attachments 1-7, 4-7

W

- workflow schemes, modifying 1-7
- workflows
 - Automatic PO Generation 3-6
 - Bundle Ordering 3-4
 - Request Approval 3-5
 - Request Status 3-7
 - Routing Request 3-4



September 28, 2001