Oracle® Scripting

Implementation Guide

Release 11*i*

Part No. B10180-03

April 2003

ORACLE®

Oracle Scripting Implementation Guide, Release 11*i*

Part No. B10180-03

Contents

3   Before You Begin

4 Implementation Tasks

5   Implementation Administration Tasks

6   Survey Resources Administration Tasks

7   Scripting Administration Tasks

8 Survey Campaign Administration Tasks

9 Campaign and List Administration Tasks

10   Integrating Oracle Scripting

11   Implementation Verification

A   Upgrading from Releases Prior to 11.5.6

B   Oracle Scripting Profile Options

C   Oracle Discoverer Workbooks

Abbreviations and Acronyms

Glossary

Send Us Your Comments

Oracle Scripting Implementation Guide, Release 11*i*

Part No. B10180-03

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Postal service:
  Oracle Corporation
  Oracle Scripting Documentation
  1900 Oracle Way
  Reston, Virginia 20190
  U.S.A.

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Audience for This Guide

Welcome to Release 11*i* of the Oracle Scripting Implementation Guide.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.

- Oracle Scripting

  If you have never used Oracle Scripting, Oracle suggests you attend one or more of the Oracle Scripting training classes available through Oracle University.

- The Oracle Applications graphical user interface.

  To learn more about the Oracle Applications graphical user interface, read the *Oracle Applications User's Guide*.

See Other Information Sources for more information about Oracle Applications product information.

How To Use This Guide

This guide contains the information you need to understand and use Oracle Scripting. Sections include:

The **Introduction** includes an overview of Oracle Interaction Center software products and indicates what is new, what has been modified, and what is obsolete in Oracle Scripting since release 11.5.7.

Detailed Product Description describes features of Oracle Scripting, discusses architecture, and lists responsibilities that may be required for using Oracle

Scripting. Concepts sections are also included for each component of Oracle Scripting.

Before You Begin mentions related product documentation, identifies various dependencies and minimum software and hardware requirements, and provides general scalability and performance guidelines.

Implementation Tasks lists an implementation task sequence to provide a map to which tasks are required for your implementation. This section then describes tasks for creating various Oracle Scripting users, and contains procedures for implementing the Scripting Engine agent and web interfaces, as well as describing details for setting up two administrative interfaces and when they are required. Finally, an informative section contains additional information about creating and administering Oracle Scripting users.

Implementation Administration Tasks addresses administration tasks that may be required when implementing Oracle Scripting.

Survey Resources Administration Tasks includes tasks required to administer survey resources, which are required for executing any script in the Scripting Engine Web interface using a Web browser. These steps are not relevant for implementations using the Scripting Engine agent interface only.

Scripting Administration Tasks documents administration performed from the Scripting Administration console. This section describes new functionality which allows users to launch the Script Author Java applet, administer Oracle Scripting files such as deployed scripts and custom Java, and generate and view a panel footprint report.

Survey Campaign Administration Tasks documents administration performed from the Survey Administration console. This section details administration of survey campaigns, how to view and interpret responses to active campaigns, how to generate and analyze summarized reports based on respondent data, and details concurrent program administration required for survey operations. Survey campaign creation and administration tasks are prerequisite to executing *any* script in a Web browser using the Scripting Engine Web interface.

Campaign and List Administration Tasks addresses administration of Oracle Marketing where this administration affects Oracle Scripting users. For example, if using Oracle Scripting from Oracle TeleSales or Oracle Collections, you must first administer campaigns in Oracle Marketing. Additionally, in order to execute survey campaigns using the survey component of Oracle Scripting, Oracle Marketing lists are required. These tasks are detailed here.

Integrating Oracle Scripting discusses integration of Oracle Scripting with other applications by Oracle Scripting component, by technical stack, and by business application integration. Specific steps required to integrate Scripting with Oracle TeleSales, Oracle Collections, and Oracle TeleService are addressed.

Implementation Verification includes a description of tasks to verify each component of Oracle Scripting, including a tutorial to create and deploy a simple script to assist in testing the Scripting Engine and Survey components.

Appendix A, Upgrading from Releases Prior to 11.5.6, provides specific information to assist you in upgrading from any pre-11.5.6 release of Oracle Scripting to any subsequent release.

Appendix B, Oracle Scripting Profile Options, provides specific information on profile options for Oracle Scripting and other Oracle applications settings required to implement various Oracle Scripting components.

Appendix C, Oracle Discoverer Workbooks, provides information on the survey reports that can be generated for analysis of survey information using Oracle Discoverer workbooks.

There are also **Glossary** and **Abbreviations and Acronyms** appendices to assist you in decoding many terms, acronyms and abbreviations.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Other Information Sources

You can choose from many sources of information, including online documentation, training, and support services, to increase your knowledge and understanding of Oracle Scripting.

If this guide refers you to other Oracle Applications documentation, use only the Release 11*i* versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF). Some online help patches are available on Oracle*MetaLink* or Oracle iSupport. Oracle Scripting online help is available for the Scripting Administration and Survey Administration components.

Related Documentation

Oracle Scripting shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other product documentation when you set up and use Oracle Scripting.

You can read the documents online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at http://oraclestore.oracle.com.

Documents Related to All Products

Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI) available with this release of Oracle Scripting (and any other Oracle Applications products). This guide also includes information on

setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Documents Related to This Product

Oracle Scripting Implementation Guide 11*I*

This guide describes how to implement Oracle Scripting components and test the implementation appropriately. This guide also details task-based steps for administering Oracle Scripting. You can access Oracle Scripting administration procedures online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Oracle Scripting User Guide 11*I*

This guide explains how to understand the various components of Oracle Scripting, how to plan for Oracle Scripting implementations, and how to use the Script Author to create scripts to be executed in the Scripting Engine (agent or Web interfaces).

Installation and System Administration

Oracle Applications Concepts

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

Installing Oracle Applications

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications, the Oracle8 technology stack, and the Oracle8*i* Server technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you need to perform to finish your installation. You should use this guide in conjunction with individual product user's guides and implementation guides.

Oracle Applications Supplemental CRM Installation Steps

This guide contains specific steps needed to complete installation of a few of the CRM products. The steps should be done immediately following that tasks given in the Installing Oracle Applications guide.

Upgrading Oracle Applications

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7 (NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

Maintaining Oracle Applications Document Set

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you need to run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

Oracle Applications System Administrator's Guide

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

Oracle Alert User's Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Applications Developer's Guide

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. It also provides information to help you build your custom Oracle Forms Developer 6*i* forms so that they integrate with Oracle Applications.

Oracle Applications User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications

products and how to apply this UI to the design of an application built by using Oracle Forms.

Other Implementation Documentation

Multiple Reporting Currencies in Oracle Applications

If you use the Multiple Reporting Currencies feature to record transactions in more than one currency, use this manual before implementing Oracle Scripting. This manual details additional steps and setup considerations for implementing Oracle Scripting with this feature.

Multiple Organizations in Oracle Applications

This guide describes how to set up and use Oracle Scripting with Oracle Applications' Multiple Organization support feature, so you can define and support different organization structures when running a single installation of Oracle Scripting.

Oracle Workflow Guide

This guide explains how to define new workflow business processes as well as customize existing Oracle Applications-embedded workflow processes.You also use this guide to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Scripting implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This manual also provides information on creating custom reports on flexfields data.

Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle*MetaLink*

Oracle Manufacturing APIs and Open Interfaces Manual

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes APIs and open interfaces found in Oracle Manufacturing.

Oracle Order Management Suite APIs and Open Interfaces Manual

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes APIs and open interfaces found in Oracle Order Management Suite.

Oracle Applications Message Reference Manual

This manual describes Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11i.

