



MERCURY QUICKTEST PROFESSIONAL™

Add-in for SAP® Solutions

VERSION 8.2

Guide

MERCURY™

QuickTest Professional Add-in for SAP Solutions

Guide

Version 8.2

QuickTest Professional Add-in for SAP Solutions Guide, Version 8.2

This manual, and the accompanying software and other documentation, is protected by U.S. and international copyright laws, and may be used only in accordance with the accompanying license agreement. Features of the software, and of other products and services of Mercury Interactive Corporation, may be covered by one or more of the following patents: United States: 5,511,185; 5,657,438; 5,701,139; 5,870,559; 5,958,008; 5,974,572; 6,137,782; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332; 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912; 6,694,288; 6,738,813; 6,738,933; 6,754,701; 6,792,460 and 6,810,494. Australia: 763468 and 762554. Other patents pending. All rights reserved.

Mercury, Mercury Interactive, the Mercury logo, the Mercury Interactive logo, LoadRunner, WinRunner, SiteScope and TestDirector are trademarks of Mercury Interactive Corporation and may be registered in certain jurisdictions. The absence of a trademark from this list does not constitute a waiver of Mercury's intellectual property rights concerning that trademark.

All other company, brand and product names may be trademarks or registered trademarks of their respective holders. Mercury disclaims any responsibility for specifying which marks are owned by which companies or which organizations.

Mercury Interactive Corporation
379 North Whisman Road
Mountain View, CA 94043
Tel: (650) 603-5200
Toll Free: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (650) 603-5300

© 2004 Mercury Interactive Corporation, All rights reserved

If you have any comments or suggestions regarding this document, please send them via e-mail to documentation@mercury.com.

Table of Contents

| | |
|--------------------------------|------|
| Welcome | vii |
| Using This Guide | viii |
| Typographical Conventions..... | xi |

PART I: SETTING UP ADD-IN SUPPORT FOR SAP SOLUTIONS

| | |
|---|-----------|
| Chapter 1: Installing the Add-in for SAP Solutions | 3 |
| Before You Install | 4 |
| Running the Setup Program | 4 |
| Chapter 2: Activating Add-in Support for SAP Solutions | 17 |
| Licensing the Add-in for SAP Solutions | 17 |
| Loading QuickTest with Add-in Support for SAP Solutions | 19 |
| Loading QuickTest without Add-in Support for SAP Solutions | 21 |
| Working with Function Library Files | 21 |
| Uninstalling and Repairing the Add-in for SAP Solutions..... | 28 |

PART II: TESTING SAP GUI FOR WINDOWS APPLICATIONS

| | |
|---|-----------|
| Chapter 3: Setting Up Your SAP Gui for Windows Environment..... | 35 |
| About Setting Up Your SAP Windows Environment | 36 |
| Installing SAP Gui Scripting Support | 37 |
| Checking Package and Patch Versions Installed on the SAP Application Server | 38 |
| Checking the Patch Version Installed on your SAP Gui for Windows Application | 43 |
| Enabling Scripting on the SAP Application (Server-Side) | 44 |
| Enabling Scripting on the SAP Application (Client-Side) | 49 |
| Setting F4 Help to Use Dialog Display Mode | 52 |
| Setting F1 Help to Use Modal Dialog Box Mode..... | 54 |
| Checking the Connection Speed on the SAP Server | 55 |

| | |
|--|------------|
| Chapter 4: Recording and Running Tests and Components on SAP Gui for Windows Applications | 57 |
| About Recording and Running Tests and Components on SAP Gui for Windows Applications | 58 |
| Before You Create SAP Gui for Windows Tests and Components | 60 |
| Understanding the SAP Windows Test Object Model | 60 |
| Recording Tests and Components on SAP Gui for Windows Applications..... | 64 |
| Configuring Active Screen Capture Options for SAP Gui for Windows Applications..... | 73 |
| Configuring Testing Options for SAP Gui for Windows Applications..... | 76 |
| Understanding Low-Level or Analog Mode Recording on SAP Gui for Windows..... | 88 |
| Using Standard Windows Recording Capabilities | 88 |
| Running Tests and Components on SAP Gui for Windows Applications..... | 89 |
| Chapter 5: Enhancing Your SAP Windows Test..... | 91 |
| About Enhancing Your SAP Windows Test..... | 91 |
| Checking SAP Windows Objects and Outputting Values..... | 93 |
| Checking SAP Windows Tables and Grids..... | 94 |
| Outputting SAP Windows Property and Table Cell Values | 103 |
| Chapter 6: Adding SAP Windows Statements to Your Test or Component | 113 |
| About Adding SAP Windows Statements to Your Test or Component..... | 113 |
| Working with SAP Windows Test Objects | 114 |
| Working with Test Object Methods and Properties..... | 128 |
| Accessing Native Methods and Properties in Your SAP Gui for Windows Application | 129 |

PART III: TESTING WEB-BASED SAP APPLICATIONS

| | |
|---|------------|
| Chapter 7: Recording and Running Tests or Components on Web-based SAP Applications | 133 |
| About Recording and Running Tests or Components on Web-based SAP Applications..... | 134 |
| Understanding the SAP Web Test Object Model | 135 |
| Recording Tests on Web-based SAP Applications | 137 |
| Running Tests on Web-based SAP Applications | 144 |

| | |
|--|------------|
| Chapter 8: Enhancing Your SAP Web Test | 145 |
| About Enhancing Your SAP Web Test..... | 145 |
| Checking SAP Web Objects and Outputting Values..... | 147 |
| Chapter 9: Adding SAP Web Statements to Your Test or Component | 153 |
| About Adding SAP Web Statements to Your Test or Component | 154 |
| Working with SAP Web Test Objects | 155 |
| Working with Test Object Methods and Properties..... | 163 |
| Accessing Native Methods and Properties in Your Web-based SAP Applications..... | 164 |

PART IV: INTEGRATING WITH ECATT

| | |
|--|------------|
| Chapter 10: Understanding QuickTest-eCATT Integration | 169 |
| About Managing Tests in eCATT..... | 169 |
| Understanding eCATT Testing Modes | 170 |
| Chapter 11: Configuring eCATT to Work with QuickTest | 173 |
| About Configuring eCATT to Work with QuickTest | 173 |
| Setting External Tool Parameters in the ECCUST_ET Table | 174 |
| Applying Necessary Roles or Profiles to eCATT-QuickTest Users | 175 |
| Chapter 12: Working with eCATT in Standalone Mode..... | 177 |
| About Working with eCATT in Standalone Mode..... | 178 |
| Connecting to and Disconnecting from eCATT | 179 |
| Saving Tests to eCATT in Standalone Mode | 182 |
| Opening Tests from an eCATT Database in Standalone Mode..... | 184 |
| Uploading Files to eCATT and Downloading Files from eCATT | 186 |
| Setting Options for Working with eCATT | 191 |
| Passing Values Between eCATT Test Scripts and QuickTest Tests | 195 |
| Running a Test Stored in an eCATT Database in Standalone Mode..... | 196 |
| Chapter 13: Working with eCATT in Integrated Mode | 199 |
| About Working with eCATT in Integrated Mode | 199 |
| Performing Basic Test Management Operations from eCATT | 200 |
| Transferring Data To and From QuickTest Tests Using Test Parameters | 208 |
| Running a Test Stored in an eCATT Database in Integrated Mode..... | 212 |
| Index | 217 |

Table of Contents

Welcome

Welcome to the QuickTest Professional Add-in for SAP Solutions.

QuickTest Professional Add-in for SAP Solutions has been certified by SAP AG.

When the QuickTest Professional Add-in for SAP Solutions is loaded, QuickTest can record and run tests on SAP applications using special test objects and methods that have been customized for SAP.

QuickTest's customized test objects, methods, and properties make scripts simple to read, maintain, enhance, and parameterize, enabling both advanced and novice users to create sophisticated tests on applications developed using SAP solutions, including:

- SAP Gui for Windows
- SAP Gui for HTML
- SAP Enterprise Portal
- SAP Workplace

QuickTest Professional Add-in for SAP Solutions is integrated with the SAP Extended Computer Aided Test Tool (SAP eCATT). This integration into eCATT, via SAP Gui for Windows 6.20, has been certified against the SAP Web AS 6.20. Now, using SAP eCATT and QuickTest Professional, customers can run quality tests in environments that span beyond Windows and SAP environments including complex, multi-platform, highly-integrated composite, legacy, and proprietary enterprise applications.



Note: You cannot use eCATT to manage components or scripted components.

Using This Guide

This guide explains how to install the QuickTest Professional Add-in for SAP Solutions. It also describes everything you need to know to work with QuickTest Professional to successfully test SAP applications using tests, business components, and scripted components. This guide should be used in conjunction with the *QuickTest Professional User's Guide*, the *QuickTest Professional New Features Guide*, and the *QuickTest Professional Object Model Reference*. All of these references can be accessed online by selecting **Help > QuickTest Professional Help** from the QuickTest main window. The guides are also available as printed books.

Note: Business components and scripted components are part of Mercury Business Process Testing, which utilizes a keyword-driven methodology for testing applications. Both types of components are maintainable, reusable modules that perform a specific task when testing your application. Business components are created in a script-free environment that employs a hierarchy-free Keyword View. Scripted components are reusable modular scripts that make use of both the Keyword View and the Expert View, as well as many options that are available for QuickTest tests. For more information, refer to the *QuickTest Professional What's New Guide*.

This information, examples, and screen captures in this guide focus specifically on working with QuickTest tests. However, much of the information also applies to business components. For example, many of the examples that show the Keyword View for tests can also be applied to business components because the Keyword View is similar for tests and business components. In addition, much of the information that is relevant for tests is also relevant for scripted components, which have functionality that is similar to tests.

Information that is relevant only for tests is preceded by  or is specified as such within the text. Note that even if a section is defined as being relevant only for tests, it may also be relevant for scripted components. Information that is relevant only for business components is preceded by  or is specified as such within the text. For more information on working with business components and scripted components, refer to the *QuickTest Professional What's New Guide*.

This guide contains the following parts:

Part I Setting Up Add-in Support for SAP Solutions

Details the process of installing the QuickTest Professional Add-in for SAP Solutions, including:

- Installing the Add-in for SAP Solutions
- Activating Add-in Support for SAP Solutions

Part II Testing SAP Gui for Windows Applications

Explains how to use the QuickTest Professional Add-in for SAP Solutions to test objects in SAP Gui for Windows applications, including:

- Setting Up Your SAP Gui for Windows Environment
- Recording and Running Tests on SAP Gui for Windows Applications
- Enhancing Your SAP Windows Test
- Adding SAP Windows Statements to Your Test

Part III Testing Web-based SAP Applications

Explains how to use the QuickTest Professional Add-in for SAP Solutions to test objects in Web-based SAP applications, including:

- Recording and Running Tests on Web-based SAP Applications
- Enhancing Your SAP Web Test
- Adding SAP Web Statements to Your Test or Component



Part IV Integrating with eCATT

Explains how to store and manage QuickTest tests in the SAP Extended Computer Aided Test Tool (SAP eCATT).

- Understanding QuickTest-eCATT Integration
- Configuring eCATT to Work with QuickTest
- Working with eCATT in Standalone Mode
- Working with eCATT in Integrated Mode

Typographical Conventions

This book uses the following typographical conventions:

| | |
|---|--|
| 1, 2, 3 | Bold numbers indicate steps in a procedure. |
| > | The greater-than sign separates menu levels (for example, File > Open). |
| Stone Sans | The Stone Sans font indicates names of interface elements (for example, the Run button) and other items that require emphasis. |
| Bold | Bold text indicates method or function names. |
| <i>Italics</i> | <i>Italic</i> text indicates method or function arguments and book titles. It is also used when introducing a new term. |
| <> | Angle brackets enclose a part of a file path or URL address that may vary from user to user (for example, < MyProduct installation path >\bin). |
| Arial | The Arial font is used for examples and text that is to be typed literally. |
| Arial bold | The Arial bold font is used in syntax descriptions for text that should be typed literally. |
| SMALL CAPS | The SMALL CAPS font indicates keyboard keys. |
| ... | In a line of syntax, an ellipsis indicates that more items of the same format may be included. In a programming example, an ellipsis is used to indicate lines of a program that were intentionally omitted. |
| [] | Square brackets enclose optional arguments. |
| | A vertical bar indicates that one of the options separated by the bar should be selected. |
|  | Indicates information that is relevant only for action-based tests. |
|  | Indicates information that is relevant only for components. |

Welcome

Part I

Setting Up Add-in Support for SAP Solutions

1

Installing the Add-in for SAP Solutions

This chapter lists the pre-installation requirements and explains how to install the Add-in for SAP Solutions.

This chapter describes:

- Before You Install
- Running the Setup Program

Before You Install

Before you begin to install the Add-in for SAP Solutions, review the system requirements listed below.

To work successfully with the Add-in for SAP Solutions, your system configuration should meet the requirements as specified for QuickTest Professional 8.2 (in the QuickTest Professional 8.2 Readme), plus the following add-in specific requirements:

| | |
|---|--|
| Prerequisites: | QuickTest Professional 8.2, including the Web Add-in. |
| Hard Disk Space: | 20 MB of free hard disk space (in addition to the QuickTest Professional installation). |
| Software for testing SAP Gui for Windows applications or integrating with eCATT: | SAP Gui for Windows 6.20, compilation CD 6, patch level 48, or SAP Gui for Windows 6.40, compilation CD 2, patch level 2 (minimum) or patch level 5. (Patch levels 3 and 4 are not recommended.) |
| Integration with SAP eCATT: | SAP Web Application Server 6.20 with support package 34 or later. |

Running the Setup Program

The setup program installs add-in support for testing SAP applications in your QuickTest Professional installation folder.

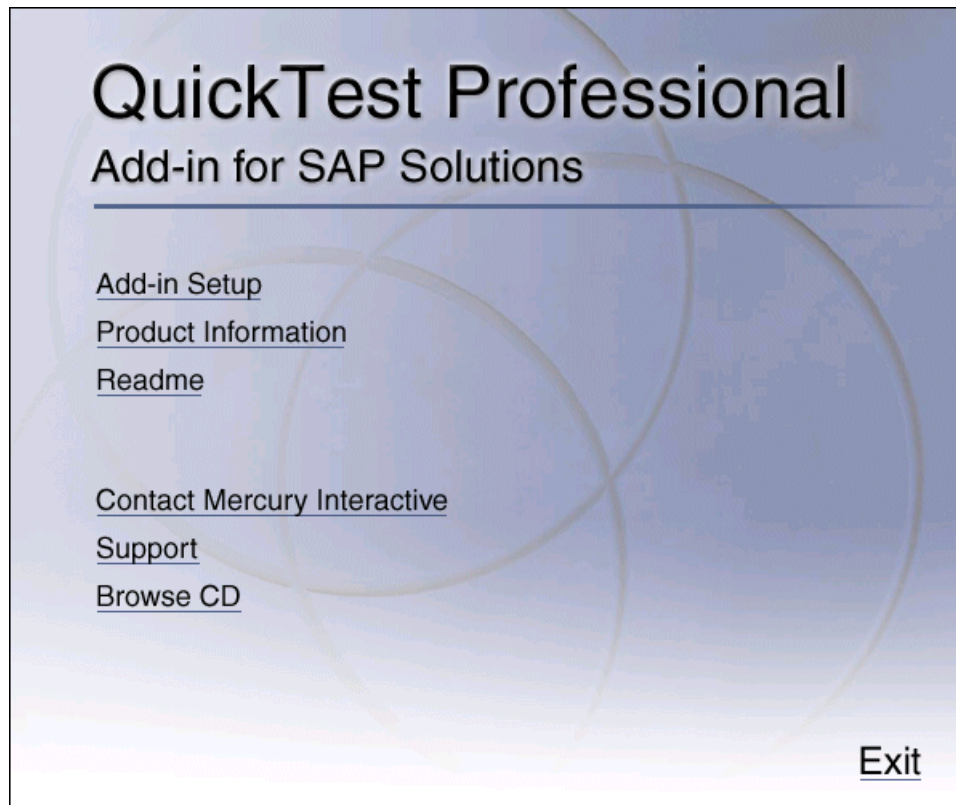
Notes:

To install the Add-in for SAP Solutions, you must be logged on with administrator privileges.

You must also have QuickTest Professional 8.2, including the Web Add-in, installed on your computer. Refer to the *QuickTest Professional Installation Guide* for information on installing QuickTest Professional.

To install the Add-in for SAP Solutions:

- 1** Close any instances of QuickTest Professional.
- 2** Insert the CD-ROM into the CD-ROM drive.
 - If the CD-ROM drive is on your local computer, the QuickTest Professional Add-in for SAP Solutions window opens.
 - If you are installing from a network drive, browse to it and double-click **autorun.exe** in the root folder of the CD-ROM. The QuickTest Professional Add-in for SAP Solutions window opens.

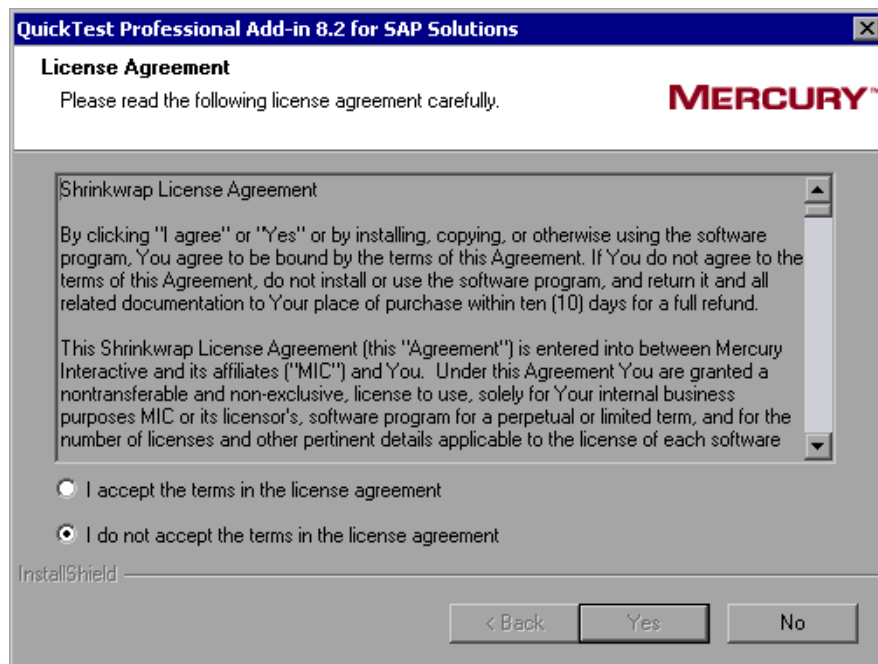


The QuickTest Professional Add-in for SAP Solutions window contains the following options:

- **Add-in Setup**—Starts the Add-in for SAP Solutions installation program.
- **Product Information**—Opens the product information site (<http://www.mercury.com/us/products/quality-center/functional-testing/quicktest-professional/>).
- **Readme**—Opens the *QuickTest Professional Add-in for SAP Solutions Readme* file.
- **Contact Mercury Interactive**—Opens the contact information page on the Mercury Interactive Web site (<http://www.mercury.com/us/company/corporate-info/contact-us/>).
- **Support**—Opens the Mercury Interactive Customer Support Web site (<http://support.mercury.com>).
- **Browse CD**—Displays the contents of the QuickTest Professional Add-in for SAP Solutions CD-ROM.
- **Exit**—Exits the Setup program.

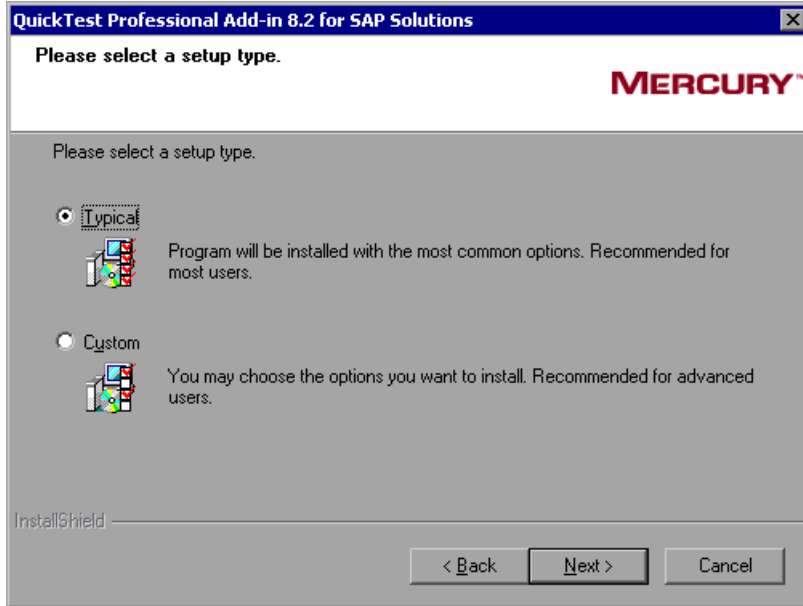
- 3 To start the Add-in for SAP Solutions setup program, click **Add-in Setup**.

- 4 The Add-in for SAP Solutions setup program starts and the License Agreement screen opens. Read the agreement.



To install QuickTest Professional add-in support for testing SAP applications, you must accept the terms of the license agreement by selecting the **I accept the terms in the license agreement** option and clicking **Yes**. If you click **No**, the setup program closes.

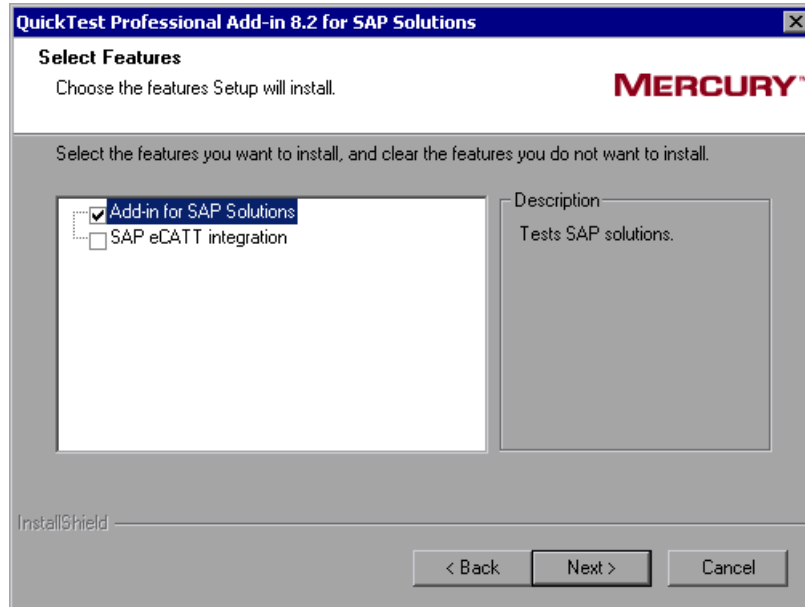
- 5 In the Setup Type screen, select **Typical** or **Custom**.



- The **Typical** installation installs the **Add-in for SAP Solutions** component.
- The **Custom** installation enables you to select the components you want to install.

Click **Next**. If you chose the **Custom** installation, proceed to step 6. Otherwise, proceed to step 7.

- 6 In the Select Features screen, select the features you want to install.



- The **Add-in for SAP Solutions** component enables you to use QuickTest to test SAP Gui for Windows applications and Web-based SAP applications, including SAP Gui for HTML and SAP Enterprise Portal.
- The **SAP eCATT integration** component enables you to store and manage QuickTest tests in the SAP Extended Computer Aided Test Tool (eCATT).

Notes:

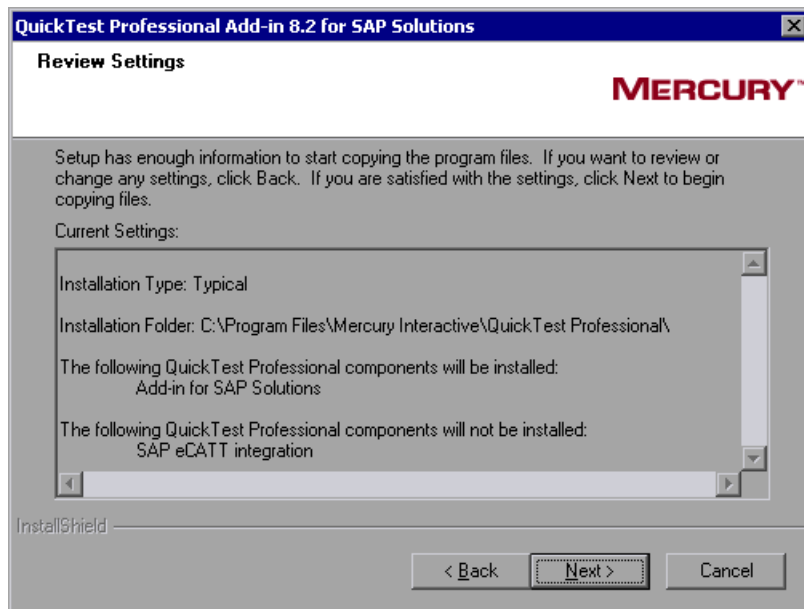
You can install the **SAP eCATT integration** component only if you already have the SAP Gui for Windows software installed on your computer, including support for Unicode. You add support for Unicode by selecting the **Unicode RFC Libraries** check box (under **Development Tools**) during the SAP installation.

You can use eCATT support only when the **SAP eCATT integration** component is installed and the SAP Add-in is loaded in the Add-in Manager.

For information on requirements for eCATT integration, refer to the *QuickTest Professional Add-in for SAP Solutions Readme*.

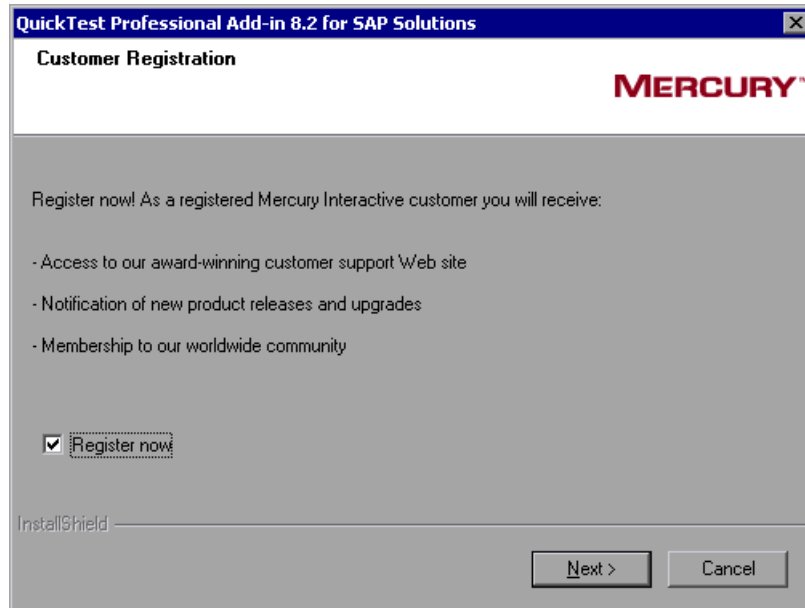
Click **Next** to proceed.

- 7 In the Review Settings screen, review the settings you selected.



To change your settings, click **Back**. To confirm the settings, click **Next** to begin the installation process.

- 8** In the Customer Registration screen, you can specify whether to register your copy of the QuickTest Professional Add-in for SAP Solutions.



If you register, you receive:

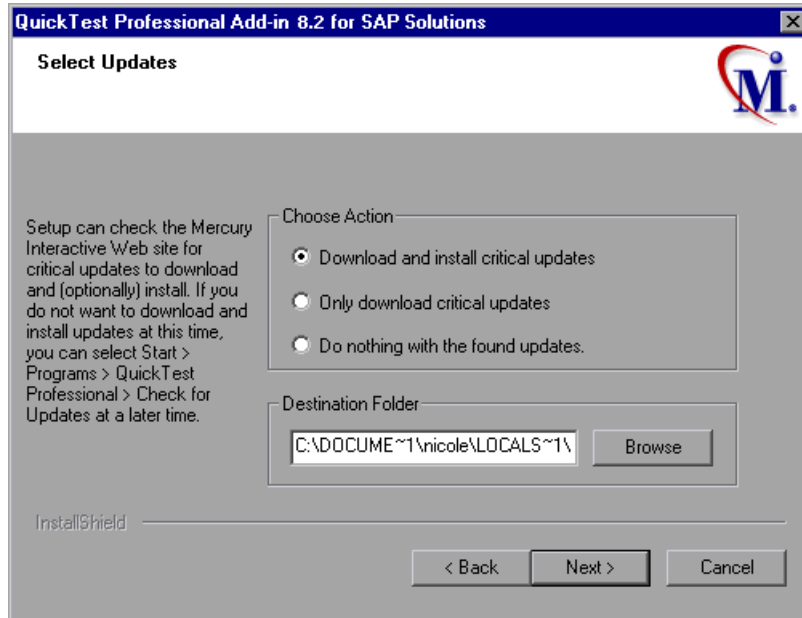
- access to the Mercury Interactive award-winning Customer Support Web site.
- notification of new product releases and upgrades.
- membership to the Mercury Interactive worldwide community of testers.

To register now, confirm that the **Register now** check box is selected. Click **Next** to proceed. Your browser opens to the Mercury Interactive Customer Support site: <http://support.mercury.com>

Click **Next** to proceed.

- 9 The Setup application checks whether there are any critical updates to be installed for the version of the Add-in for SAP Solutions you are installing. It also checks whether there are any critical updates to be installed for the version of QuickTest Professional that is installed. If no critical updates are found, the installation continues with step 11.

If the Setup application finds any critical updates to be installed, the Select Updates screen opens.



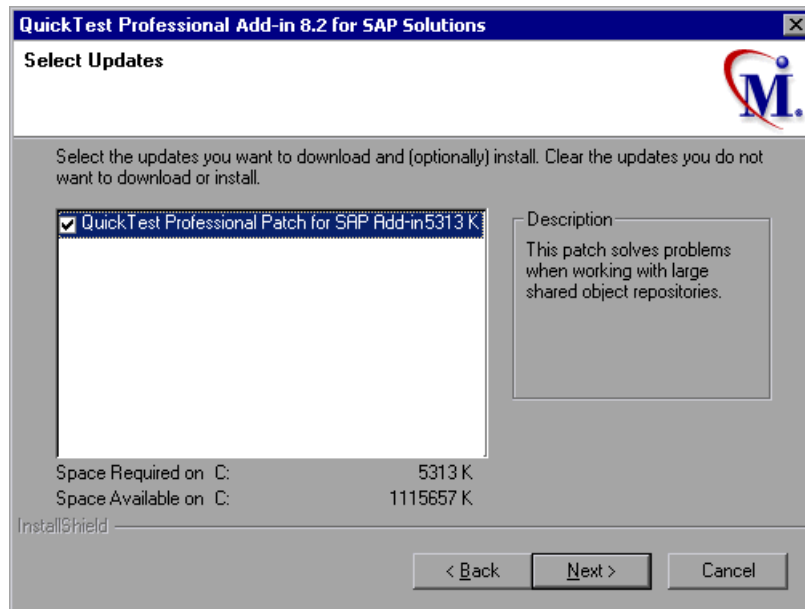
Choose one of the following options:

- **Download and install critical updates**—Downloads any critical updates to the specified destination folder and then installs them. This option is selected by default.
- **Only download critical updates**—Downloads any critical updates to the specified destination folder. You then need to double-click the downloaded files to install them.
- **Do nothing with the found updates**—Does not download or install critical updates.

Note: If Setup finds any critical updates to be installed, it is highly recommended to install them immediately. (You can also check for updates at a later time by choosing **Start > Programs > QuickTest Professional > Check for Updates.**)

A default download folder is displayed in the **Destination Folder** box. To select a different location to which to download the file, click **Browse**, choose a folder, and click **OK**.

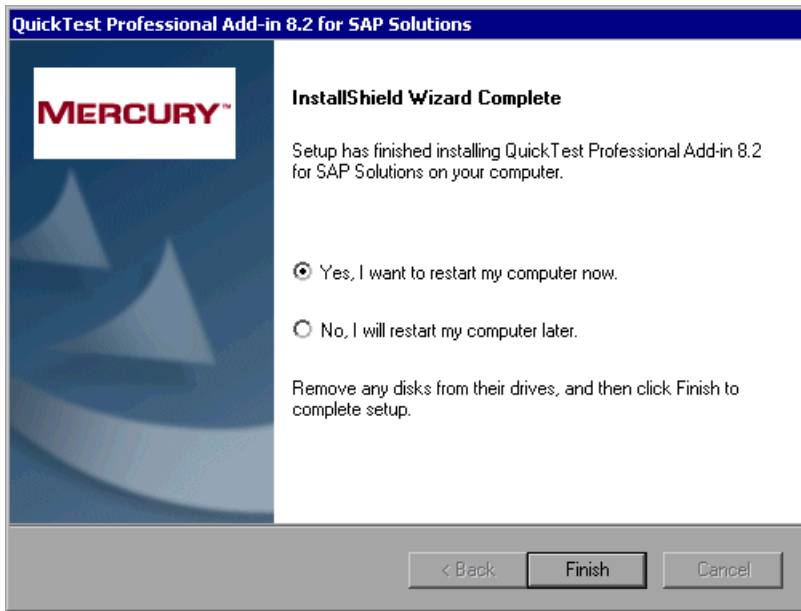
Click **Next** to proceed. If you chose to download and/or install critical updates, a list of all available critical updates opens. Otherwise, continue with step 11.



- 10** Select the updates that you want to download and (optionally) install. Clear the updates that you do not want to download or install.

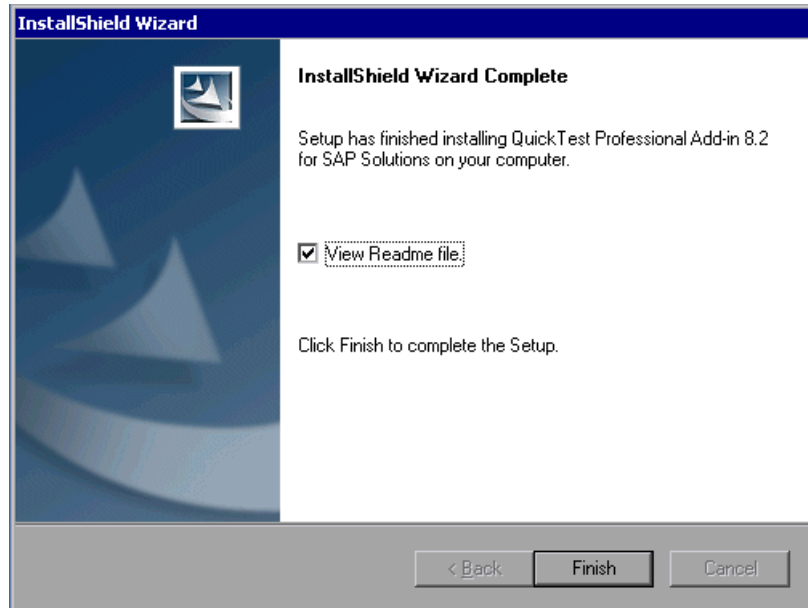
Click **Next** to proceed. QuickTest downloads the specified critical updates, according to the option you selected in step 9. If you chose to download and install the updates, they are installed at the end of the add-in installation process, after you restart your computer.

- 11** If the InstallShield Wizard Complete screen prompts you to restart your computer, you can choose to restart your computer at a later time, but you must restart your computer before you use QuickTest Professional.



Note: If you are prompted to restart, it is strongly recommended that you restart your computer as soon as possible. Delaying the system restart could result in unexpected system problems.

- 12** After you restart the computer, Setup continues and the InstallShield Wizard Complete screen opens. If you want to open the *Add-in for SAP Solutions Readme* file at the end of the setup process, select the **View Readme file** check box. Click **Finish** to complete the setup process.



The Readme file contains the latest technical and troubleshooting information. To open the Readme file at another time, choose **Start > Programs > QuickTest Professional > Add-ins > Add-in for SAP Solutions Readme**.

2

Activating Add-in Support for SAP Solutions

Before you can work with the QuickTest Professional Add-in for SAP Solutions, you must make sure that the Add-in for SAP Solutions is licensed, and then use the Add-in Manager to load QuickTest Professional with add-in support for SAP. You can load QuickTest without add-in support for SAP if you do not want to test SAP applications.

This chapter describes:

- Licensing the Add-in for SAP Solutions
- Loading QuickTest with Add-in Support for SAP Solutions
- Loading QuickTest without Add-in Support for SAP Solutions
- Working with Function Library Files
- Uninstalling and Repairing the Add-in for SAP Solutions

Licensing the Add-in for SAP Solutions

QuickTest add-ins that are installed separately from the QuickTest Professional installation (referred to as *external* add-ins) require a separate seat or concurrent license code.

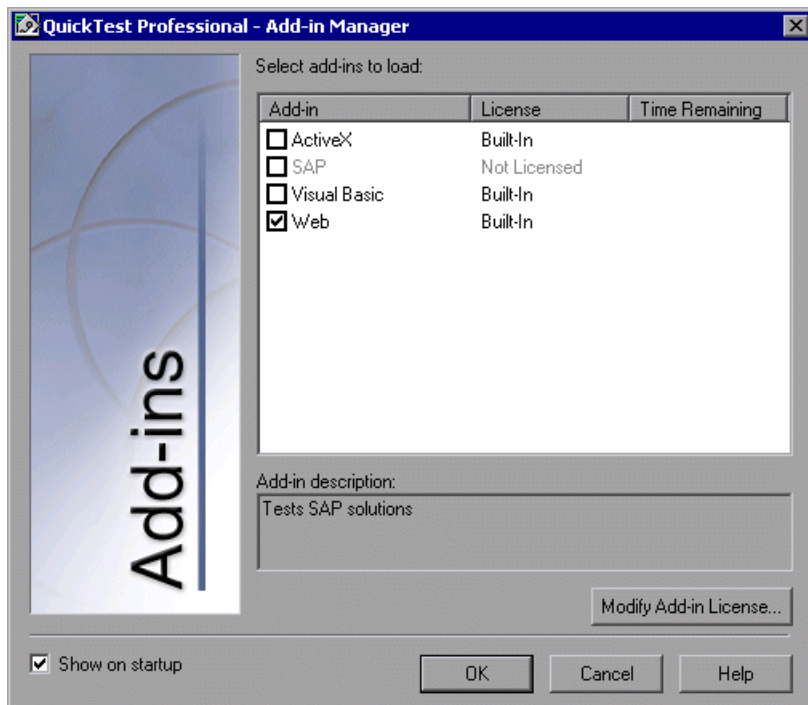
For seat licenses, you install the QuickTest Professional Add-in License by clicking the **Modify Add-in License** button in the Add-in Manager dialog box, as described in this section.

For concurrent licenses, you install the QuickTest Professional Add-in License on the Mercury Functional Testing Concurrent License Server computer. For information on installing a concurrent license, refer to the *Mercury Functional Testing Concurrent License Server Installation Guide*.

To access the add-in license installation wizard:

- 1 Choose **Start > Programs > QuickTest Professional > QuickTest Professional**. The QuickTest Professional - Add-in Manager dialog box opens.

(If the Add-in Manager dialog box does not open, see “Displaying the Add-in Manager Dialog Box” on page 20.)



Note: If the add-in license has not yet been installed, the **SAP** option is displayed as **Not Licensed** in the Add-in Manager dialog box.

If the add-in license has already been installed, the license category is displayed (**Time-Limited** or **Permanent**). For **Time-Limited** licenses, the number of days and hours remaining until the license expires is also displayed in the **Time Remaining** column. For more information on the Add-in Manager dialog box, refer to the *QuickTest Professional User's Guide*.

The Add-in Manager displays the **Modify Add-in License** button and **Time Remaining** column only when a QuickTest seat license is installed on your computer.