Oracle CRM Common Application Components Implementation Guide

Many CRM products use components from CRM Common Application Components (formerly referred to as the CRM Application Foundation). Use this guide to correctly implement CRM Common Application Components.

Training and Support

Training

Oracle offers training courses to help you and your staff master Oracle Scripting and reach full productivity quickly. You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many Education Centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization structure, terminology, and data as examples in a customized training session delivered at your own facility.

Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Scripting working for you. This team includes your Technical Representative, Account Manager, and Oracle's large staff of consultants and support specialists with expertise in your business area, managing an Oracle8*i* server, and your hardware and software environment.

Oracle*MetaLink*

Oracle*MetaLink* is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With Oracle*MetaLink*, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use Oracle*MetaLink*, register at (http://metalink.oracle.com).

Alerts: You should check Oracle*MetaLink* alerts before you begin to install or upgrade any of your Oracle Applications. Navigate to the Alerts page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade/Alerts.

Self-Service Toolkit: You may also find information by navigating to the Self-Service Toolkit page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade.

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using Oracle Applications can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

About Oracle

Oracle Corporation develops and markets an integrated line of software products for database management, applications development, decision support, and office automation, as well as Oracle Applications, an integrated suite of more than 160 software modules for financial management, supply chain management, manufacturing, project systems, human resources and customer relationship management.

Oracle products are available for mainframes, minicomputers, personal computers, network computers and personal digital assistants, allowing organizations to integrate different computers, different operating systems, different networks, and even different database management systems, into a single, unified computing and information resource.

Oracle is the world's leading supplier of software for information management, and the world's second largest software company. Oracle offers its database, tools, and applications products, along with related consulting, education, and support services, in over 145 countries around the world.

1

Introduction

1.1 Oracle Interaction Center Overview

Every customer interaction (a telephone call, an e-mail message, or a Web chat conversation) presents an opportunity to win new business or improve customer satisfaction. The Oracle Interaction Center supports the management and processing of customer relationship activity across all channels of customer contact.

The Oracle Interaction Center integrates with several customer relationship business applications in the Oracle eBusiness Suite. The Oracle Interaction Center consists of several modules. The modules relating to inbound telephony and outbound telephony are bundled separately.

The Oracle Interaction Center allows access to centralized customer information and business application functionality. Oracle Interaction Center integrates with front office applications (known as Customer Relationship Management or CRM), and back office applications (Enterprise Relationship Planning or ERP), thereby enabling a workflow powered, end-to-end strategic e-business solution.

 The Oracle Interaction Center products include:

- Section 1.1.1, "Oracle Advanced Inbound"
- Section 1.1.2, "Oracle Advanced Outbound"
- Section 1.1.3, "Oracle Email Center"
- Section 1.1.4, "Oracle Scripting"
- Section 1.1.5, "Oracle Interaction Center Intelligence"
- Section 1.1.6, "Oracle One-to-One Fulfillment"
- Section 1.1.7, "Oracle Interaction History"

1.1.1 Oracle Advanced Inbound

Oracle Advanced Inbound is designed to consistently and effectively handle customer interactions by intelligently routing, queuing and distributing media items. Oracle Advanced Inbound offers CTI support for market-leading traditional ACD/PBX and IP Telephony platforms, and provides enhanced screen pops on customer data into the Oracle e-Business suite application. Oracle Advanced Inbound is fully integrated with Oracle TeleSales, Oracle TeleService and Oracle Collections, thereby minimizing integration time and deployment costs. Oracle Advanced Inbound also provides the Oracle Telephony Adapter SDK, which can be used to integrate other PBX/ACD and CTI middleware combinations that are not supported by an Oracle telephony adapter.

Oracle Advanced Inbound is required to telephony enable business applications in the Oracle eBusiness Suite. "Telephony-enabled" means that the application can communicate with a telephone system for inbound calls, outbound calls, or both by way of the CTI middleware that handles the messaging between the customer's PBX/ACD and the business application.

The Oracle Advanced Inbound bundle consists of the following products: Oracle Interaction Center Server Manager, Oracle Universal Work Queue, Oracle Telephony Manager, Oracle Interaction Center Intelligence and Oracle Interaction Blending.

See Also

Section 1.1.2, "Oracle Advanced Outbound"

Section 1.1.3, "Oracle Email Center"

Section 1.1.4, "Oracle Scripting"

Section 1.1.5, "Oracle Interaction Center Intelligence"

Section 1.1.6, "Oracle One-to-One Fulfillment"

Section 1.1.7, "Oracle Interaction History"

1.1.2 Oracle Advanced Outbound

Oracle Advanced Outbound is another key part of the Oracle eBusiness Suite of applications. It is the module of Oracle Interaction Center that addresses outbound telephony. Oracle Advanced Outbound consists of two main components:

- A tactical list manager, which determines who to call and when to call them

- An outbound dialing engine, which dials numbers and transfers live contacts to call center agents

Oracle Advanced Outbound integrates with and relies on Oracle Marketing to create campaigns and lists to execute. Oracle Advanced Outbound serves as the execution arm for these marketing lists to maximize both outbound list penetration and agent productivity. Oracle Advanced Outbound also integrates with desktop applications like Oracle TeleSales and Oracle Collections to handle the actual customer interactions. Oracle Advanced Outbound can be used any time agents need to contact parties via the telephone.

Oracle Advanced Outbound also integrates with Oracle Interaction History to provide feedback that marketers can use to analyze and measure the success of the marketing campaign, thereby providing a closed-loop marketing process.

Oracle Advanced Outbound does not include any other telephony management modules, and thus requires the use of Oracle Advanced Inbound.

See Also

Section 1.1.1, "Oracle Advanced Inbound"

Section 1.1.3, "Oracle Email Center"

Section 1.1.4, "Oracle Scripting"

Section 1.1.5, "Oracle Interaction Center Intelligence"

Section 1.1.6, "Oracle One-to-One Fulfillment"

Section 1.1.7, "Oracle Interaction History"

1.1.3 Oracle Email Center

Oracle Email Center is designed to satisfy requirements for inbound customer support, e-mail interaction management, and outbound sales and marketing e-mail message processing.

Oracle Email Center helps your business respond to e-mail queries with clear and comprehensive replies in a much more efficient manner. Oracle Email Center automatically generates suggested responses and scores them according to how closely they match the requirements.

See Also

Section 1.1.1, "Oracle Advanced Inbound"

1.1.4 Oracle Scripting

Oracle Scripting is a set of tools to facilitate the process of gathering of information for the benefit of the enterprise. Oracle Scripting is composed of several components: the Script Author, the Scripting Engine, the Scripting Administration console, and the Survey Administration console.

The Script Author is the development tool with which customized business requirements are translated into miniature programs known as "scripts." Each implementation of Oracle Scripting employs at least one customized script built by Oracle Consulting, consulting partners, or the enterprise. There are various ways in which scripts can be employed to gather or distribute data for an enterprise. For example, a script can serve to unify an agent's desktop by integrating aspects of various applications, or as a survey questionnaire to solicit specific information from the sample or target population. The Script Author offers two ways to create a script, including graphical layout tools and a Script Wizard component.

The Scripting Engine is responsible for displaying the script to the end user, interpreting the end user's responses to questions and answers, and processing custom code developed in support of the script. The Scripting Engine includes two interfaces (one for agents, and one for executing a script using a Web browser). Any script executed in the Web interface requires survey campaign administration.

The Scripting Administration console provides the user interface with which script developers can launch the Script Author as a Java applet, and script administrators can administer Oracle Scripting files, as well as generate, view and analyze a panel footprint report.

The Survey Administration console provides the user interface with which survey administrators establish and maintain survey campaign information, define and manage survey deployments, and view responses from data received.

See Also

1.1.5 Oracle Interaction Center Intelligence

Oracle Interaction Center Intelligence is a Web-based reporting solution that provides intelligent reports that facilitate day-to-day operational and long-term strategic decisions.

The data is presented to the user in a easy-to-use portal format. This format gives the user a unified, role-based, easily customized view of Interaction Center information, including Oracle Universal Work Queue information, key performance measures relating to agent productivity, speed to answer, and abandon rate.

The product is built on an Oracle proprietary Java-based technology stack (Oracle CRM Foundation, sometimes referred to as Java Technology Framework or JTF).Users of Oracle Interaction Center Intelligence require minimal training, and no additional software is needed on the user's machine other than a Web browser.

Oracle Interaction Center Intelligence is based on a three-tier architecture:

- The front end (client) using the system via an Oracle Applications 11*i*-certified Web browser.

- The middle tier, which contains the Apache Web server and application server, included as part of the installation of Oracle Applications release 11*i*.

- The database tier, using an Oracle8*i* or Oracle9*i* database.

See Also

1.1.6 Oracle One-to-One Fulfillment

Oracle One-to-One Fulfillment is a framework for compiling and distributing fulfillment information to customers in electronic format. Fulfillment information includes generic collateral or personalized cover letters sent to customers or prospective customers. Oracle One-to-One Fulfillment automates this process, providing the ability to immediately satisfy a requests for information, literature, and other correspondence. Interaction center agents handle a variety of requests ranging from product and service inquiries, pricing questions, billing inquiries, and general customer care issues. Many of these requests result in some dissemination of literature, collateral, forms of application, letters, or correspondence to the customer. Oracle One-to-One Fulfillment provides the ability for interaction center administrators, mobile field representatives, marketing managers, customer care representatives, and other service agents to respond to different customer needs quickly and easily using e-mail. Oracle One-to-One Fulfillment is also used by marketing groups to send marketing information to customers. Fulfillment requests can also be generated by e-Commerce applications that send information to customers using Web clicks.

See Also

Section 1.1.1, "Oracle Advanced Inbound"

Section 1.1.2, "Oracle Advanced Outbound"

Section 1.1.3, "Oracle Email Center"

Section 1.1.4, "Oracle Scripting"

Section 1.1.5, "Oracle Interaction Center Intelligence"

Section 1.1.7, "Oracle Interaction History"

1.1.7 Oracle Interaction History

Oracle Interaction History provides a real-time repository for recording contact interactions and relevant business events between businesses and customers. Oracle Interaction History also provides user interfaces for setup administration and for viewing and querying these stored interactions and events.

See Also

Section 1.1.1, "Oracle Advanced Inbound"

Section 1.1.2, "Oracle Advanced Outbound"

2

Detailed Product Description

This section includes the following topics:

2.1 New in this Release

This section includes the following topics:

See Also

2.1.1 Script Wizard

To make the Script Author more accessible to non-technical users, Oracle Corporation has added a Script Wizard feature to the Script Author Java applet. Users can quickly and easily create simple scripts or surveys by providing script information in a series of windows known as a wizard. This feature can help reduce the time it takes to train a non-technical Script Author user to create and modify simple scripts. While hiding complexity from non-technical users, the Script Wizard feature makes it possible to re-use questions and answers, reduce keystrokes and data entry errors, and reduce some repetitive work needed to create and modify a script.

Scripts created by the Script Wizard are referred to as *wizard scripts*. In contrast, scripts created using the Script Author's graphical editing tools are now referred to as *graphical scripts*.

Wizard scripts can be listed, created, edited, and deployed only from the Script Wizard feature of the Script Author. You can also generate a graphical copy of any wizard script, to view or edit using the Script Author's visual layout tools. The Script Wizard cannot open any graphical script for editing. For this reason, when creating a graphical copy, the original wizard script is retained.

Just like graphical scripts, a wizard script can include start and termination nodes, panels, and default and distinct branching. Groups, which in a graphical script serve the dual roles of grouping related functionality and containing shortcut

properties, are not supported in wizard scripts, with one notable exception. Each wizard script automatically includes a single, empty group with the name Disconnect group and the WrapUpShortcut value in the shortcut property. This enables the Disconnect button in the agent interface at runtime.

Additional groups, as well as conditional or indeterminate branching, or custom code cannot be added to a wizard script, nor does the Script Wizard provide functionality to create shortcut buttons or to populate the script information area.

Despite its limitations, the Script Wizard can create fairly complex scripts. When used in the agent interface, each wizard script automatically contains a valid Disconnect button and (unless disabled through other means) a valid Suspend button. Wizard scripts can also include answer validation using best practice Java methods, created with a few clicks in the question manager portion of the wizard. Additionally, panels meant to contain information only can be automatically provided with a Continue button without requiring the Script Wizard user to specify a panel answer.

If you want to develop a script using the Script Wizard, but wish to add other features, you must save a graphical copy and make modifications using Script Author's standard visual tools. Since graphical scripts cannot be opened in the Script Wizard, ensure that you have made all changes to the wizard script before converting a copy. This approach is recommend if the script must include any of the following:

- Groups

- Conditional branching

- Indeterminate branching

- Custom code

- Actions, pre-actions or post-actions of panels or of the global script

- Panels with more than one style in the panel layout

- Customized panel layout, including HTML tables

Using this process, script development work can be split between more and less technical users. For example, a non-technical user can create the basic script text, questions, and flow strictly using the Script Wizard. Trained users of the Script Author can then create a graphical copy of the script and add more advanced functions, including custom or best practice Java commands, PL/SQL, conditional or indeterminate branching, groups containing functionality and shortcuts, and so forth.

2.1.2 Suspension and Resumption of Script Transactions

Oracle Scripting now provides the ability to suspend a Scripting Engine agent interface transaction, saving all the information collected thus far, and the current location of the script, in the applications database. As soon as business applications integrate this ability into their user interfaces, then agents using Oracle Scripting will be able to suspend a script transaction, and later resume the suspended transaction.

- When a script is suspended, a snapshot in time of the Oracle Scripting transaction is saved to the database. The flow through the script, answers provided, and current state are all saved.

- Upon resumption of the script, the conversation (and any data exchange) continues where it ended in the flow during the previous conversation and associated Oracle Scripting transaction.

- The resumed transaction will then be associated with a new interaction.

Reserved for Future Functionality

Currently, no applications take advantage of the script suspension and resumption capabilities. At this time, you can suspend any script transaction executed in the Scripting Engine agent interface. In the future, business applications such as Oracle TeleSales, Oracle Collections, or the Customer Support module of Oracle TeleService are expected to incorporate into their user interfaces a method to resume a suspended script.

Current Functionality

In Oracle Applications release 11.5.9 (CRM Family Pack Q) you can only resume a suspended transaction from Oracle Scripting's Suspended Transactions Chooser form. This is functionality that has been added to the Navigator for the Scripting Agent, Vision Enterprises and Scripting User responsibilities.

- At least one transaction must be suspended before a transaction can be resumed.

- You must know how to identify your script transaction in order to resume it.

Steps to Resume a Suspended Script in the Agent Interface

1. From the Navigator, select **Suspended Transactions Chooser** and click **Open**.

   The Restart Suspended Transactions window appears.

   From the View menu, select **Find All**

2. To view all suspended transactions, from the View menu, select **Find All.**

3. To locate a specific transaction:

   a. From the **View** menu, select **Query By Example > Enter**.

   b. Enter your search criteria in the appropriate field.

   c. From the **View** menu, select **Query By Example > Run**.

      The Restart Suspended Transactions window refreshes, listing all suspended transactions that match your search criteria.

4. Click in the row representing the appropriate Oracle Scripting transaction to resume.

5. Click **Start Script.**

   The Script Chooser automatically selects the appropriate script and minimizes.