- 2 Click **Modify Add-in License**. The QuickTest Professional License Installation - Welcome window opens. Install your add-in license.

The procedure for installing an add-in license is the same as the procedure for installing a QuickTest Professional license. Refer to the *QuickTest Professional Installation Guide* for more information.

Loading QuickTest with Add-in Support for SAP Solutions

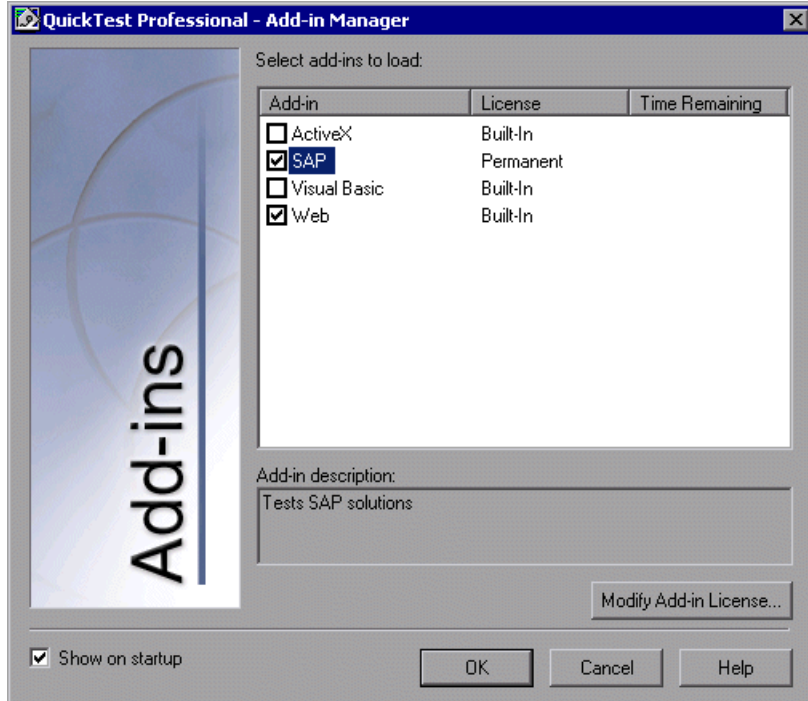
You use the Add-in Manager to load support for testing SAP Solutions applications.

To start QuickTest with add-in support for testing SAP applications:

- 1 Choose **Start > Programs > QuickTest Professional > QuickTest Professional**. The QuickTest Professional - Add-in Manager dialog box opens.

(If the Add-in Manager dialog box does not open, see “Displaying the Add-in Manager Dialog Box” on page 20.)

- 2 In the add-in list, select **SAP** and **Web**.



Click **OK**.

For more information about the Add-in Manager dialog box, refer to the *QuickTest Professional User's Guide*.

Displaying the Add-in Manager Dialog Box

If the Add-in Manager dialog box does not open when you open QuickTest:

- 1 Choose **Start > Programs > QuickTest Professional > QuickTest Professional** to start QuickTest.
- 2 From the QuickTest menu, choose **Tools > Options** and click the **General** tab.
- 3 Select **Display Add-in Manager on startup**.
- 4 Click **OK** to close the Options dialog box.
- 5 Close and reopen QuickTest.

Loading QuickTest without Add-in Support for SAP Solutions

If you want to work with QuickTest without support for SAP, you can load QuickTest without the Add-in for SAP Solutions.

To load QuickTest without add-in support for SAP:

- 1** Choose **Start > Programs > QuickTest Professional > QuickTest Professional**. The QuickTest Professional Add-in Manager dialog box opens. (If the Add-in Manager dialog box does not open, see “Displaying the Add-in Manager Dialog Box” on page 20.)
- 2** Clear the **SAP** check box and click **OK**. QuickTest opens without add-in support for SAP.

Working with Function Library Files

QuickTest provides out of the box resource function library files that are specially designed to provide additional functionality for business components when working with Business Process Testing to test your SAP application using the QuickTest Professional Add-in for SAP Solutions. This enables you or Subject Matter Experts creating business process tests in Quality Center to choose functions that perform complex operations. In particular, these files contain functions that enable you to open a specific application, write results of specific component steps to the test results, and verify and retrieve object property values.

These function library files are installed locally on your computer when you install QuickTest and the QuickTest Professional Add-in for SAP Solutions. To use these functions in your business components, you attach the **SAP.txt** and **Common.txt** files to your Quality Center project and then associate them with your application area or specific business component. For more information on working with application areas and business components, refer to the *QuickTest Professional New Features Guide*.

Notes:

The functions included in the **SAP.txt** file only work in conjunction with the **Common.txt** file. Therefore, you must make sure that both files are added to your project and associated with your application area or business component.

It is recommended not to modify these function library files, as QuickTest may overwrite them at a later stage, for example, when you upgrade QuickTest or the QuickTest Professional Add-in for SAP Solutions.

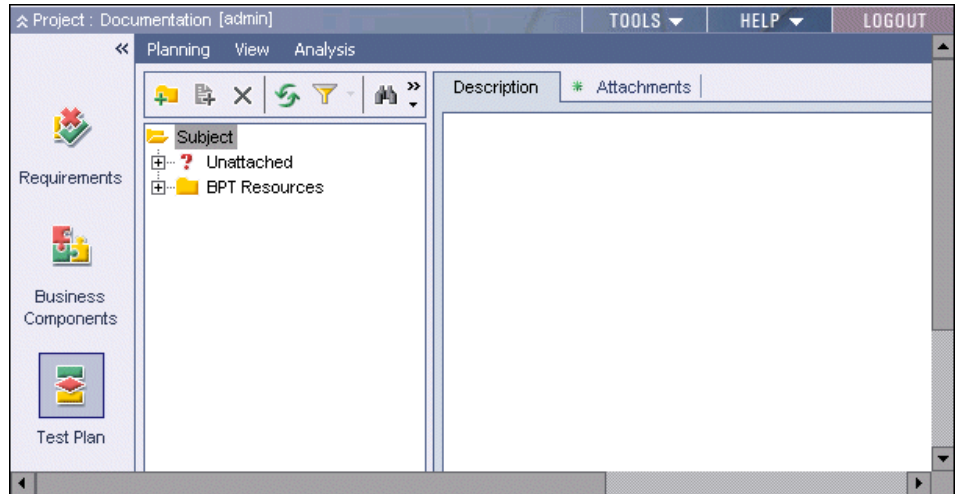
Attaching Function Library Files to Your Quality Center Project

Before you can use the functions stored in the **SAP.txt** and **Common.txt** function library files in your component steps, you must attach both of these files to your Quality Center project. By default, the first time that you connect to a Quality Center project with Business Process Testing support, QuickTest sets up default Business Process Testing folders and files (including **Common.txt**) in your Quality Center project. If, for any reason **Common.txt** does not appear in the list (for example, if the file was inadvertently deleted), you can attach it manually, as described in this section.

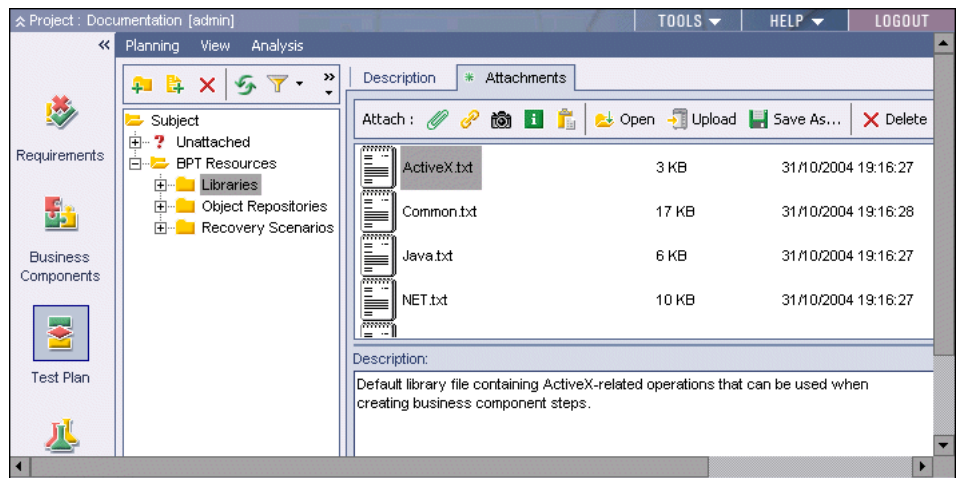
Note: You can also attach and use function library files that contain user-defined functions. For more information, refer to the *QuickTest Professional New Features Guide*.

To add the function library file to your Quality Center project:

- 1 Open the Test Plan module in a Quality Center project that has Business Process Testing support. For more information on opening Quality Center projects, refer to the *Mercury Quality Center User's Guide*.



- 2 Expand the **BPT Resources** branch in the test plan tree and select **Libraries**.
- 3 Click the **Attachments** tab in the right pane. The Attachments tab opens and displays the function library files that are attached to your project.





- 4** In the right pane, click the **File** button. The Open dialog box opens.
- 5** Browse to the QuickTest Professional installation folder on your local computer, expand the **dat\BPT_Resources** folder, and select **SAP.txt**. Click **Open**. The **SAP.txt** library file name is added to the Attachments list.
- 6** In the **Description** box, add text that clearly describes what the file can be used for, for example: Default library file containing SAP-related operations that can be used when creating business process components.
- 7** If the **Common.txt** file is not included in the Attachments tab, repeat steps 4 to 6 and attach the **Common.txt** file to your Quality Center project.
- 8** Close Quality Center, if desired.

You can now associate the library file with your application area or component. For more information, see “Associating Add-in Function Library Files with Your Application Area or Component”.

Associating Add-in Function Library Files with Your Application Area or Component

After you attach the **SAP.txt** (and **Common.txt**, if not already attached) function library files to your Quality Center project, you can associate both of these resource files with an application area or a component. This enables you or Subject Matter Experts working in Quality Center to use the functions included in these files when creating, modifying, or running component steps. If you associate these resource files with an application area, all associated components can access their functions. If you associate these files directly with a component, the functions are accessible only to that component.

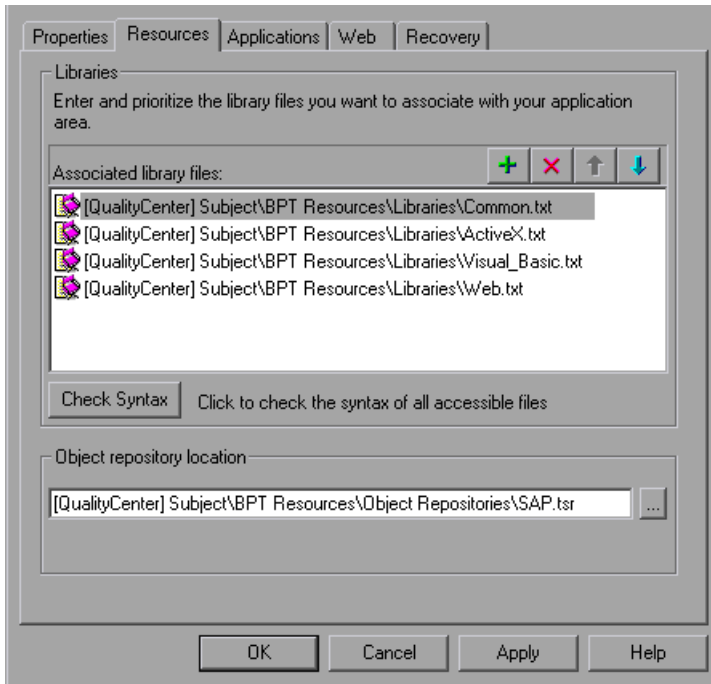
To associate the add-in library files with an application area or a specific component:

- 1 Verify that the **SAP.txt** and **Common.txt** files are both attached to your Quality Center project.

Tip: You can check that these files are attached to your project by opening the Test Plan module of your Quality Center project, clicking the **Attachments** tab, and expanding the branches to display the files stored in **Subject\BPT Resources\Libraries**. If the **Common.txt** and/or the **SAP.txt** files are not displayed, you can add them to your project, as described in “Attaching Function Library Files to Your Quality Center Project” on page 22.

- 2 Open QuickTest and connect to your Quality Center project with Business Process Testing support. For information on connecting to a Quality Center project, refer to the *QuickTest Professional New Features Guide*.
- 3 Open the application area on which business components testing SAP solutions applications will be based or create a new application area (**File > Open > Application Area** or **File > New > Application Area**). Alternatively, you can open a specific business component used in business process tests for SAP solutions applications. For more information on opening and creating application areas and business components, refer to the *QuickTest Professional New Features Guide*.
- 4 Choose **Application Area > Settings** (or **Component > Settings**) to open the QuickTest Application Area Settings dialog box (or the Business Component Settings dialog box if you want to associate the files with a specific component), and click the **Resources** tab.

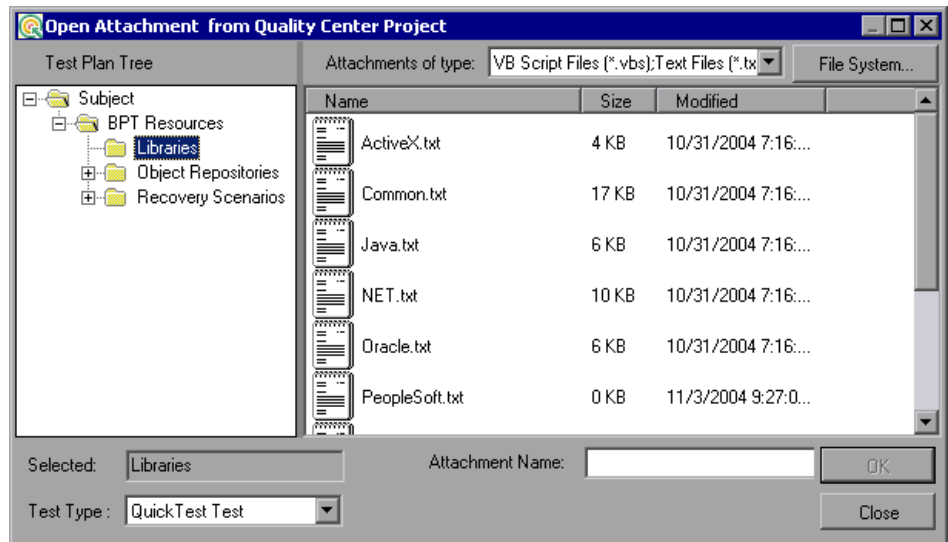
Note: If you are creating a new application area, the Application Area Settings dialog box opens automatically.



- 5 Click the **Add** button. QuickTest adds [QualityCenter] and displays a browse button so that you can locate the Quality Center path.

Note: You can also type the entire Quality Center path manually. If you do, you must add a space after [QualityCenter], for example:
[QualityCenter] Subject\BPT Resources.

- 6 Click the browse button. The Open Attachment from Quality Center Project dialog box opens, displaying the files located in the **Subject\BPT Resources\Libraries** folder.



- 7 Select the **SAP.txt** file and click **OK**. The file is added to the **Associated library files** area in the Resources tab.
- 8 If the **Common.txt** file is not displayed in the **Associated library files** area, repeat steps 5 to 7 and associate it with your application area or component.

You can now view and use the functions located in the **SAP.txt** and **Common.txt** function library files from any component that references these resource files.

Uninstalling and Repairing the Add-in for SAP Solutions

You can uninstall the QuickTest Professional Add-in for SAP Solutions 8.2 without uninstalling QuickTest or any other add-ins. You can also repair a QuickTest Professional Add-in for SAP Solutions 8.2 installation that has become corrupted.

Uninstalling the Add-in for SAP Solutions

You can uninstall the Add-in for SAP Solutions using either the **Add/Remove Programs** option in the Windows Control Panel, or using the QuickTest Professional Add-in for SAP Solutions CD-ROM.

Note: If you uninstall QuickTest, the uninstall program also removes all installed features, including any external add-ins that are installed. For more information on uninstalling QuickTest, refer to the *QuickTest Professional Installation Guide*.

To uninstall the Add-in for SAP Solutions using the Add/Remove Programs option:

- 1** Make sure that QuickTest is closed.
- 2** Choose **Start > Settings > Control Panel**.
- 3** Double-click the **Add/Remove Programs** option.
- 4** In the Add/Remove Programs dialog box, select **QuickTest Professional Add-in for SAP Solutions** and then click **Change/Remove**.
- 5** A message prompts you to confirm your decision to uninstall the QuickTest Professional Add-in for SAP Solutions. Click **Yes** to uninstall the Add-in for SAP Solutions. The uninstall program removes the Add-in for SAP Solutions from your computer. QuickTest Professional and any other installed add-ins remain on your computer.

Note: Clicking **No** keeps the Add-in for SAP Solutions installed on your computer.

- 6** The Maintenance Complete screen may prompt you to restart your computer. If it does, you can choose to restart your computer at a later time, but you must restart your computer before you use QuickTest Professional.

Note: You can save any open files, but you should restart your computer as soon as possible. Delaying the system restart could result in unexpected system problems.

- 7** Click **Finish** to complete the uninstall process.

To uninstall the Add-in for SAP Solutions using the QuickTest Professional Add-in for SAP Solutions CD-ROM:

- 1** Insert the QuickTest Professional Add-in for SAP Solutions CD-ROM into the CD-ROM drive. If the CD-ROM drive is on your local computer, the QuickTest Professional Add-in for SAP Solutions Setup window opens.

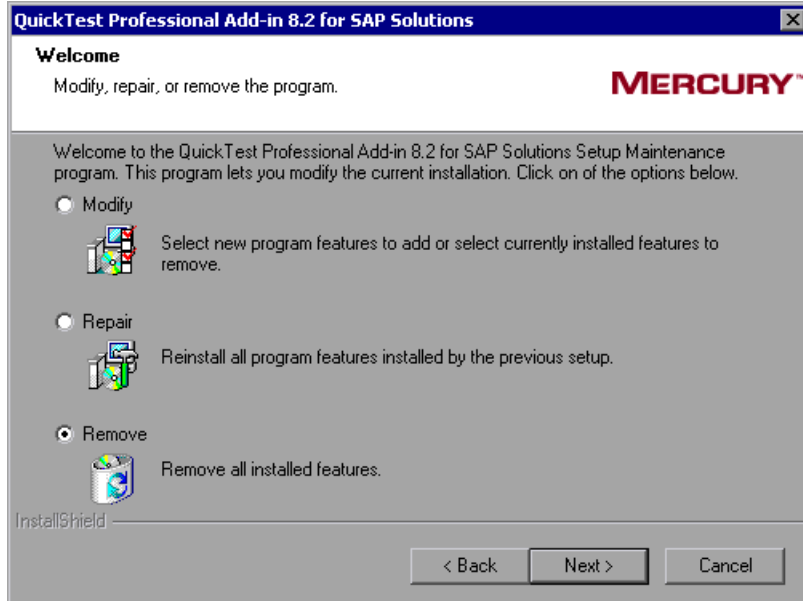
If the CD-ROM is in a network drive, double-click **autorun.exe** in the root folder of the CD-ROM.

Note: You must use the exact same version of the Add-in for SAP Solutions that you used for the original installation.

The License Agreement screen opens.

- 2** Read the license agreement, select **I accept the terms in the license agreement**, and click **Yes**. If you click **No**, the setup program will close.

- 3** In the Setup Maintenance screen, select **Remove** and click **Next**.



- 4** A message prompts you to confirm your decision to uninstall the QuickTest Professional Add-in for SAP Solutions. Click **OK** to uninstall the Add-in for SAP Solutions. The uninstall program removes the Add-in for SAP Solutions from your computer. QuickTest Professional and any other installed add-ins remain on your computer.

Note: Clicking **Cancel** keeps the Add-in for SAP Solutions installed on your computer and returns to the previous screen.

- 5 The Maintenance Complete screen may prompt you to restart your computer. If it does, you can choose to restart your computer at a later time, but you must restart your computer before you use QuickTest Professional.

Note: You can save any open files, but, if prompted to do so, you should restart your computer as soon as possible. Delaying the system restart could result in unexpected system problems.

- 6 Click **Finish** to complete the uninstall process.

Repairing the Add-in for SAP Solutions Installation

Your QuickTest Professional Add-in for SAP Solutions 8.2 CD-ROM enables you to repair an existing Add-in for SAP Solutions 8.2 installation, by replacing any missing or damaged files from your previous Add-in for SAP Solutions installation.

To repair your Add-in for SAP Solutions installation:

- 1 Insert the QuickTest Professional Add-in for SAP Solutions CD-ROM into the CD-ROM drive. If the CD-ROM drive is on your local computer, the QuickTest Professional Add-in for SAP Solutions Setup window opens.

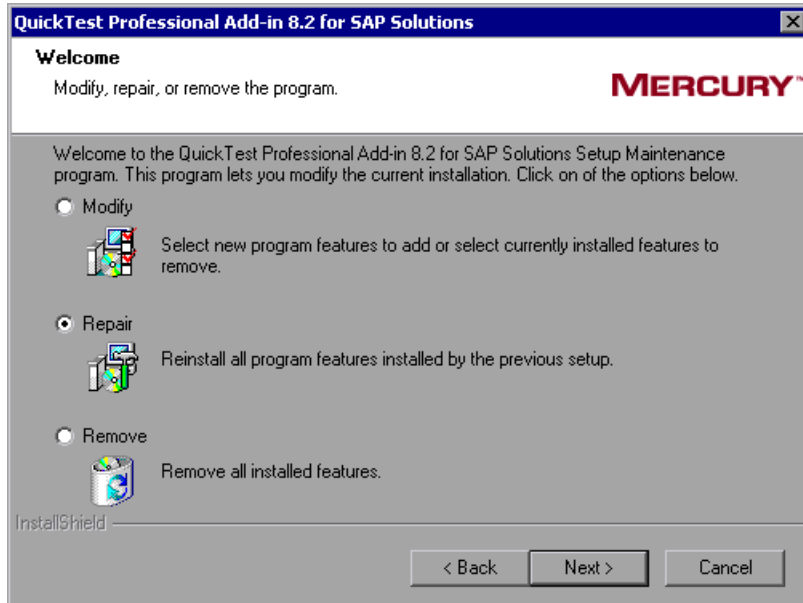
If the CD-ROM is in a network drive, double-click **autorun.exe** in the root folder of the CD-ROM.

Note: You must use the exact same version of the Add-in for SAP Solutions that you used for the original installation.

The License Agreement screen opens.

- 2 Read the license agreement, select **I accept the terms in the license agreement**, and click **Yes**. If you click **No**, the setup program will close.

- 3 In the Setup Maintenance screen, select **Repair** and click **Next**.



The maintenance program repairs your QuickTest Professional Add-in for SAP Solutions installation.

- 4 The Maintenance Complete screen may prompt you to restart your computer. If it does, you can choose to restart your computer at a later time, but you must restart your computer before you use QuickTest Professional.

Note: You can save any open files, but, if prompted to do so, you should restart your computer as soon as possible. Delaying the system restart could result in unexpected system problems.

- 5 Click **Finish** to complete the repair process.

Part II

Testing SAP Gui for Windows Applications

3

Setting Up Your SAP Gui for Windows Environment

Before you can start testing your SAP Gui for Windows applications, you must make sure that your server and client are installed and configured with the correct versions and support options. This chapter provides the specific setup information that you need to successfully use the QuickTest Professional Add-in for SAP Solutions. This chapter is not relevant if you only test Web-based SAP applications.

This chapter describes:

- About Setting Up Your SAP Windows Environment
- Installing SAP Gui Scripting Support
- Checking Package and Patch Versions Installed on the SAP Application Server
- Checking the Patch Version Installed on your SAP Gui for Windows Application
- Enabling Scripting on the SAP Application (Server-Side)
- Enabling Scripting on the SAP Application (Client-Side)
- Setting F4 Help to Use Dialog Display Mode
- Setting F1 Help to Use Modal Dialog Box Mode
- Checking the Connection Speed on the SAP Server

About Setting Up Your SAP Windows Environment

QuickTest Professional's support for SAP Gui for Windows version 6.20 and 6.40 is based on the SAP Gui Scripting API. This API is available only in recent versions of the SAP Kernel and is disabled by default.

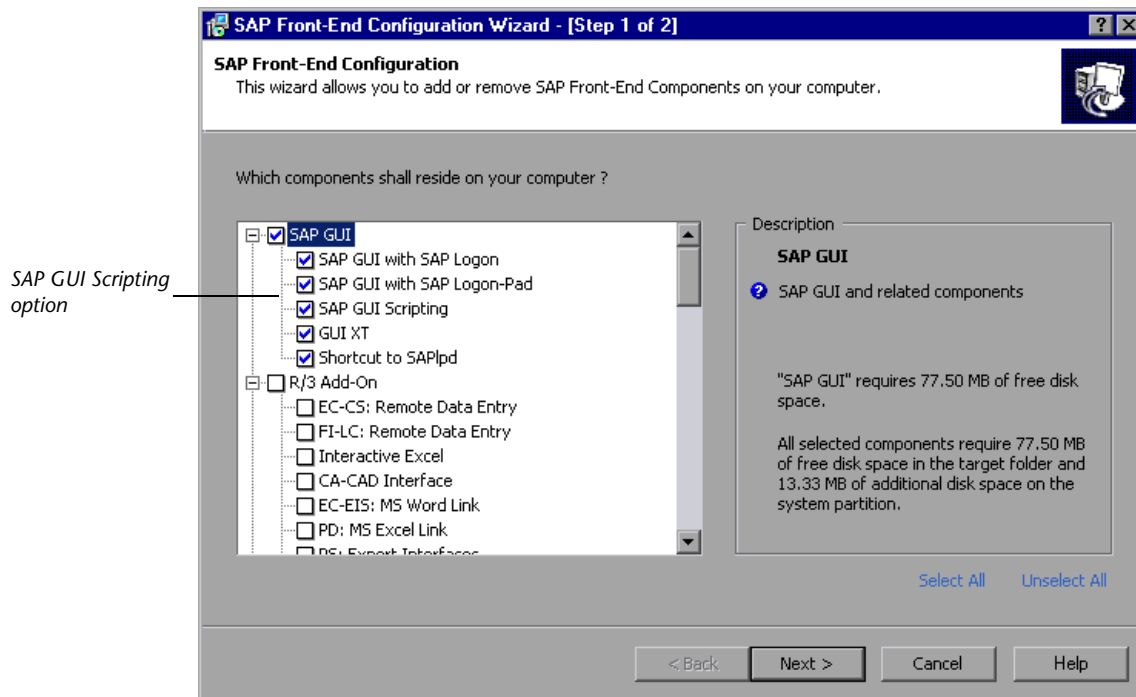
To test your SAP Gui for Windows application using the QuickTest Professional Add-in for SAP Solutions, you must confirm that:

- The SAP Gui Scripting option is installed.
- Your server and client have the proper package and patch versions installed.
- Your server supports the Scripting API.
- The Scripting API is enabled on both the server and clients.
- Your client is configured to use the **Dialog** display mode for F4 Help screens and that it is not set to use a **Low speed connection**.
- The F1 and F4 Help display setting is configured correctly, to support testing the use of the F1 and F4 Help screens in your SAP Gui for Windows application.

Note: If you plan to use the QuickTest-eCATT integration features, you must also install the appropriate support package and configure the eCATT server to work with QuickTest. For more information, see Chapter 11, "Configuring eCATT to Work with QuickTest."

Installing SAP Gui Scripting Support

When you install your SAP Gui for Windows application, you must select the **SAP GUI Scripting** installation option.



If you did not select this option when you installed the SAP Gui for Windows application, it is essential that you reinstall it and select this option before setting the other configuration options described in this chapter.

Note: SAP provides a range of security mechanisms that allow the administrator to limit the use of SAP Gui Scripting by system, by group, by user, and by scripting functionality. To test SAP Gui for Windows applications, you must ensure that these security mechanisms are not activated for the application you are testing. For more information about the various security options, refer to the online SAP Gui Scripting Security Guide at the SAP Service Marketplace.

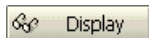
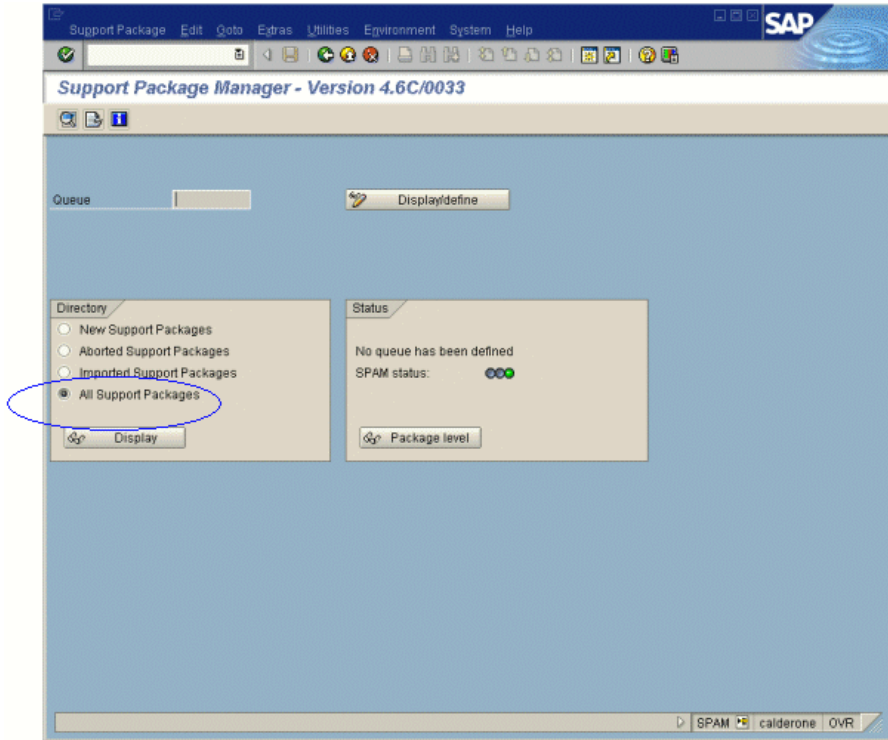
Checking Package and Patch Versions Installed on the SAP Application Server

To use the QuickTest Professional Add-in for SAP Solutions, you must confirm that you have the correct support package and kernel patch levels for your software component release. The following table shows the **minimum** required. You must have these versions/levels or higher:

| Software Component | Release | Support Package | Kernel Patch Level |
|--------------------|---------|-----------------|-----------------------|
| SAP_APPL | 31I | SAPKH31I96 | Kernel 3.1I level 650 |
| SAP_APPL | 40B | SAPKH40B71 | Kernel 4.0B level 903 |
| SAP_APPL | 45B | SAPKH45B49 | Kernel 4.5B level 753 |
| SAP_BASIS | 46B | SAPKB46B37 | Kernel 4.6D level 948 |
| SAP_BASIS | 46C | SAPKB46C29 | Kernel 4.6D level 948 |
| SAP_BASIS | 46D | SAPKB46D17 | Kernel 4.6D level 948 |
| SAP_BASIS | 610 | SAPKB61012 | Kernel 6.10 level 360 |

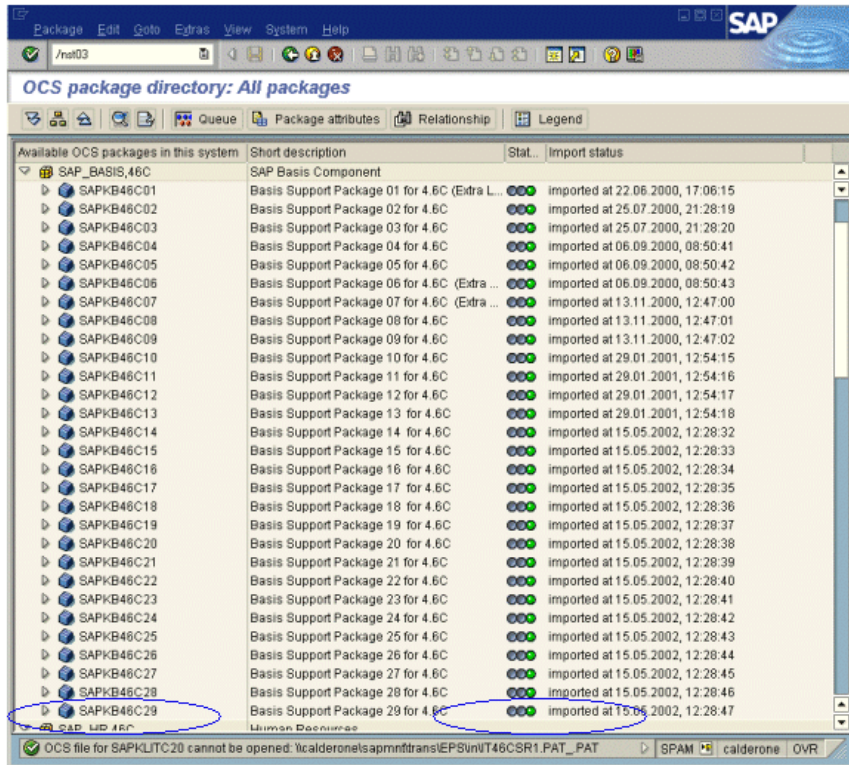
To check the support package:

- 1** Log on to your SAP server.
- 2** Run the SPAM transaction.



- 3** In the **Directory** section, select **All Support Packages** and click the **Display** button.

- 4 The All packages window opens. Verify that the correct package is installed for the SAP release you are using (see “Checking Package and Patch Versions Installed on the SAP Application Server” on page 38).



If the correct package is installed, a green light icon is displayed in the **Status** column.

If you do not have the required package installed, download and install it.

For more information on downloading and installing the required package, refer to SAP OSS note # 480149.

To check the kernel patch level:

- 1** Log on to your SAP server.
- 2** Choose **System > Status**. The System: Status dialog box opens.

The screenshot shows the 'System: Status' dialog box with the following data:

| Usage data | | | |
|------------|------|----------------|---------------------|
| Client | 800 | Previous logon | 05.11.2004 16:12:38 |
| User | QA01 | Logon | 07.11.2004 08:53:10 |
| Language | EN | System time | 16:37:39 |
| | | Time zone | CET 15:37:39 |

| SAP data | |
|---------------------|------------------|
| Repository data | SAP System data |
| Transaction | PA30 |
| Program (screen) | MP000600 |
| Screen number | 2001 |
| Program (GUI) | MP000600 |
| GUI status | DIS |
| Component version | R/3 Release 4.6C |
| Installation number | 0120033759 |
| License expiry date | 31.12.9999 |

| Host data | | Database data | |
|------------------|---------------|---------------|-----------|
| Operating system | Windows NT | System | ORACLE |
| Machine type | 2x Intel 8 | Release | 8.1.7.0.0 |
| Server name | calderone_MI6 | Name | MI6 |
| Platform ID | 560 | Host | CALDERONE |
| | | Owner | SAPR3 |

At the bottom, there is a 'Navigate' button with a green checkmark icon and a red X icon.



- 3 Click the **Other kernel information** button. The System: Kernel information dialog box opens.

| Kernel information | |
|--------------------|-------------------|
| Kernel release | 46D |
| Compilation | NT 4.0 1381 Servi |
| Sup.Pkg lvl. | 1729 |
| ABAP Load | 1258 |
| CUA load | 12 |
| Mode | opt |
| Rsyn file | |

| Database information | |
|----------------------|------------------|
| DB client lib. | OCI_805_SHARE |
| DB releases | ORACLE 8.0.5.*.* |
| DBSL version | 46D.00 |

| System information | |
|--------------------|--------------------|
| IP address | 212.199.95.5 |
| SAP versions | 46A, 46B, 46C, 46D |
| Operat. system | Windows NT 4.0, W |
| OP release | 5.0 |

- 4 In the **Kernel information** section, check the value of the **Sup. Pckg. lvl.**

If the level is lower than the required level for the SAP release you are using (see “Checking Package and Patch Versions Installed on the SAP Application Server” on page 38), you must download the latest kernel version and upgrade your existing one.


For more information on downloading and installing the required kernel patch level, refer to SAP OSS note #480149.

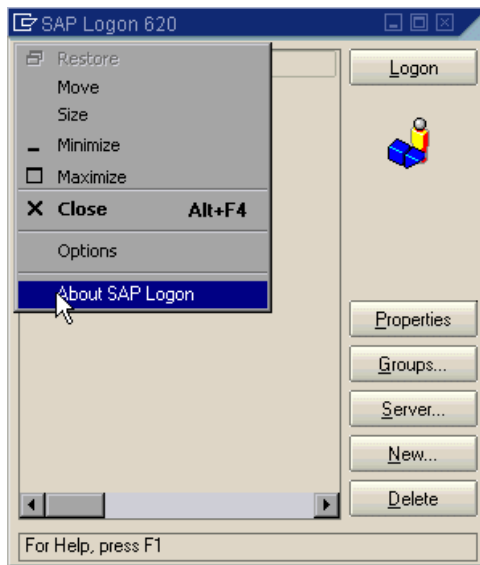
Checking the Patch Version Installed on your SAP Gui for Windows Application

If you want to test your SAP Gui for Windows application with the QuickTest Professional Add-in for SAP Solutions, make sure that the minimum required patch level is installed. For more information about required patch levels, refer to the *QuickTest Professional Add-in for SAP Solutions Readme*.

Note: If the minimum required patch level is not installed, an error message displays when you try to record on your SAP Gui for Windows application.

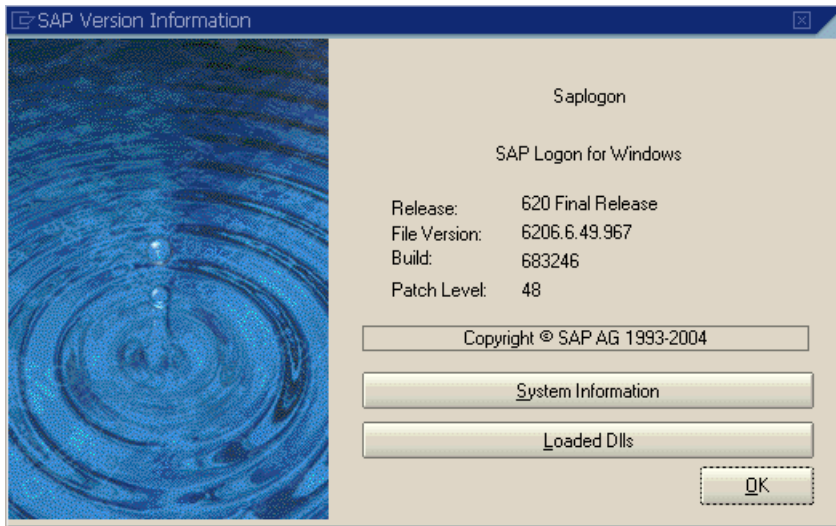
To check the patch level on your SAP Gui for Windows application:

- 1 Open the SAP Logon dialog box and click the  button on the left side of the SAP Logon dialog box's title bar. Then choose **About SAP Logon** from the menu.



The SAP Version Information dialog box opens.

- 2 In the SAP Version Information dialog box, confirm that the minimum required patch level is installed.



Enabling Scripting on the SAP Application (Server-Side)

After you confirm that you have the proper support package and kernel patch levels installed, you must enable scripting on your SAP application. By default, scripting is disabled.