   A new window appears, containing the resumed script. State is returned for all parameters to the equivalent state at the point when the script was suspended.

   - The active panel upon resumption is the same panel that was active when the script was suspended.

   - Responses for all questions already answered appear in the Progress area.

   - Blackboard values for any blackboard commands and all answers up to this point in the script are restored to static memory.

   - Panel footprinting and answer collection information (if enabled) is enabled.

2.1.3 New Global Script Properties

Global script properties are those properties that apply to an entire script. Oracle Scripting release 11.5.9 introduces two new Script Author global properties: Script Type (wizard script or graphical script) and the boolean Suspendable option (on or "true" by default).

Script Type

The script type property itself is not visible from the user interface; however, this property is determined by the method you use to view or modify a script. You cannot view a wizard script from a script of "graphical script" script type, nor can you view a script of type "graphical script" from the wizard tool. Note that wizard

scripts can be converted or saved to a graphical script format; doing so retains the wizard script, generating a copy that is a graphical script.

For more information on Script Type, see Section 2.4.1.4, "Script Wizard".

Suspendable Option

The Suspendable check box option is visible only from a graphical script, in the script properties dialog (**File** > **Script Properties**). This option is selected by default. Wizard scripts also default this option to "True." There is currently no method for changing the suspendable feature for a wizard script other than to convert the script to a graphical script.

When the Suspendable option is checked, and the **IES : Display Suspend Button on Script Frame** profile option is set to **True**, the Scripting Engine agent interface displays a Suspend button on the bottom of the script frame. This button is hidden if the profile is set to False (or remains null). When this button is clicked by an agent during script runtime, the current script interaction is suspended.

In the future, when business applications integrated with Oracle Scripting incorporate the ability to resume scripts, this feature will provide the much requested ability to resume an interrupted script transaction. A resumed transaction retains all information collected in a script up to the point at which it was suspended, including footprinting and answer collection information. Additionally, for reporting purposes (if relevant), the resumed transaction will then be associated with a new interaction.

In the current release, suspended scripts can be resumed for scripts executed in the Scripting Engine agent interface in stand-alone mode (using no business applications). From the Navigator, select **Suspended Transactions Chooser**, populate the list (**View > Query By Example > Run**), and click **Start Script**.

Only scripts executed in the Scripting Engine agent interface currently support suspension of script interactions.

2.1.4 Integration with Oracle Interaction History

Oracle Scripting is now integrated with Oracle Interaction History. In order to discriminate in Oracle Interaction History between scripts executed in the agent interface or from a Web browser as a survey or Web script, there are two new activity types: script and survey.

Script activity types indicate that the interaction included a script type activity (the execution of a script using the Scripting Engine agent interface). The Interaction History record can be used to allow the user to drill down to a source script.

Survey activity types indicate that the interaction included a survey or Web script activity (execution of a script in a Web browser, based on existing survey campaign information). The Interaction History record can be used to allow the user to drill down to a source survey questionnaire script.

If an interaction has already been started by an application that invokes a script or a survey, Oracle Scripting will add a script or survey activity to the Interaction History table for the current interaction.

This functionality requires that the application that launches the script or survey pass the Interaction ID to the Scripting Engine as it launches the script.

Applications that have committed to implementing the new methodology for launching the script and passing the Interaction ID for release 11.5.9 include:

- Oracle Telesales
- Oracle Marketing

2.1.5 Access to Survey Administration

Due to changes in the Oracle Scripting survey administration architecture, login to the Survey Administration console must be performed specifically through the personal homepage (PHP) login. This login method provides access to standard and self-service Oracle Applications. Survey administration is considered a self-service application. Users with the Survey Administrator responsibility can select this responsibility from the navigator, and then access the Survey Administration console by clicking Survey Home. For more information, see Section 2.2, "Modified in this Release" > Section 2.2.2, "Survey Administration Architecture Modifications".

2.1.6 Scripts Locked in Active Survey Campaigns

Survey questionnaire scripts used in active survey campaigns are now locked. This prevents script developers from making any changes to the script that could accidentally overwrite survey-related data, changing the questions that are asked as part of the survey, or otherwise invalidating the planning assumptions used to establish the survey campaign.

Note that users with the Scripting Administrator responsibility can still delete the script specified as the survey questionnaire from the Scripting Administration console.

2.1.7 Integration with Oracle Discoverer Reporting

The panel footprint report previously accessible from the Survey Administration console is now only accessible through the Scripting Administration Console.

The criteria for the remaining reports previously accessible from the Survey Administration console have been enhanced. These reports and some additional reports are now available as workbooks available through Oracle Discoverer. For more information, see Section C, "Oracle Discoverer Workbooks".

2.1.8 New System Profile Options for Oracle Scripting

Oracle Scripting now makes use new system profiles to control its behavior, as described below.

2.1.8.1 Control Initial Agent Interface Script Frame Size

The new **IES : Initial Script Frame Size** profile option controls the initial size of the Scripting Engine frame window in the Forms/Java agent user interface. Before the introduction of this profile, the initial Oracle Forms frame containing a script running in the agent interface was always 800 pixels in width and 600 pixels in height. Administrators can now change this 800 X 600 default setting to 1024 X 768, 1280 X 1024, or Maximized, to establish the frame size upon launch to accommodate various monitor resolutions.

2.1.8.2 Control Display of Suspend Button in Agent Interface

The new **IES : Display Suspend Button on Script Frame** profile option supports Oracle Scripting's new feature in which a scripting transaction in the agent interface can be suspended, and later resumed. When this profile is set to True, and the Suspendable global script property is enabled, the agent interface displays a Suspend button. This option is hidden if this profile is set to False (or remains null), or for scripts in which the Suspendable attribute is set to False.

2.1.8.3 Control Script Author Java Applet Messages

The new **IES : AUTHOR DEBUG MODE** profile option controls display of informational and error messages for Script Author debugging. When enabled, this profile option provides additional informational and error messages for Script Author debugging, generated in the Apache server logs. Note that you must administer (configure) your Apache Jserv to send console output to the Apache error log.

2.1.9 Scripting Administration Console

The Scripting Administration console provides script administrators the interface to launch the Script Author as a Java applet, to view and delete deployed scripts, to view and administer custom Java archive files, to map JAR files to specific scripts, and to view agent interface reports.

This console is accessed by logging into Oracle HTML-based applications using a user account with the Scripting Administrator responsibility.

2.1.10 Scripting Administrator Responsibility

Scripting Administrator is a new Oracle Applications responsibility that provides users of Oracle HTML-based applications with access to the Scripting Administration console.

2.1.11 Script Author Java Applet

From the Home tab of the Scripting Administration console, you can launch the Script Author as a Java applet.

What Has Changed?

In all previous releases, the Script Author component was a two-tiered stand-alone Java application. In releases 11.5.1 through 11.5.5, this application was available via shipped product CDs or on installable patches distributed through Oracle*Meta*Link.

Using 11.5.6, 11.5.7, and baseline 11.5.8 releases, the installable Script Author component is downloaded from the IES_TOP directory on the applications server (after patching to the appropriate level).

In all cases, the Script Author was installed on Windows operating system clients only, using the Oracle Universal Installer. This application was executed using a command line batch file script.

How Does It Work Now?

As of Interaction Center Family Pack P and later or Oracle Scripting release 11.5.9 or later, the Script Author can be launched by any user accessing the Scripting Administration console by clicking **Launch Script Author**. A separate Oracle JInitiator window appears. The Script Author then executes as a Java applet from the Oracle JInitiator session, using the current logged-in user's database authentication information. The Scripting Administration console is therefore accessible to any operating system supported by Oracle Applications. The new

Scripting Administration Java Server Pages (JSP) user interface is accessible by an Oracle Applications user with the new Scripting Administrator responsibility.

Benefits

Benefits of running the Script Author from an existing Oracle Applications session as a Java applet include:

- Ability to create, modify and deploy scripts through a firewall using HTTP and secure HTTP.

- Single login and authentication to the current production environment. Logging into the Scripting Administration console and subsequently launching the Script Author enables the user to access the database instance, obtain reusable commands from the command library, deploy scripts, and access PL/SQL stored procedures or packages without specifying database connection and login information (providing the same instance is accessed).

- No requirement to implement (locate, download, unpackage and install) the Script Author component.

- Apps user name and password no longer required to deploy scripts.

Other Ramifications

Since the Script Author is no longer installed on a script developer's client installation, there is no longer a home directory for Script Author files. As a result:

- Script Author stand-alone online help files, previously available in the Docs directory, is no longer available. For assistance using Script Author, refer to *Oracle Scripting User Guide.*

- The CCTBUILDER.PROPERTIES file is now written to the Oracle JInitiator home directory.

  This file, created automatically the first time the Script Author is used, and updated each time the application is closed, stores user settings such as Script Author window size and location and database connection parameters.

- The EN.CONTENTS and BUNDLE.CONTENTS files are no longer created or required.

  The default English strings for the Oracle Scripting application are stored in the EN.CONTENTS file. Previously, when the Script Author stand-alone application was run for the first time after installation, the user was asked to download strings for a specific language from the database. If the user decided

not to connect to the database, BUNDLE.CONTENTS was created from EN.CONTENTS.

In current releases, using the Java applet, localized strings for the Script Author are downloaded from the database each time the Script Author is launched. The applet uses the default language setting for the user and attempts to load the localized strings from the database for that language. If the strings for that language are not found, then the user receives a warning message, and the default strings in English are used.

2.1.12 Administering Oracle Scripting Files

The new Oracle Scripting Administration console allows Oracle Scripting administrative users to manage Oracle Scripting-related files using the Administration tab of this HTML-based administrative user interface.

As before, the only way to deploy a script to the applications database is from the Script Author. Scripts deployed using the Script Author Java applet can now be managed (viewed and deleted) using the Deployed Scripts subtab. Using filter options from the View list, you can display lists of records meeting the following criteria:

- Active scripts only, deployed by the current user name

- Active scripts only, deployed by any user

- Active and inactive scripts, deployed by the current user name

- Active and inactive scripts, deployed by any user

2.1.13 Passing Parameters to the Web Interface

When executing scripts in the Scripting Engine Web interface, you can pass parameters into an Oracle Scripting session from external applications. These parameters are stored in the Scripting blackboard, and are available for use throughout the script session. This method is appropriate regardless of whether you run scripts as surveys, or from a self-service Web application such as Oracle iSupport.

Using methods appropriate to your source application, pass each parameter to the Scripting blackboard by concatenating each key and value to the runtime survey URL required to initiate a script session.

After the last parameter required as part of the URL to initiate the script transaction, pass in values from external applications by concatenating an ampersand (&), the

keyname, an equal sign (=), and the key value to the URL required to initiate the survey.

Thus, if you want to pass an Oracle iSupport transaction ID (for example, 999) to a Web script launched from iSupport, you can append to the survey URL the key value pair **transaction_id=999**. This will then be accessible as a blackboard value from the script with and the key/value pair (transaction_id, 999).

Initial URL May Differ

The initial runtime survey URL used in your environment may differ based on method of execution for your environment. For example:

- The initial JSP page called may be IESSVYMENUBASED.JSP if hosted on a Web-based application.

  This uses the Oracle Applications authentication information from the current session to launch an Oracle Scripting transaction in a Web browser.

- The initial JSP page called may be IESSVYMAIN.JSP if executed as a survey.

  This uses a guest Oracle Applications account for which the only function in an Oracle Applications session is to execute a script once per login in a Web browser. The Oracle Applications session is automatically terminated when the final page resource is displayed.

- Each Scripting Engine transaction using the Web interface requires at least one parameter, dID (deployment ID). List-based surveys also require a unique rID (respondent ID) parameter.

Syntax and Examples

The appropriate syntax follows. This includes two additional parameters (Keyname 1 and Keyname 2):

```
http://<hostname>:<port>/OA_HTML/<hosted or stand-alone survey jsp>?<Deployment
ID>&<Respondent ID>&<Keyname1>=<Value1>&<Keyname2>=<Value2>
```

An example of a list-based stand-alone survey URL passing a customer ID of 12345 and a customer name of Smith appears as follows:

```
http://server1.company.com:7777/OA_HTML/iessvymain.jsp?dID=263&CUSTOMER_
ID=12345&CUSTOMER_NAME=Smith
```

An example of a hosted survey URL passing a Party ID of 1000 and a customer name of Chan appears as follows:

```
http://server2.company.com:8888/OA_HTML/iessvymenubased.jsp?dID=374&PARTY_
ID=1000&CUSTOMER_NAME=Chan
```

The limit to parameters that can be passed is restricted by the Web browser's supported limit of URL characters.

2.1.14 Script Author Servlet Validation Page

The Script Author Servlet provides Oracle Scripting with access to the database to perform Script Author activities such as deploying a script, accessing the command library, saving scripts in progress to the database, and saving or retrieving wizard scripts. This servlet can be verified by entering the following URL in a Web browser:

```
http://<hostname>:<port>/<servlet_zone>/
oracle.apps.ies.builder.httpservice.BuilderHttpServlet
```

The servlet zone entered into the URL should be the value set in the Apps Servlet Agent system profile. If the environment is properly configured, a page with the following message will display:

"I am up and running but I only talk POST"

2.1.15 Requirement for ICX: Forms Launcher Profile

In order to use the Scripting Administration console to launch the Script Author Java applet, the ICX: Forms Launcher profile must be set with the appropriate URL. This profile setting specifies the path of the executable CGI program used to launch the Forms applet via Oracle JInitiator, and is also used by the Script Author applet.

The syntax for this profile is: http://<server>.<domain>:<port>/dev60cgi/<cgi executable program>. The executable program name may differ based on platform. For UNIX, an example is:

```
http://server1.vision.com:8032/dev60cgi/f60cgi
```

If a forms session (for example, an Oracle Applications session) can launch in your environment, this profile is appropriately set.

2.2 Modified in this Release

This section includes the following topics:

Section 2.2.1, "Footprint Report Accessibility"

See Also

2.2.1 Footprint Report Accessibility

As of release 11.5.9 and later, the Scripting footprint report is only accessible through the Scripting Administration Console. This HTML user interface requires the Scripting Administrator responsibility.

Other survey reports, previously available from the Analysis tab of the Survey Administration console, are now accessible to users of Oracle Discoverer workbooks. For more information, see Section C, "Oracle Discoverer Workbooks".

2.2.2 Survey Administration Architecture Modifications

Survey administrators now have access to an improved Survey Administration console. This HTML user interface is built on the Oracle Applications Framework using JRAD technology, replacing the JTT (CRM Foundation) technical stack.

This HTML user interface is accessed only from the Oracle Personal Homepage (PHP) login by a user with the Survey Administrator responsibility. The syntax for the Personal Homepage URL is typically as follows:

```
http://<server>.<domain>:<port>/OA_HTML/<country_code>/ICXINDEX.htm
```

A valid PHP URL might look like the following sample:

```
http://server1.vision.com:8000/OA_HTML/US/ICXINDEX.htm
```

Once you log into the PHP, the Navigator appears, listing the responsibilities assigned to your Oracle Applications account. Survey Administration is listed in the Self Service category.

After selecting Survey Administrator, the Navigator refreshes, displaying in a new column on the right all functions associated with the selected responsibility. At this time, the only function is named Survey Home. Upon selecting this function, the Survey Administration console appears.