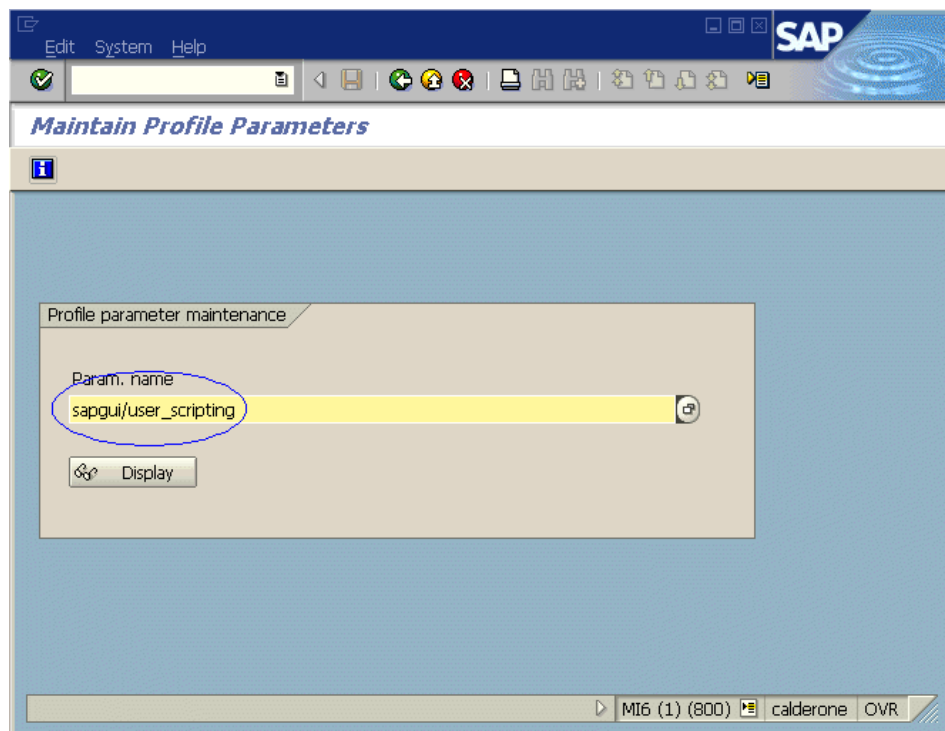
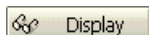
You enable scripting by entering the Maintain Profile Parameters window with administrative permissions and setting the *sapgui/user_scripting* profile parameter to TRUE on the application server.

To enable scripting for all users, set this parameter on all application servers. To enable scripting for a specific group of users, set the parameter only on application servers with the appropriate access restriction settings.

Note: If you connect to a server on which scripting is disabled, an error message displays when you try to record on your SAP Gui for Windows application.

To change the profile parameter:

- 1 Enter **/nrz11** in the OKCode edit box to open transaction **rz11**.
- 2 In the **Param. Name** box of the Maintain Profile Parameters window, enter **sapgui/user_scripting** and click the **Display** button.



Note: If the message **Parameter name is unknown** is displayed in the status bar, your client lacks the required support package (see “Checking the Patch Version Installed on your SAP Gui for Windows Application” on page 43). Download and install the support package that corresponds to the SAP release you are using and then repeat steps 1 and 2.

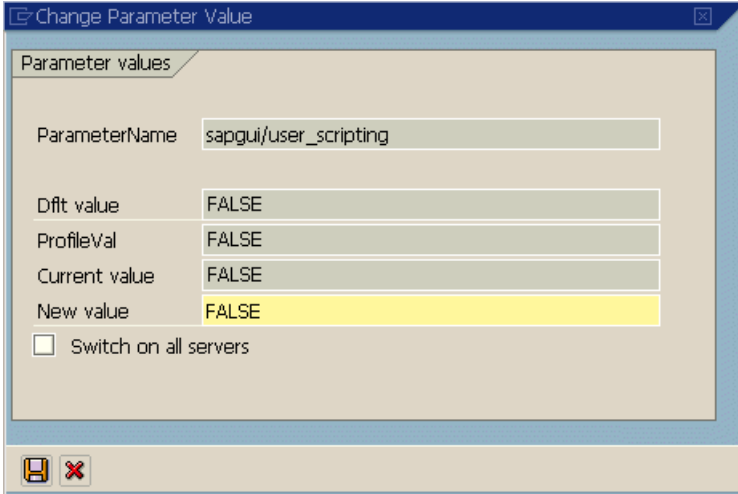
The Display Profile Parameter Attributes window opens.

The screenshot shows the SAP 'Display Profile Parameter Attributes' window. The title bar includes 'Edit', 'Goto', 'System', and 'Help' menus, along with standard window controls and the SAP logo. Below the title bar is a toolbar with various icons. The main content area has a tabbed interface with 'Documentation' and 'Change value' tabs. The 'Documentation' tab is active, showing the parameter details for 'sapgui/user_scripting'. The parameter name is entered in a text field. Below this, several attributes are listed in a table-like format, each with a value and a small icon for editing or documentation. The attributes include: Short description(Engl), Appl. area, ParameterType, Changes allowed, Valid for oper. system, DynamicallySwitchable, Same on all servers, Dflt value, ProfileVal, and Current value. The 'DynamicallySwitchable' checkbox is checked, while 'Same on all servers' is unchecked. The default, profile, and current values are all set to 'TRUE'.

| Attribute | Value |
|-------------------------|---|
| ParameterName | sapgui/user_scripting |
| Short description(Engl) | Enable or disable user scripting on the frontend. |
| Appl. area | GUI Frontend |
| ParameterType | Logical value |
| Changes allowed | Change permitted |
| Valid for oper. system | All operating systems |
| DynamicallySwitchable | <input checked="" type="checkbox"/> |
| Same on all servers | <input type="checkbox"/> |
| Dflt value | TRUE |
| ProfileVal | TRUE |
| Current value | TRUE |

Change value

- 3 If **ProfileVal** is FALSE, you must modify its value. To modify it, click the **Change value** button. The Change Parameter Value window opens.



The image shows a Windows-style dialog box titled "Change Parameter Value". It has a tab labeled "Parameter values". Inside the dialog, there are five text input fields, each with a label to its left: "ParameterName" (containing "sapgui/user_scripting"), "Diff value" (containing "FALSE"), "ProfileVal" (containing "FALSE"), "Current value" (containing "FALSE"), and "New value" (containing "FALSE" and highlighted in yellow). Below these fields is a checkbox labeled "Switch on all servers" which is currently unchecked. At the bottom left of the dialog are two icons: a floppy disk (Save) and a red X (Close).

| Field | Value |
|---------------|-----------------------|
| ParameterName | sapgui/user_scripting |
| Diff value | FALSE |
| ProfileVal | FALSE |
| Current value | FALSE |
| New value | FALSE |



- 4 Enter **TRUE** in the **ProfileVal** box and click the **Save** button.

Note: You must enter **TRUE** in all capital letters. Entering **True** or **true** has no effect.

When you save the change, the window closes and the value of the parameter is displayed as **TRUE**. However, this change takes effect only when you log on to the system. Therefore, before beginning to work with the QuickTest Professional Add-in for SAP Solutions, you must log off and log on again. You may also need to restart the SAP Service from the SAP Console.

If you find that even after restarting the SAP Service from the SAP Console and logging on again to the client, your change to the ProfileVal parameter was not saved, you may have an outdated kernel version. In this case, either restart the application server or download and import the required kernel patch, as specified below.

| Release | Kernel Version | Patch Level |
|---------|----------------|-------------|
| 6.10 | 6.10 | 391 |
| 6.20 | all versions | all levels |
| 6.40 | all versions | all levels |

For more information and download guidelines, refer to SAP OSS note # 480149.

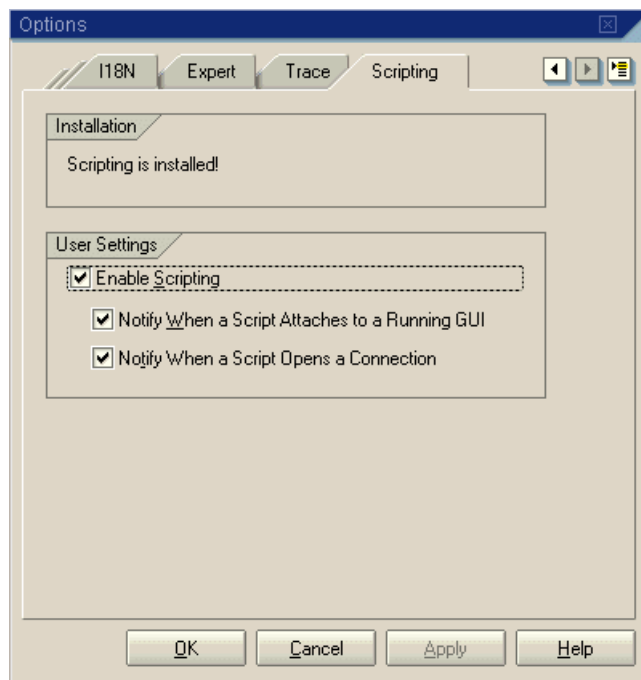
Enabling Scripting on the SAP Application (Client-Side)

To test SAP Gui for Windows applications with QuickTest Professional, you must confirm that scripting is enabled on the SAP Gui for Windows client.

It is also recommended to disable warning messages in the SAP Gui for Windows environment when working with QuickTest Professional.

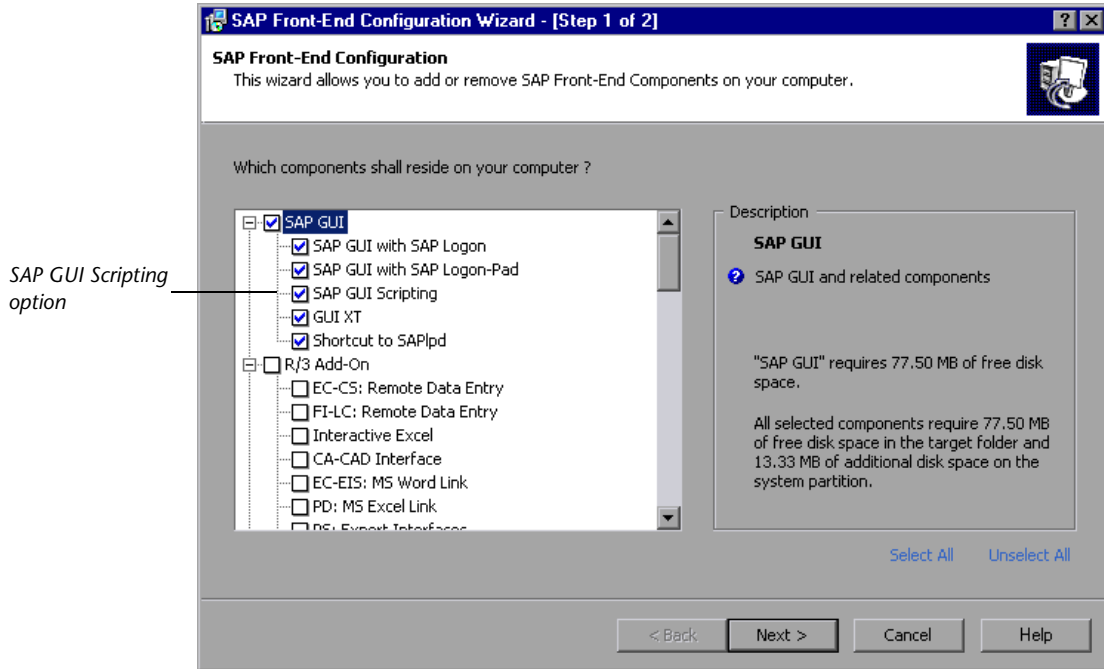
To ensure that scripting is enabled on the SAP Gui for Windows client:

- 1 Log on to your SAP server.
- 2 Click the **Customizing of Local Layout**  SAP toolbar button and then choose **Options**. The Options dialog box opens.
- 3 Click the **Scripting** tab.



- 4 Confirm that the **Enable Scripting** check box is selected. If the **Enable Scripting** check box is cleared, select it.

- 5 If the **Scripting is installed** message is not displayed in the **Installation** area, or the **Enable Scripting** check box is disabled, then the **SAP GUI Scripting** option is not installed. Reinstall your SAP Gui for Windows application and be sure to select the **SAP GUI Scripting** check box.



Eliminating Warning Messages

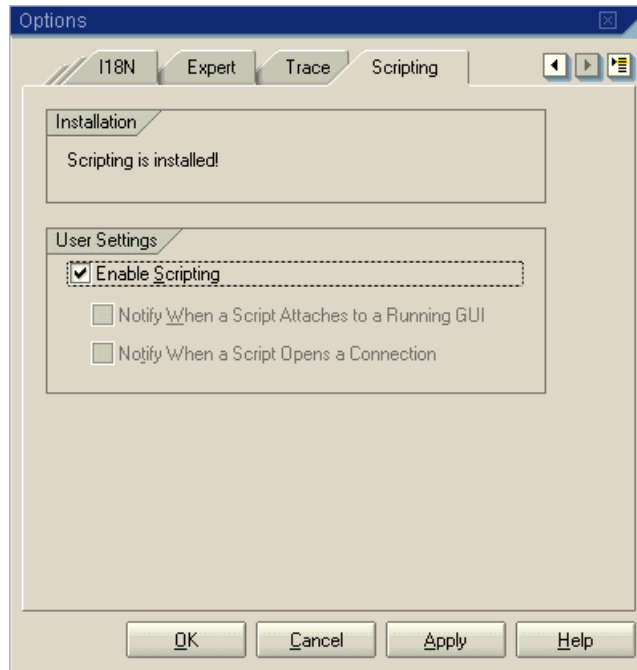
By default, you regularly receive two warning messages when using QuickTest Professional with an SAP Gui for Windows application:

- When QuickTest Professional connects to the Scripting API, the following warning message is displayed: A script is trying to attach to the gui.
- When QuickTest Professional opens a new connection using the Scripting API, the following warning message is displayed: A script is opening a connection to system <system_name>.

It is recommended to disable these warning messages in the SAP Gui for Windows application when working with QuickTest Professional.

To eliminate the display of warning messages:

- 1** Log on to your SAP server.
- 2** Click the **Customizing of Local Layout**  SAP toolbar button and then choose **Options**. The Options dialog box opens.
- 3** Click the **Scripting** tab.



- 4** Clear the **Notify When a Script Attaches to a Running GUI** and **Notify When a Script Opens a Connection** check boxes and click **OK**.

Setting F4 Help to Use Dialog Display Mode

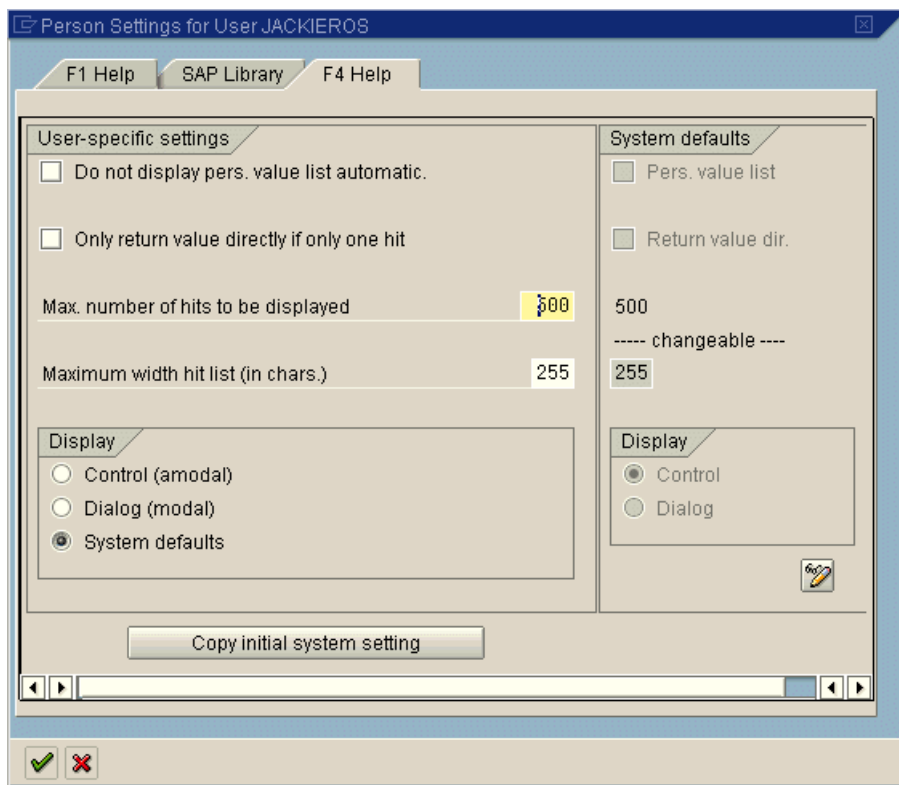
When the SAP Gui for Windows application uses the SAP Gui Scripting API (Enable Scripting option), it cannot load the F4 Help screens in **Control** mode. Therefore, you must ensure that your client is set to load F4 Help screens in **Dialog** mode.

Note: This is a per-user setting. You must set this option on each client that you want to test using the QuickTest Professional Add-in for SAP Solutions. Alternatively, the system default can be changed by the SAP system administrator.

To set F4 Help to use Dialog mode:

- 1** Log on to your SAP server.
- 2** Choose **Help > Settings** from the SAP menu bar. Your Person Settings dialog box opens.

3 Click the **F4 Help** tab.



4 In the **System defaults** section on the right side of the tab, view the **Display** setting. This setting indicates the default server setting for all clients.

- If the **Display** setting is **Dialog**, then you can set your **Display** settings in the **User-specific** section on the left side of the tab to **System defaults** or **Dialog (modal)**.
- If the **Display** setting is **Control** (as in the example above), then you must change the **Display** setting in the **User-specific** section on the left side of the tab to **Dialog (modal)**.



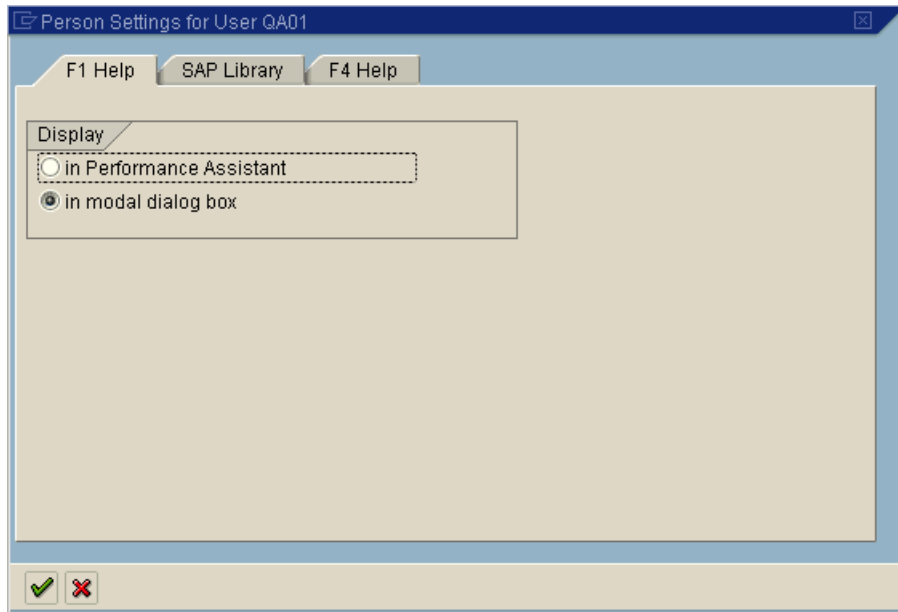
5 Click the **Copy** (**Ctrl + S**) button to save your changes and close the dialog box.

Setting F1 Help to Use Modal Dialog Box Mode

The F1 Help in your SAP Gui for Windows application can be displayed using either the Performance Assistant or as a modal dialog box. QuickTest Professional can record the displaying of the F1 Help only if the modal dialog box option is selected. If you want to include F1 Help access in your tests, you should select the **in modal dialog box** option.

To set FI Help to use modal dialog box mode:

- 1 Log on to your SAP server.
- 2 Choose **Help > Settings** from the SAP menu bar. Your Person Settings dialog box opens.
- 3 Click the **F1 Help** tab.



- 4 In the **Display** section, select **in modal dialog box**.



- 5 Click the **Copy (Ctrl + S)** button to save your changes and close the dialog box.

Checking the Connection Speed on the SAP Server

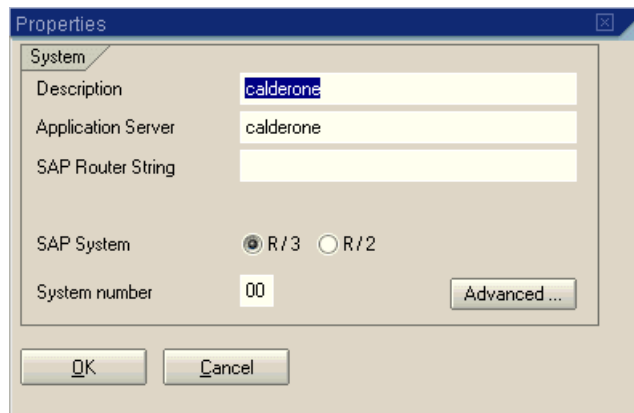
When you log on to SAP using the **Low speed connection** option to communicate with the server, the SAP server does not send sufficient information for QuickTest to properly record and run tests. (QuickTest displays an error message if the **Low speed connection** option is selected.) Therefore, confirm that this option is not selected for the server to which you are connecting before recording and running QuickTest tests.

For more information, refer to SAP OSS note #587202.

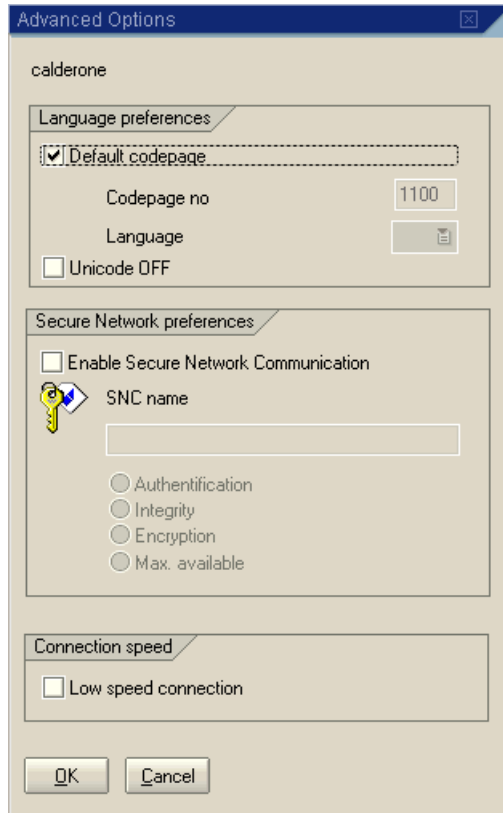
Note: Depending on the SAP Gui for Windows version you are working with, the dialog boxes shown in this section may or may not appear the same as those on your screen. However the instructions in this section are the same for all supported SAP Gui for Windows versions.

To check the connection speed setting on your SAP Gui for Windows client:

- 1** Open the SAP Logon dialog box and select the server to which you want to connect.
- 2** According to the version you are using, either click the **Properties** button or right-click a server and choose **Properties**. The Properties dialog box for the selected server opens.



- 3 Click the **Advanced** button. The Advanced Options dialog box opens.



- 4 In the **Connection speed** section, confirm that the **Low speed connection** check box is cleared.
- 5 Repeat steps 1 to 4 for each server you want to use in conjunction with QuickTest.

4

Recording and Running Tests and Components on SAP Gui for Windows Applications

The QuickTest Professional Add-in for SAP Solutions supports testing on SAP Gui for Windows applications. This chapter explains how to use QuickTest to record and run tests and components on SAP Gui for Windows applications. For more information about working with QuickTest, refer to the *QuickTest Professional User's Guide*. For information about recording and running tests and components on Web-based SAP applications, see “Recording and Running Tests or Components on Web-based SAP Applications” on page 133.

Note: The QuickTest Active Screen, Record and Run Settings dialog box, and auto-parameterization options described in this chapter are relevant only for tests.

This chapter describes:

- About Recording and Running Tests and Components on SAP Gui for Windows Applications
- Before You Create SAP Gui for Windows Tests and Components
- Understanding the SAP Windows Test Object Model
- Recording Tests and Components on SAP Gui for Windows Applications
- Configuring Active Screen Capture Options for SAP Gui for Windows Applications


- Configuring Testing Options for SAP Gui for Windows Applications
- Understanding Low-Level or Analog Mode Recording on SAP Gui for Windows
- Using Standard Windows Recording Capabilities
- Running Tests and Components on SAP Gui for Windows Applications

About Recording and Running Tests and Components on SAP Gui for Windows Applications


QuickTest Professional can record and run tests and components on SAP Gui for Windows applications using special test objects and methods that have been customized for SAP.


As you record a test or component on your SAP Gui for Windows application, QuickTest records the operations you perform. QuickTest works directly with the SAP Gui Scripting API to record your operations. Therefore, although QuickTest records a step for each operation you perform, it adds the steps to your test only when API events are sent to QuickTest (when information is sent to the SAP server).

The QuickTest Professional Add-in for SAP Solutions recognizes special SAP Windows objects such as OKCode edit boxes, table and grid controls, simple, list, and column trees, and toolbars. For more information on the supported objects, see “Adding SAP Windows Statements to Your Test or Component” on page 113.

Notes:  You can also record in low-level recording mode to record operations in terms of coordinates instead of objects. For more information, see “Understanding Low-Level or Analog Mode Recording on SAP Gui for Windows” on page 88.

You can also record specific operations in your SAP Gui for Windows Application in Standard Windows recording mode, if required. For more information, see “Using Standard Windows Recording Capabilities” on page 88.

 Each time you begin recording a test, you can use the SAP tab of the Record and Run Settings dialog box to instruct QuickTest to connect to a specified SAP server. Alternatively, you can instruct QuickTest to record on any open SAP session. For more information, see “Recording Tests and Components on SAP Gui for Windows Applications” on page 64.

 The Record and Run Settings dialog box is not used for components. When you record a component on an SAP Gui for Windows session, you cannot instruct QuickTest to open or connect to a specified SAP server. You must open and connect to it manually or include statements in your component (using the **SAPGuiUtil** test object) that open and connect to the SAP server.



You can use the Active Screen (for tests and scripted components only) and the SAP tab of the Options dialog box to set additional preferences that control the way QuickTest records and runs your test. For more information, see “Configuring Active Screen Capture Options for SAP Gui for Windows Applications” on page 73 and “Configuring Testing Options for SAP Gui for Windows Applications” on page 76.

You run tests or components containing SAP Windows test objects in the same way as you run any other test or component. You can view the details of your run session in the Test Results window.

Before You Create SAP Gui for Windows Tests and Components

Before you begin to design tests or components on your SAP Gui for Windows application, confirm that you have properly configured your SAP server and client as described in Chapter 3, “Setting Up Your SAP Gui for Windows Environment”.

As you prepare to create your test or component, keep in mind the following guidelines:

- QuickTest works directly with the SAP Gui Scripting API to record your operations. Therefore, although QuickTest records a step for each operation you perform, it adds the steps to your test only when API events are sent to QuickTest (when information is sent to the SAP server).
-  The Record and Run Settings dialog box in QuickTest enables you to specify a server and client to open at the beginning of every test record and run session. The servers available in the dialog box are the same as those available in the SAP Logon Pad and SAP Logon dialog box.
-  When you record a component on an SAP Gui for Windows session, the Record and Run Settings dialog box is not available. Instead, you need to open the SAP session manually or include statements in your component that connect to the SAP server (using the **SAPGuiUtil** test object).
- The steps that QuickTest records are based on the events sent by the SAP Gui Scripting API. Therefore, you may see steps in your test or component that do not directly correspond to an operation you performed, or an operation you performed may be shown as two or more steps. For more information on the SAP Gui Scripting API events, refer to your SAP documentation.

Understanding the SAP Windows Test Object Model

The *test object model* is the set of object types or *classes* that QuickTest uses to represent the objects in your application. Each test object class has a list of properties that can uniquely identify objects of that class and a set of relevant methods that QuickTest can perform during a run session.

A *test object* is an object that QuickTest creates in the test or component to represent the actual object in your application and to store information about that object. This information, stored in the object repository (separate or shared), helps QuickTest identify and check the object during a run session.

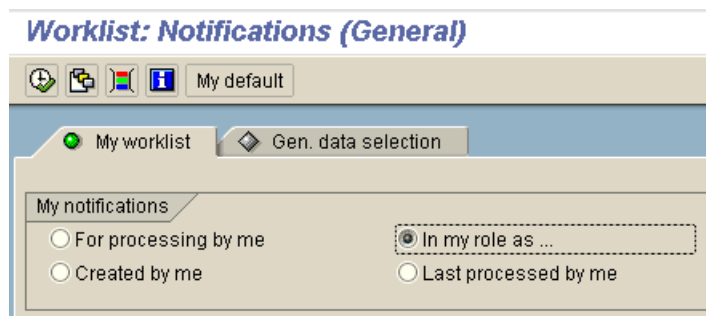
A *run-time object* is the actual object in your SAP Gui for Windows application on which methods are performed during a run session.

When you perform an operation on your SAP Gui for Windows application while recording a test or component, QuickTest:

- identifies the run-time object on which you performed the operation and creates the corresponding test object.
- reads the current property values of the object in your application and stores them in the object repository as the test object's property values.
- chooses a unique name for the object, generally using the value of one of its prominent properties.
- records the operation (method) that you performed on the object.
- If the operation you performed resulted in an information exchange with the SAP server, the steps performed since the last information exchange are added as steps in the Keyword View and statements in the Expert View.

Note: Each time information is exchanged with the SAP server, the steps you recorded up to that point are added to your test or component. Therefore, while each operation you perform has a corresponding step in your test or component, you may perform several operations before the corresponding steps are actually added to your test or component.

For example, suppose you select a radio button in the **My worklist** tab of your SAP Gui for Windows application. This radio button is labeled **In my role as...**



QuickTest uses the SAP Gui component type (41) to identify the object as an SAPGuiRadioButton object. It creates an SAPGuiRadioButton test object with the name **In my role as...** and records the following properties and values as the description for the radio button.

| Type | Property | Value |
|------|------------------|-------------------|
| ABC | guicomponenttype | 41 |
| ABC | name | MEL_ROLE |
| ABC | text | In my role as ... |

Note: The **guicomponenttype** and **name** property values are supplied by the SAP Gui Scripting API.

QuickTest also records that you performed a **Set** method to turn ON the radio button.

QuickTest displays your step in the Keyword View like this:

| | | | |
|-------------------------|----------|--------|--|
| Worklist: Notifications | Resize | 135,25 | Resize the "Worklist: Notifications" window to 135 by 25 characters. |
| In my role as ... | Set | | Select the "In my role as ..." radio button. |
| In my role as ... | SetFocus | | Set the focus on the "In my role as ..." radio button. |

QuickTest displays your step in the Expert View like this:

```
SAPGuiSession("Session").SAPGuiWindow("Worklist: Notifications").  
  SAPGuiRadioButton("In my role as ...").Set
```

When you run a test or component, QuickTest identifies each object in your application by its test object class and its *description*—the set of test object properties and values used to uniquely identify the object. In the above example, during the run session, QuickTest looks up the description for the `SAPGuiRadioButton` object with the name **In my role as...** by searching the object repository. QuickTest finds the following description:

```
guicomponenttype = 41  
name = MEL_ROL  
text = In my role as...
```

QuickTest then looks in the application for an `SAPGuiRadioButton` object that matches the above description. When it finds the object, it performs the **Set** method on it to change the value of the field to ON (selects the radio button).

Tip: You can use Object Spy to view all of the properties and methods of any object in your SAP Gui for Windows application.

Note: The QuickTest Professional Add-in for SAP Solutions provides an alternative recording mechanism for specific SAP Gui for Windows objects that do not have built-in test object support. For more information, see “SAP Gui for Windows Alternative Recording Mechanism” on page 127.

For more information on the test object model, the object repository, and the Object Spy, refer to the *QuickTest Professional User's Guide*.

For more information on SAP Gui for Windows test objects and methods, see the **SAP Windows** section of the *QuickTest Object Model Reference*.

Recording Tests and Components on SAP Gui for Windows Applications


You record steps on your SAP Gui for Windows application by activating the Record option in QuickTest and then performing operations on your application just as you normally do.



When you begin recording a new component on an SAP Gui for Windows application (by clicking **Record** or choosing **Component > Record**), the Applications dialog box opens (unless you previously specified a Windows environment in the Application Area Settings or Business Component Settings dialog box). Simply close the dialog box to begin recording.









When you begin recording a new test, the Record and Run Settings dialog box opens automatically (unless you open the dialog box and set your preferences manually before you begin recording). You can also open it by selecting **Test > Record and Run Settings**. You can use it to define certain settings. For example, when you record a test, you can instruct QuickTest to connect to a specific SAP server. For more information, see “Setting Record and Run Settings for SAP Gui for Windows Tests” on page 66.

Note:  Components do not require specific record and run settings to work with SAP Gui for Windows applications. To record a component on an SAP application, you need to first open the SAP session manually or include steps in your component that connect to the SAP server. When you begin recording a new component, the Applications dialog box opens (unless you previously specified a Windows environment in the Application Area Settings or Business Component Settings dialog box). Simply close the dialog box to begin recording. For more information on the Applications tab and Applications dialog box, refer to the *QuickTest Professional User's Guide*.

You can use the Active Screen tab (for tests only) and the SAP tab of the Options dialog box to set additional preferences that control the way QuickTest records and runs your test or component. For more information, see “Configuring Active Screen Capture Options for SAP Gui for Windows Applications” on page 73 and “Configuring Testing Options for SAP Gui for Windows Applications” on page 76.


QuickTest works directly with the SAP Gui Scripting API to record your operations. Therefore, QuickTest adds steps to your test or component only when API events are sent to the server. This means that while recording a test or component, you may perform several operations on your application before the corresponding steps are added. When you perform a step that sends information to the server, QuickTest inserts steps with the relevant SAP Windows objects in the Keyword View (tests and components) and adds corresponding statements in the Expert View (tests only).

For example, suppose you record the steps of filling in a Price Simulation for Material form. You select the three check boxes in the form (**Incl. cash discount**, **Delivery costs**, and **Effective price**) and click **Continue**. When you click the **Continue** button, information is sent to the SAP server, and the steps in which you select the check boxes and click the **Continue** button are added to your test at once. In the Keyword View, the process described above is displayed as follows.

| | | | |
|---|----------|------|---|
|  Price Simulation for Material | | | |
|  Incl. cash discount | Set | "ON" | Set the state of the "Incl. cash discount" check b |
|  Delivery costs | Set | "ON" | Set the state of the "Delivery costs" check box to |
|  Effective price | Set | "ON" | Set the state of the "Effective price" check box to |
|  Effective price | SetFocus | | Set the focus on the "Effective price" check box. |
|  Continue (Enter) | Click | | Click the "Continue (Enter)" button. |


QuickTest records these steps in the Expert View as follows:

```
SAPGuiSession("Session").SAPGuiWindow("Price Simulation for
Material").SAPGuiCheckBox("Incl. cash discount").Set "ON"
SAPGuiSession("Session").SAPGuiWindow("Price Simulation for
Material").SAPGuiCheckBox("Delivery costs").Set "ON"
SAPGuiSession("Session").SAPGuiWindow("Price Simulation for
Material").SAPGuiCheckBox("Effective price").Set "ON"
SAPGuiSession("Session").SAPGuiWindow("Price Simulation for
Material").SAPGuiCheckBox("Effective price").SetFocus
SAPGuiSession("Session").SAPGuiWindow("Price Simulation for
Material").SAPGuiButton("Continue (Enter)").Click
```

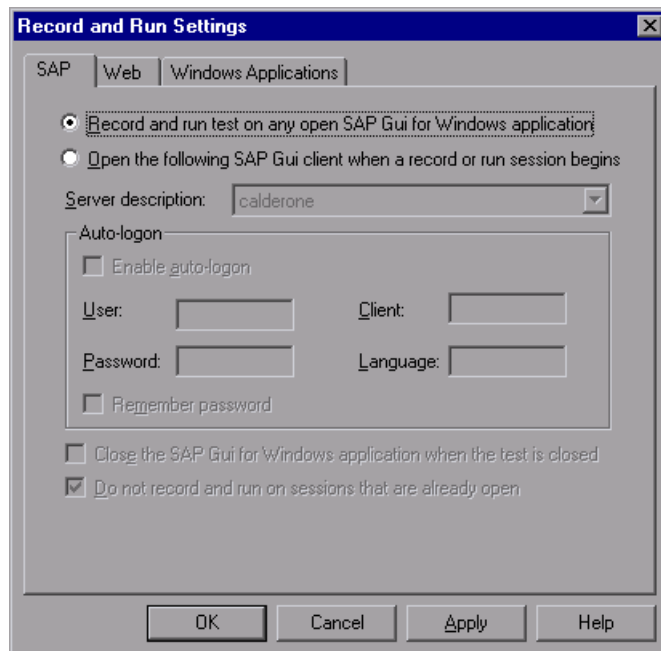
Note:  When you select a step in the Keyword View or the Expert View, the corresponding object is highlighted in the Active Screen (unless you chose not to capture Active Screen information when you recorded your test). However, the values of the object properties stored with the Active Screen are the values of the properties at the time that the steps were added to the test (when you performed the step that sent information to the SAP server). These values may potentially be different than the values of the properties at the time that the selected step was actually performed.

For more information on Active Screen capture levels, see “Configuring Active Screen Capture Options for SAP Gui for Windows Applications” on page 73.

Setting Record and Run Settings for SAP Gui for Windows Tests

 You can use the SAP tab of the Record and Run Settings dialog box to instruct QuickTest to connect to a particular SAP server and open your SAP Gui for Windows application each time you begin a recording session. Alternatively, you can instruct QuickTest to record on any open SAP Gui for Windows application.

If you do not modify the Record and Run settings before you begin recording, the Record and Run Settings dialog box opens automatically when you begin recording a new test (by clicking **Record** (or choosing **Test > Record**). You can also open it by choosing **Test > Record and Run Settings**.



If you load only the QuickTest Professional Add-in for SAP Solutions and the Web add-in, then only the SAP, Web, and Windows Applications tabs are displayed in the Record and Run Settings dialog box (as shown above). If other add-in(s) are loaded, the corresponding tabs (if any) are also displayed.

You can use the SAP tab to instruct QuickTest to connect to a specified SAP server and open your SAP Gui for Windows application using specified user settings. Alternatively, you can instruct QuickTest to record and run the test on any open SAP Gui for Windows application. If you select to connect to a specific server, you can specify details that will enable QuickTest to automatically log on to the server each time a record or run session begins (instead of recording the log on steps).

You can also use application details environment variables to specify these parameters. For more information, see “Defining Application Details Environment Variables” on page 71.

Note: If you begin a new recording session on an existing test, the Record and Run Settings dialog box does not open. Instead, QuickTest uses the existing record and run settings for the test. You can change the settings by choosing **Test > Record and Run Settings**. Before running the test, though, confirm that the options in the SAP tab are appropriate for the first step of your test.