Improved Flow and Access to Administration Tasks

Functions previously dispersed throughout the console in subtab menus are now accessible from root pages for a tab. For example, the Survey Campaigns page provides access to survey and cycle creation, deployment definition, deployment activation, and review of responses received.

A new Survey Resources tab provides access to defined JSP resources and the capability of defining new resources that map to physical files in $OA_HTML.

For each page in this administration console, you can now create definitions without having all information required to execute at runtime, without losing your work.

Other applications are still hosted in this administrative console. The Audience tab provides access to Oracle Marketing functionality, and the Invitations tab provides access to Oracle One-to-One Fulfillment functionality. These are required for targeted survey campaigns using marketing lists distributed by the fulfillment engine.

2.2.3 Class Path Specification No Longer Required

Custom code loaded from the Scripting Administration console to the applications database need not be added to the class path of any configuration file.

Conversely, custom code deployed manually to the APPL_TOP must still have the class path specified (in the JSERV.PROPERTIES file for enterprises using the Apache mid-tier architecture, or in the of the APPSWEB.CFG file for enterprises using the caching architecture).

Manually deployed Java archives will not appear in the list of JAR files on the Administration > Jar Listings page.

2.2.4 Method of Connecting to the Database from Script Author

Previously, using the Script Author stand-alone Java application, the Script Author client made a direct JDBC connection with the database. This required the Script Author user to log into the database separately, with apps privileges, in order to deploy scripts, to access the command library, or to access database commands.

In release 11.5.9 and later, you must log into Oracle Applications (with a user assigned the Scripting Administrator responsibility) in order to launch and use the Script Author. The Script Author Java applet uses the current Oracle Applications authentication information to connect to the database without requesting the script developer to log in. Apps level privileges are no longer required to deploy scripts.

The Script Author Java applet connects to an Apache mid-tier servlet via HTTP. This Author servlet makes a JDBC connection to the database, and all database-based operations in the Script Author are executed by this servlet, including script deployment, saving in-progress scripts to the database, and accessing the command library.

2.2.5 Embedded Values Stored in Script MetaData Instead of Panel HTML

The properties of embedded values created using the Script Author are no longer stored in the HTML of specific panels in which the embedded values apply. Instead, they are stored in the metadata of the script itself, similar to the properties of all other Script Author script objects such as panels and branches.

As a result, the behavior of building or modifying scripts containing embedded values differs from previous releases. It is no longer sufficient to export a panel's HTML text (containing embedded values) and import the text into another panel. This action will cause an embedded value to be created in the subsequent panel, but without defined properties (since those properties are no longer stored in the panel HTML). To execute this embedded value successfully (or to save the script), you must use the Script Author panel layout editor, highlight the embedded value, access the command properties associated with the embedded value, and reassociate the desired properties for the embedded value. Obviously, this eliminates time savings for using the export/import HTML method to duplicate embedded values from panel to panel. As a workaround, you can duplicate on the canvas a panel containing an embedded value, and modify the properties of the cloned panel. (At minimum, change each answer name in the cloned panel, to maintain answer name uniqueness in the script.)

The Script Author's syntax checking engine now ensures that all embedded values have properties associated with them. Thus, for example, importing panel HTML containing an embedded value requires the script developer to associate the

embedded value properties prior to saving the panel HTML. If you attempt to save panel HTML without defining embedded value properties, the error message indicates that previous embedded values and their properties will be deleted from the script metadata, and will ask the user to confirm this operation. Answering "Yes" will save the panel HTML without the embedded value, while answering "No" will cancel the save operation, providing the user the opportunity to associate the embedded value properties prior to saving again.

2.2.6 Improved End-of-Script Processing

Scripting is a stateful application which incorporates the concept of the script "transaction." A script transaction begins when a script is launched in the Scripting Engine (agent or Web interface), and completes when the script runs to completion. When the transaction ends, the Scripting Engine must execute several tasks. These include:

- Update record in IES_TRANSACTIONS table with script end time

- Write answers collected during transaction into appropriate table (IES_ QUESTION_DATA), assuming answer collection is enabled for the script

- Write footprinting data into appropriate table (IES_PANEL_DATA), assuming answer collection or footprinting is enabled for the script

In previous releases of Oracle Scripting, at the end of a script transaction in the agent interface, the window in which the script appeared closes, and then the application is locked until all of the above tasks are complete. Large scripts could conceivably require a minute or more for end-of-script processing to complete. In these circumstances, the user must wait before launching the next script (even if an agent has already received another call).

End-of-script processing now occurs in separate thread using a different database connection. This allows the next script transaction to be started (in the existing thread) before the data from the prior session is completely written to the database. The data from the previous transaction is concurrently written to the database using the newly instantiated thread. The net end user effect is the ability of agent interface users to start a new Scripting transaction much sooner.

2.2.7 Answer Collection Property Now Forces Footprinting Data Collection

Answer collection is the recording of end user responses ("answers") to all answer controls ("questions") that are marked in the script as collectable. Answers are collected for each transaction or session of the script running in the Scripting

Engine, in either interface. If enabled (at the script level), answer collection data is collected in table IES_QUESTION_DATA.

Footprinting is the recording in the database of which panels in a script transaction were visited, and the duration of time in milliseconds before the next panel is requested. Footprint data is stored in table IES_PANEL_DATA.

For each script, you can designate the collection of footprint data by enabling the Footprinting option as a global script property. For each new script created, this option is selected by default.

Additionally, regardless of whether the Footprinting option is enabled, footprinting data is now also automatically saved to the database when the Answer Collection option is enabled. This change is effected in this release to improve the quality of data saved from a script transaction for reporting purposes. This ensures a link between each response provided at runtime, and the specific panel instance from which that response was provided. Accordingly, a new column (PANEL_DATA_ID) has been added to the answer collection table, IES_QUESTION_DATA. This column contains the foreign key reference to the footprinting table, IES_PANEL_DATA.

If the Footprinting option is selected but the Answer Collection option is not, only footprinting data for each session or transaction of that script will be saved to footprinting tables. Obviously, if neither option is selected for a specific script, no footprinting or answer collection data is saved.

If neither footprinting nor answer collection are enabled for a script:

- No data will be available from which to view individual responses from the Responses tab of the Survey Administration console.

- No data will display in the panel footprint report executed from the Reports tab of the Scripting Administration console.

If answer collection is disabled for a script, but footprinting is enabled:

- No data will be available from which to view individual responses from the Responses tab of the Survey Administration console.

- Footprinting data will still be recorded in the IES_PANEL_DATA table if enabled for the script.

References
- For more information, see Footprinting and Answer Collection.

2.3 Obsolete in this Release

This section includes the following topics:

Section 2.3.1, "Analysis Tab Obsolete in Survey User Interface"

Section 2.3.2, "Quick Find Menu Obsolete"

Section 2.3.3, "Default Survey Resources and Profiles Obsolete"

Section 2.3.5, "Script Author Implementation Obsolete"

Section 2.3.6, "Cannot Launch Script Author Independently"

Section 2.3.7, "Database Connection Information No Longer Required"

Section 2.3.8, "Connecting to Multiple Scripting Environments Not Supported"

See Also

Section 2.1, "New in this Release"

Section 2.2, "Modified in this Release"

2.3.1 Analysis Tab Obsolete in Survey User Interface

The Analysis tab has been removed from the Survey Administration console. Reports previously available through the survey user interface, as well as additional reports, are now accessible using Oracle Discoverer workbooks. For more information, see Section C, "Oracle Discoverer Workbooks".