The SAP tab includes the following options:

| Option | Description |
|--|--|
| Record and run tests on any open SAP Gui for Windows application | <p>Instructs QuickTest to use any open SAP Gui for Windows application to record and run the test.</p> <p>Note: This option supports sessions opened using the SAP Logon dialog box or the SAP Logon Pad.</p> |
| Open the following SAP Gui client when a record or run session begins | <p>Instructs QuickTest to connect to the specified server.</p> |
| Server description | <p>Indicates the server to which you want to connect.</p> <p>Note: The Server description box lists the servers available in the SAP Logon Pad or the SAP Logon dialog box.</p> <p>To add a server to the list in the Record and Run Settings dialog box, close the Record and Run Settings dialog box, define an appropriate entry using your SAP Logon dialog box, and then reopen the Record and Run Settings dialog box.</p> |

| Option | Description |
|--------------------------|---|
| Enable auto-logon | Instructs QuickTest to open the specified SAP Gui for Windows application using the specified logon details. Enabled only when Open the following SAP Gui client when a record or run session begins is selected. |
| User | The user name used to log on to the specified server. Enabled only when Enable auto-logon is selected. |
| Password | The password for the specified user name. Enabled only when Enable auto-logon is selected. |
| Client | The client number. Enabled only when Enable auto-logon is selected. |
| Language | The language that you want the specified SAP Gui for Windows application to display. Enabled only when Enable auto-logon is selected. |
| Remember password | Saves the password information in this dialog box so that you do not have to enter it each time you begin to record or run the test. Enabled only when Enable auto-logon is selected. |

| Option | Description |
|--|---|
| Close the SAP Gui for Windows application when the test is closed | <p>Instructs QuickTest to close the SAP Gui for Windows session specified in the Record and Run dialog box when the test is closed.</p> <p>Any other SAP Gui for Windows session that was opened before, during, or after the run session is not affected.</p> <p>The Session cleanup option in the SAP tab of the Options dialog box (Tools > Options > SAP) overrides this option. For more information, see “Configuring Testing Options for SAP Gui for Windows Applications” on page 76.</p> |
| Do not record and run on sessions that are already open | <p>Instructs QuickTest not to record or run tests on any SAP Gui for Windows sessions that were already open prior to the start of the record or run session. This is to ensure that steps are not inadvertently recorded on other SAP Gui for Windows sessions that may also be running on the same computer.</p> |

Note: In addition to saving all values set in the Record and Run Settings dialog box with the test, the values you enter for the **User**, **Client**, **Password**, and **Language** in the **Auto-logon** area of the dialog box are saved with the selected server. If you select the same server in the **Server description** box for a new test, the saved values are automatically displayed in the auto-logon area.

Defining Application Details Environment Variables



You can use application details environment variables to specify the applications you want to use for recording and running your test. These variables can also be used in external library files for automation scripts.

If you define any of these application details environment variables, they override the values in the **Server description**, **User**, **Password**, **Client**, and **Language** boxes in the SAP tab of the Record and Run Settings dialog box. For more information, see “Setting Record and Run Settings for SAP Gui for Windows Tests” on page 66.

Use the variable names listed in the table below to define SAP application details:

| Option | Variable Name | Description |
|---------------------------|------------------|--|
| Server description | SAP_SERVER_ENV | The description of the server to which you want to connect. |
| User | SAP_USER_ENV | The user name used to log on to the specified client number. |
| Password | SAP_PASSWORD_ENV | The encrypted password for the specified user name. |
| Client | SAP_CLIENT_ENV | The client number. |
| Language | SAP_LANGUAGE_ENV | The language that you want the specified SAP Gui for Windows application to display. |

For more information on defining and working with environment variables, refer to the *QuickTest Professional User's Guide*.

Optimizing Settings for Other Record and Run Settings Dialog Box Tabs



In addition to setting the appropriate settings in the SAP tab, confirm that the other tabs in the dialog box have the appropriate settings. The following settings are recommended:


- **Windows tab**—Choose **Record and run on these applications (opened when a session begins)** and confirm that the list of Windows applications is empty.
- **Web and other tabs (if displayed)**—Choose the option to record and run on any open browser/application (upper radio button of each tab).

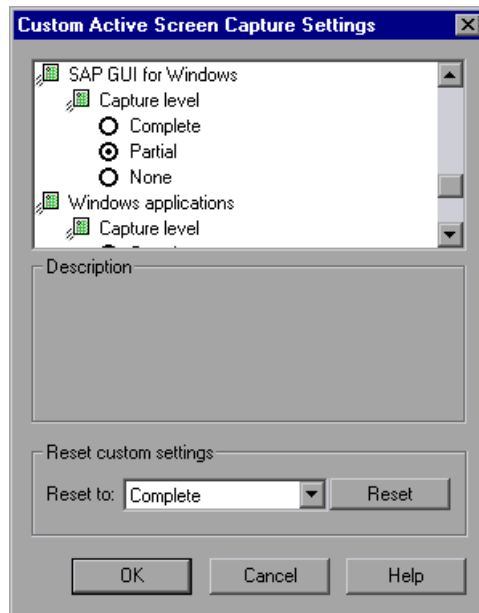
Note: You use the Web tab to record on SAP Gui for HTML or SAP Enterprise Portal applications. For more information, see “Recording Tests on Web-based SAP Applications” on page 137.

While these settings do not directly affect your record or run sessions when working with SAP Gui for Windows applications, these settings prevent you from inadvertently recording operations performed on Windows applications (such as e-mail) during your recording session. These settings also prevent QuickTest from opening unnecessary applications when you record or run tests on SAP Gui for Windows applications.

For more information on the Record and Run settings dialog box, refer to the *QuickTest Professional User's Guide*.

Configuring Active Screen Capture Options for SAP Gui for Windows Applications

 The Active Screen tab of the Options dialog box (**Tools > Options > Active Screen**) enables you to specify Active Screen capture settings for all environments. The Custom Active Screen Capture Settings dialog box (**Tools > Options > Active Screen > Custom Level**) enables you to customize how QuickTest captures and saves Active Screen information for specific environments.



Note: Changes made to settings in this dialog box for a specific environment override those same settings in the Active Screen tab for that specific environment. Changing the capture level in the Active Screen tab from **Custom** to a specific level changes the level for all environments accordingly.

In addition to the core options described in the *QuickTest Professional User's Guide*, the Custom Active Screen Capture Settings dialog box contains the following SAP Gui for Windows-specific options:

| Option | Description |
|------------------------|--|
| <p>Complete</p> | <p>Instructs QuickTest to save the property values of all objects in the application's open window/dialog box in the Active Screen of each step.</p> <p>This option makes it possible for you to insert checkpoints and perform other operations on any object in the window/dialog box from the Active Screen of any step. However, it may result in longer recording times and require more disk space.</p> <p>Note: The properties for inner objects of some container objects (such as table cells or tree nodes) are not captured in the Active Screen.</p> <p>Use the appropriate SAPGuiTable or SAPGuiTree methods to access information for these objects. For more information, refer to the SAP Windows section of the <i>QuickTest Professional Object Model Reference</i>.</p> |
| <p>Partial</p> | <p>Default. Instructs QuickTest to save properties of the recorded object and of its parent in the Active Screen of each step.</p> <p>This option enables speedy recording and requires relatively little disk space. However, you can insert checkpoints and perform other operations only on the recorded object and on the window/dialog box itself. You cannot perform operations on the other objects displayed in the Active Screen.</p> |

| Option | Description |
|--------|--|
| None | <p>Disables the capture of Active Screen files for SAP Gui for Windows applications.</p> <p>This option allows extremely fast recording and requires only minimum disk space. However, you cannot perform post-recording test editing (such as inserting checkpoints, output values, and so forth) from the Active Screen.</p> |

Notes:

The property values of the objects in the Active Screen reflect the values at the time that the steps are added to your test (when information is sent to the SAP server). These values may potentially be different than the property values at the time that a particular step is performed.

The Active Screen captures only the visible part of the SAP Gui for Windows applications window at the time that the step is added to the test or component.

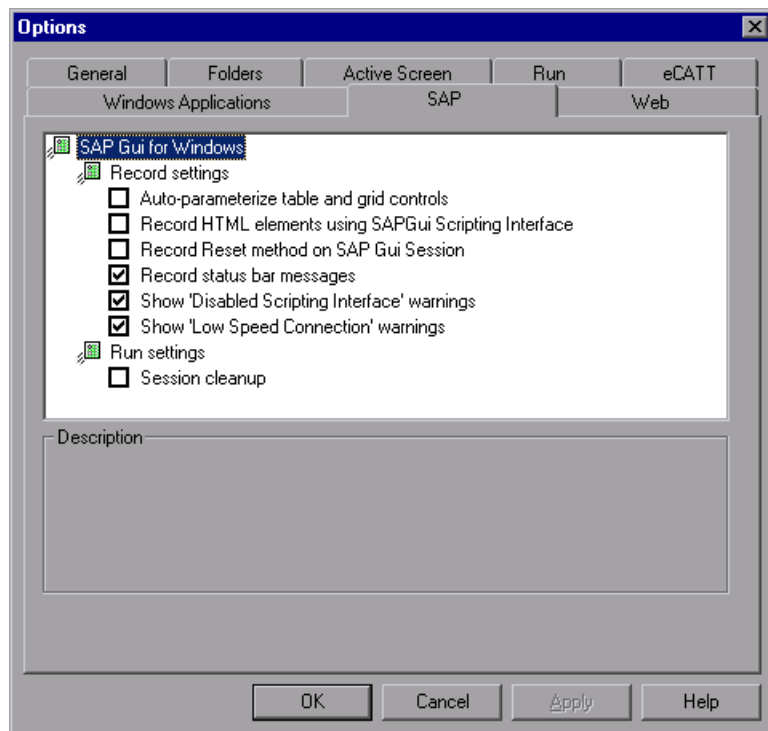
The more information saved in the Active Screen, the easier it is to edit the test or component after it is recorded. However, saving more information in the Active Screen adds to the recording time and to the amount of required disk space.

Tip: You can use the **Save Active Screen files** option in the Save dialog box or use the **Update Run** option to increase or decrease the amount of information saved in the Active Screen after you record the test. For more information, refer to the *QuickTest Professional User's Guide*.

Note: While you are recording or running components, the Active Screen is not available.

Configuring Testing Options for SAP Gui for Windows Applications


The SAP tab of the Options dialog box (**Tools > Options > SAP**) enables you to configure how QuickTest records and runs tests and components on SAP applications.



Note: The SAP tab is available only when the QuickTest Professional Add-in for SAP Solutions is installed and loaded.

The options in this tab apply only to steps performed on SAP Gui for Windows applications.

The SAP tab contains the following options:


| Option | Description |
|--|--|
|  Auto-parameterize table and grid controls | While recording, automatically captures the data you set in table and grid cells and stores it in a new data sheet in the Data Table. QuickTest inserts an Input statement into your test, which refers to the new data sheet. Using this option enables you to set values of multiple cells in a single test step and easily parameterize the cell values. For more information, see “Using the Auto-Parameterize Option to Parameterize Table and Grid Cell Values” on page 80. |
| Record HTML elements using SAPGui Scripting interface | Specifies whether QuickTest should use the SAP Gui Scripting API when recording HTML elements within SAP applications, or use the built-in Web support to record these HTML elements. You can use this option to handle synchronization issues that may arise from recording on Web elements inside an SAP Gui for Windows session. Note: Changes to this option take effect only after you start recording a new test or component. |

| Option | Description |
|---|--|
| Record Reset Method on SAP Gui Session | <p>Records a Reset method as the first step when recording a test or component. When the test or component is run, the first step resets the application session so that it starts at the initial SAP screen. This ensures that the test or component starts with the same application state each time it is run.</p> <p>Note: This option is applicable only if the Open the following SAP Gui client when a record or run session begins and the Auto-logon options are both selected in the SAP tab of the Record and Run Settings dialog box.</p> |
| Record status bar messages | <p>Records a SAPGuiStatusbar.Sync step each time a status bar message is displayed in the SAP application.</p> <p>Note: This option is selected by default.</p> |
| Show 'Disabled Scripting Interface' warnings | <p>Instructs QuickTest to display warnings if the SAP Gui Scripting API is disabled on the SAP application. If this is the case, you cannot record or run steps until you enable the SAP Gui Scripting API. For information on enabling the SAP Gui Scripting API, see “Enabling Scripting on the SAP Application (Client-Side)” on page 49.</p> <p>Note: This option is selected by default.</p> |

| Option | Description |
|---|--|
| Show 'Low Speed Connection' Warnings | <p>Instructs QuickTest to display warnings if the connection speed to the server is set to Low speed connection.</p> <p>Note: This option is selected by default.</p> <p>If this option is selected, one of the following occurs if the connection speed is low:</p> <ul style="list-style-type: none"> • If the session was opened by QuickTest when recording started (as defined in the Record and Run Settings dialog box), the error message appears and the recording stops. • If the session was opened by the user before recording started, the error message appears and recording continues in Standard Windows mode. For information on Standard Windows Recording mode, see “Using Standard Windows Recording Capabilities” on page 88. <p>If this option is not selected, one of the following occurs if the connection speed is low:</p> <ul style="list-style-type: none"> • If the session was opened by QuickTest when recording started (as defined in the Record and Run Settings dialog box), the error message does not appear and the recording stops. • If the session was opened by the user before recording was begun, the error message does not appear and recording continues in Standard Windows mode. For information on Standard Windows Recording mode, see “Using Standard Windows Recording Capabilities” on page 88. <p>The connection speed can be checked using the SAP client. For more information, see “Checking the Connection Speed on the SAP Server” on page 55.</p> |

| Option | Description |
|------------------------|--|
| Session cleanup | <p>Instructs QuickTest to close all SAP Gui for Windows sessions opened by QuickTest during the current run session when the test is closed. This includes all SAP Gui for Windows sessions that were invoked from the Record and Run dialog box, plus any sessions that may have been invoked during the run session using a SAPGuiUtil statement or the Open New Session button in the SAP Gui for Windows application that was being recorded.</p> <p>SAP Gui for Windows sessions that were opened during a previous run session, or opened manually before or during the current run session are not affected.</p> <p>This option overrides the Close the SAP Gui for Windows application when the test is closed option in the SAP tab of the Record and Run Settings dialog box. For more information, see “Setting Record and Run Settings for SAP Gui for Windows Tests” on page 66.</p> |

Using the Auto-Parameterize Option to Parameterize Table and Grid Cell Values

 By default, QuickTest records a **SetCellData** statement each time you modify the value of a cell in a table or grid. However, if you want to modify the values of several cells in a single table or grid, and then parameterize your test so that different values are entered into the cells each time your test action runs, you would have to parameterize each statement individually.

Alternatively, you can enable the **Auto-parameterize table and grid controls** option. When this option is selected, QuickTest automatically captures all values you set for a particular table or grid during a recording session and stores them in a special data sheet in the Data Table. QuickTest inserts a single **SAPGuiTable.Input**, **SAPGuiGrid.Input**, or **SAPGuiAPOGrid.Input** statement into your test, which refers to this new data sheet. Before running the test, you can easily modify the values or add additional sets of data to the data sheet for each action iteration.

To record in auto-parameterize mode:

- 1** Choose **Tools > Options** and click the **SAP** tab.
- 2** Select the **Auto-parameterize table and grid controls** option.
- 3** Begin a recording session.
- 4** Set the value of one or more cells in a table or grid.
- 5** Press the ENTER key or perform another operation that sends data to the SAP server.

For information on auto-parameterization and the **Input** statement, see:

- “Understanding How QuickTest Records in Auto-Parameterize Mode” on page 81
- “Parameterizing Cell Values in the Input Data Sheet” on page 84
- “Working with Auto-Parameterization—Tips and Guidelines” on page 85
- “Entering Data in Rows Requiring Scrolling” on page 87

Understanding How QuickTest Records in Auto-Parameterize Mode



When you record with the **Auto-parameterize table and grid controls** option and you perform an operation that sends data to the SAP server after setting table or grid cell values, QuickTest:

- Creates a new data sheet to represent the table or grid. Each data sheet is a sub-sheet of the action in which the table or grid operations were recorded. The data sheet name is always the action name followed by a period (.) and the internal name of the table or grid. For example: Action1.FLIGHT_TABLE
- Adds a column to the data sheet for each table or grid column in which you record. (Columns in which you did not set any cell data are not added to the data sheet.)

The name of the column in the data sheet is generally the same as the name of the column in your application.

If a column in the application does not have a header, or more than one column header has the same name, QuickTest inserts a column with a name in the format: **__<index>**, where **<index>** represents the column number according to its location when you record the **Input** step.

- Inserts the values you set during the recording session into the appropriate cells in the data sheet. Each row in which you entered data is represented by a row in the data sheet. Place-holder (empty) rows are added for rows above the rows in which you recorded. For example, if you set data in rows 2, 4, and 7, seven rows are added to the data sheet. The cells in rows 1, 3, 5, and 6 do not contain any data.
- Inserts an additional **end row** where the value of the first cell in the row is .END

| | Airline_carrier | Flight_number |
|---|-----------------|---------------|
| 1 | AB | |
| 2 | AB | |
| 3 | | 553 |
| 4 | END | |

End Row

- Inserts an **Input** *DataSetName* statement (followed by a **SelectCell** statement) into your test.

| | | | | |
|-----|------------------------|------------|------------------------------------|---------------------------|
| SAP | Table control tc spfli | Input | "Action1.Table control tc spfli_3" | Add a sheet named "Ac |
| SAP | Table control tc spfli | SelectCell | 1,"Airline" | Select the cell in row 1, |
| SAP | | SendKey | ENTER | Press the ENTER keyb |

QuickTest records these steps in the Expert View as follows:

```
SAPGuiSession("Session").SAPGuiWindow("SAP").SAPGuiTable("Table
control tc spfli").Input "Action1.Table control tc spfli_3"
```

```
SAPGuiSession("Session").SAPGuiWindow("SAP").SAPGuiTable("Table
control tc spfli").SelectCell 1,"Airline"
```

```
SAPGuiSession("Session").SAPGuiWindow("SAP").SendKey ENTER
```

The **Input** statement instructs QuickTest to enter values from the data sheet into the table or grid corresponding to the data sheet name. This **Input** statement is, in essence, an automatically parameterized statement referring to a special sheet in the Data Table.

Suppose you update values in a table control containing airline flight information. For example, you update some airline codes, add state and country names to some of the departure and destination cities, update one of the destination airport codes, and update some of the departure times. The edited table in your application may look something like this.

| Ai... | Fli... | Depart.city | D... | Destination cit | D... | Flight ti... | Departur |
|-------|--------|-------------------|------|-------------------|------|--------------|----------|
| AB | 17 | NEW YORK CITY | LGA | SAN FRANCISCO, CA | SFO | | 13:30:0 |
| AB | 64 | SAN FRANCISCO, CA | SFO | NEW YORK | JFK | | 09:35:0 |
| AZ | 553 | ROME, ITALY | FCO | FRANKFURT | FRK | | 19:00:0 |
| AZ | 788 | ROME, ITALY | FCO | TOKYO, JAPAN | TYO | | 12:00:0 |
| AZ | 789 | TOKYO, JAPAN | TYO | ROME, ITALY | FCO | | 11:45:0 |
| AZ | 790 | ROME, ITALY | FCO | OSAKA | KIX | | 10:35:0 |

QuickTest inserts the following **Input** statement in your test to represent the data input:

```
SAPGuiSession("Session").SAPGuiWindow("SAP R/3").SAPGuiTable("SPFLI").
    Input "Action1.SPFLI"
```

Note: If you record on a table or grid that scrolls using the ENTER key rather than the PAGEDOWN key, you may need to manually add the **ScrollMethod** optional argument. For more information, see “Entering Data in Rows Requiring Scrolling” on page 87.

The corresponding data sheet in your Data Table looks like this:

| | | | | | | |
|-------------------------------|-----------------|-------------------|--------------|-----------------|---------------|-----------|
| A7 | END | | | | | |
| | Airline_carrier | Depart.city | Dep. airport | Destination_cit | Dest. airport | Departure |
| 1 | AB | NEW YORK CITY | LGA | SAN FRANCISCO | | |
| 2 | AB | SAN FRANCISCO, CA | | | | 09:35:00 |
| 3 | | ROME, ITALY | | | FRK | |
| 4 | | ROME, ITALY | | TOKYO, JAPAN | | |
| 5 | | TOKYO, JAPAN | | ROME, ITALY | | |
| 6 | | ROME, ITALY | | | | |
| 7 | END | | | | | |
| 8 | | | | | | |
| Global Action1 Action1. SPFLI | | | | | | |

There are six rows in the data sheet, because data was modified in the first six rows of the table or grid in the application. Note that the data sheet does not contain columns for the **Flight Number** and **Flight time** columns, because no values were modified in those columns during the recording session.

Parameterizing Cell Values in the Input Data Sheet



After you record an **Input** statement to create an input data sheet, you can modify the values to be used in the run session, and you can create multiple sets of table or grid cell data to be used in different iterations of an action.

As described above, when you record the **Input** statement, QuickTest records the values you set in the appropriate rows and columns in the input data sheet for that table or grid. Below the data it adds an end row (shaded in blue) with the text **.END** in the first cell of the row. This row indicates the end of the first set of data for the table or grid. This set of data and its corresponding end row represents a single *data set*.

To supply different data values for each action iteration, you add new data sets. You add a new data set for a table or grid by entering the values in the appropriate rows and columns below the previous end row. To indicate the end of the new data set, copy and paste the end row from the first set of data to the row below the new set of data. You can include a different number of rows in each data set.

Note: The **Input** statement can run successfully only if it can find the end row. Therefore, the first cell of the end row, must contain only the text **.END**. You can enter text into other cells in that row, if needed. For example, you can enter a number in the second cell of the end row to indicate the iteration number corresponding to that set of data.

Because the input data sheets are added as a sub-sheet of the current action, the **Input** statement uses the data set corresponding to the current action iteration. For example, if you set the action to run on all iterations and your action sheet includes five rows of data, then your input data sheet should also include five data sets (and five **.END** rows).

The input data sheet below contains three sets of data. The first set contains data for the top three rows of the table or grid. The second set contains data for the top two rows of the table or grid. The third set contains data for rows 2-5. The blank first row (row 8 in the data sheet), indicates that no data should be entered or modified in the first row of the table or grid. Note that a number was manually entered into the second cell of each **END** row to make it easier to identify the action iteration to which each data set corresponds.

| | Airline_carrier | Flight_number | Depart.city | Dep. airport |
|----|-----------------|---------------|-------------------|--------------|
| 1 | AB | | NEW YORK CITY | LGA |
| 2 | AB | | SAN FRANCISCO, CA | |
| 3 | | 553 | ROME, ITALY | |
| 4 | END | 1 | | |
| 5 | AA | 551 | NYC | LGA |
| 6 | | 552 | ORLANDO, FL | |
| 7 | END | 2 | | |
| 8 | | | | |
| 9 | QR | 123 | LOUISVILLE, KY | SDF |
| 10 | ST | 101 | MADISON, WI | |
| 11 | YY | 110 | BALTIMORE, MD | BWI |
| 12 | | 552 | ORLANDO, FL | |
| 13 | END | 3 | | |

Manually added numbers indicate the iteration that corresponds to each data sheet

Working with Auto-Parameterization—Tips and Guidelines

Consider the following tips and guidelines when using the **Auto-parameterize table and grid controls** option:

- QuickTest inserts an **Input** statement and a new input data sheet each time information including modified table or grid cell data is sent to the server. If you set data in the cells of a particular table or grid both before and after sending information to the server, you will have more than one input data sheet (and more than one **Input** statement) representing the same table or grid.
- It is recommended to enter data only in the visible rows of the table or grid while recording, especially if scrolling results in sending information to the server. You can add additional rows to the recorded data set while editing your test.
- It is recommended to perform sorting, calculations, and other such operations either before beginning or after you finish entering data in a table or grid.

- The end of each data set in the input table or grid must be indicated by an end row with only the text `.END` in the first cell of the row.
- You can enter additional text, such as comments or an iteration number, in other cells of the `.END` row.
- You can include a different number of rows in each data set.


If you enter data for rows that require scrolling in order to display them in your application, you may need to modify your **Input** statement. For more information, see “Entering Data in Rows Requiring Scrolling” on page 87.

- When recording, QuickTest adds a column to the input data sheet only for table or grid columns in which you set data. You can add additional columns from your table or grid to the data sheet while editing your test. Double-click the column header in the data sheet to rename it. Enter the name of your table or grid column. If the table or grid column name has spaces, replace the spaces with underscores.
- In general, the columns in your data sheet can be in any order, as long as the column names match the column names in your table or grid. However, if you record data in a column without a column header name or if more than one column in the table or grid has the same header name, QuickTest adds a column to the data sheet in the format: `__<index>`, where `<index>` indicates the number of the column in the table or grid when you record the **Input** statement, for example, `__1` or `__2`. You can also use this format for columns in the data sheet if the column header names in your table or grid may change from iteration to iteration.
- To use multiple sets of data from an input data sheet, you must have at least one other Data Table parameter in your action that is set to use **Current action sheet (local)**. Also, confirm that the action is set to run multiple iterations in the Run tab of the Action Properties dialog box.
- The number of data sets in your input data sheet should match the number of rows in the corresponding action data sheet.

If your input data sheet contains fewer data sets than the number of rows in the action sheet, no data will be inserted in the table or grid during those action iterations. For example, if the action runs five iterations, and your input data sheet contains only four data sets, during the fifth iteration no data will be entered into the table or grid when the **Input** statement runs.

If your data sheet contains more data sets than the number of rows in the action sheet, those data sets will not be used.

Entering Data in Rows Requiring Scrolling

 QuickTest inserts a new **Input** statement and creates a new input data sheet each time you send information to the server that includes table or grid cell data. Therefore, if scrolling results in sending data to the server, it is recommended to add data only to visible cells during the recording session. If you want to enter data into additional rows during the run session, you can add those rows to the data sheet manually while editing your test.

If you create an input data set for rows that are not visible on the table or grid in your application, then QuickTest must scroll the table or grid during the run session in order to insert the data for those rows. If you create an input data set for a row that needs to be added to the table or grid, QuickTest must send a command to add the row. By default, QuickTest sends a `PAGEDOWN` command if the rows in the data sheet exceed those currently displayed in the application. If QuickTest needs to use the `ENTER` key to add additional rows to the table or grid, then you need to manually add the optional *ScrollMethod* argument (with the value `ENTER`) to your **Input** statement before running your test. For example:

```
SAPGuiSession("Session").SAPGuiWindow("Create Standard").  
  SAPGuiTable("SAPMV45ATCTRL_V_ERF_").  
    Input "Action1.All items", ENTER
```

Understanding Low-Level or Analog Mode Recording on SAP Gui for Windows



If you are unable to record on an object in the normal recording mode, or if you want to record mouse clicks and keyboard input with the exact x- and y-coordinates, you can record on those objects using low-level or analog recording (choose **Test > Low Level Recording** or **Test > Analog Recording** during a recording session).

Note that when recording in one of these modes, your steps are added to your test (or to the analog file) as you record them rather than when information is sent to the server. However, if you begin recording in low-level or analog mode, do not switch back to the normal recording mode until you perform a step that results in communication with the SAP server. If you switch between one of these modes and the normal recording mode before the server communication, your steps may be recorded twice (once in low-level/analog mode and once in normal mode).

For more information on low-level and analog recording, refer to the *QuickTest Professional User's Guide*.

Using Standard Windows Recording Capabilities



There are some operations in SAP Gui for Windows applications that open Standard Windows controls. To record steps on these controls while recording a test on an SAP Gui for Windows application, you need to switch to Standard Windows Recording mode. If you do not switch to Standard Windows Recording mode, then nothing is recorded in your test when you perform operations on these controls.



To switch to Standard Windows Recording mode while recording a test in an SAP Gui for Windows application, either click the **Standard Windows Recording Mode** button in the Testing toolbar, or choose **Test > Standard Windows Recording**.



To record steps as SAP Gui for Windows objects again, click the **Standard Windows Recording Mode** button in the Testing toolbar, or choose **Test > Standard Windows Recording** to deselect the option.

Note: If you switch to Standard Windows Recording mode after performing an operation on a Standard Windows control, in some cases this may cause both QuickTest and the SAP application to become unresponsive. To avoid this, make sure that you switch to Standard Windows Recording mode before you use the Standard Windows control in your SAP application.

Running Tests and Components on SAP Gui for Windows Applications

You run tests and components containing SAP Windows test objects in the same way as you run any other test or component. You can view the details of your run session in the Test Results window. For more information, refer to the *QuickTest Professional User's Guide*.



You can use the SAP tab of the Record and Run Settings dialog box to instruct QuickTest to open your SAP Gui for Windows application each time you begin running your test, or you can instruct QuickTest to run only on open SAP Gui for Windows sessions.

The test results tree displays the same SAP Windows test object icons as those used in the Keyword View. If you choose to save screen captures to the test results (**Tools > Options > Run** tab), the bottom right pane of the Test Results window displays the page that was captured during the run session and highlights the SAP Windows object for each step in the test results tree.

For more information on running tests and components and analyzing the results, refer to the *QuickTest Professional User's Guide*.

5

Enhancing Your SAP Windows Test



After you create your test, you can enhance it by adding checkpoints, retrieving output values, parameterizing values, and inserting SAP Windows objects, methods, and properties.

Note: All of the information in this chapter is relevant for tests and scripted components only.

This chapter describes:

- About Enhancing Your SAP Windows Test
- Checking SAP Windows Objects and Outputting Values
- Checking SAP Windows Tables and Grids
- Outputting SAP Windows Property and Table Cell Values

About Enhancing Your SAP Windows Test

After you record a test, you can use a variety of options to enhance your test. This section describes some of the options you can use to create comprehensive tests for your SAP Gui for Windows application.

Note: Checkpoints and output values are not available for components.

- You can add *checkpoints* to your test to check the properties of an SAP Windows object. A checkpoint is a step in your test that compares the values of the specified property during a run session with the values stored for the same test object property within the test. This enables you to identify whether or not your SAP Gui for Windows application is functioning correctly. For more information about checkpoints, see “Outputting SAP Windows Property and Table Cell Values” on page 103.

The results of the checkpoint can be viewed in the QuickTest Test Results window. For more information about the Test Results window, refer to the *QuickTest Professional User's Guide*.

- You can *parameterize* your test to replace fixed values with values from an external source during your run session. The values can come from a Data Table, environment variables you define, or values that QuickTest generates during the run session. For more information about parameterizing tests, refer to the *QuickTest Professional User's Guide*.
- You can specify test, action, or component parameters to pass values into and from your test or component, and between actions in your test. For more information, refer to the *QuickTest Professional User's Guide*. For components, you can also use *local* parameters, which are parameters that are accessible only to steps within the current component. (You cannot use them to pass data between components.) For more information, refer to the *QuickTest Professional New Features Guide*.
- You can retrieve values from your test and store them in the Data Table as *output values*. You can subsequently use these values as input parameters in your test. This enables you to use data retrieved during one part of a test in later parts of the test. For more information about output values, see “Outputting SAP Windows Property and Table Cell Values” on page 103.
- You can insert statements containing SAP Windows test object methods in the Expert View or in the Keyword View by using the **Insert > Step > Step Generator** option to open the Step Generator. This enables you to include other operations in your test that may not have been added while recording, such as retrieving property values, checking that objects exist, and checking other elements of your application. For more information about SAP Windows test object methods and properties, see “Working with SAP Windows Test Objects,” on page 114, and the **SAP Windows** section of the *QuickTest Professional Object Model Reference*.

Checking SAP Windows Objects and Outputting Values

After you record a test, you can use a variety of options to enhance it, including adding checkpoints and retrieving output values. Checkpoints confirm that your application behaves as you expect. Output values enable you to retrieve object property or table cell values for use later in your test.

Note: You can use checkpoints and output values to enhance tests only. Checkpoints and output values are not available for components.

This chapter describes how to use checkpoints and output values with your SAP Gui for Windows application. For more information on parameterizing values, refer to the *QuickTest Professional User's Guide*.





















For more information on working with SAP Windows objects, methods and properties, see Chapter 6, “Adding SAP Windows Statements to Your Test or Component” and

The following checkpoints and output values are supported when running tests on SAP Windows objects:

- **Standard** checkpoints check the property value of an object in your SAP Gui for Windows application. You create checkpoints for SAP Windows test objects in the same way you create checkpoints for standard Windows objects. You can insert standard checkpoints while recording or editing your test. You can also create a standard output value from any object property value.
- **Table** checkpoints check information within tables and grids in your SAP Gui for Windows application. You can also create a table output value from the contents of a cell in a table or grid.
- **Bitmap** checkpoints check an area of your SAP Gui for Windows application as a bitmap. You can use bitmap checkpoints to compare an object or selected area with the bitmap stored when the test or component was recorded and check that the area is displayed accurately in your SAP Gui for Windows application.

For more information about table checkpoints, see “Checking SAP Windows Tables and Grids” on page 94. For more information about standard and bitmap checkpoints and about output values, refer to the *QuickTest Professional User’s Guide*.

The following table lists the SAP Windows test objects on which you can insert a checkpoint or output value:

| Icon | Test Object Class | Icon | Test Object Class |
|---|-------------------|---|-------------------|
|  | SAPGuiSession |  | SAPGuiLabel |
|  | SAPGuiWindow |  | SAPGuiMenuBar |
|  | SAPGuiAPOGrid |  | SAPGuiOKCode |
|  | SAPGuiButton |  | SAPGuiRadioButton |
|  | SAPGuiCalendar |  | SAPGuiStatusBar |
|  | SAPGuiCheckBox |  | SAPGuiTable |
|  | SAPGuiComboBox |  | SAPGuiTabStrip |
|  | SAPGuiEdit |  | SAPGuiTextArea |
|  | SAPGuiElement |  | SAPGuiToolbar |
|  | SAPGuiGrid |  | SAPGuiTree |

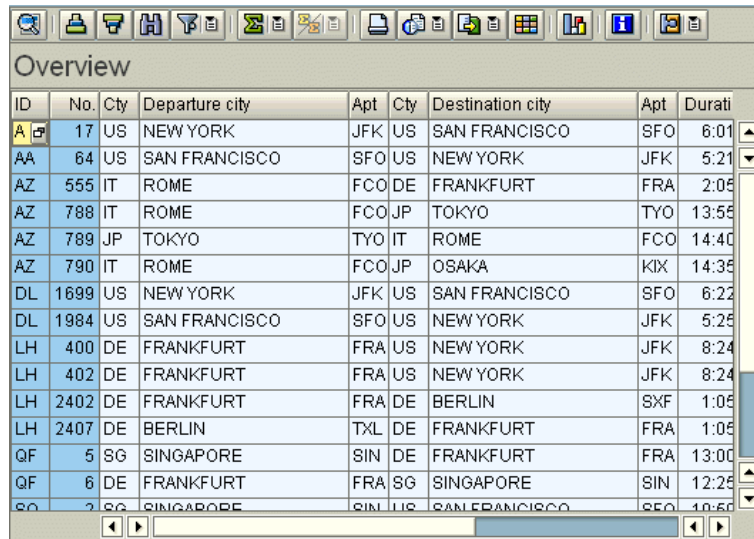
For a detailed list of object properties, see the **SAP Windows** section of the *QuickTest Professional Object Model Reference*.

Checking SAP Windows Tables and Grids

You can check the contents and properties of simple table controls and ActiveX grid controls in your SAP Gui for Windows application by inserting table checkpoints into your test. You can add a table checkpoint while recording or editing a test.

You can check content (like a regular table checkpoint) and/or properties (like a regular standard checkpoint) for SAP Gui for Windows table and grid controls. For example, you can check that the value of a specific table cell is as expected (content check), and also check that a table has the expected number of columns and rows (properties check). For general information on checking tables, refer to the *QuickTest Professional User's Guide*.

For example, you can check the contents and/or the properties of the following grid control:



| ID | No. | Cty | Departure city | Apt | Cty | Destination city | Apt | Durati |
|----|------|-----|----------------|-----|-----|------------------|-----|--------|
| A | 17 | US | NEW YORK | JFK | US | SAN FRANCISCO | SFO | 6:01 |
| AA | 64 | US | SAN FRANCISCO | SFO | US | NEW YORK | JFK | 5:21 |
| AZ | 555 | IT | ROME | FCO | DE | FRANKFURT | FRA | 2:05 |
| AZ | 788 | IT | ROME | FCO | JP | TOKYO | TYO | 13:55 |
| AZ | 789 | JP | TOKYO | TYO | IT | ROME | FCO | 14:40 |
| AZ | 790 | IT | ROME | FCO | JP | OSAKA | KIX | 14:35 |
| DL | 1699 | US | NEW YORK | JFK | US | SAN FRANCISCO | SFO | 6:22 |
| DL | 1984 | US | SAN FRANCISCO | SFO | US | NEW YORK | JFK | 5:25 |
| LH | 400 | DE | FRANKFURT | FRA | US | NEW YORK | JFK | 8:24 |
| LH | 402 | DE | FRANKFURT | FRA | US | NEW YORK | JFK | 8:24 |
| LH | 2402 | DE | FRANKFURT | FRA | DE | BERLIN | SXF | 1:05 |
| LH | 2407 | DE | BERLIN | TXL | DE | FRANKFURT | FRA | 1:05 |
| QF | 5 | SG | SINGAPORE | SIN | DE | FRANKFURT | FRA | 13:00 |
| QF | 6 | DE | FRANKFURT | FRA | SG | SINGAPORE | SIN | 12:25 |
| SO | 2 | SG | SINGAPORE | SIN | US | SAN FRANCISCO | SFO | 10:50 |

When creating a checkpoint on simple table controls, QuickTest captures all rows and columns of the table (whether or not they are visible). You can choose to create your checkpoint on a specific range of rows in the Define Row Range dialog box.

When creating a checkpoint on ActiveX grid controls (these generally have toolbars), QuickTest captures the data from all columns and all rows in the grid. If you do not need to check data from all rows in your grid, you can specify the rows you want to include in the checkpoint in the Define Row Range dialog box. You can also increase or decrease the number of rows included in the checkpoint at a later time.

Tip: When inserting a table checkpoint into your test, consider how other steps performed on the table may affect the checkpoint.

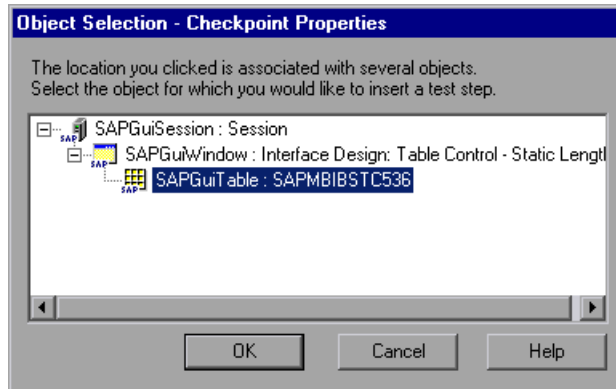
Example 1: If you have a step in your test that clicks the **Total** toolbar button on a grid control, that click refreshes all data in the table. The refresh could potentially cause a table checkpoint on a cell in the table to fail.

Example 2: If you click on a toolbar button in a grid control that adds rows to your table before creating a table checkpoint, the extra rows are captured as part of the grid checkpoint (if you capture all rows). Therefore, confirm that the same rows are displayed during the run session.

To add a table checkpoint while recording:

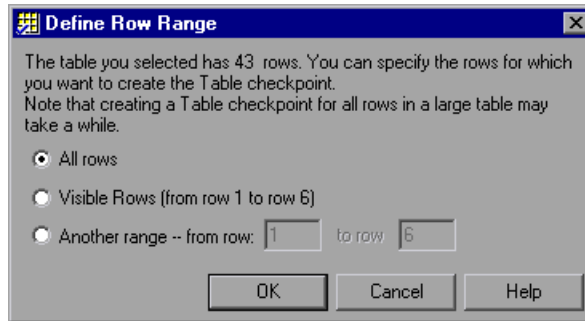


- 1 Choose **Insert > Checkpoint > Standard Checkpoint** or click the **Insert Checkpoint** button. The QuickTest window is minimized and the mouse pointer turns into a pointing hand.
- 2 Click the table or grid you want to check. If the location you select is associated with more than one object, the Object Selection - Checkpoint Properties dialog box opens.



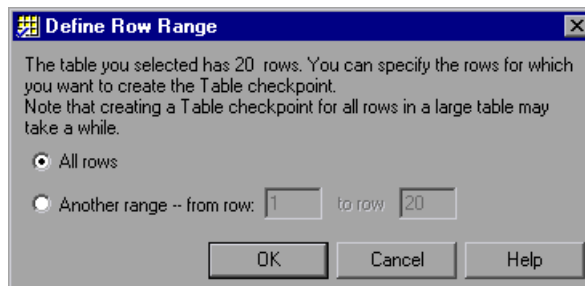
- 3 Select an **SAPGuiTable**, **SAPGuiGrid**, or **SAPGuiAPOGrid** object from the displayed object tree and click **OK**. The Define Row Range dialog box opens. The Define Row Range dialog box differs depending on whether a table object, a grid, or an APO grid object is selected.