2.3.2 Quick Find Menu Obsolete

Prior to this release, the Survey Administration console included a Quick Find menu feature to locate survey campaigns, cycles, or deployments. For any CRM Technology Foundation (JTT) application associated with product code IES (Oracle Scripting), this results in the appearance of the Quick Find menu below navigation tabs. Thus, in Oracle Scripting Release 11.5.8 plus Interaction Center Family Pack P implementations, this menu appears in both the Survey Administration console (where it provides quick access to survey campaigns, cycles, and deployments) and the Scripting Administration console (where it serves no function).

As of Oracle Scripting release 11.5.9 (Interaction Center Family Pack Q or later), the Quick Find menu is now obsolete. The redesigned Survey Administration console is streamlined in a new technical stack for improved flow and rapid survey campaign administration. New implementations will exclude this obsolete feature. For

implementations upgrading from previous releases, you can remove the association between Oracle Scripting and this menu, as described in Removing the Quick Find Menu.

2.3.3 Default Survey Resources and Profiles Obsolete

Prior to this release, the Survey Administration console included the ability to define default survey resources in the Profile section for a specific Oracle Applications account. With the architecture changes for survey administration functionality, this feature is obsolete. This includes the following profiles that previously held values for default survey resources: IES : SVY ERROR PAGE, IES : SVY HEADER PAGE, and IES: SVY FINAL PAGE.

2.3.4 IES: Time Format Profile Obsolete

The IES: TIME FORMAT profile was required for the previous architecture supporting Oracle Scripting's survey functionary. This profile is no longer required. Leaving this profile set to any previous setting will have no negative consequences.

2.3.5 Script Author Implementation Obsolete

Since Script Author is now launched as a Java applet from the Scripting Administration console, the requirement to implement the Script Author is obsolete.

2.3.6 Cannot Launch Script Author Independently

Using the separate client application, previous releases of the Script Author could be launched without connection to an Oracle Applications 11*i* environment.

Using the Script Author Java applet, Script Author users must now connect to a functioning Oracle Applications 11*i* environment before launching the Script Author Java applet.

Launching the Script Author without being connected to an Oracle Applications 11*i* environment is no longer supported with Script Author release 1.6.2.01 and later.

2.3.7 Database Connection Information No Longer Required

Since starting a valid Oracle Applications session is now a prerequisite of launching the Script Author Java applet, the requirement to provide database connection information is now obsolete. The Script Author applet uses the authenticated user session in the middle tier. Whenever a database connection is needed, the Script

Author requests a database connection from the mid-tier pool via the authenticated session.

Thus, no connection dialog information is required to be provided by the Script Author user to deploy scripts, access the command library, or access database commands.

2.3.8 Connecting to Multiple Scripting Environments Not Supported

With the previous Script Author application, the user could connect from one Script Author session to any number of Oracle Applications 11*i* environments, simply by supplying the appropriate database information and apps database password. Using the Script Author Java applet, the user is only allowed to connect to the one environment from which the Applet was launched. In order to connect to a different environment, the user must launch the Author from that environment's Scripting Administration console.

2.4 Features

Oracle Scripting is a set of tools to facilitate the process of gathering or exchanging information for the benefit of the enterprise. Oracle Scripting is composed of four components: the Script Author, the Scripting Engine, the Scripting Administration console, and the Survey Administration console.

This section includes the following topics:

- Section 2.4.1, "Script Author Features"

- Section 2.4.2, "Scripting Engine Features"

- Section 2.4.3, "Scripting Administration Console Features"

- Section 2.4.4, "Survey Administration Console Features"

- Section 2.4.5, "Accessibility"

2.4.1 Script Author Features

The component of Oracle Scripting that provides script developers with graphical layout tools to create, modify, and deploy scripts is referred to as the Script Author. The Script Author is a Java applet launched from the Scripting Administration console. Using this authoring environment, trained developers build scripts to be executed using the Scripting Engine. Scripts created using the graphical tools are referred to as graphical scripts. Scripts created using the Script Wizard feature of

the Script Author are referred to as wizard scripts. Custom code can be associated with graphical scripts globally, or at certain points in the script (corresponding with flow of the script at runtime). This code can execute a variety of customized or predefined commands to extend the functionality of the script.

This section includes the following topics:

- Section 2.4.1.1, "Script Author Graphical Layout"

- Section 2.4.1.2, "Graphical Script Object and Branch Properties"

- Section 2.4.1.3, "Panel Layout Editor"

- Section 2.4.1.4, "Script Wizard"

- Section 2.4.1.5, "Script Author File Management"

See Also

- Section 2.4.2, "Scripting Engine Features"

- Section 2.4.3, "Scripting Administration Console Features"

- Section 2.4.4, "Survey Administration Console Features"

- Section 2.4.5, "Accessibility"

2.4.1.1 Script Author Graphical Layout

The Script Author is a graphical layout environment. Script developers build graphical scripts by selecting objects from one of two tool palettes, and placing them on the work area (the canvas). Objects are connected by branches.

Project and Syntax Tab

The left side of the Script Author development environment displays either a Project tab or a Syntax tab.

The Project tab contains a table displaying properties of the currently selected object. Above this table, you can list objects on the canvas using one of four views:

- The script view displays all objects as defined on the canvas, including their hierarchy.

- The group view displays an alphabetical listing of groups defined in the script only.

- The block view displays an alphabetical listing of blocks defined in the script only.

■　The panel view displays an alphabetical listing of panel defined in the script only.

The Syntax tab provides a list of each object on the canvas that breaks Script Author syntax rules.

Debug Pane

Clicking on any listed error will cause a section of the canvas to be replaced with the Debug pane. The debug pane lists any available information about the selected problem.

Status Bar

At the very bottom of the Script Author work area, below the canvas and the Project and Syntax tabs, is a status bar. This indicates the status after a defined action such as checking syntax, saving a script, or deploying a script to the database.

Hierarchical Graphs

Scripts consist of one or more graphs (collections of Script Author objects). Each graph automatically includes a Start node which indicates where runtime flow begins. No properties can be associated with a Start node.

A script can contain any number of graphs. A graph created at any level of a script is known as a child graph. The canvas on which that sub-graph is created is known as the parent graph. The graph shown when you first create or open a script is referred to as the root graph, and has no parent. The script session at runtime ends when the Termination node on the root graph is reached.

New graphs are created using group or block objects. After creating a group or block object, you must explicitly select it and navigate down into it to view the child graph. Any graph can contain any number of other Script Author objects.

When an object in the flow of the script contains a child graph in the form of a group or block, any objects included in that child graph are evaluated before the flow on the root graph continues to be evaluated. A child graph at any level is exited at runtime when the Termination node on that graph is reached in the flow of the script. Processing (flow of the script) then returns to the child graph's next object in the parent graph flow.

Certain syntactical rules apply to every graph. All objects in a script must be connected with branches. At minimum, each graph must contain a Start node and a Termination node, connected by a default branch. When you include a group or

block in a script, its child graph must meet the same syntactic rules as the root graph.

See Also

2.4.1.2 Graphical Script Object and Branch Properties

The two non-configurable objects in every graph of a graphical script are the Start node and the Termination node. These have no names or configurable properties associated with them.

Each configurable object and every branch created using the Script Author graphical tools has properties associated with it. These include a name property and a comments property. The comments property is blank by default. In each graphical script, every configurable object and branch placed on a graph is automatically assigned a unique default name. Script developers should replace default names for objects with meaningful names. It is also recommended to provide meaningful names to distinct, conditional, or indeterminate branches to provide an indication of each branch's function. This is not required for default branches; names assigned to default branches are not displayed in the Script Author.

The script itself is considered a configurable object. Thus, it too has a name and comment property. These are referred to as global script properties. The global script name is the name under which the script is stored in the database when deployed from the Script Author. This must be distinguished from the name assigned to the script file, which may differ. Also please note, when saving a graphical script, the Script Author applet will automatically provide a file extension of .SCRIPT if no file extension is provided by the user. Script Author can open files saved with both .SCR and .SCRIPT file extensions.