- **Table controls**—The Define Row Range dialog box for table controls appears as follows:



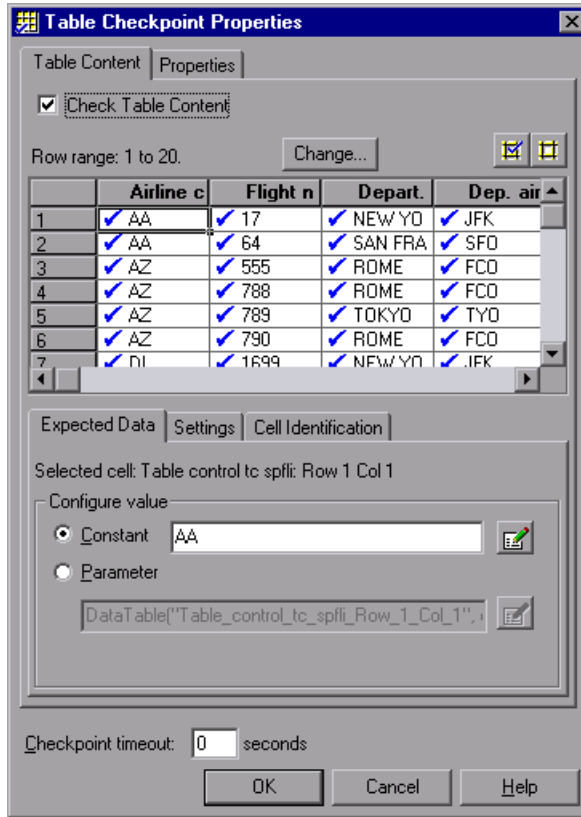
Note: The total number of rows indicated in the first sentence of the dialog box for table controls is only an approximation. This is because only the data from visible rows is actually available for SAP Windows table controls.

- **Grid and APOGrid controls**—The Define Row Range dialog box for grid controls and APOGrid controls appears as follows:



Note: The total number of rows indicated in the first sentence of the dialog box for grid controls is exact. The **Visible Rows** option is not available when checking grid controls.

- 4 Select the range of rows you want to include in your checkpoint. You can include all the rows in the table, only the visible rows (for SAP Gui for Windows table controls only), or another range that you specify.
- 5 Click **OK**. The Table Checkpoint Properties dialog box opens.



Note: If you selected **All rows** or you specified a large row range in the Define Row Range dialog box, it may take a few moments for the Table Checkpoint Properties dialog box to open.

Tip: The Table Checkpoint Properties dialog box also has a **Change** button that enables you to modify the number of rows captured for the checkpoint. For more information, see “Modifying a Table Checkpoint” on page 101.

- 6 Choose one or both of the following options, depending on the type of information you want to check:
 - To check the content of the table object, select the **Check Table Content** check box in the Table Content tab. (This check box is selected by default.) The Table Content tab enables you to specify the content of the table object to check and which verification method to use. You can also edit or parameterize the expected data for the values included in the check. The options in this tab are similar to the options in the regular Table Checkpoint Properties dialog box. For information on the options in this tab, refer to the chapter that describes checking tables and databases in the *QuickTest Professional User's Guide*.
 - To check the property values of the table object, select the **Check Properties** check box in the Properties tab. (This check box is not selected by default.) The Properties tab enables you to specify which properties of the object to check. You can also edit or parameterize the expected data for the values included in the check. The options in this tab are similar to the options in the regular Checkpoint Properties dialog box. For information on the options in this tab, refer to the chapter that describes checking object property values in the *QuickTest Professional User's Guide*. For information on the specific test object properties that are available for each tabular object, refer to the **SAP Windows** section of the *QuickTest Professional Object Model Reference*.
- 7 Specify the options for your checkpoint. For more information about table checkpoint options, refer to the *QuickTest Professional User's Guide*.
- 8 Click **OK** to close the dialog box. A checkpoint step is added to your test or component.

To add a table checkpoint while editing your test:

- 1 Open the SAP Gui for Windows application containing the table you want to check and display the table in the application. You must have your SAP Gui for Windows application open to the screen that contains the table object to be able to create a checkpoint on the content of the object.

Note: Even if your Active Screen contains a capture of the table or grid, you must have the actual table or grid open to the appropriate level or view to insert a new table checkpoint while editing your test or component. In general, it is not necessary to open the table or grid to edit an existing checkpoint. However, if you want to modify the row range for the checkpoint, the actual table or grid must be open to the appropriate level or view.

- 2 Depending on whether the table object is already in a step in your test or component, perform one of the following:
 - If you have already recorded a step on the table object you want to check, right-click the step for the table object in the Keyword View or Expert View and choose **Insert Standard Checkpoint**.



- If you have not recorded a step on the table object you want to check, but you have an Active Screen capture that displays the table object, you can add a table checkpoint only if the Active Screen **Capture level** was set to **Complete** when the object was captured and the **Active Screen** button is currently selected. (You set the **Capture level** in the Active Screen tab of the Options dialog box (**Tools > Options**). For more information, refer to the *QuickTest Professional User's Guide*.)

Click the step in your test or component where you want to add a checkpoint. The Active Screen displays the screen corresponding to the highlighted step. Right-click the table object in the Active Screen and choose **Insert Standard Checkpoint**. The Object Selection - Checkpoint Properties dialog box opens. Select an SAP table object from the displayed object tree and click **OK**.

- 3 The remainder of the procedure is similar to that for creating a checkpoint while recording. For more information, continue with step 3 on page 96.

Modifying a Table Checkpoint

You can change the expected data, settings, and cell identification options for an existing table checkpoint. You can also change the rows that are included in the checkpoint.

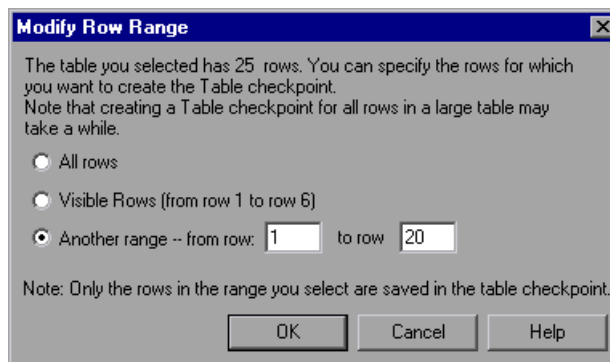
To modify settings for an existing table checkpoint:

- 1 In the Keyword View or Expert View, right-click the table checkpoint that you want to modify and select **Checkpoint Properties**. The Checkpoint Properties dialog box opens.
- 2 Modify the settings as required. For more information, refer to the *QuickTest Professional User's Guide*.

To modify the number of rows in an existing table checkpoint:

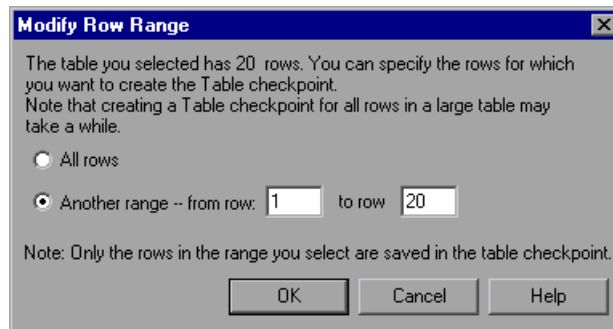
- 1 Open the SAP Gui for Windows application containing the table or grid you want to check and display the table or grid in the application.
- 2 In the Keyword View or Expert View, right-click the table checkpoint that you want to modify and select **Checkpoint Properties**. The Checkpoint Properties dialog box opens.
- 3 In the Table Content tab, click the **Change** button at the top of the dialog box. The Modify Row Range dialog box opens. The Define Row Range dialog box differs depending on whether a table object, grid, or APOGrid object is selected.

► **Table controls**—The Modify Row Range dialog box appears as follows:



Note: The total number of rows indicated in the first sentence of the dialog box for table controls is only an approximation. This is because only the data from visible rows is actually available for SAP Windows table controls.

- **Grid and APOGrid controls**—The Modify Row Range dialog box for grid controls and APOGrid controls appears as follows:



Note: The total number of rows indicated in the first sentence of the dialog box for grid controls is exact. The **Visible Rows** option is not available when checking values for grid controls.

- 4** Select the range of rows you want to include in your checkpoint. You can include all the rows in the table or grid, only the visible rows (for SAP table controls only), or another range that you specify.

- 5 Click **OK**. The Modify Row Range dialog box closes, and the Table Checkpoint Properties dialog box displays the rows you specified in the Modify Row Range dialog box.
- If your modified row range includes new rows, QuickTest captures the current values of the new rows from the open table in your SAP Gui for Windows application.
 - If your modified row range includes some or all of the rows that were already included in the checkpoint, the expected values of those cells are not changed. This enables you to modify the row range without losing parameterization, regular expressions, or other changes you may have made to the expected cell values in your checkpoint. Therefore, you cannot use the Modify Row Range dialog box to update the expected values of an existing table checkpoint. To update the expected values of your checkpoint, use the **Update Run** option. For more information, refer to the *QuickTest Professional User's Guide*.
 - If your modified row range excludes some or all of the rows that were previously included in your checkpoint, those rows (and any modifications you made to the expected values) are deleted from the checkpoint.

Outputting SAP Windows Property and Table Cell Values

You can retrieve object property values or table cell values from your test during each iteration of the run session and store them in the Data Table. You can subsequently use these *output values* as input variables. This enables you to use data retrieved during a run session in other parts of your test.

Outputting Object Property Values

You can output the property values of objects in your application while recording or editing your test. You output the property values of SAP Windows objects just as you do for any other application.

For example, consider a purchase order transaction. You design a test in which you place a new order and then view the status of the new order. Each time you run the test, the application generates a unique purchase order ID for the new order. To view the status of a purchase order, you must enter the purchase order ID that was generated when the order was created. You cannot know the purchase order ID before you run the test.

To solve this problem, you create an output value for the unique number that is generated when you create a new order and store it in an OrderID column in the Data Table. Then you parameterize the step that specifies an order ID to view. You use the Order ID Data Table column (created in the output value step) as the parameter column.

When you run the test, QuickTest retrieves the unique order ID that was generated for the new order and inserts it in the run-time Data Table as an output value. When the test reaches the step for setting the value of the **Order ID** edit box in the View Order Status window, QuickTest inserts the unique order ID from the run-time Data Table into the **Order ID** edit box.

For more information about output values, refer to the *QuickTest Professional User's Guide*.

Outputting Table Cell Values

You can output the contents of simple table controls and ActiveX grid controls in your SAP Gui for Windows application by inserting output values in your test. You can insert an output value for a table or grid cell while recording or editing a test.

When inserting an output value for simple table controls, QuickTest captures all rows and columns of the table (whether or not they are visible), in the SAP Gui for Windows application. You can choose to create your output value on a larger range of rows in the Define Row Range dialog box.

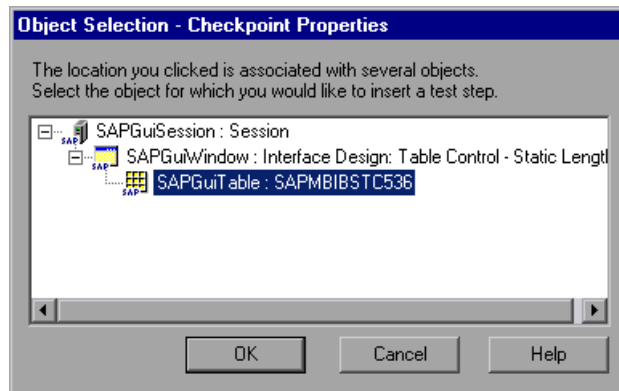
When creating an output value for ActiveX grid controls (these generally have toolbars), QuickTest captures the data from all columns and all rows in the grid. If you do not need to output data from all rows in your test, you can specify the rows you want to include in the output value in the Define Row Range dialog box. You can also increase or decrease the number of rows included Table Output Value Properties dialog box at a later time.

Tip: You can also spool all of the available data from a table into an external file. For more information, see “Spooling Data from a Table” on page 112.

To create an output value for a table cell while recording:

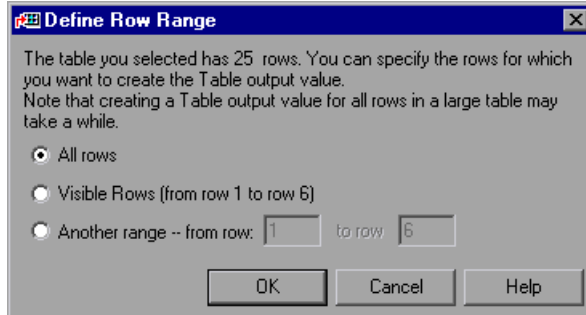


- 1** In the Keyword View or Expert View, choose **Insert > Output Value > Standard Output Value**. Alternatively, click the arrow next to the **Insert Checkpoint** toolbar button and choose **Standard Output Value**. The QuickTest window is minimized and the mouse pointer turns into a pointing hand.
- 2** Click the table or grid for which you want to insert an output value. If the location you select is associated with more than one object, the Select an Object dialog box opens.



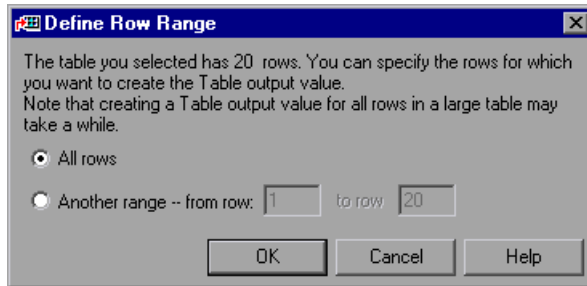
- 3** Select an **SAPGuiTable**, **SAPGui Grid**, or **SAPGuiAPOGrid** object from the displayed object tree and click **OK**. The Define Row Range dialog box opens.

- **Table controls**—The Define Row Range dialog box appears as follows



Note: The total number of rows indicated in the first sentence of the dialog box for table controls is only an approximation. This is because only the data from visible rows is actually available for SAP Windows table controls.

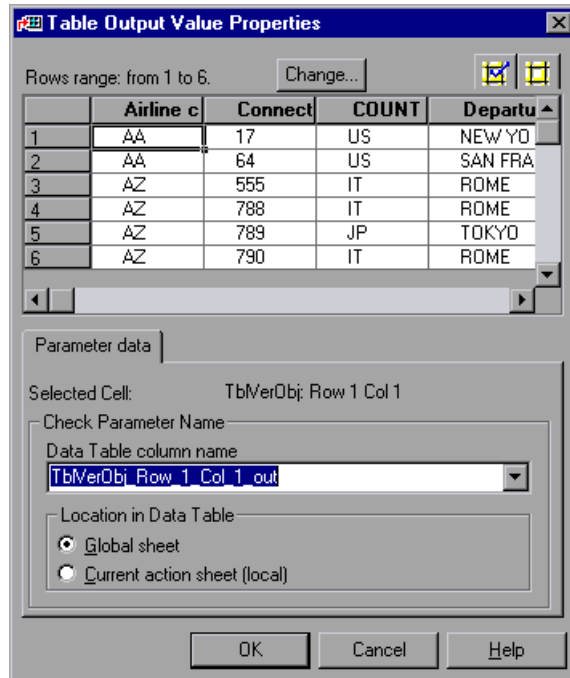
- **Grid and APOGrid controls**—The Define Row Range dialog box for grid controls and APOGrid controls appears as follows:



Note: The total number of rows indicated in the first sentence of the dialog box for grid controls is exact.

The **Visible Rows** option is not available when outputting values for grid controls.

- 4 Select the range of rows from which you want to output values. You can include all the rows in the table or grid, only the visible rows (for SAP table controls only), or another range that you specify.
- 5 Click **OK**. The Table Output Value Properties dialog box opens.



Note: If you selected **All Rows** or you specified a large row range in the Define Row Range dialog box, it may take a few moments for the Table Output Value Properties dialog box to open.

- 6 Specify your preferences for the cells you want to output. For more information on the Table Output Value Properties dialog box, refer to the *QuickTest Professional User's Guide*.

Note: The Table Output Value Properties dialog box also has a **Change** button that enables you to modify the number of rows captured for the output value. For more information, see “Modifying a Table Output Value” on page 111.

- 7 Click **OK** to close the dialog box. QuickTest inserts an output value step in your test.

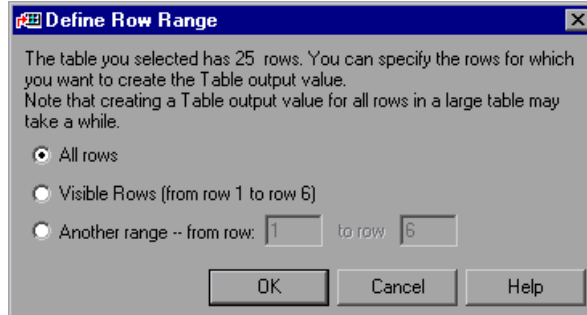
To create a table output value while editing your test:

- 1 Open the SAP Gui for Windows application containing the table or grid from which you want to output values and display the table in the application.
- 2 Confirm that the **Active Screen** button is selected.
- 3 In the Keyword View or Expert View, click the step whose Active Screen contains the table or grid for which you want to specify an output value.
- 4 In the Active Screen, right-click the table or grid for which you want to create an output value and select **Insert Output Value**. If the location you select is associated with more than one object, the Object Selection - Output Value Properties dialog box opens.
- 5 Select the table or grid for which you want to create an output value and click **OK**. The Define Row Range dialog box opens.



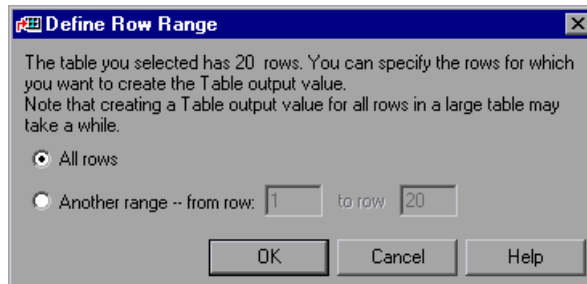
Tip: Instead of performing steps 2 to 4 above, you can right-click a table or grid object in the Keyword View or Expert View and choose **Insert Output Value** to open the Table Output Value Properties dialog box for the selected table or grid.

- **Table controls**—The Define Row Range dialog box appears as follows:



Note: The total number of rows indicated in the first sentence of the dialog box for table controls is only an approximation. This is because only the data from visible rows is actually available for SAP Windows table controls.

- **Grid and APOGrid controls**—The Define Row Range dialog box for grid controls and APOGrid controls appears as follows:

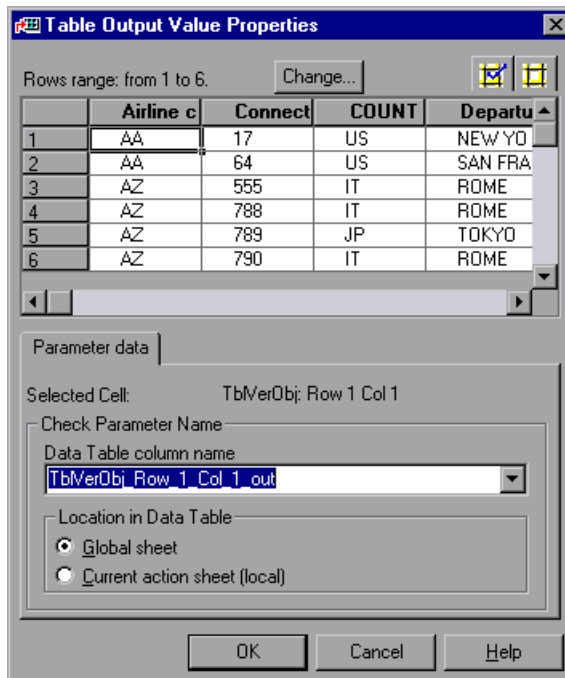


Note: The total number of rows indicated in the first sentence of the dialog box for grid controls is exact.

- 6 Select the range of rows from which you want to output values. You can include all the rows in the table or grid, only the visible rows (for SAP table controls only), or another range that you specify.

Note: The **Visible Rows** option is not available when outputting values for ActiveX grid controls.

- 7 Click **OK**. The Table Output Value Properties dialog box opens.



Note: If you selected **All Rows** or you specified a large row range in the Define Row Range dialog box, it may take a few moments for the Table Output Value Properties dialog box to open.

- 8 Specify your preferences for the cells you want to output. For more information on the Table Output Value Properties dialog box, refer to the *QuickTest Professional User's Guide*.

Note: The Table Output Value Properties dialog box also has a **Change** button that enables you to modify the number of rows captured for the output value. For more information, see “Modifying a Table Output Value” on page 111.

- 9 Click **OK** to close the dialog box. QuickTest inserts an output value step in your test.

Modifying a Table Output Value

You can modify existing table output value steps. You can change the cells to output or the options for selected cells. You can also change the rows that are available to output.

To modify settings for an existing table output value step:

- 1 In the Keyword View or Expert View, right-click the table output value step that you want to modify and select **Output Value Properties**. The Output Value Properties dialog box opens.
- 2 Modify the settings as required. For more information, refer to the *QuickTest Professional User's Guide*.

To modify the number of rows in an existing table output value step:

- 1 Open the SAP Gui for Windows application containing the table or grid for which you want to modify output values and display the table or grid in the application.
- 2 In the Keyword View or Expert View, right-click the output value step that you want to modify and select **Output Value Properties**. The Output Value Properties dialog box opens.
- 3 Click the **Change** button at the top of the dialog box. The Modify Row Range dialog box opens.

- 4 Select the range of rows you want to make available to output. You can include all the rows in the table or grid, only the visible rows (for SAP table controls only), or another range that you specify.
- 5 Click **OK**. The Modify Row Range dialog box closes. The Table Output Value Properties dialog box displays the rows you specified in the Modify Row Range dialog box.
 - If your modified row range includes new rows, QuickTest captures the current values of the new rows from the open table in your SAP Gui for Windows application.
 - If your modified row range includes some or all of the rows that were already available for your output value, the Data Table settings of those cells are not changed.
 - If your modified row range excludes some or all of the rows that were previously available for your output value, those rows (and any output value settings you configured on cells in those rows) are deleted from the output value.

Spooling Data from a Table

If you want to pool all the data from an SAP Gui for Windows table into an external file, use the **GetCellData** method to loop through each cell in the table. You can then save the information to an external file.

The following example uses the **GetCellData** method to list the data of each cell in a table of 10 rows and 10 columns:

```
For i=1 to 10
    for j=1 to 10
        col="#" & j
        Dat=SAPGuiSession("Session").SAPGuiWindow("Create Standard").
            SAPGuiTable("SAPMV45ATCTRL_U_ERF_").GetCellData(i, col)
        'Enter line(s) of code that use the value of the returned Dat variable
    next
next
```

For more information on the **GetCellData** method, see the SAP Gui for Windows section of the *QuickTest Professional Object Model Reference*.

6

Adding SAP Windows Statements to Your Test or Component

In addition to recording the steps that you perform on your application, you can add statements to your test or component using the Keyword View. You can also add statements to your test using the Expert View or the Step Generator. This enables you to create a more comprehensive test or component for your SAP Gui for Windows application.


This chapter describes:

- About Adding SAP Windows Statements to Your Test or Component
- Working with SAP Windows Test Objects
- Working with Test Object Methods and Properties
- Accessing Native Methods and Properties in Your SAP Gui for Windows Application

About Adding SAP Windows Statements to Your Test or Component




A test or component consists of statements coded in Microsoft VBScript. These statements are composed of objects, methods, and/or properties that instruct QuickTest to perform operations or retrieve information. When you record, these statements are generated automatically in response to input to the application.

In addition, for tests, you can add and modify statements manually in the Keyword View or the Expert View. You can also mix recorded and programmed statements in the same test. For components, you can record statements, or you can add and modify them using the Keyword View.

 After you record a basic test and enhance it with checkpoints, parameters, and output values, you may also want to add statements containing test object methods using the Step Generator (**Insert > Step > Step Generator**). You can also enter VBScript programming statements in the Expert View. This enables you to include operations in your test that you did not, or could not, add while recording. These can include retrieving property values, checking that objects exist, enumerating objects, and checking other elements of your application.

Working with SAP Windows Test Objects

The basic SAP Windows test object hierarchy consists of three levels: **SAPGuiSession** > **SAPGuiWindow** > **SAPGuiObject**, where *SAPGuiObject* represents an object in your SAP Windows application. For example, if you record a step on an edit box, QuickTest records a step with the following hierarchy:

| | | | |
|---|--------|--------|--|
|  Session | | | |
|  SAP R/3 | Resize | 135,33 | Resize the "SAP R/3" window to 135 by 33 characters. |
|  Fixed_cols | Set | "4" | Enter "4" in the "Fixed_cols" edit box. |

```
SAPGuiSession("Session").SAPGuiWindow("SAP R/3").
    SAPGuiEdit("Fixed_cols").Set "4"
```

QuickTest has a set of SAP Windows test object classes that represent objects in your application on which you can record operations. There are also other objects that you can add to your test or component manually.

QuickTest also has an alternative recording mechanism that can be used to record objects in your application that are not represented by a specific test object class. For more information, see “SAP Gui for Windows Alternative Recording Mechanism” on page 127.

This section describes how QuickTest identifies the objects in an SAP Gui for Windows application and provides information on each of the following objects:

- SAPGuiSession Object (see page 116)
- Basic SAP Gui for Windows UI Controls
 - SAPGuiButton Object (see page 116)
 - SAPGuiCheckBox Object (see page 116)
 - SAPGuiComboBox Object (see page 118)
 - SAPGuiEdit Object (see page 116)
 - SAPGuiMenubar Object (see page 116)
 - SAPGuiOKCode Object (see page 116)
 - SAPGuiRadioButton Object (see page 116)
 - SAPGuiTextArea Object (see page 118)
 - SAPGuiWindow Object (see page 116)
- SAPGuiAPOGrid Object (see page 118)
- SAPGuiCalendar Object (see page 120)
- SAPGuiElement Object (see page 120)
- SAPGuiGrid Object (see page 119)
- SAPGuiLabel Object (see page 121)
- SAPGuiStatusBar Object (see page 121)
- SAPGuiTable Object (see page 122)
- SAPGuiTabStrip Object (see page 122)
- SAPGuiToolbar Object (see page 123)
- SAPGuiTree Object (see page 125)
- SAPGuiUtil Object (see page 127)

SAPGuiSession Object



The SAPGuiSession object represents the SAP Gui for Windows session on which an operation is performed. For tests and components, you can record on the SAPGuiSession object or add it manually using the Keyword View. For tests, you can also insert SAPGuiSession statements using the Step Generator or manually in the Expert View to create, reset, synchronize, or close a session.

Basic SAP Gui for Windows UI Controls

The example window on the next page contains the following objects:



- **SAPGuiMenubar**—Represents the menu bar at the top of the main SAP window.



- **SAPGuiButton**—Represents push-buttons in your application. The SAPGuiButton object is also recorded when you perform operations on buttons in the toolbar of the main SAP window.

Note that the SAPGuiToolbar object is used only for toolbar objects within your SAP Gui for Windows application. For more information, see “SAPGuiToolbar Object” on page 123.



- **SAPGuiOKCode**—Represents the edit box in which you enter commands to navigate to the desired transaction.



- **SAPGuiWindow**—Represents the main SAP window and the dialog boxes in your SAP Gui for Windows application.



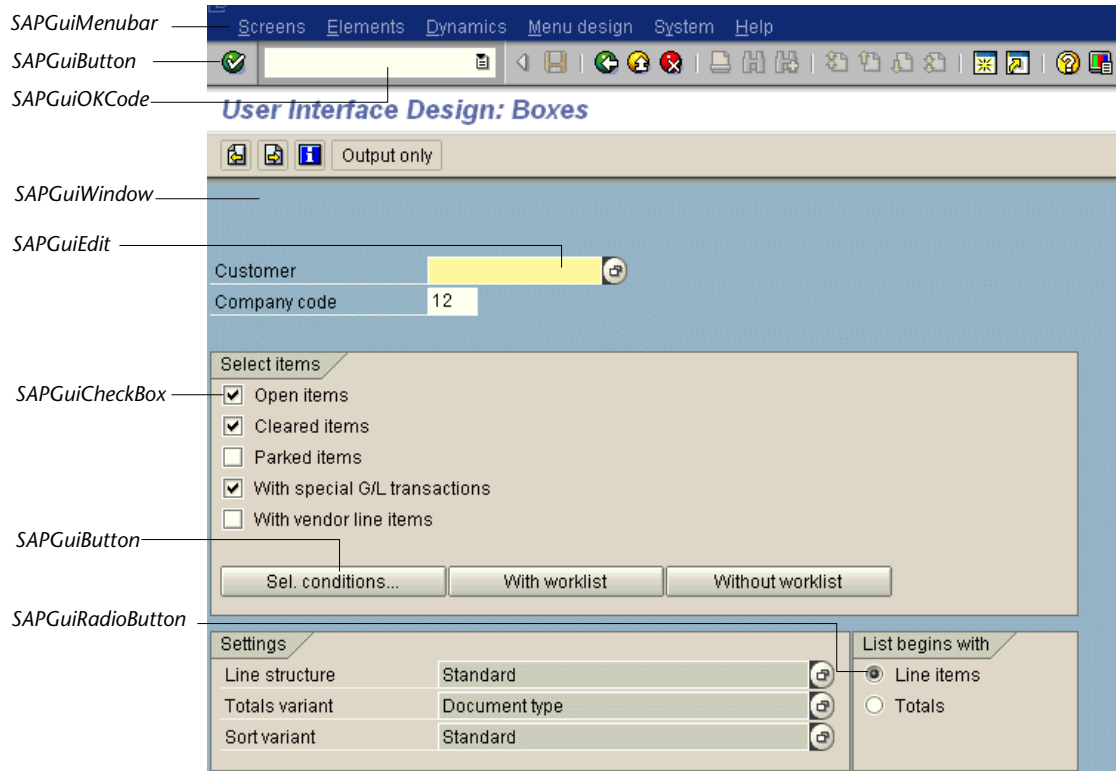
- **SAPGuiEdit**—Represents fields in which you can enter a single line of text.



- **SAPGuiCheckbox**—Represents toggle check box objects in your application.



- **SAPGuiRadioButton**—Represents radio button objects in your application.



The example below contains the following objects:



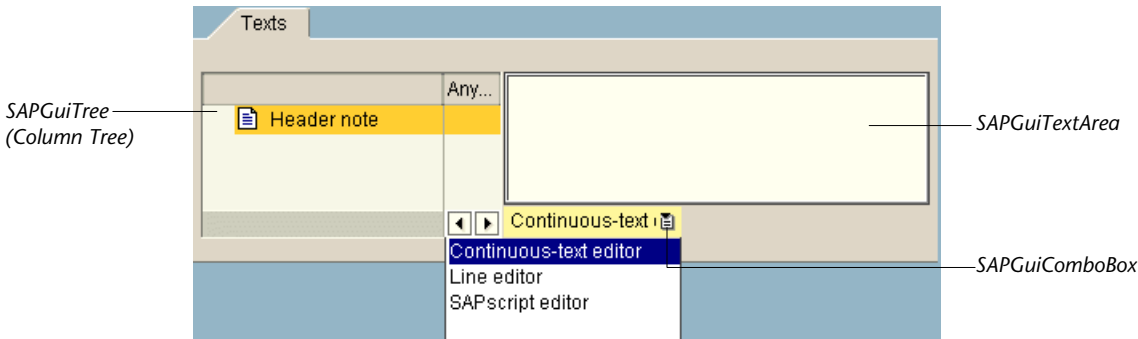
SAPGuiTree—Represents simple or table tree controls. For more information, see “SAPGuiTree Object” on page 125.



SAPGuiTextArea—Represents multi-line text areas.



SAPGuiComboBox—Represents single- or multiple-selection combo boxes.



SAPGuiAPOGrid Object



The SAPGuiAPOGrid object represents APO Grid controls in your SAP Gui for Windows application. APO Grid controls are similar to Grid controls, with some additional functionality. The SAPGuiAPOGrid is available only in the SAP APO module.

The screenshot shows the SAP GUI APO Grid window. It has a menu bar with 'Design' and 'Graphic' options. Below the menu bar is a toolbar with various icons. The main area is a grid with columns for months: 'M 09.2004', 'M 10.2004', 'M 11.2004', and 'M 12.2004'. The grid rows include 'HISTORICAL DATA', 'Last Year's Quantity', 'Last Year's Sales', 'Last Year's Cost of Goods Ordered', 'Last Year's Price/1000 Units', 'Last Year's Costs/1000 Units', and 'FORECAST'. The 'Last Year's Costs/1000 Units' row is highlighted. The 'FORECAST' row is highlighted in green.

| | Un | M 09.2004 | M 10.2004 | M 11.2004 | M 12.2004 |
|-----------------------------------|-----|-----------|-----------|-----------|-----------|
| HISTORICAL DATA | KG | | | | |
| Last Year's Quantity | KG | | | | |
| Last Year's Sales | KG | | | | |
| Last Year's Cost of Goods Ordered | EUR | | | | |
| Last Year's Price/1000 Units | EUR | | | | |
| Last Year's Costs/1000 Units | EUR | | | | |
| ===== | KG | | | | |
| FORECAST | KG | | | | |

SAPGuiGrid Object



The SAPGuiGrid object represents ActiveX grid controls in your SAP Gui for Windows application. Grid controls usually have toolbars that enable you to manipulate or perform operations on the values in the grid. The following image shows an SAPGuiGrid object after a row has been added to the grid as a result of highlighting the **Distance** column and clicking the **Total** button in the toolbar.

SAPGuiGrid

| ID | No. | Apt | Durati... | Departure | Arrival time | z | Distance | Dis. | ... | Σ | ... |
|----|------|------|-----------|-----------|--------------|---|----------|------|-----|---|-----|
| LH | 400 | JFK | 8:24 | 10:10:00 | 11:34:00 | | 6.162 | KM | | 0 | |
| LH | 402 | JFK | 8:24 | 13:30:00 | 15:05:00 | | 6.162 | KM | X | 0 | |
| LH | 2402 | SXF | 1:05 | 10:30:00 | 11:35:00 | | 555 | KM | | 0 | |
| QF | 6 | SIN | 12:25 | 23:45:00 | 18:10:00 | | 10.000 | KM | | 1 | |
| SQ | 2 | SFO | 10:50 | 09:30:00 | 03:35:00 | | 4.218 | MI | | 0 | |
| SQ | 158 | JKT | 2:25 | 11:30:00 | 13:55:00 | | 560 | MI | | 0 | |
| SQ | 866 | H... | 4:30 | 07:10:00 | 11:40:00 | | 1.625 | MI | | 0 | |
| SQ | 988 | TYO | 5:40 | 16:35:00 | 00:15:00 | | 3.125 | MI | | 1 | |
| UA | 941 | SFO | 12:36 | 14:30:00 | 21:06:00 | | 5.685 | MI | | 0 | |
| UA | 3504 | FRA | 13:35 | 15:00:00 | 10:30:00 | | 5.685 | MI | | 1 | |
| | | | | | | | 33.434 | KM | | 5 | |
| | | | | | | | 56.331 | MI | | 0 | |

A row added to the grid as a result of toolbar operations.

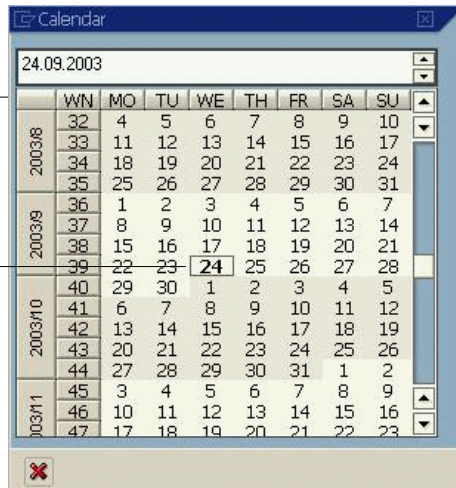
SAPGuiCalendar Object



The SAPGuiCalendar object represents a calendar control in your SAP Gui for Windows application. The following image shows an SAPGuiCalendar object with a date selected.

SAPGuiCalendar
for a calendar
control

Date selected
in a calendar
control



SAPGuiElement Object

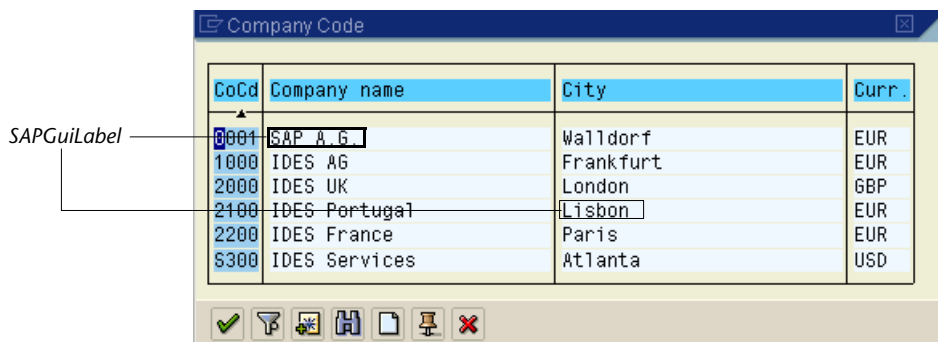


The SAPGuiElement object can represent any SAP Windows object. This object may be recorded if you insert a checkpoint or output value on an unrecognized SAP Windows object. You can also enter this object into your test manually to refer to any SAP Windows object that matches a specified programmatic description. For more information on programmatic descriptions, refer to the *QuickTest Professional User's Guide*.

SAPGuiLabel Object



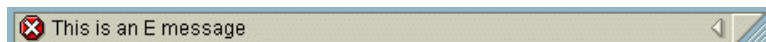
The following image shows a Possible Entries dialog box containing SAPGuiLabel (static text) objects. When you record on an SAPGuiLabel object, QuickTest always records the cursor position (**SetCaretPos** method) within the label object.



SAPGuiStatusBar Object



The following image shows an SAPGuiStatusBar object. You cannot record operations performed on a status bar, but you can check the entire text or any reserved parameter within the status bar's message text. When the **Record status bar messages** option is selected in the SAP tab of the Options dialog box, a step is automatically recorded each time a message is sent from the server. If this option is not selected, you can add **SAPGuiStatusBar** steps to the script only by inserting checkpoints or output values while recording.



SAPGuiTable Object





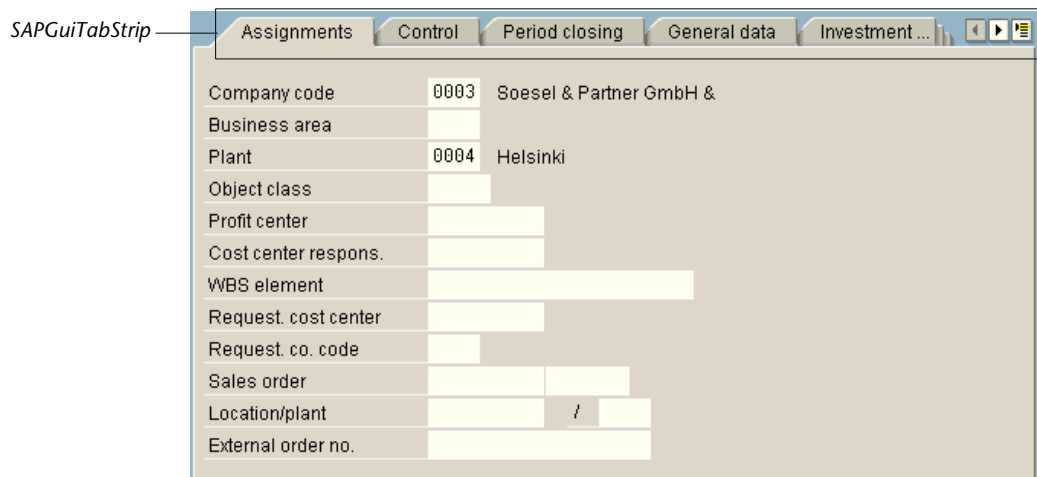
The SAPGuiTable object represents table controls in your SAP Gui for Windows application. SAPGuiTable objects are generally simple and may not have toolbar buttons (other than the **Table Settings** button).



SAPGuiTabStrip Object



The following image shows an SAPGuiTabStrip object. Note that using the tab navigation button  to select a tab is recorded the same way as actually clicking the tab in a tab strip. Clicks on the tab rotation buttons  are not recorded.



Note: There may be cases where a running test or component needs to find an object in a tab strip that is not currently selected. In such an event, QuickTest finds the correct tab strip in which the object is located, selects that tab strip, and activates the object. Then the test or component continues to run. A comment that summarizes this automatic selection of the correct tab strip will appear in the results.

SAPGuiToolbar Object



The SAPGuiToolbar object represents toolbar objects in your SAP Gui for Windows application.

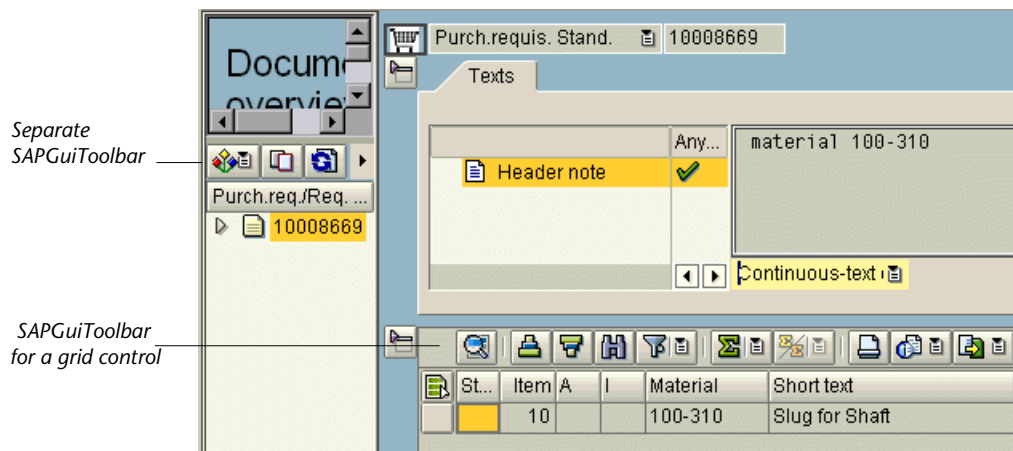
SAPGuiToolbars are recorded for:

- **GuiComponentType - 202**—separate toolbar controls (toolbars that are not an integral part of another object).
- **GuiComponentType - 204**—toolbars that are inside grid controls.

Note: The Object Spy does not locate SAPGuiToolbars that are part of grid controls. They are treated as part of the SAPGuiGrid object.

The toolbar buttons on the main SAP window toolbar are recorded as SAPGuiButton objects.

The example below shows a separate toolbar above a tree control that enables operations on the tree control, but is not associated with it. It also shows a toolbar inside a grid control.

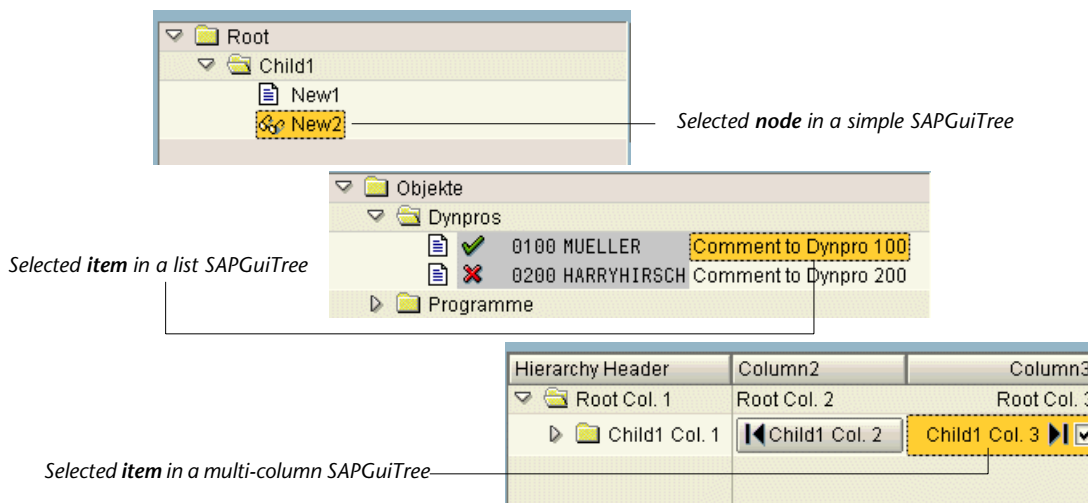


SAPGuiTree Object



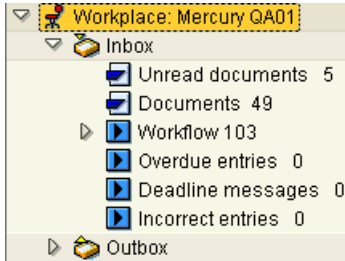
The SAPGuiTree object represents all tree controls in your application, including simple tree controls, list tree controls, and column tree controls. Based on information received from the SAP API, QuickTest recognizes the elements in simple trees as *nodes* and the elements in list and column trees as *items*. Therefore, you may record an **ActivateItem** step for one tree and an **ActivateNode** step for another tree, for example.

The samples below show the three different SAPGuiTree object types:



The names of some nodes or items may change dynamically, for example the number of unread documents in an inbox may change as documents are added or read. Therefore, when you run a step from a test or component on an SAP tree control, the name of the node or item may have changed since the step was recorded.

The tree below contains examples of elements whose names may change over time.



For example, the **Workplace: Mercury QA01** node may have a different name at another time (when a different user logs in). Similarly, all the nodes and items in the **Inbox**, whose names include quantities (numbers of documents or table entries), are likely to change over time.

In order to enable QuickTest to identify these nodes and items when running a test or component, you can use regular expressions when you specify the path of the item or node.

The syntax when writing the value of the node or item name as a regular expression is:

RegExp:=<regular expression>

For example, a value for the **Unread documents** node of the above tree (that uses a regular expression) would look similar to this:

RegExp:=Workplace: .*;Inbox;RegExp:=Unread Documents \d*

The majority of the value is simple text. The regular expression parts of the value in this example are the two asterisks—one before **Inbox** and one at the end of the line—that represent the Workplace name and the number of unread documents respectively.

Each level of the tree can have its own RegExp= value.

The following example illustrates a regular expression in the Keyword View and in the Expert View.

| Item | Operation | Value |
|------------------|------------|--|
| TableTreeControl | SelectNode | "RegExp:=Workplace: .*;Inbox;RegExp:=Unread Documents \d*" |

```
SAPGuiSession("Session").SAPGuiWindow("Business Workplace of").
  SAPGuiTree("TableTreeControl").SelectNode "RegExp:=Workplace: .*;Inbox;
  RegExp:=Unread Documents \d*"
```

For more information on regular expressions, refer to the *QuickTest Professional User's Guide*.

SAPGuiUtil Object



You can enter SAPGuiUtil statements that perform connection operations on your SAP Gui for Windows application during the run session. The SAPGuiUtil object is a reserved object and is not recorded.

SAP Gui for Windows Alternative Recording Mechanism

The QuickTest Professional Add-in for SAP Solutions provides an alternative recording mechanism for specific SAP Gui for Windows objects that do not have built-in test object support. This mechanism uses the **SAPGuiElement** test object and the **Object** method to record all the SAP Gui for Windows API events.

For example, using this recording mechanism, when you double-click an image object, the following statement is recorded in the Expert View:

```
SAPGuiSession("Session").SAPGuiWindow("SAP R/3").
SAPGuiElement("ImageCtrl").Object.doubleClickPictureArea "90","30"
```

The following SAP Gui for Windows objects are automatically recorded using this mechanism:

- Picture or Image controls
- BarChart controls

You can also specify that additional SAP Gui for Windows objects be recorded using this mechanism. This may be useful in cases where QuickTest is not recording an object in your SAP Gui for Windows application as you would like. For more information, refer to the *QuickTest Professional Add-in for SAP Solutions Readme*.

Working with Test Object Methods and Properties

QuickTest provides test object methods and properties that you can use when testing your SAP Gui for Windows applications.

You can record some test object methods while recording on SAP Gui for Windows applications. You can add additional functionality to your test or component by entering statements manually in the Keyword View. In addition, for tests, you can enter statements manually in the Expert View or using the Step Generator.



The QuickTest Professional Add-in for SAP Solutions supports IntelliSense and statement completion in the Expert View, including the display of run-time (native) methods and properties when using the **Object** property in a statement in the Expert View. The add-in also supports generating statements that access run-time methods and properties in the Step Generator.

For more information about the Keyword View, Expert View, and Step Generator, refer to the *QuickTest Professional User's Guide*.



You can use SAP Windows test object property values to confirm that objects in your application look and behave as expected. You can check test object property values using standard checkpoints, or you can retrieve the values of these properties during the run session using the **GetROProperty** method. For more information on creating checkpoints or using the **GetROProperty** method, refer to the *QuickTest Professional User's Guide*.

For detailed information, syntax, and examples of all test object methods and properties, refer to the **SAP Windows** section of the *QuickTest Professional Object Model Reference*.

Accessing Native Methods and Properties in Your SAP Gui for Windows Application



In addition to the SAP Windows test objects methods and properties, you can also use the **Object** property to access native (internal) methods and properties of objects in your SAP Gui for Windows application. The **Object** property is available for all SAP Windows objects.

Tip: You can use the Object Spy to view the native (run-time) methods and properties of an object in your application.

For example, you can use the grid's **setCurrentCell** method to set a row number to -1 (according to the SAP API).

```
SAPGuiSession("Session").SAPGuiWindow("Organization").
    SAPGuiGrid("GridViewCtrl").Object.setCurrentCell -1, "ADD_FIELD3"
```

The **Object** property is also useful for checking the value of properties that are not available using a standard checkpoint in your SAP Gui for Windows application.

For example, you can use the grid control's native **selectionMode** property to determine the types of selections that the grid supports. If the method returns the value **RowsAndColumns**, this indicates that the grid supports multiple row selection. In this case, you can use the **SelectRowsRange** test object method to select multiple rows in the grid.

```
GridMode = SAPGuiSession("Session").SAPGuiWindow("Vorg_nge").
    SAPGuiGrid("GridViewCtrl").Object.selectionMode
msgbox GridMode
If GridMode = "RowsAndColumns" then
    SAPGuiSession("Session").SAPGuiWindow("Vorg_nge").
        SAPGuiGrid("GridViewCtrl").SelectRowsRange 2,6
    SAPGuiSession("Session").SAPGuiWindow("Vorg_nge").
        SAPGuiButton("Enter").Click
End If
```

For more information on using the **Object** property, refer to the *QuickTest Professional User's Guide*.

Part III

Testing Web-based SAP Applications

7

Recording and Running Tests or Components on Web-based SAP Applications

The QuickTest Professional Add-in for SAP Solutions supports testing on Web-based SAP applications, including SAP Gui for HTML and SAP Enterprise Portal.

This chapter explains how to use QuickTest to record and run tests on SAP Web applications. For more information about working with QuickTest, refer to the *QuickTest Professional User's Guide*. For information about recording and running tests and components on SAP Gui for Windows applications, see “Recording and Running Tests and Components on SAP Gui for Windows Applications” on page 57.

This chapter describes:

- About Recording and Running Tests or Components on Web-based SAP Applications
- Understanding the SAP Web Test Object Model
- Recording Tests on Web-based SAP Applications
- Running Tests on Web-based SAP Applications

About Recording and Running Tests or Components on Web-based SAP Applications

QuickTest Professional can record and run tests and components on SAP Gui for HTML and SAP Enterprise Portal applications using special test objects and methods that have been customized for SAP.

As you record a test or component on your SAP Gui for HTML or SAP Enterprise Portal application, QuickTest records the operations you perform.

The QuickTest Professional Add-in for SAP Solutions recognizes special SAP Windows objects such as frames, table controls, iViews, and portals. For more information on the supported objects, see “Adding SAP Web Statements to Your Test or Component,” on page 153.

You can use the Active Screen (for tests and scripted components only) and the Web tab of the Options dialog box to set additional preferences that control the way QuickTest records and runs your test. For more information, refer to the *QuickTest Professional User's Guide*.

You run tests or components containing SAP Web test objects in the same way as you run any other test or component. You can view the details of your run session in the Test Results window.

Note: When you load the QuickTest Professional Add-in for SAP Solutions, the settings in the Web Event Recording Configuration dialog box (**Tools > Web Event Recording Configuration**) are automatically customized. You do not need to make any Web event configuration changes.

Understanding the SAP Web Test Object Model

The *test object model* is the set of object types or *classes* that QuickTest uses to represent the objects in your application. Each test object class has a list of properties that can uniquely identify objects of that class and a set of relevant methods that QuickTest can perform during a run session.

A *test object* is an object that QuickTest creates in the test or component to represent the actual object in your application and to store information about that object. This information, stored in the object repository, helps QuickTest identify and check the object during the run session.

A *run-time object* is the actual object in your SAP Gui for HTML or SAP Enterprise Portal application on which methods are performed during the run session.

When you perform an operation on your SAP Gui for HTML or SAP Enterprise Portal application while recording a test, QuickTest:

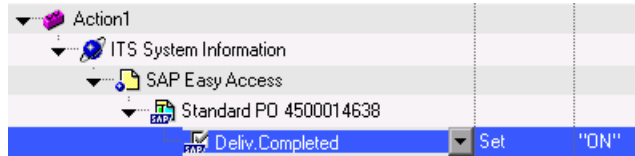
- identifies the object on which you performed the operation and creates the appropriate test object in the test or component.
- reads the current property values of the object in your application and stores them in the object repository as the test object's property values.
- chooses a unique name for the object, generally using the value of one of its prominent properties.
- records the operation (method) that you performed on the object.
- displays the operation as a step in the Keyword View (tests and components) and as a statement in the Expert View (tests only).

For example, suppose you check a **Deliv.Completed** check box. QuickTest creates an SAPCheckBox test object with the name **Deliv.Completed** and records the following properties and values as the description for the **Deliv.Completed** SAPCheckBox.

| Type | Property | Value |
|------|-------------------|-----------------|
| ABC | html tag | INPUT |
| ABC | sap attached text | Deliv.Completed |

It also records that you performed a **Set** method to turn ON the check box.

QuickTest displays your step in the Keyword View like this:



QuickTest displays your step in the Expert View like this:

```
Browser("ITS System Information").Page("SAP Easy  
Access").SAPFrame("Standard PO 4500014638").  
    SAPCheckBox("Deliv.Completed").Set "ON"
```

When you run a test or component, QuickTest identifies each object in your application by its test object class and its *description*—the set of test object properties and values used to uniquely identify the object.

In the above example, during the run session, QuickTest searches the object repository for the SAPCheckBox object with the name `Deliv.Completed`. Based on the description it finds for the object (html tag = INPUT, sap attached text = Deliv.Completed), QuickTest then looks in the application for an SAPCheckBox object that matches this description. When it finds the object, it performs the **Set** method on it to change the value of the field to ON (checks the check box).

Tip: You can view all the properties and methods of any object in your SAP Gui for HTML or SAP Enterprise Portal application using the Object Spy.


For general information on the test object model, the object repository, and the Object Spy, refer to the *QuickTest Professional User's Guide*.

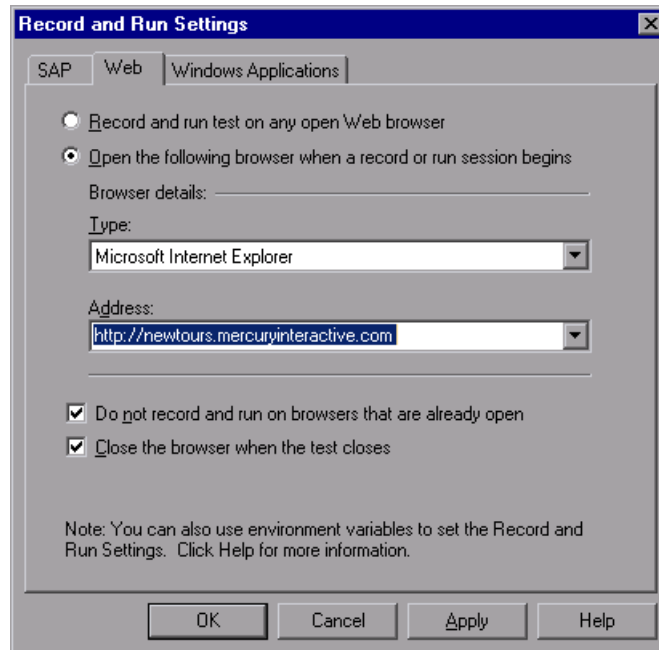
For more information on SAP Web test objects and methods, refer to the **SAP Web** section of the *QuickTest Professional Object Model Reference*.

Recording Tests on Web-based SAP Applications

Before you begin recording tests on Web-based SAP applications, you can define your required recording settings. This enables you to specify the browser on which you want QuickTest to record, specify any environment variables, and select the required Web options to optimize performance.

Setting Record and Run Settings for Tests on Web-based SAP Applications

 You can use the Web tab of the Record and Run Settings dialog box to instruct QuickTest to open the URL of your SAP Gui for HTML or SAP Enterprise Portal application each time you begin a recording session. Alternatively, you can instruct QuickTest to record on any open browser.



Note: The Web Add-in must be loaded to view the Web tab of the Record and Run Settings dialog box. For information on loading add-ins, see “Activating Add-in Support for SAP Solutions” on page 17. The SAP Tab is not used for recording and running tests on SAP Gui for HTML or SAP Enterprise Portal applications.

In addition to setting the appropriate settings in the Web tab, you should confirm that the other tabs in the dialog box have the appropriate settings. The following settings are recommended:

- **Windows tab**—Choose **Record and run on these applications (opened when a session begins)** and confirm that the list of Windows applications is empty.
- **Other tabs (if displayed)**—Choose the option to record and run on any open application (upper radio button of each tab).

While these settings do not directly affect your record or run sessions when working with SAP Gui for HTML or the SAP Enterprise Portal application, these settings prevent you from inadvertently recording operations performed on Windows applications (such as e-mail) during your recording session. These settings also prevent QuickTest from opening unnecessary applications when you record or run tests on an SAP Gui for HTML or SAP Enterprise Portal application.

For more information on the Record and Run settings dialog box, refer to the *QuickTest Professional User's Guide*.

When you record an operation on an SAP Gui for HTML or SAP Enterprise Portal object, QuickTest inserts a step with the relevant SAP Web icon in the Keyword View and adds a corresponding statement in the Expert View.

For example, if you use the SAP Enterprise Portal Drag & Relate feature to drag an item to the iPanel while recording, QuickTest displays this test in the Keyword View, as follows:

| | | | |
|-------------------------|-----------|--------------------|---|
| ▼ SAP Enterprise Portal | | | |
| ▼ Desktop | | | |
| Pete | FireEvent | "ondragstart" | Simulate the "ondragstart" event on the "Pete" link. |
| ▼ ToolBar_2 | | | |
| SAPNavigationBar | Relate | "My Portal;Tra..." | Drag the selected item and drop it onto the last node of item "My Porti |

QuickTest records this step in the Expert View as:

```
Browser("Browser").Page("SAP Enterprise Portal").Frame("Desktop").
    Link("Pete").FireEvent "ondragstart"
Browser("Browser").Page("SAP Enterprise Portal").
    Frame("ToolBar_2").SAPNavigationBar("SAPNavigationBar").Relate
        "My Portal;Trash"
```

Note: When you select a step in the Keyword View or the Expert View, the corresponding object is highlighted in the Active Screen.

For more information, refer to the *QuickTest Professional User's Guide*.

Defining Application Details Environment Variables



You can use environment variables to specify the applications you want to use for recording and running your test. These variables can also be used in external library files for automation scripts. You use environment variables in SAP Web applications in the same way as you use them in any other Web application.

Use the variable name listed in the table below to define the Web browser and URL to open:

| Option | Variable Name | Description |
|---------|---------------|--|
| Type | BROWSER_ENV | The browser type to open. Possible values: IE, NS, NS6 |
| Address | URL_ENV | The SAP Web application address to display in the browser. |

For more information on defining and working with environment variables, refer to the *QuickTest Professional User's Guide*.

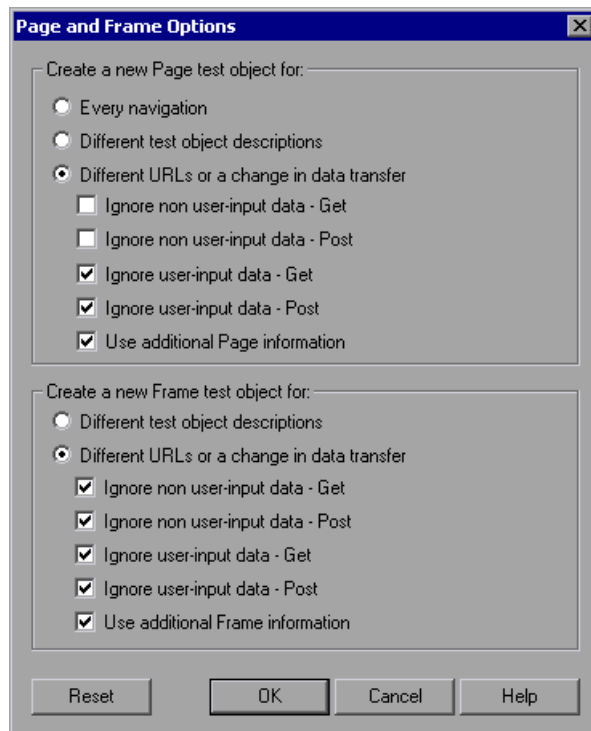
Setting Web Testing Settings

Before you begin to record and run tests, you can configure the settings that are best-suited for testing Web-based SAP applications. Applying these recommended settings helps to optimize QuickTest performance.

Page and Frame Options

Page/Frame Options..

In the Page and Frame Options dialog box (**Tools > Options > Web tab > Page/Frame Options** button), select the following options:



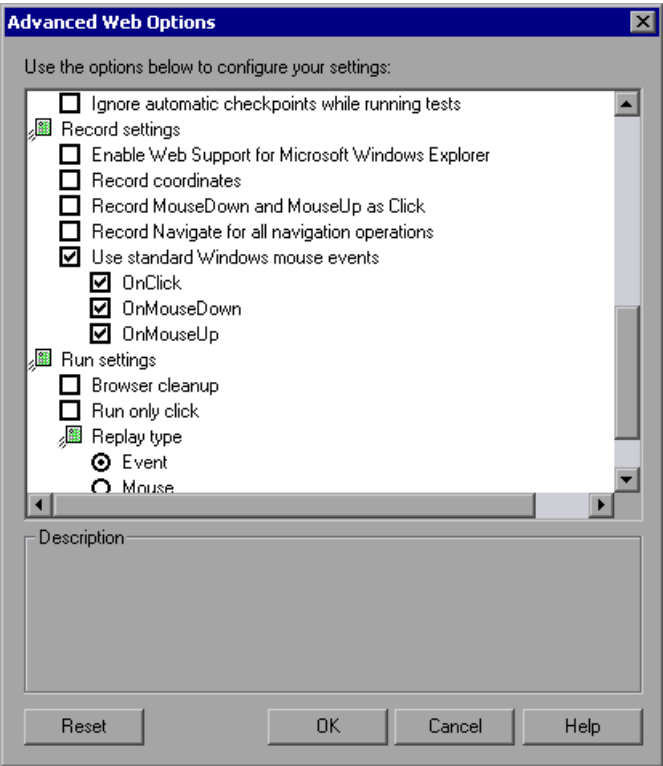
| Area | Options |
|--|---|
| Create a new Page test object for: | Different URLs or a change in data transfer Instructs QuickTest to create a new Page object only when the page URL changes, or if the URL stays the same and data that is transferred to the server changes, according to the data types and transfer methods you select. Make sure that only the following check boxes are selected: <ul style="list-style-type: none"> • Ignore user-input data - Get • Ignore user-input data - Post • Ignore additional Page information |
| Create a new Frame test object for: | Different URLs or a change in data transfer Note: This option is selected by default. Instructs QuickTest to create a new Frame object only when the page URL changes, or if the URL stays the same and data that is transferred to the server changes, according to the data types and transfer methods you select. Make sure that all of the check boxes in this section are selected. |

These Page and Frame settings are especially suited to testing Web-based SAP applications. For general information on setting Web testing options, refer to the *QuickTest Professional User's Guide*.

Advanced Web Options

Advanced...

In the Advanced Web Options dialog box (**Tools > Options > Web tab > Advanced Web Options** button), select the following **Record setting** options:



| Area | Options |
|-----------------|---|
| Record settings | Select the Use standard Windows mouse events check box, as well as the following check boxes: <ul style="list-style-type: none">• OnClick• OnMouseDown• OnMouseUp <p>Note: These options are selected by default. This instructs QuickTest to use these standard Windows mouse events instead of browser events.</p> |

These recording settings are especially suited to testing Web-based SAP applications. For general information on setting Web testing options, refer to the *QuickTest Professional User's Guide*.

Running Tests on Web-based SAP Applications

You run tests containing SAP Web test objects in the same way as you run any other test or component. You can view the results of your run session in the Test Results window.



You can use the Web tab of the Record and Run Settings dialog box to instruct QuickTest to open the URL of your SAP Gui for HTML or SAP Enterprise Portal application each time you begin running your test, or you can instruct QuickTest to run only on open browsers.

For more information on running tests or components, refer to the *QuickTest Professional User's Guide*.

8

Enhancing Your SAP Web Test



After you create your test, you can enhance it by adding checkpoints, retrieving output values, parameterizing values.

Note: All of the information in this chapter is relevant for tests and scripted components only.

This chapter describes:

- About Enhancing Your SAP Web Test
- Checking SAP Web Objects and Outputting Values

About Enhancing Your SAP Web Test

After you record a test, you can use a variety of options to enhance your test. This section describes some of the options you can use to create comprehensive tests for your SAP Gui for HTML or SAP Enterprise Portal application.

Note: Checkpoints and output values are not available for components.

- You can add *checkpoints* to your test to check the properties of an SAP Web object. A checkpoint is a step in your test that compares the values of the specified property during a run session with the values stored for the same test object property within the test. This enables you to identify whether or not your SAP Gui for HTML or SAP Enterprise Portal application is functioning correctly. For more information about checkpoints, see “Checking SAP Web Objects and Outputting Values” on page 147.

The results of the checkpoint can be viewed in the QuickTest Test Results window. For more information about the Test Results window, refer to the *QuickTest Professional User's Guide*.

- You can *parameterize* your test to replace fixed values with values from an external source during your run session. The values can come from a Data Table, environment variables you define, or values that QuickTest generates during the run session. For more information about parameterizing tests, refer to the *QuickTest Professional User's Guide*.
- You can specify test, action, or component parameters to pass values into and from your test or component, and between actions in your test. For more information, refer to the *QuickTest Professional User's Guide*. For components, you can also use *local* parameters, which are parameters that are accessible only to steps within the current component. (You cannot use them to pass data between components.) For more information, refer to the *QuickTest Professional New Features Guide*.
- You can retrieve values from your test and store them in the Data Table as *output values*. You can subsequently use these values as input parameters in your test. This enables you to use data retrieved during one part of a test in later parts of the test.
- You can insert statements containing SAP Web test object methods in the Expert View or in the Keyword View by using the **Insert > Step > Step Generator** option to open the Step Generator. This enables you to include other operations in your test that may not have been added while recording, such as retrieving property values, checking that objects exist, and checking other elements of your application. For more information about SAP Web test object methods and properties, see “Working with SAP Web Test Objects,” on page 155, and the **SAP Web** section of the *QuickTest Professional Object Model Reference*.









Checking SAP Web Objects and Outputting Values









You use checkpoints to check the properties of SAP Web objects similar to the way you check the properties of corresponding Web objects. You can also output property or text values from the objects in your SAP Gui for HTML or SAP Enterprise Portal application to use in your test.

The following checkpoint and output value types are supported when testing SAP Web objects:

- **Standard** checkpoints check the property value(s) of an object in your SAP Gui for HTML or SAP Enterprise Portal application. You insert standard checkpoints for SAP Web objects in the same way you create standard checkpoints for Web objects. You can insert standard checkpoints while recording or while editing your test. You can also create a standard output value from an object property.
- **Table** checkpoints are a special type of standard checkpoint that enable you to check the cells of a table in your SAP Gui for HTML or SAP Enterprise Portal application.
- **Text** checkpoints check that a text string is displayed accurately in the appropriate place in your SAP Gui for HTML or SAP Enterprise Portal application. You can also create a text output value from a text string.
- **Bitmap** checkpoints check an area of your SAP Gui for HTML or SAP Enterprise Portal application as a bitmap. You can use bitmap checkpoints to compare an object or selected area with the bitmap stored when the test or component was recorded and check that the area is displayed accurately in your SAP Gui for HTML or SAP Enterprise Portal application.

The following table lists the SAP Web test objects on which you can insert a checkpoint or output value:

| Icon | Test Object Class | Icon | Test Object Class |
|---|-------------------|---|-------------------|
|  | SAPButton |  | SAPNavigationBar |
|  | SAPCheckBox |  | SAPOKCode |
|  | SAPDropDownMenu |  | SAPPortal |
|  | SAPEdit |  | SAPRadioGroup |

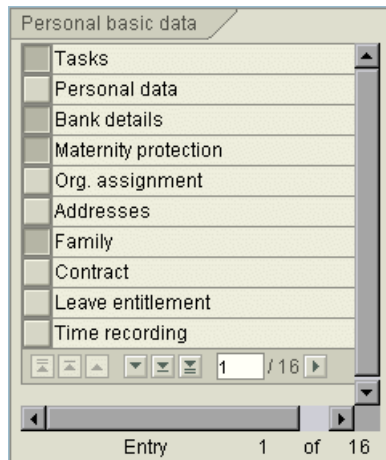
| Icon | Test Object Class | Icon | Test Object Class |
|---|-------------------|---|-------------------|
|  | SAPFrame |  | SAPStatusBar |
|  | SAPiView |  | SAPTable |
|  | SAPList |  | SAPTabStrip |
|  | SAPMenu |  | SAPTreeView |

For more information about table checkpoints, see “Checking SAP Web Tables” on page 148. For more information about standard, text, and bitmap checkpoints and output values, refer to the *QuickTest Professional User’s Guide*.

Checking SAP Web Tables

You check tables in your SAP Gui for HTML application using the Table Checkpoint Properties dialog box. You can add a table checkpoint while recording or editing your test.

For example, you can check the content of the following table:



When working with tables in your SAP Gui for HTML application, you can also pool all of the available data from a table into an external file. For more information, see “Spooling Data from an SAP Gui for HTML Application Table” on page 152.

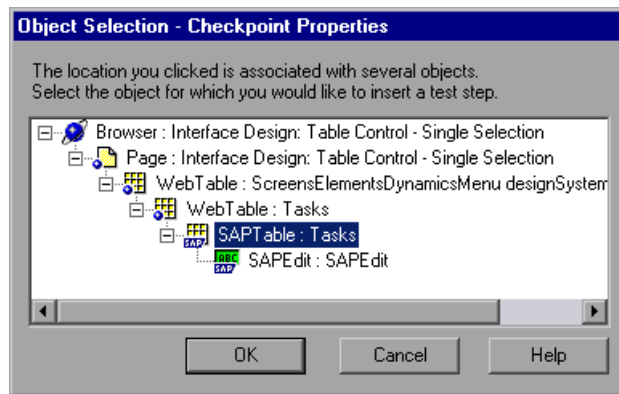
To create a checkpoint on an SAP Gui for HTML application table while recording:



- 1 Choose **Insert > Checkpoint > Standard Checkpoint** or click the **Insert Checkpoint** button.

The QuickTest window is minimized and the mouse pointer turns into a pointing hand.

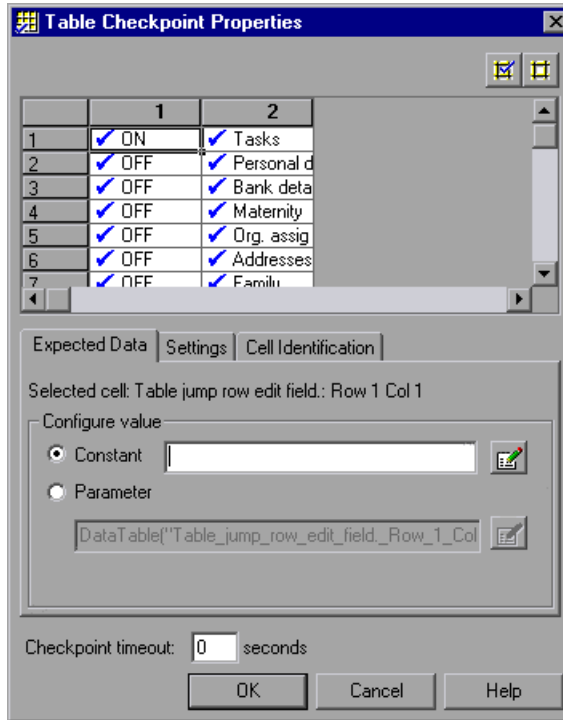
- 2 Click a table you want to check. If the location you select is associated with more than one object, the Object Selection - Checkpoint Properties dialog box opens.



- 3 Select an **SAPTable** object from the displayed object tree.

Note: You cannot insert a checkpoint on a child object (inner object) of a table.

- 4 Click **OK**. The Table Checkpoint Properties dialog box opens.



Note: If a table has a column header row, it is counted as the first row in the table.

- 5 Specify the options for your checkpoint. For more information about table checkpoint options, refer to the *QuickTest Professional User's Guide*.
- 6 Click **OK** to close the dialog box. A checkpoint statement is added for the selected object in the Keyword View and in the Expert View.

To create a checkpoint on an SAP Gui for HTML or SAP Enterprise Portal application table while editing your test:

- 1** Depending on whether the table object is already in your step or component, perform one of the following:
 - If you have already recorded a step on the table object you want to check, right-click the step for the table object in the Keyword View or Expert View and choose **Insert Standard Checkpoint**.



- If you have not recorded a step on the table object you want to check, but you have an Active Screen capture that displays the table object, you can add a table checkpoint only if the Active Screen **Capture level** was set to **Complete** when the object was captured and the **Active Screen** button is currently selected. (You set the **Capture level** in the Active Screen tab of the Options dialog box (**Tools > Options**). For more information, refer to the *QuickTest Professional User's Guide*.)

Click the step in your test or component in which you want to add a checkpoint. The Active Screen displays the page corresponding to the highlighted step. Right-click the table object in the Active Screen and choose **Insert Standard Checkpoint**. The Object Selection - Checkpoint Properties dialog box opens. Select the SAP Web table object from the displayed object tree and click **OK**.

- 2** The remainder of the procedure is similar to that for creating a checkpoint while recording. For more information, continue with step 3 on page 149.

Note: If the table object is not open in your SAP Web application when you create the checkpoint, the Table Checkpoint Properties dialog box contains only the Properties tab and the option to select the type of information to check (content or properties) is disabled.

Spooling Data from an SAP Gui for HTML Application Table



If you want to spool all the available data from an SAP Gui for HTML application table into an external file, use the **GetCellData** method to loop through each cell in the table. You can then save the information to an external file.

The following example uses the **GetCellData** method to list the data of each cell in a table of 10 rows and 10 columns:

```
For i=1 to 10
  For j=1 to 10
    Dat=Browser("ITS System Informati").Page("Table control").
    SAPTable("MySAPTable").GetCellData (i, j)
    'Enter line(s) of code that use the value of the returned Dat variable
  Next
Next
```

For more information on the **GetCellData** method, see the **SAP Web** section of the *QuickTest Professional Object Model Reference*.

9

Adding SAP Web Statements to Your Test or Component

After you create your test or component, you can add SAP Web objects, methods, and properties to it. You can add statements to your test or component using the Keyword View, and you can add statements to your test using the Expert View or the Step Generator. This enables you to create a more comprehensive test or component.

This chapter describes:

- About Adding SAP Web Statements to Your Test or Component
- Working with SAP Web Test Objects
- Working with Test Object Methods and Properties
- Accessing Native Methods and Properties in Your Web-based SAP Applications

About Adding SAP Web Statements to Your Test or Component

A test or component consists of statements coded in Microsoft VBScript. These statements are composed of objects, methods, and/or properties that instruct QuickTest to perform operations or retrieve information. When you record, these statements are generated automatically in response to input to the application.

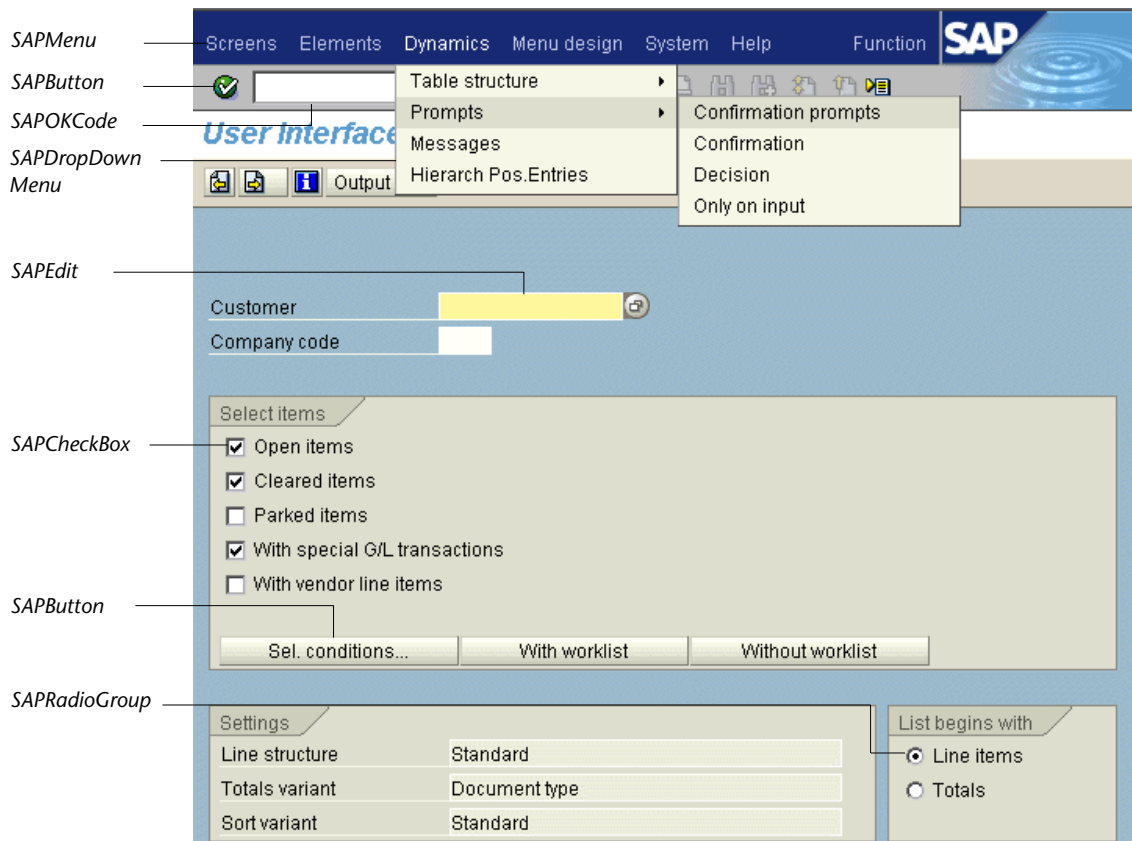
In addition, you can add and modify statements manually in the Keyword View (tests and components) or in the Expert View (tests only). You can also mix recorded and programmed statements in the same test.



After you record a basic test and enhance it with checkpoints, parameters, and output values, you may also want to add statements containing test object methods using the Step Generator (**Insert > Step > Step Generator**). You can also enter VBScript programming statements in the Expert View. This enables you to include operations in your test that you did not, or could not, add while recording. These can include retrieving property values, checking that objects exist, enumerating objects, and checking other elements of your application.

Working with SAP Web Test Objects

QuickTest has a set of SAP Web test object classes that represent objects in your application on which you can record operations, or add to your test or component manually. SAP Web test objects are customized to make it easy for you to identify and work with the objects in your test. As part of this customization, the methods and properties recorded for these objects are somewhat different than those recorded for comparable Web objects. The example below shows a window containing several common SAP Web objects:



This section describes how QuickTest identifies the objects in an Web-based SAP application and provides information on each of the following objects:

- SAPButton (see page 157)
- SAPCheckBox (see page 157)
- SAPDropDownMenu (see page 157)
- SAPEdit (see page 157)
- SAPFrame (see page 157)
- SAPiView (see page 158)
- SAPList (see page 158)
- SAPMenu (see page 158)
- SAPNavigationBar (see page 159)
- SAPOKCode (see page 159)
- SAPPortal (see page 160)
- SAPRadioGroup (see page 160)
- SAPStatusBar (see page 160)
- SAPTable (see page 161)
- SAPTabStrip (see page 162)
- SAPTreeView (see page 163)

SAPButton



The SAPButton test object represents SAP Gui for HTML and SAP Enterprise Portal application buttons, including icons, toolbar buttons, regular buttons, buttons with text, and buttons with text and an image.

SAPCheckBox



The SAPCheckBox test object represents SAP Gui for HTML and SAP Enterprise Portal application toggle buttons, including check boxes and images that can be pressed and released.

SAPDropDownMenu



The SAPDropDownMenu test object represents menus that are opened by clicking a menu icon within an SAP Gui for HTML or SAP Enterprise Portal application.

SAPEdit



The SAPEdit test object represents SAP Gui for HTML and SAP Enterprise Portal application edit boxes, including single-line edit boxes and multi-line edit boxes (text area).

SAPFrame



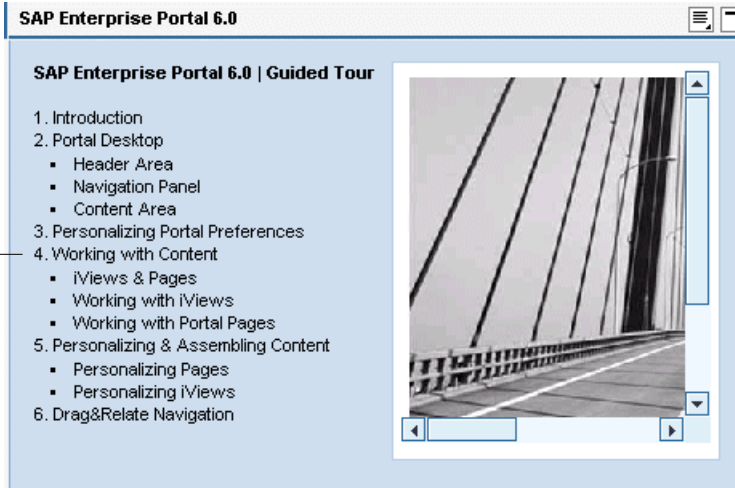
The SAPFrame test object represents SAP Gui for HTML application frames.

SAPiView



The SAPiView test object represents iView frame objects within SAP Enterprise Portal desktops.

SAPiView



SAPList



The SAPList test object represents SAP Gui for HTML and SAP Enterprise Portal application drop-down boxes and multiple selection lists.



SAPMenu

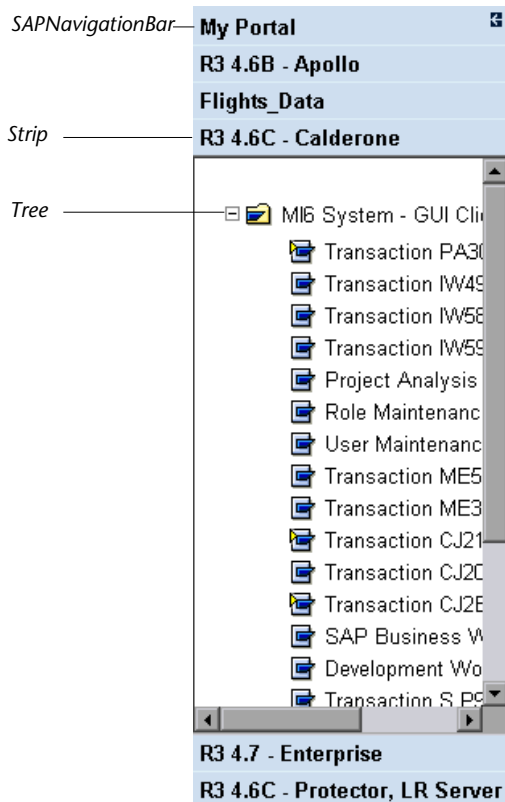


The SAPMenu test object represents SAP Gui for HTML application top-level menus. When you click a menu item, the SAPMenu test object records the full path of the selected item.

SAPNavigationBar



The SAPNavigationBar test object represents an iPanel—the navigation pane displayed in an SAP Enterprise Portal 5.0 desktop. An iPanel is divided into sections, represented by expandable strips. Each strip contains a hierarchical tree of items. When you record an operation on the SAPNavigationBar test object, it records the strip as part of the path, for example, `SAPNavigationBar("SAPNavigationBar").Select "R3 4.6C - Calderone;MI6 System - GUI Client Roles;Transaction PA30"`.



SAPOKCode

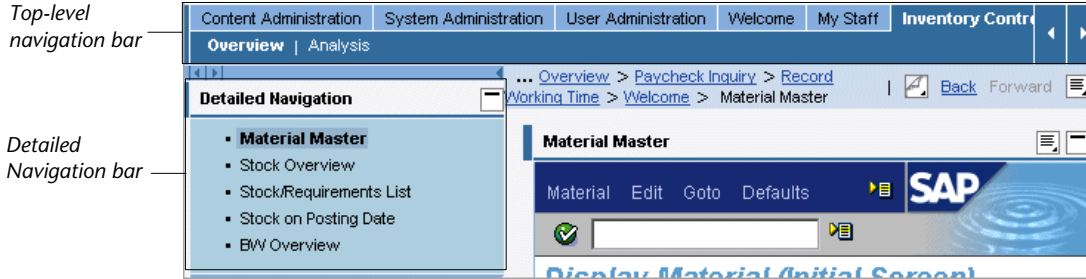


The SAPOKCode test object represents the edit box in a SAP Gui for HTML application in which you enter commands to navigate to the desired transaction.

SAPPortal



The SAPPortal test object represents the SAP Enterprise Portal 6.0 desktop objects—top-level navigation bar and Detailed Navigation bar.



SAPRadioGroup



The SAPRadioGroup test object represents SAP Gui for HTML and SAP Enterprise Portal application radio button groups.

When possible, QuickTest records radio button selections using the attached text property of the selected radio button. If two or more radio buttons in the group have identical attached text values (or no attached text), QuickTest records the radio button index instead. For example:

'This radio button selection uses the attached text property.
 Browser("System Messages").Page("User Interface_52").SAPRadioGroup("Basic personal").Select "Personal data"

'This radio button selection uses the radio button index.
 Browser("System Messages").Page("User Interface_46").SAPRadioGroup("Address_2").Select "#0"

When writing **SAPRadioGroup.Select** statements manually in the Expert View, you can use either argument type to identify the radio button.

SAPStatusBar



The SAPStatusBar test object represents a status bar in an SAP Gui for HTML application.

Standard Order 9596 has been saved

SAPTable



The SAPTable test object represents SAP Gui for HTML application table objects. Each cell can contain an SAP Web or Web object, such as a check box or combo box. The value of this object determines the value in the cell.

| | A... | Fl... | Depart.city | D... | Destination cit | D... | Flight... | Depart... |
|--|------|-------|---------------|------|-----------------|------|-----------|-----------|
| | AA | 17 | NEW YORK | JFK | SAN FRANCISCO | SFO | | 13:30:00 |
| | AA | 64 | SAN FRANCISCO | SFO | NEW YORK | JFK | | 09:00:00 |
| | AZ | 555 | ROME | FCO | FRANKFURT | FRA | | 19:00:00 |
| | AZ | 788 | ROME | FCO | TOKYO | TYO | | 12:00:00 |
| | AZ | 789 | TOKYO | TYO | ROME | FCO | | 11:45:00 |
| | AZ | 790 | ROME | FCO | OSAKA | KIX | | 10:35:00 |

When you perform an operation on an object within a table cell, QuickTest records the changes to the data in the cell, rather than recording the method performed on the object within the cell.

For example, if you select a table cell containing the text **San Francisco** from a list of departure cities (in the **Depart.city** column), QuickTest records a **SetCellData** method for the table cell, indicating the data that was set in the cell. If the object in the cell is later changed to an edit object, QuickTest enters the data (**San Francisco**) into the cell (in the edit object) during the run session. Thus, you do not need to modify your test even when the object inside the cell changes.

Note: You perform operations on a table object and not on the inner objects contained in the table object, for example, a check box or edit box.

SAPTabStrip



The SAPTabStrip test object represents SAP Gui for HTML tab strip objects (objects that enable switching between multiple tabs). You choose the required tab by clicking its title. If a tab is not visible, you can display it by clicking the left or right arrows.

SAPTabStrip

| | | |
|----------------------|------|-------------------------|
| Company code | 0003 | Soesel & Partner GmbH & |
| Business area | | |
| Plant | 0004 | Helsinki |
| Object class | | |
| Profit center | | |
| Cost center respons. | | |
| WBS element | | |
| Request. cost center | | |
| Request. co. code | | |
| Sales order | | |
| Location/plant | | / |
| External order no. | | |

Note: While recording a test, arrow clicks are ignored. However, while running a test, a tab that is not visible will still be selected correctly by QuickTest.

Selecting a tab through the dropdown menu (the rightmost button on the tab strip) is handled exclusively via the **SAPButton.SelectMenuItem** method.

SAPTreeView

The SAPTreeView test object represents SAP Gui for HTML and SAP Enterprise Portal application tree objects.



QuickTest can record selection and activation operations on any item in an SAPTreeView object. Although QuickTest does not record expand and collapse operations, it can select and activate items in an SAPTreeView object during the run session, regardless of whether the tree is expanded or collapsed.

For more information on all the SAP Web test objects and methods, refer to the **SAP Web** section of the *QuickTest Professional Object Model Reference*.

Working with Test Object Methods and Properties


QuickTest provides test object methods and properties that you can use when testing your Web-based SAP applications.

You can record some test object methods while recording on Web-based SAP applications. You can add additional functionality to your test or component by entering statements manually in the Keyword View. In addition, for tests, you can enter statements manually in the Expert View or using the Step Generator.




The QuickTest Professional Add-in for SAP Solutions supports IntelliSense and statement completion in the Expert View, including the display of run-time (native) methods and properties when using the **Object** property in a statement in the Expert View. The add-in also supports generating statements that access run-time methods and properties in the Step Generator.

For more information about the Keyword View, Expert View, and Step Generator, refer to the *QuickTest Professional User's Guide*.

 You can use SAP Web test object property values to confirm that objects in your application look and behave as expected. You can check test object property values using standard checkpoints, or you can retrieve the values of these properties during the run session using the **GetROProperty** method. For more information on creating checkpoints or using the **GetROProperty** method, refer to the *QuickTest Professional User's Guide*.

For detailed information, syntax, and examples of all test object methods and properties, refer to the **SAP Web** section of the *QuickTest Professional Object Model Reference*.

Accessing Native Methods and Properties in Your Web-based SAP Applications

 In addition to the SAP Web-specific test object methods and properties, you can also use the **Object** property to access native (run-time) methods and properties of the objects in your SAP Gui for HTML or SAP Enterprise Portal application. The **Object** property is available for all SAP Web objects.

Tip: You can use the Object Spy to view the native (run-time) methods and properties of an object in your application.

For example, suppose the **clearAttributes** method is supported as a native (run-time) method for an SAPList object. You can use the following statements to activate the native (run-time) **clearAttributes** method.

```
dim SAPList
set SAPList = Browser("ITS System Informati").Page("User Interface_19").
SAPList("Language").Object
SAPList.clearAttributes
```

The **Object** property is also useful for checking the value of properties that are not available using a standard SAP Web checkpoint.

The following example uses the **Object** property to access the **Search** button, retrieve its **title** and **readyState** property values, and display this information in message boxes.

```
set obj = Browser("ITS System Informati").Page("System Messages").  
SAPButton("Search").Object  
msgbox obj.title  
msgbox obj.readyState
```

For more information on using the **Object** property, refer to the *QuickTest Professional User's Guide*.

Part IV

Integrating with eCATT

10

Understanding QuickTest-eCATT Integration

In addition to Quality Center (formerly known as TestDirector), Mercury Interactive's Web-based test management tool, you can also store and manage QuickTest tests in the SAP Extended Computer Aided Test Tool (eCATT).

Note: You cannot save or maintain business components or scripted components using eCATT.

This chapter describes:

- About Managing Tests in eCATT
- Understanding eCATT Testing Modes

About Managing Tests in eCATT

After creating tests for your SAP application, you can store and manage them in a test management tool. Depending on your needs, you may choose to use Mercury Interactive's Quality Center, or SAP eCATT.

For more information on integrating with Quality Center, refer to the *QuickTest Professional User's Guide* and your Quality Center documentation.

You must configure your eCATT server to work with QuickTest. Once the server is configured, you can connect to eCATT from QuickTest (standalone mode) and you can connect to QuickTest from eCATT (integrated mode).

From QuickTest or from eCATT you can create QuickTest tests, store tests and associated resource files in the eCATT database, edit tests, run tests, and review test results. You can also call and pass values from an eCATT test script to a QuickTest test.

Notes:

eCATT support is available only when the SAP Front End software is installed on your computer (including support for Unicode), both the Add-in for SAP Solutions and **SAP eCATT integration** components are installed from the QuickTest Professional Add-in for SAP Solutions CD, and the SAP Add-in is loaded. For more information, see Chapter 1, “Installing the Add-in for SAP Solutions.”

You cannot connect to both eCATT and Quality Center simultaneously.

Understanding eCATT Testing Modes

You can work with tests stored in eCATT in standalone or integrated mode. The current mode is indicated in the QuickTest title bar.

Standalone mode—The test is stored in eCATT, but was opened from within the QuickTest interface.

Integrated mode—The test was opened for editing or running from within the eCATT interface.

The table below describes the basic differences between these two modes.

| | Standalone Mode | Integrated Mode |
|------------------------------|--|--|
| eCATT-QuickTest connection | Connect to eCATT from QuickTest using the eCATT Connection dialog box. | eCATT automatically establishes the eCATT-QuickTest connection. |
| Available QuickTest features | All QuickTest features are available. You can open and work with any test in eCATT or in the file system. | You can work only with the currently open test. File > Open , File > New , and the Recent files list options are disabled. If you choose File > Save As , QuickTest warns you that it will disconnect from eCATT and switch QuickTest to standalone mode. |
| Resource files | When you open the test, you can also edit and save all the test's resource files, including those stored in eCATT. | When you open the test, test resources stored in eCATT are opened in read-only mode. |
| Save location | Tests and uploaded files are automatically saved to the local package (\$TMP) in eCATT. | You can save tests to any package (including non-local packages). |
| eCATT dependence | Although QuickTest is connected to eCATT, you can work and navigate in eCATT independently. | eCATT is locked while the test is open in QuickTest. To release eCATT, close QuickTest. |
| Test results | All test results are stored in the file system. They cannot be accessed from your eCATT log list. | Test results are stored to the network drive you specify in the eCATT tab of the Options dialog box. You can access the test results from the eCATT log. |

For more information on working in standalone mode, see Chapter 12, “Working with eCATT in Standalone Mode.”

For more information on working in integrated mode, see Chapter 13, “Working with eCATT in Integrated Mode.”

11

Configuring eCATT to Work with QuickTest

Before you can use the eCATT-QuickTest integration features available with the QuickTest Professional Add-in for SAP Solutions, you (or an eCATT system administrator) must install the appropriate support package and configure the eCATT server to work with QuickTest.

For information on specific SAP server version and support package requirements, refer to the *QuickTest Professional Add-in for SAP Solutions Readme*.

This chapter describes:

- About Configuring eCATT to Work with QuickTest
- Setting External Tool Parameters in the ECCUST_ET Table
- Applying Necessary Roles or Profiles to eCATT-QuickTest Users

About Configuring eCATT to Work with QuickTest

To enable eCATT-QuickTest integration, an SAP user with administrative privileges must:

- Update the eCATT server with the appropriate support package. Contact SAP or your SAP representative to receive the necessary support package.
- Set external tool parameters in the ECCUST_ET table.
- Apply the necessary roles or profiles to each user who wants to work with QuickTest and eCATT.

Setting External Tool Parameters in the ECCUST_ET Table

To enable eCATT to communicate with QuickTest, certain values must be set for the external tool parameters in the ECCUST_ET table. This procedure should be performed only once in the system.

To set parameter values in the ECCUST_ET Table:

- 1** Navigate to transaction **se17**. The General Table Display window opens.
- 2** In the **Table Name** box, enter ECCUST_ET and press ENTER.
- 3** The Display Table ECCUST_ET window opens and displays an empty table with the required parameter names.
- 4** Enter the values exactly as shown below:

| | |
|---------------|----------------------------|
| TOOL_NAME | QuickTest Professional |
| PROG_ID | MERCURY.ECATTAGENT |
| TOOL_DESC | QuickTest Professional |
| TOOL_DATABASE | QUICKTEST DATABASE |
| TOOL_RUN_DB | QUICKTEST RUNTIME DATABASE |
| TOOL_NO_PWD | X |
| TOOL_NO_DB | X |

Note: You can also use the function module **SET_EXTERNAL_TOOL** to create entries in the customizing table. For more information, refer to your eCATT documentation.

Applying Necessary Roles or Profiles to eCATT-QuickTest Users

To perform all of the operations described in the following chapters, you must have permission to:

- run eCATT scripts
- edit eCATT scripts
- work with an external tool (QuickTest) in integrated mode
- connect to eCATT from an external tool (QuickTest) in standalone mode

Each of these tasks requires special roles or profiles. Before you begin working with the QuickTest-eCATT integration, you should confirm with your system administrator that the user name you use is assigned the necessary roles or profiles to perform the above tasks. For example, to work with QuickTest in standalone mode, you must be assigned the role **S_ECET** or the profile **SAP_ECET** in the eCATT system.

For more information, contact your system administrator or refer to your SAP and eCATT documentation.

12

Working with eCATT in Standalone Mode

You can connect to an eCATT database from QuickTest. This is called *standalone mode*. When you work in standalone mode, you have access to all standard QuickTest features.

This chapter describes:

- About Working with eCATT in Standalone Mode
- Connecting to and Disconnecting from eCATT
- Saving Tests to eCATT in Standalone Mode
- Opening Tests from an eCATT Database in Standalone Mode
- Uploading Files to eCATT and Downloading Files from eCATT
- Setting Options for Working with eCATT
- Passing Values Between eCATT Test Scripts and QuickTest Tests
- Running a Test Stored in an eCATT Database in Standalone Mode

About Working with eCATT in Standalone Mode

When you open QuickTest with the **SAP** Add-in loaded, you can connect to eCATT, store tests in the eCATT database, open existing tests from the eCATT database, and upload or download files to or from eCATT. QuickTest provides a special set of eCATT-specific options that enable you to control certain elements of the eCATT-QuickTest integration.

You can also perform many of these operations from the eCATT interface (integrated mode).

For more information on working in integrated mode, see Chapter 13, “Working with eCATT in Integrated Mode.”

For more information on eCATT testing modes, see “Understanding eCATT Testing Modes” on page 170.

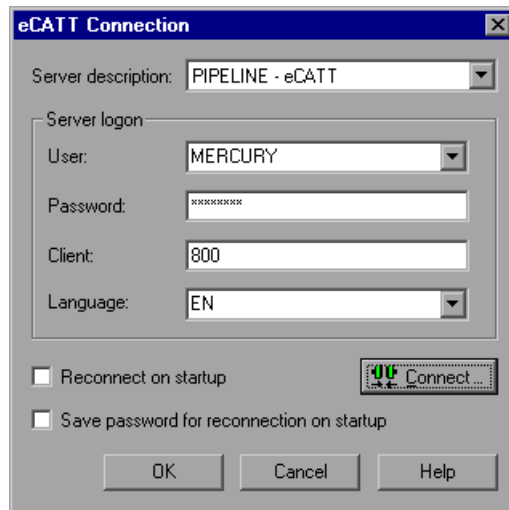
Note: You can pass values from an eCATT test script to a QuickTest test, or vice versa, in the form of QuickTest test parameters. For more information on parameters, refer to the *QuickTest Professional User's Guide*.

Connecting to and Disconnecting from eCATT

From QuickTest, you can connect or disconnect to or from eCATT at any time during the testing process. However, you should not disconnect QuickTest from eCATT while a QuickTest test that is stored in eCATT is open or while QuickTest is using a shared resource stored in eCATT (such as a shared object repository or Data Table file).

Connecting QuickTest to eCATT

You connect to eCATT using the eCATT Connection dialog box (**Tools > eCATT Connection**).




The eCATT Connection dialog box contains the following options:

| Option | Description |
|--|--|
| Server description | <p>The eCATT server to which you want to connect.</p> <p>Note: The Server description box lists the servers available in the SAP Logon Pad or the SAP Logon dialog box.</p> <p>To add a server to the list in the eCATT Connection dialog box, close the dialog box, define an appropriate entry using your SAP Logon dialog box, and then reopen the eCATT Connection dialog box.</p> |
| User | The user name used to log on to the specified server. |
| Password | The password for the specified user name. |
| Client | The client number. |
| Language | The language that you want to use. |
| Reconnect on startup | Instructs QuickTest to automatically reconnect to the eCATT server the next time you open QuickTest. |
| Save password for reconnection on startup | <p>Instructs QuickTest to save your password for reconnection on startup.</p> <p>If you select Reconnect on startup, but do not select this option, you are prompted to enter it each time QuickTest opens.</p> <p>Enabled only when Reconnect on startup is selected.</p> |
| Connect | Connects QuickTest to eCATT. |

The eCATT icon is displayed in the QuickTest status bar to indicate that QuickTest is currently connected to an eCATT server.



Tip: To open the eCATT Connection dialog box, double-click the **eCATT** icon  in the QuickTest status bar.

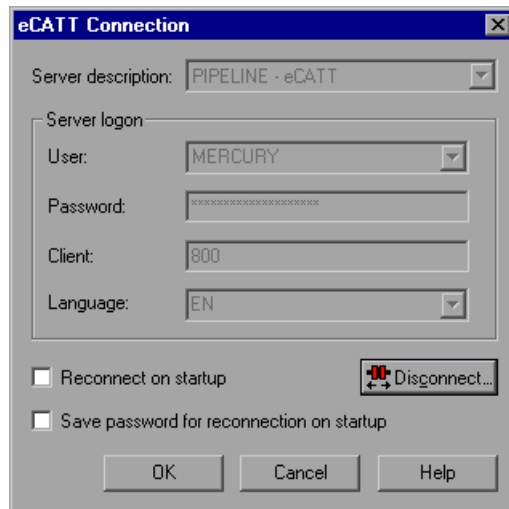
Disconnecting QuickTest from eCATT

When you are finished working with eCATT, you can disconnect from it.

Note: If a test or shared file (such as a shared object repository or Data Table file) that is stored in eCATT is open when you disconnect from eCATT, then QuickTest informs you that the test will be closed.

To disconnect QuickTest from eCATT:

- 1 Choose **Tools > eCATT Connection**. The eCATT Connection dialog box opens.



- 2 Click **Disconnect** to disconnect QuickTest from the selected server.
- 3 Click **OK** to close the eCATT Connection dialog box.

Saving Tests to eCATT in Standalone Mode

When QuickTest is connected to an eCATT server in standalone mode, you can create new tests in QuickTest and save them directly to your eCATT database. You can also open, edit, and save existing tests that are stored in eCATT, and you can save existing tests with a new name in the eCATT database or in the file system.

Note: When saving a test to eCATT in standalone mode, it is automatically saved to the local package (**\$TMP**) in eCATT.

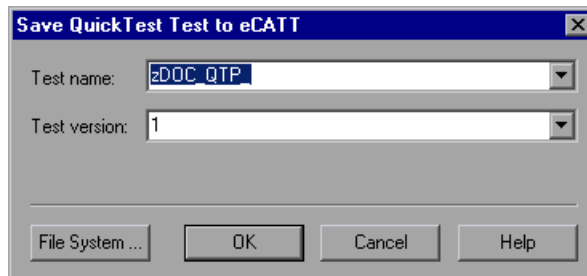
To save a test to an eCATT database in standalone mode:

- 1 From QuickTest, connect to an eCATT server. QuickTest connects to eCATT in standalone mode. For more information, see “Connecting QuickTest to eCATT” on page 179.



- 2 In QuickTest, click **Save** or choose **File > Save** to save the test.

The Save QuickTest Test to eCATT dialog box opens. If you defined a **New test prefix** in the eCATT tab of the QuickTest Options dialog box, the Save QuickTest Test to eCATT dialog box displays the defined prefix. For more information, see “Setting Options for Working with eCATT” on page 191.



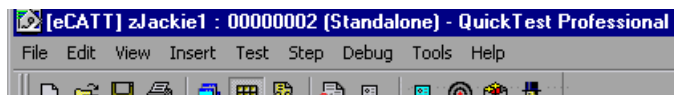
Note: The Save QuickTest Test to eCATT dialog box opens when QuickTest is connected to eCATT in standalone mode. To save a test directly in the file system while you are connected to eCATT, click the **File System** button to open the Save QuickTest Test dialog box.

- 3** In the **Test name** box, enter a valid name for the test. Use a descriptive name that will help you easily identify the test. Ensure that the test name begins with a prefix that matches your eCATT server naming conventions. For example, your eCATT server may require all test names to begin with the letter **z**.
- 4** In the **Test version** box, enter a version number. The version number can be any number that you choose. For example, each time you open and modify a test, you can increment the version number by 1, instead of overwriting the existing version of the test if you want to keep a record of all versions of a test. The test name and version number together form a unique ID for the test.
- 5** Click **OK** to save the test and close the dialog box. Note that the words **Saving**, and then **Uploading**, are displayed in the QuickTest status bar. When QuickTest completes the save process, the status bar displays the word **Ready**.

When the save process is complete, the QuickTest title bar displays the test information in the following format:

[eCATT] TestName : VersionNumber (Mode)

For example:

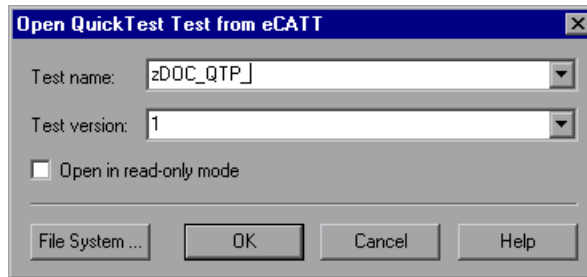


Opening Tests from an eCATT Database in Standalone Mode

When QuickTest is connected to an eCATT server in standalone mode, you can open and edit QuickTest tests that are saved in your eCATT database. You can also edit and save all of a test's resources (external files associated with the test), even if they are stored in eCATT.

To open a test from an eCATT database:

- 1 Connect to an eCATT server. For more information, see “Connecting QuickTest to eCATT” on page 179.
- 2 In QuickTest, click **Open** or choose **File > Open** to open the test. The Open QuickTest Test from eCATT dialog box opens.



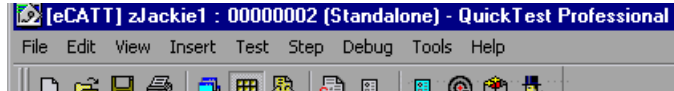
Note: The Open QuickTest Test from eCATT dialog box opens only when QuickTest is connected to an eCATT server. To open a test directly from the file system while you are connected to eCATT, click the **File System** button to open the Open Test dialog box.

- 3 In the **Test name** box, enter a valid test name or select one from the list of recently opened eCATT tests. Do not specify a folder path or other location.
- 4 In the **Test version** box, select a version number from the list.
As QuickTest downloads and opens the test, the operations it performs are displayed in the status bar.
- 5 If you want to open the test in read-only mode, select the **Open in read-only mode** check box.

When the test opens, the QuickTest title bar displays the test information in the following format:

[eCATT] *TestName* : *VersionNumber* (*Mode*)

For example:



Opening Tests from the Recent Tests List

When working in standalone mode, you can open eCATT tests from the recent tests list in the **File** menu. If you select a test located in an eCATT database, but QuickTest is currently not connected to eCATT or to the correct eCATT server for the test, the eCATT Connection dialog box opens.

Enter the **Password** and click **OK**.

The eCATT Connection dialog box also opens if you choose to open a test that was last edited on your computer using a different eCATT user name. You can either log on using the displayed **User** or you can click **Cancel** to prevent the opening of the selected test and remain logged in with your current user name.

Note: You must disconnect from eCATT before opening a Quality Center test from the recent tests list.

Uploading Files to eCATT and Downloading Files from eCATT

You can store a test's external resource files in eCATT. For example, you can store shared object repository files, Data Table files, library files, environment variable files, and recovery files in your eCATT database.

You can also download resource files already stored in eCATT and save them in the file system.

Uploading Files to eCATT

When you save a QuickTest test in eCATT, it is also recommended to store all associated resource files in eCATT so that any user who opens the test from eCATT will have access to all the test's resource files.

Like test names, all test resource files stored in eCATT must begin with a valid prefix. You can set the default prefix for files in the eCATT tab of the QuickTest Options dialog box. For more information, see “Setting Options for Working with eCATT” on page 191.

For files that you create outside QuickTest, such as Data Table files, library files, and environment variable files, you upload the files to eCATT using the **eCATT Upload File** option. You can also use this dialog box to upload existing shared object repository or recovery files from the file system to eCATT.

When you create a new shared object repository or recovery file, you can create the file as you normally would in QuickTest and then save the file directly to eCATT.

Note: When uploading a file to eCATT in standalone mode, it is automatically saved to the local package (**\$TMP**) in eCATT.

To upload external resource files from the file system to eCATT:

- 1 Create and save the file in the file system.
- 2 Connect to eCATT. For more information, see “Connecting QuickTest to eCATT” on page 179.
- 3 Choose **Tools > eCATT Upload File** option. The eCATT Upload File dialog box opens.



- 4 Browse or enter the file path of the **Local file** you want to upload.
- 5 Specify the **eCATT name** and **Version** number you want to assign to the uploaded file.
- 6 Associate the uploaded file with your test in the appropriate QuickTest dialog box. For more information on associating Data Table files, library files, environment variable files, and shared object repository with your test, refer to the *QuickTest Professional User's Guide*.

To create a new shared object repository file and store it in eCATT:

- 1 Open a blank test.
- 2 Choose **Test > Settings > Resources** tab.
- 3 Select the **Shared** radio button.
- 4 Enter the name you want to use for the shared object repository in the format: [eCATT] *SORFileName* : *VersionNumber*. For example:

[eCATT] z_SOR_dwdm : 00000001

- 5 If a warning message opens, click **Yes** to create the new object repository file in eCATT.

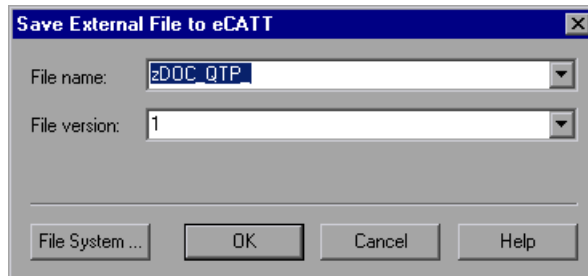
For more information on creating object repository files, refer to the *QuickTest Professional User's Guide*.

To copy or export an object repository to eCATT:

- 1 Open the test whose object repository you want to copy or export.
- 2 Choose **Tools > Object Repository** to open the Object Repository dialog box.
- 3 If the test is in object repository per action mode, click the **Export** button to export the object repository to a shared object repository file in eCATT.

If the test is in shared object repository mode, click the arrow next to the **Save** button and select **Save As** to save a copy of the object repository file with a new name in eCATT.

The Save External File to eCATT dialog box opens.



- 4 Enter the **File name** and **File version** for the shared object repository.
- 5 Click **OK** to save the file.

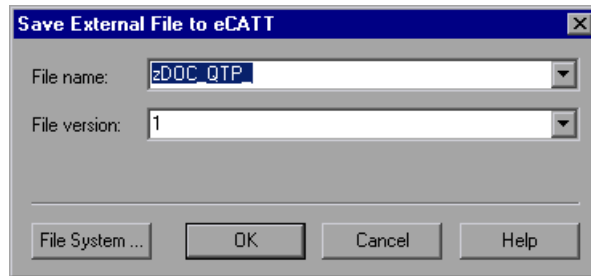
For more information on exporting and saving object repository files, refer to the *QuickTest Professional User's Guide*.

To create a new recovery file in eCATT:

- 1 Choose **Tools > Recovery Scenario Manager**. The Recovery Scenario Manager opens.
- 2 Click the **New Scenario** button. The Recovery Scenario Wizard opens. Follow the instructions in the wizard to create a new scenario. When you are finished, the scenario is displayed in the Recovery Scenario Manager.

If you want to add more scenarios to the new scenario file, repeat step 2.

When you are ready to save the scenario file, click **Save**. The Save External File to eCATT dialog box opens.

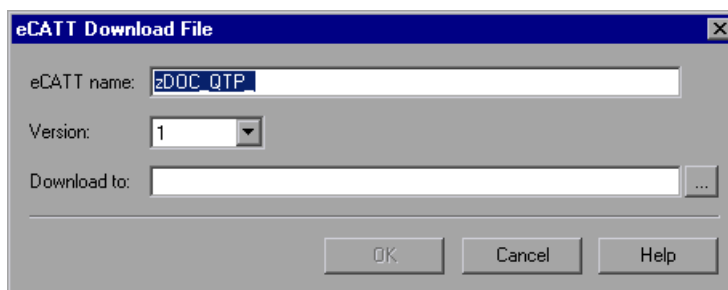


- 3** Enter the **File name** and **File version** for the recovery file.
- 4** Click **OK** to save the file.

For more information on creating and saving recovery files, refer to the *QuickTest Professional User's Guide*.

Downloading Files from eCATT

If you upload a file to eCATT and then associate that file with a test as a resource file, the resource file is automatically downloaded each time you open the test. You can also download files stored in eCATT and save them in the file system using the eCATT Download File dialog box (**Tools > eCATT Download File**).

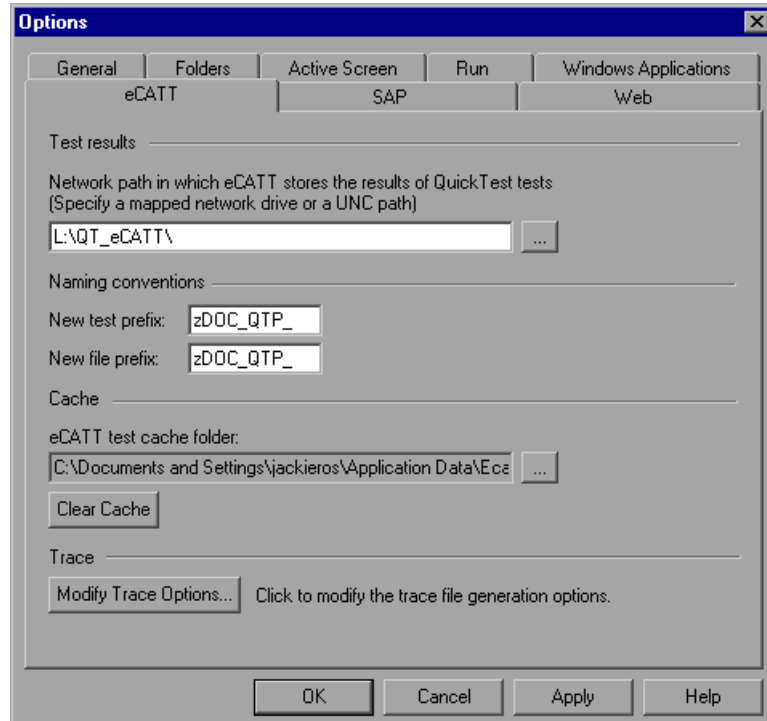


The eCATT Download File dialog box includes the following options:

| Option | Description |
|-------------|---|
| eCATT name | The name of the file stored in eCATT. Note: When the eCATT Download File dialog box opens, it displays the default file prefix in the eCATT name box. You can define or modify this prefix in the eCATT tab of the Options dialog box. For more information, see “Setting Options for Working with eCATT” on page 191. |
| Version | The version number of the file to download. |
| Download to | The complete path and file name of the location to which you want to download the file. You can enter or browse to the folder path. |

Setting Options for Working with eCATT

The eCATT tab of the Options dialog box enables you to configure how QuickTest behaves when you are connected to eCATT.



Note: The eCATT tab is available only when the QuickTest Professional Add-in for SAP Solutions is installed and loaded.

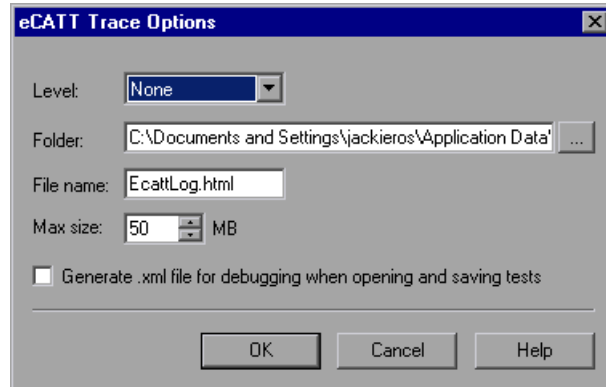
The eCATT tab contains the following options:

| Option | Description |
|--------------------------------|--|
| Test results | The location in which test results are stored when the test is run from eCATT. Note: This folder must be a mapped network drive or a path in Universal Naming Convention (UNC) format. |
| New test prefix | The prefix that is displayed by default in the Save QuickTest Test in eCATT dialog box. |
| New file prefix | The prefix that is displayed by default when specifying a file to store in eCATT. |
| eCATT test cache folder | The location in which a test from eCATT is temporarily stored when it is open for editing or running in QuickTest. Default location = C:\Documents and Settings\<UserName>\Application Data\Ecatt Note: RunCache and EditCache folders are automatically created under the specified folder. |
| Clear Cache | Deletes all files from the eCATT test cache folder. |
| Modify Trace Options | Opens the eCATT Trace Options dialog box, which enables you to configure if and how QuickTest generates the QuickTest-eCATT communication trace log (used for troubleshooting communication errors). |

Configuring the eCATT Trace File

You can instruct QuickTest to generate a QuickTest-eCATT communication trace file each time eCATT runs a QuickTest test in order to troubleshoot communication errors.

You use the eCATT Trace Options dialog box to configure whether and how QuickTest generates the trace file.



The eCATT Trace Options dialog box contains the following options:

| Option | Description |
|---|---|
| Level | <p>The level of detail to include in the trace file that is created when eCATT runs a QuickTest test.</p> <p>None—(default) No trace file is created.</p> <p>Low—The trace file lists any eCATT-QuickTest communication errors.</p> <p>Medium—The trace file includes eCATT-QuickTest communication errors and information on other major operations that result in eCATT-QuickTest communication.</p> <p>High—The trace file includes all available information related to eCATT-QuickTest communications.</p> |
| Folder | <p>The folder path for storing the trace file. Required if a trace file level other than None is specified in the Level option.</p> <p>Default location = C:\Documents and Settings\<UserName>\Application Data\Ecatt\Trace</p> |
| File name | <p>The file name for the trace file.</p> <p>Default = EcattLog.html</p> |
| Max size | <p>The maximum file size you want to allow for the trace file.</p> |
| Generate .xml file for debugging when opening and saving tests | <p>Generates an .xml file each time you open or save a test that is stored in eCATT. The files are saved in Open and Save folders under the trace folder.</p> <p>Note: Selecting this option results in slower response times for editing and saving tests in QuickTest. In general, you should select this option only when instructed to do so in order to debug eCATT connectivity issues.</p> |

Passing Values Between eCATT Test Scripts and QuickTest Tests

You can pass values from an eCATT test script to a QuickTest test, or vice versa, using QuickTest test parameters.

Passing values to QuickTest involves:

- Defining QuickTest test parameters and using them in your QuickTest Test. For more information, refer to the *QuickTest Professional User's Guide*.
- Calling a QuickTest Test and Specifying Arguments from eCATT (see page 208)

Running a Test Stored in an eCATT Database in Standalone Mode

When you run the test from QuickTest (standalone mode), the test results are stored in the location you specify in the file system. You cannot access these results from eCATT.

(For information on running QuickTest tests from eCATT, see “Running a Test Stored in an eCATT Database in Integrated Mode” on page 212.)

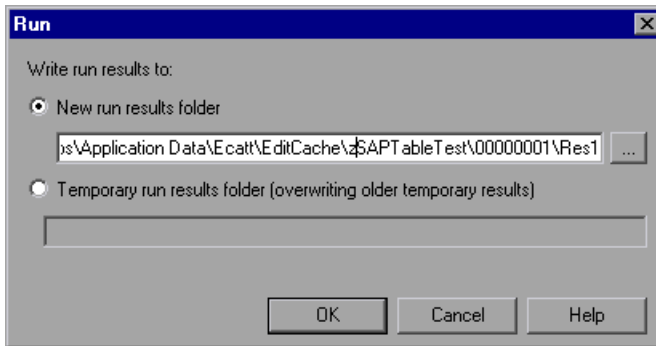
Running Tests from QuickTest

When working with QuickTest in standalone mode, you run a test stored in an eCATT database just like any other QuickTest test.

To run a test stored in eCATT (in Standalone mode):



- 1 In QuickTest, click the **Run** button or choose **Test > Run**. The Run dialog box opens.



- 2 Accept the default results folder or browse to select another one.

Notes:

The default results folder is created under the folder where the cache (local) copy of your test is stored. You set the location of your **eCATT test cache folder** in the eCATT tab of the QuickTest Options dialog box.

When running tests in standalone mode, no eCATT log is created. For more information on the eCATT log for QuickTest run sessions, see “Viewing Results of a QuickTest Test Run in Integrated Mode” on page 215.

To run the test and overwrite the previous run session results, select the **Temporary run results folder (overwriting older temporary results)** option.

Note: QuickTest stores temporary run session results for all tests in <**System Drive**>:\%Temp%\TempResults. The path in the text box of the **Temporary run results folder (overwriting older temporary results)** option is read-only and cannot be changed.

- 3** Click **OK**. The Run dialog box closes and QuickTest begins running the test.

When the run session ends, the Test Results window opens (unless the **View results when run session ends** check box is cleared in the Run tab of the QuickTest Options dialog box). For more information on running QuickTest tests and analyzing test results, refer to the *QuickTest Professional User's Guide*.

13

Working with eCATT in Integrated Mode

You can connect to QuickTest from eCATT. This is called *integrated mode*. When you work in integrated mode, only QuickTest features related to the eCATT test are available in QuickTest. When you run tests in integrated mode, your run session results are accessible in the eCATT log.

This chapter describes:

- About Working with eCATT in Integrated Mode
- Performing Basic Test Management Operations from eCATT
- Transferring Data To and From QuickTest Tests Using Test Parameters
- Running a Test Stored in an eCATT Database in Integrated Mode

About Working with eCATT in Integrated Mode

When you log on to an eCATT server that has been configured to integrate with QuickTest, you can view, edit, and run QuickTest tests that are stored in eCATT. You can also use the standard eCATT commands to copy, rename, and delete QuickTest tests, just as you would with any other file stored in eCATT.

When you open a QuickTest test from eCATT, QuickTest opens in integrated mode. In this mode, you can use all QuickTest features that are associated with the open test. You cannot open another test or save the open test with another name.

You can run a test in integrated mode in any of the following ways. You can use the **Run** option in QuickTest. You can use the **Execute Test Script** (F8) option for a selected QuickTest test in eCATT. You can also execute an eCATT test script (or *blob*—Binary Large Object) that calls a QuickTest test.

Creating eCATT scripts that call QuickTest tests is useful if you want to pass or retrieve values to or from a QuickTest test. For more information on configuring eCATT to work with QuickTest, see Chapter 11, “Configuring eCATT to Work with QuickTest.”

You can also work with tests stored in eCATT from the QuickTest interface (standalone mode). For more information on working in standalone mode, see Chapter 12, “Working with eCATT in Standalone Mode.” For more information on eCATT testing modes, see “Understanding eCATT Testing Modes” on page 170.

Performing Basic Test Management Operations from eCATT

From eCATT, you can perform basic QuickTest test management operations, such as displaying or opening existing QuickTest tests that are stored in eCATT, making copies of tests, renaming tests, deleting tests, and creating new QuickTest tests.

For more information, see:

- Displaying or Editing a QuickTest Test from eCATT, below
- Copying QuickTest Tests from eCATT, on page 203
- Renaming and/or Changing Packages for QuickTest Tests from eCATT, on page 204
- Deleting QuickTest Tests from eCATT, on page 206
- Creating QuickTest Tests from eCATT, on page 206

Displaying or Editing a QuickTest Test from eCATT

You can select to display any existing QuickTest test that is stored in eCATT. When you open the test, QuickTest opens in integrated and read-only mode.

Alternatively, you can select to open the QuickTest test for editing. When the test opens in integrated mode, you can use many QuickTest options. For example, you can edit the test and run the test from the QuickTest interface. However, any external resource files (for example, shared object repository files or external Data Table files) open in read-only mode. Resource files that are saved with the test (for example, per-action object repository or the test’s

local Data Table file) are editable. To edit external resource files, open the test in standalone mode.

When you display or open a QuickTest test in integrated mode, you can work only with the open test. You cannot open another test or save the open test with another name.

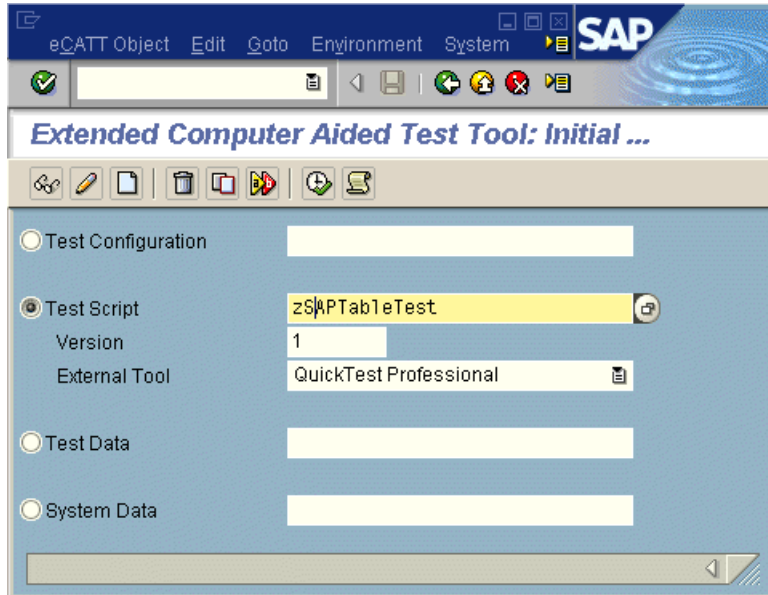
Note: If you choose the QuickTest **File > Save As** menu command when working in integrated mode, QuickTest displays a warning message indicating that you can save a test with a new name in the file system, but doing so disconnects QuickTest from eCATT and switches QuickTest to standalone mode.

For more information on integrated and standalone modes, see “Understanding eCATT Testing Modes” on page 170.

To display or open a QuickTest Test from eCATT:

- 1** Log on to eCATT.
- 2** In the eCATT initial window, select **Test Script**.
- 3** In the **Test Script** box, enter the name of the test.
- 4** In the **Version** box, enter the test version number.

5 Select QuickTest Professional as the External Tool.



- 6** To display the test in read-only mode, click the **Display Object** button. The Display Test Script window opens.



To open the test for editing, click the **Change Object** button. The Change Test Script window opens.



- 7** Click the **Script (Call External Tool)** button (SHIFT + F12). If QuickTest is not already open, it opens. If the test has external resource files and/or if you chose to open the test in read-only (Display) mode, QuickTest reminds you that the resources and/or test will open in read-only mode.
- 8** Click **OK** on the message boxes. QuickTest displays the test.
- 9** If you opened the test for editing, you can use most QuickTest options. For more information, refer to the *QuickTest Professional User's Guide*.
- 10** Close QuickTest to return to eCATT.

Copying QuickTest Tests from eCATT

You can create a copy of a QuickTest test that is stored in the eCATT database and store it with a different name.

To copy a QuickTest Test:

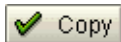
- 1 Repeat steps 1 through 5 of “Displaying or Editing a QuickTest Test from eCATT” on page 201 to specify information about the test you want to copy.
- 2 Click the **Copy Object** button. The Copy dialog box opens.



| | Name | Version |
|------------------|------------------|---------|
| from Test Script | ZDOC_QTP_SAPDWDN | 1 |
| to Test Script | ZDOC_QTP_SAPDWDN | 1 |

✓ Copy ✗

- 3 The **from Test Script** box displays the current QuickTest test **Name** and **Version** number.
- 4 In the **to Test Script** box, enter the **Name** and **Version** of the new copy of the test.



- 5 Click **Copy** to save the copy of the test.
 - If you modified only the version number, proceed to step 8.
 - If you entered a new test name, the Create Object Directory Entry dialog box opens.

Object: R3TR ECAT ZDOC_QTP_SAPDEMO

Attributes

Package: \$TMP

Person Responsible: HOPE

Original System: MI7

Original language:

Local Object Lock Overview ✗

- 6 In the **Package** box, specify the package in which you want to store the copy of the test. Modify other edit boxes as necessary.

Note: If the test has external resource files, they are stored separately in the **\$TMP** (local) package.



- 7 Click **Save** to save your settings and copy the test.
- 8 The eCATT initial window reopens and displays the name and version of the new copy.

Renaming and/or Changing Packages for QuickTest Tests from eCATT

You can rename a QuickTest test that is stored in the eCATT database and store it with another name and/or store it in a different package.

To rename and/or change packages for a QuickTest Test:

- 1 Repeat steps 1 through 5 of “Displaying or Editing a QuickTest Test from eCATT” on page 201 to specify information about the test you want to rename or move.

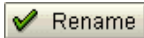


- 2 Click the **Rename Object** button. The Rename dialog box opens.

A screenshot of the 'Rename' dialog box in eCATT. The dialog has a title bar 'Rename' with a close button. It contains two rows of input fields. The first row is labeled 'from Test Script' and contains 'ZDOC_QTP_SAPDWDN' in the 'Name' field and '2' in the 'Version' field. The second row is labeled 'to Test Script' and contains 'ZDOC_QTP_SAPDWDN' in the 'Name' field and '2' in the 'Version' field. At the bottom, there are two buttons: 'Rename' (with a green checkmark icon) and a cancel button (with a red X icon).

| | Name | Version |
|------------------|------------------|---------|
| from Test Script | ZDOC_QTP_SAPDWDN | 2 |
| to Test Script | ZDOC_QTP_SAPDWDN | 2 |

- 3 The **from Test Script** box displays the current QuickTest test **Name** and **Version** number.
- 4 In the **to Test Script** box, enter the new **Name** and/or **Version** for the test.



5 Click **Rename.**

- If you modified only the version number, proceed to step 8.
- If you entered a new test name, the Create Object Directory Entry dialog box opens.

- 6** In the **Package** box, specify the package in which you want to store the test. Modify other edit boxes as necessary.

Note: If the test has external resource files, they are stored separately in the **\$TMP** (local) package.



- 7** Click **Save** to save your settings and rename the test.

- 8** The eCATT initial window reopens and displays the name and version of the renamed test.

Deleting QuickTest Tests from eCATT

You can delete a QuickTest test from the eCATT database.

To delete a QuickTest test from the eCATT database:

- 1 Repeat steps 1 through 5 of “Displaying or Editing a QuickTest Test from eCATT” on page 201 to specify information about the test you want to delete.
- 2 Click the **Delete Object** button. A message box opens.
- 3 Click **Yes** to confirm that you want to delete the test. The test is deleted from the database.



Creating QuickTest Tests from eCATT

You can create a new QuickTest from eCATT.

To create a new QuickTest test:

- 1 Repeat steps 1 through 5 of “Displaying or Editing a QuickTest Test from eCATT” on page 201 to specify information about the test you want to create.
- 2 Click the **Create Object** button. The Create Test Script window opens. Ensure that the **General Data** tab is selected within the **Attributes** tab.
- 3 In the **Title** box, enter a title for your eCATT test script. The title is a short description of your test script.



4 Enter or select a Component.

Test Script: ZDOC_QTP_SAPDEMO Version: 1

Attributes

General Data | Versioning Data | Extras | Restrictions

Header Data

| | | | |
|--------------------|--|------------------------------|---|
| Title | Test that the SAP demo behaves as expected | | |
| External Tool | QuickTest Professional | | |
| Package | | | |
| Person Responsible | HOPE | Type | B |
| Component | CA | Cross-Application Components | |

Script

5 Click the **Script button. QuickTest opens with a blank test.****6 Create the test in QuickTest. For information on creating tests in QuickTest, refer to the other chapters in this guide and to the *QuickTest Professional User's Guide*.****7 In QuickTest, click **Save**. The eCATT Create Object Directory Entry dialog box opens.**

Create Object Directory Entry

Object: R3TR ECAT ZDOC_QTP_SAPDEMO

Attributes

| | |
|--------------------|-------|
| Package | \$TMP |
| Person Responsible | HOPE |
| Original System | MI7 |
| Original language | |

Local Object Lock Overview

8 In the **Package box, specify the package in which you want to store the test. Confirm that the other edit boxes contain correct values.**

Note: If the test has external resource files, they are stored by default in the **\$TMP** (local) package. If you select another package for the test, you must manually move any external resource files to the same package.



- 9 Click **Save** to close the dialog box and save the test. QuickTest is restored in integrated mode and displays the saved test for additional editing.
- 10 When you are finished with the test, close QuickTest to return to eCATT.

Transferring Data To and From QuickTest Tests Using Test Parameters

You can pass values from an eCATT test script to a QuickTest test, or vice versa, using QuickTest test parameters.

Passing values to QuickTest involves:

- Defining QuickTest test parameters and using them in your QuickTest Test. For more information, refer to the *QuickTest Professional User's Guide*.
- Calling a QuickTest Test and Specifying Arguments from eCATT (see below)

Calling a QuickTest Test and Specifying Arguments from eCATT

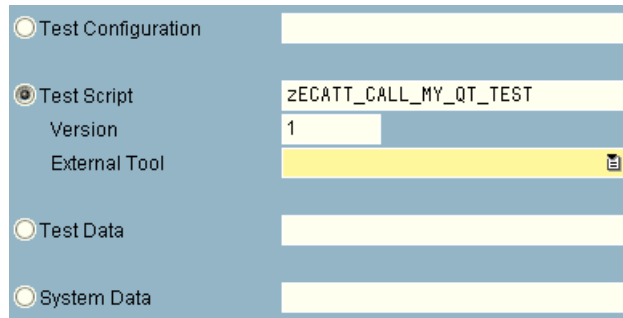
To send values to your QuickTest input arguments, you must run your test via a call from an eCATT test script.

Once you have defined input and output arguments for your QuickTest test, you can insert a call to that test from an eCATT script and specify argument values for the input arguments.

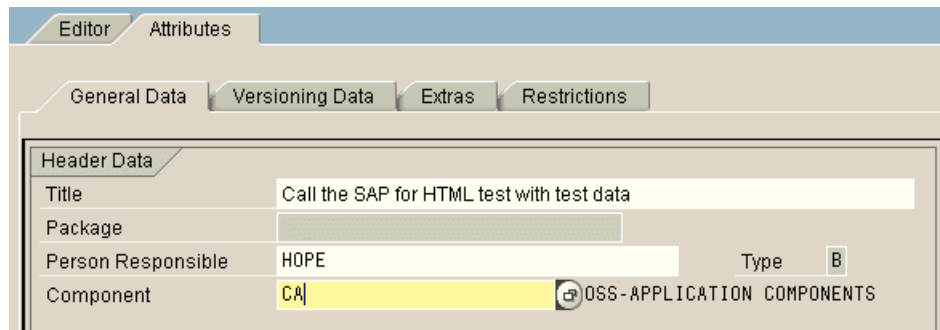
To create an eCATT test script that calls and sends argument values to a QuickTest test:

- 1 Log on to eCATT.
- 2 In the eCATT initial window, select **Test Script**.

- 3 Enter a test script name and version number. Do not specify an **External Tool**.




- 4 Click the **Create Object** button. The Create Test Script window opens. Ensure that the **General Data** tab is selected within the **Attributes** tab.
- 5 In the **Title** box, enter a title for your eCATT test script. The title is a short description of your test script.
- 6 Enter or select a **Component**.

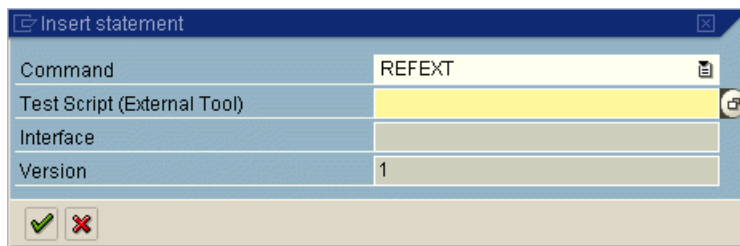


- 7 Click the **Editor** tab. The eCATT command editor is displayed.
- 8 If the **Command Interface** is not displayed, click the **Parameter<->Command Interface** toggle button to display the **Command Interface**.



- 9 Click the **Pattern** button in the Create Test Script toolbar. The Insert Statement window opens.

- 10** In the **Command** box, select **REFEXT** to call an external test. The window changes to display the options corresponding to an external call.



| | |
|-----------------------------|--------|
| Command | REFEXT |
| Test Script (External Tool) | |
| Interface | |
| Version | 1 |

- 11** In the **Test Script (External Tool)** box, enter the name of the QuickTest test you want to call.

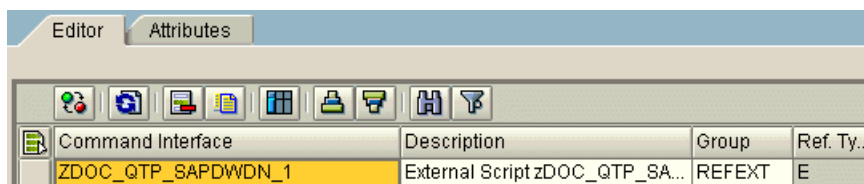


- 12** Click the **Continue (Enter)** button. eCATT enters the default interface value in the **Interface** box. Modify this value if necessary.

- 13** If you want to call a test version other than version 1, enter the version number of the test you want to call in the **Version** box.



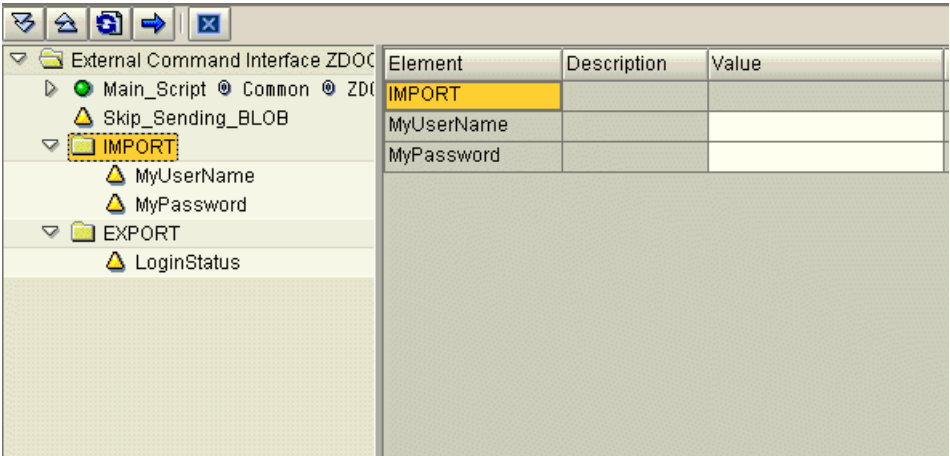
- 14** Click the **Continue (Enter)** button again to add the test to the **Command Interface**.



| Command Interface | Description | Group | Ref. Ty... |
|--------------------|--------------------------------|--------|------------|
| ZDOC_QTP_SAPDWDN_1 | External Script zDOC_QTP_SA... | REFEXT | E |

- 15** Double-click the row header for the test in the **Command Interface**. The External Command Interface table is displayed.

- 16 Double-click the **Import** folder. If input test arguments have been defined for the test, they are displayed in the far-right grid.



Tip: You can also double-click the **Export** folder to view the output arguments defined for the test.

- 17 Set the value of each input argument in the **Value** column of the far-right grid.

Tip: You can enter the name of an eCATT parameter from the eCATT script as the value of a QuickTest input parameter.

Running a Test Stored in an eCATT Database in Integrated Mode

When you run the test from eCATT (integrated mode), the test results are stored in the network drive specified for eCATT test results (as defined in the eCATT tab of the QuickTest Options dialog box). Although the test results are not stored in eCATT, you can access the results from the eCATT log.

For information on running tests in standalone mode, see “Running a Test Stored in an eCATT Database in Standalone Mode” on page 196.

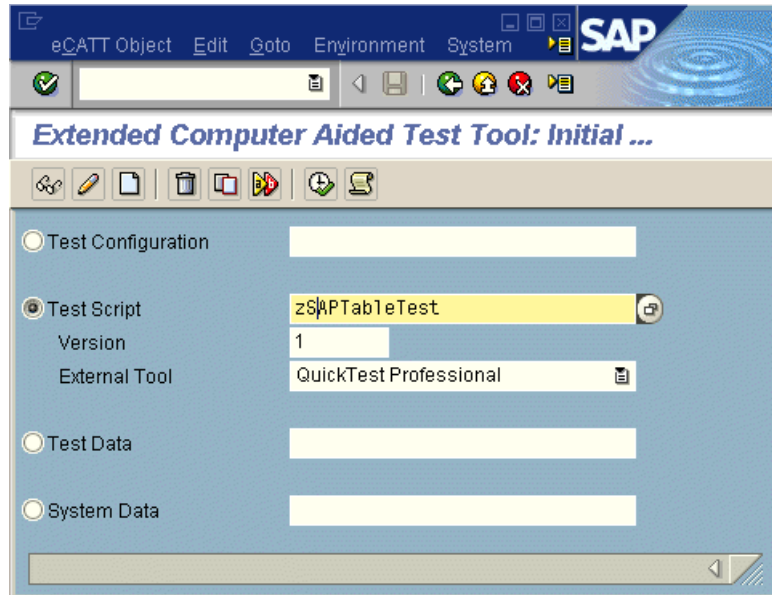
Running QuickTest Tests from eCATT Test Scripts (Integrated Mode)

To run a QuickTest test from eCATT, you can:

- Display the test in QuickTest and use the standard QuickTest **Run** option in QuickTest. For more information on displaying the test, see “Displaying or Editing a QuickTest Test from eCATT” on page 200. For information on running a test using the QuickTest **Run** option, refer to the *QuickTest Professional User’s Guide*.
- Create and run an eCATT test script that calls a QuickTest test. For information on creating eCATT test scripts that call QuickTest tests, see “Transferring Data To and From QuickTest Tests Using Test Parameters” on page 208. For information on running eCATT test scripts, refer to your eCATT documentation.
- Use the **Execute Test Script** (F8) option for a selected QuickTest test in eCATT. For more information, see the procedure below.

To run a QuickTest test from eCATT using the Execute Test Script option:

- 1 Log on to eCATT.
- 2 In the eCATT initial window, enter the test name in the **Test Script** box and the version number in the **Version** box. Select **QuickTest Professional** as the **External Tool**.





- 3 Click the **Execute Test Script (F8)** button. The Start Options window opens.

Start Options (Test Script ZSAPTABLETEST Version 1 -)

Error behavior: S No Termination, Continue with Next Script Command

System Data:

Test System:

☒ Log Display

☐ Archiving

☐ Close RFC connection

TCD

Start Mode for Command TCD: N Process in Background, Synchronous Local

☒ Start Mode Overwrites Mode If TCD Is Run

SAPGUI

Proc Mode SAPGUI: N Optimized Performance

Error Mode for SAPGUI: N Standard (Terminate on Any Error)

Stop When: N Do Not Stop

Close GUIs: N Close Generated Sessions After Script

External Tool

Mode for Ext. Tool: N Normal

User Name:

Password: *****

Execute (Green Arrow) Cancel (Red X)

- 4 Select the options you want to use for the run session. For example, select the **Log Display** check box if you want the test log to display when the run session ends and select **A With Surface of External Tool** in the **Mode for Ext. Tool** box, if you want QuickTest to be displayed during the run session. For information on other options in this window, refer to your eCATT documentation.



- 5 Click the **Execute** button to start the run session. If you selected the **Log Display** check box in the Start Options window, then the eCATT log is displayed when the run session ends.

```

0000001291 Test Script zSAPTableTest Version 1 - SECATT [Without Interruption]
MI7 800 HOPE E 620 PIPELINE Windows NT ORACLE 02.11.2003 11:07:15
  zSAPTableTest Version 1 External Script zSAPTableTest (00000001)
    The test case has been active for [02:34 min] minutes
  
```

Viewing Results of a QuickTest Test Run in Integrated Mode

You can view the results of a QuickTest test that was run from eCATT in one of three ways:

- **In the QuickTest Test Results window**—Because all tests run from eCATT are stored in a network drive, you can open QuickTest (or the QuickTest Test Results Viewer application), and browse to the desired test results. You can view the folder in which the results are saved in the **UNCPathToLocalLog** line of the eCATT log.

```

0000001291 Test Script zSAPTableTest Version 1 - SECATT [Without Interruption]
MI7 800 HOPE E 620 PIPELINE Windows NT ORACLE 02.11.2003 11:07:15
  zSAPTableTest Version 1 External Script zSAPTableTest (00000001)
    EXT_SCRIPT 00000000 EXECUTED WITH QuickTest Professional XML-DATA-02
      External Command Interface (00000001)
        UNCPathToLocalLog = 'L:\QuickTest\Ver_6_5\Sample_Tests_4_Docs\zSAPTableTest\00000001\0000001291
  
```

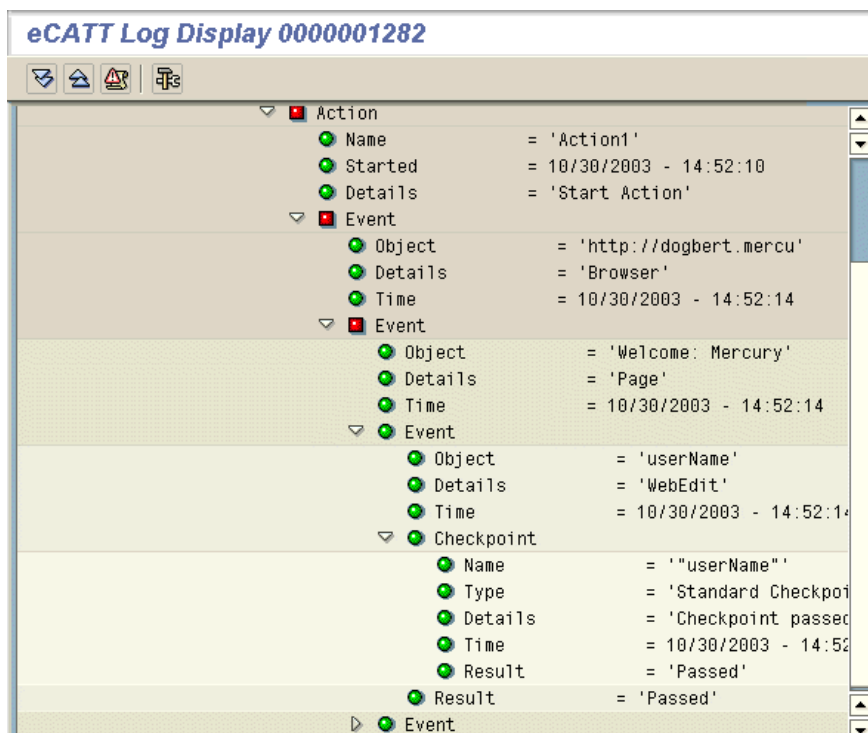
For more information on opening and analyzing run session results in QuickTest, refer to the *QuickTest Professional User's Guide*.

- **Via the generated XML Report**—Each time you run a QuickTest test from eCATT, an .xml file is generated. This file contains all details of the run session. To view the file, click the line containing the text: XML-DATA in the eCATT log.

```

0000001291 Test Script zSAPTableTest Version 1 - SECATT [Without Interruption]
MI7 800 HOPE E 620 PIPELINE Windows NT ORACLE 02.11.2003 11:07:15
  zSAPTableTest Version 1 External Script zSAPTableTest (00000001)
    EXT_SCRIPT 00000000 EXECUTED WITH QuickTest Professional XML-DATA-02
      External Command Interface (00000001)
  
```

- **In the eCATT Log Display**—You can expand the log to view the results of the test or the results of a specific event, such as a checkpoint.



Index

A

- action parameters 92, 146
- Active Screen tab, Options dialog box 73
- Active Screen, capture level 73
- Add-in Manager dialog box 18, 19
- add-in support
 - loading with 19
 - loading without 21
- adding checkpoints
 - bitmaps 147
 - for text in SAP Web applications 147
- add-ins
 - licensing 17
 - repairing 31
 - uninstalling 28
- administrator privileges for installing 4
- Advanced Web Options dialog box 140
- application details environment variables 71
- application details environment variables, Web 140
- associated library files 24
- attaching library files 22

B

- bitmap checkpoint
 - for SAP Web objects 147
 - for SAP Windows objects 93
- buttons 157

C

- calling QuickTest Tests from eCATT 208
- checking
 - appearance of SAP Web objects 147
 - SAP Windows tables and grids 94
 - text in SAP Web applications 147

checkpoints

- bitmap, for SAP Web objects 147
- for SAP Web tables and grid controls 148
- for SAP Windows table and grid controls 94
- for text in SAP Web applications 147
- Complete (Active Screen capture level) 73
- component parameters 92, 146
- connecting to eCATT 179
- connection speed, SAP server 55
- Control display mode, for Help screens 52
- Customer Registration screen 11
- customer support site 11

D

- Define Row Range dialog box 96, 100, 105, 108
- Dialog display mode, for Help screens 52
- disconnecting from eCATT 181
- disk space requirements 4
- downloading files from eCATT 190

E

- eCATT 169
 - applying privileges for QuickTest integration 175
 - calling QuickTest tests 208
 - configuring to work with QuickTest 173
 - connecting to 179
 - copying tests, integrated mode 203
 - creating tests, integrated mode 206
 - deleting tests, integrated mode 206
 - disconnecting from 181

eCATT (*cont'd*)

- displaying tests (read-only),
 - integrated mode 200
- downloading files from 190
- external tool parameters 174
- integrated mode 199
- integration overview 169
- managing tests from 200
- opening tests
 - from recent test list 185
 - integrated mode 200
 - standalone mode 184
- packages, moving tests between 204
- renaming tests, integrated mode 204
- running tests stored in
 - integrated mode 212
 - standalone mode 196
- saving tests
 - integrated mode 200
 - standalone mode 182
- sending test arguments 208
- setting testing options 191
- standalone mode 177
- testing modes 170
- uploading files to 186
 - recovery files 188
 - shared object repository files 187

eCATT Connection dialog box 179

eCATT Download File dialog box 190

eCATT Trace Options dialog box 193

eCATT Upload File dialog box 186

ECCUST_ET table 174

Enable Scripting option, Scripting tab 49, 51

enabling tests, setting F1Help 54

environment variables, application details 71

Expert View 128, 163

F

F1 Help 54

F4 Help 52

G

grid objects 161

guicomponenttype test object property 60

H

hard disk space requirements 4

Help screens 52, 54

I

icons 157

inner objects 73

installation requirements 4

installing the add-in license 17

InstallShield Wizard Complete screen 14

integrated mode 170

IntelliSense 128, 163

internal methods and properties 129

K

kernel patches

- checking 41
- levels 38

L

library files

- associating with Quality Center
 - project 24
- attaching to Quality Center project 22
- working with 21

License Agreement screen 7

licensing add-ins 17

loading

- with add-in support 19
- without add-in support 21

Low speed connection option, SAP server 55

low-level recording 88

M

methods
 accessing native 129, 164
 test object 128, 163
 modal dialog box mode, for Help screens 54
 Modify Row Range dialog box 101, 111

N

name test object property 60
 native methods and properties 129
 None (Active Screen capture level) 73

O

Object property 129, 164
 Object Selection - Checkpoint Properties
 dialog box 108, 151
 Object Spy 129, 164
 Open QuickTest Test from eCATT dialog box 184
 opening tests
 in eCATT integrated mode 200
 in eCATT standalone mode 184
 Options dialog box
 eCATT tab 191
 eCATT Trace Options, eCATT tab 193
 SAP tab 76
 output values
 for SAP Windows objects 103
 inserting while editing your test 108
 inserting while recording 105
 modifying 111
 object property values 103
 SAP Windows property and table cell values 103
 SAP Windows table cell values 104

P

Page and Frame Options dialog box 140
 parameters 92, 146
 Partial (Active Screen capture level) 73
 patch version, SAP Gui for Windows client 43
 prerequisites, installation 4

properties
 accessing native for SAP Web 164
 accessing native for SAP Windows 129

Q

Quality Center project
 associating library files 24
 attaching library files 22
 working with library files 21
 QuickTest
 loading with add-in support 19
 loading without add-in support 21

R

recent tests list, opening eCATT tests from 185
 Record and Run Settings dialog box
 optimizing other tab settings 72
 SAP tab 66, 89, 137
 Web tab 72
 Windows Applications tab 72
 recording
 low-level mode 88
 relative to coordinates 88
 recording and running tests
 on SAP Web objects 137
 on SAP Windows objects 58
 recording tests
 on SAP Gui for Windows applications 64
 on SAP Web objects 137, 140
 recording, standard Windows capabilities 88
 renaming tests in eCATT 204
 repairing add-ins 31
 requirements, installation 4
 Run dialog box 196
 running tests
 on SAP Web objects 144
 on SAP Windows objects 89
 running tests stored in eCATT
 integrated mode 212
 standalone mode 196
 rz11 transaction 44

S

- SAP application server
 - configuring 60
 - enabling the scripting interface 44
 - package and patch versions, checking 38
- SAP Gui for Windows
 - configuring 60
 - enabling scripting 49
 - installing 37
 - patch versions, checking 43
- SAP Gui Scripting API 60
- SAP Gui Scripting option 37
- SAP Logon dialog box 43, 64
- SAP Logon Pad 64
- SAP OSS note #480149 39, 41, 44
- SAP support packages 38
- SAP tab
 - Options dialog box 76
 - Record and Run Settings dialog box 72
- SAP Web
 - test object model 135
 - test objects 155
- SAP Windows
 - supported checkpoints 93
 - supported output values 93
 - test object model 60
- SAP Windows objects, overview 114
- SAP_CLIENT_ENV variable 71
- SAP_ECET role 175
- SAP_LANGUAGE_ENV variable 71
- SAP_PASSWORD_ENV variable 71
- SAP_SERVER_ENV variable 71
- SAP_USER_ENV variable 71
- SAPButton test object 157
- SAPCheckBox test object 157
- SAPDropDownMenu test object 157
- SAPEdit test object 157
- SAPFrame test object 157
- sapgui/user_scripting profile 44
- SAPGuiButton object 116
- SAPGuiCalendar object 120
- SAPGuiCheckbox object 116
- SAPGuiComboBox object 116
- SAPGuiEdit object 116
- SAPGuiElement object 120
- SAPGuiGrid object 119
- SAPGuiLabel object 121
- SAPGuiMenubar object 116
- SAPGuiOKCode object 116
- SAPGuiRadioButton object 116
- SAPGuiSession object 116
- SAPGuiStatusBar object 121
- SAPGuiTable object 122
- SAPGuiTabStrip object 122
- SAPGuiTextArea object 116
- SAPGuiToolbar object 123
- SAPGuiTree object 125
- SAPGuiUtil object 127
- SAPGuiWindow object 116
- SAPiView test object 158
- SAPList test object 158
- SAPLogon.exe 64
- SAPlpd.exe 64
- SAPMenu test object 158
- SAPNavigationBar test object 159
- SAPOKCode test object 159
- SAPPortal test object 160
- SAPRadioGroup test object 160
- SAPStatusBar test object 160
- SAPTable test object 161
- SAPTabStrip test object 162
- SAPTreeView test object 163
- Save External File to eCATT dialog box 188
- Save QuickTest Test to eCATT dialog box 182
- Save step screen capture to test results option 73
- saving tests
 - in eCATT integrated mode 200
 - in eCATT standalone mode 182
- Select Updates screen 12
- SET_EXTERNAL_TOOL function module 174
- settings for testing Web-based SAP applications 140
- setup
 - Customer Registration screen 11
 - InstallShield Wizard Complete screen 14
 - License Agreement screen 7
 - running 4
- SPAM transaction 39

- spooling data
 - from an SAP Gui for Windows table 112
 - from an SAP Web table 152
- standalone mode, eCATT 170
- standard Windows recording 88
- statements, inserting 113
- Step Generator 114, 128, 154, 163
- support packages, checking 39
- system requirements 4

T

- table checkpoints 94
 - inserting while editing your test 100
 - inserting while recording 96
 - modifying 101
- table output values, modifying 111
- tables, spooling data 112, 152
- test arguments
 - specifying from eCATT 208
- test object model
 - for SAP Web objects 135
 - for SAP Windows objects 60
- test parameters 92, 146
- test results 89
- testing
 - SAP Web objects 134
 - SAP Windows objects 58
- tests
 - copying in eCATT 203
 - creating eCATT tests 206
 - creating QuickTest tests in eCATT 206
 - deleting from eCATT 206
 - opening in integrated mode (in eCATT) 200
 - opening in standalone mode (in eCATT) 184
 - renaming in eCATT 204
 - saving in eCATT (integrated mode) 200
 - saving in eCATT (standalone mode) 182
- toggle buttons 157
- toolbar buttons 157

- trace file, configuring for eCATT run sessions 193
- tree objects 163

U

- uninstalling, add-ins 28
- Update Run option 73
- uploading files to eCATT 186
 - recovery files 188
 - shared object repository files 187

V

- values, outputting property and table cell for
 - SAP Windows objects 103
- VBScript 113

W

- warning messages, eliminating 51
- Web environment variables, application
 - details 140
- Web tab, Record and Run Settings dialog box 72
- Web test settings 140
- Windows Applications tab, Record and Run
 - Settings dialog box 72
- working with library files 21