Object properties for every object and branch can be accessed in one of two ways:

1. By double-clicking on the object.

2. By right-clicking on the object and selecting **Edit Blob Properties** from the pop-up menu. This option is not available on the Macintosh platform.

Additionally, properties can be assigned to the global script by selecting **File > Script Properties**.

In order to be syntactically correct, all objects on a particular graph must have properties associated with them, and be connected with appropriate branching from the Start node to the Termination node. Panels must contain at least one "question" or answer definition. Groups and blocks must contain proper termination of their child graphs.

See Also

- Section 2.4.1.1, "Script Author Graphical Layout"

- Section 2.4.1.3, "Panel Layout Editor"

- Section 2.4.1.4, "Script Wizard"

- Section 2.4.1.5, "Script Author File Management"

2.4.1.3 Panel Layout Editor

The panel layout editor feature of the Script Author allows script developers to quickly define simple HTML layouts for a panel within a graphical script without writing HTML code. This simple graphic interface provides users with the ability to:

- Enter and format simple panel content

- Manipulate typeface, font size and color

- Create bulleted and ordered lists

- Import existing HTML or export panel content as HTML

- Import GIF or JPG images

In addition, the panel layout editor produces syntactically correct HTML content for Scripting-specific features such as embedded values without having to learn custom Oracle Scripting HTML tags and syntax.

Questions (or answer definitions), images, and text can be interspersed and formatted as desired.

The panel layout editor includes one-click formatting of text into two distinct styles. *Instructional text* indicates to the script end user specific instructions. *Spoken text* indicates text an agent would speak (using the agent interface) or that a customer or survey respondent would read for communication of primary information for that panel.

The panel text editor is not intended to be used for complex layout, creation of use of HTML tables, or the use of cascading style sheets, which are not supported with

Oracle Scripting. To create complex layout, or to use features of HTML such as tables, it is expected that script developers will use full-function third-party HTML editors (or code HTML by hand) and simply import the HTML content into the Script Author.

See Also

-

-

-

-

2.4.1.4 Script Wizard

As of Oracle Scripting release 11.5.9 or later, or Interaction Center Family Pack Q or later, the Script Author includes a Script Wizard. Using this feature, script developers can quickly and easily create simple scripts or surveys by providing script information in a series of windows known as a wizard.

Scripts created with the Script Wizard are called wizard scripts. All features of wizard scripts are compatible with those of graphical scripts, and wizard scripts can be graphed or converted to a copy of a graphical script for viewing or modification using standard graphical tools. At this time, there is no backward compatibility, nor can graphical scripts be converted to wizard scripts.

The Script Wizard makes script development more accessible to non-technical users, although an understanding of business rules and business process flow is still absolutely essential to developing successful scripts using this feature. Since wizard scripts are created from the Script Author, access to the Scripting Administrator responsibility is also required.

Wizard scripts can be created, edited, saved to the database without deployment, or deployed to the database. Unlike graphical scripts, wizard scripts cannot be saved locally in a computer's file system. Wizard scripts can only be listed in a user interface from within the Script Wizard.

Using a SQL tool such as SQL*Plus, you can list properties for saved or deployed scripts. The metadata for saved scripts are stored in table IES_DEV_SCRIPTS, and the metadata for deployed scripts are stored in table IES_DEPLOYED_SCRIPTS. To differentiate between script types, wizard scripts are assigned a SCRIPT_CATEGORY value of 0, whereas graphical scripts are assigned a SCRIPT_CATEGORY value of 1.

Highlights of wizard scripts include the following:

- A wizard script cannot be saved or deployed until it is syntactically correct. When you complete a wizard script, you can save it and continue working in the wizard, you can save and exit the script, or you can save, deploy, and exit. The product of progressing through the Script Wizard and saving your work is always a valid, deployable wizard script.

- From the Script Wizard, existing wizard scripts can be edited, copied, deleted, graphed for use in the graphical Script Author tools, or deployed.

- Each wizard script automatically contains a valid Disconnect button. This is only visible in the Scripting Engine agent interface.

- Each wizard script automatically contains a valid Suspend button, unless the Suspend feature is disabled through other means. This is only visible in the Scripting Engine agent interface.

- When defining question detail, wizard script developers can include default answers to questions by providing a value in the Default Value field. These can be accepted or changed at runtime.

- When defining question detail, using a few clicks, wizard script developers can include answer validation for responses provided at runtime. You can check that the response provided is either (a) an integer, (b) an integer within a designated start and end range, (c) a valid date, or (d) a date that is not in the past (e.g, the date is equal to SYSDATE or in the future).

- Flow of a script can be controlled in part by providing an exit panel sequence. This method allows panels meant to contain information only (in contrast to panels designated to collect information) to automatically be provided with a Continue button without requiring the Script Wizard user to specify a panel answer.

Wizard scripts can contain panels, default branches, distinct branches, start and termination nodes, and a Disconnect group. Panels contain questions which may have predefined answer choices (lookup values).

For advanced wizard script features, any question may contain a default answer. Additionally, questions requiring text input may have the aforementioned answer validation. Corresponding features in graphical scripts are provided, respectively, by defining constant commands or Java commands referencing custom or best practice Java methods. These are defined for graphical scripts in the data dictionary for the particular "answer definition" or question.

See Also

- Section 2.4.1.1, "Script Author Graphical Layout"

- Section 2.4.1.2, "Graphical Script Object and Branch Properties"

- Section 2.4.1.3, "Panel Layout Editor"

- Section 2.4.1.5, "Script Author File Management"

2.4.1.5 Script Author File Management

As of Oracle Scripting release 11.5.9 or later, or Interaction Center Family Pack P or later, the Script Author is a Java applet launched from a valid Oracle Applications session. In previous releases, the Script Author was a two-tiered Windows-only client application that required separate database login and authentication at the APPS user level. With the introduction of the Java applet version of the Script Author, these restrictions are eliminated.

Script Author Editing Forward and Backward Compatibility

Scripts saved using client versions of the Script Author (versions 1.6.1.02 and earlier) *can* be opened in Java applet (1.6.2 and later) versions of the Script Author. However, scripts created or modified and saved with the Java applet version (1.6.2 or later) version of the author are *not* backward compatible, and cannot be opened in previous Script Author versions. A warning message will display for users attempting to save an older script in the applet version.

Note that older scripts can be imported as groups without change to the original older script file. This method is recommended when older files must retain their editability in client application versions of the Script Author.

Runtime Compatibility

Oracle Corporation recommends that all scripts deployed to the database from the Script Author stand-alone application (Script Author versions prior to 1.6.2) be redeployed using the Script Author Java applet, accessed from the Scripting Administration console.

Single Script Editing

The Script Author application can only open a single script at a given time. Opening a second script causes any open scripts to close, prompting the user to save any changes made to the file. This restriction applies to client application and Java applet versions of the Script Author.

- To modify two scripts simultaneously, you must open two separate instances of the Script Author (from separate Oracle Applications sessions, using two different compatible Web browsers). Objects or data will not be able to be conveyed using the clipboard when working in this fashion, although import and export capabilities are available.

- To deploy the same script to multiple environments (for example, to move a fully tested script from a development to a production environment), you must first save a copy of the script from the original environment to a local or shared file system. Then you must log out, log into the target environment, launch the Script Author applet, open the script, and deploy it to the environment.

Database Connection Information

The Script Author Java applet is accessed from a valid Oracle Applications session, and is automatically aware of the database location of scripts for that environment. Thus, opening Script Author seeded or stored commands in the command library, opening scripts from the database, and saving deployed scripts to the database all occur without the need to establish database connections.

To open deployed scripts from or deploy scripts to a different database instance, you must log into the Oracle Applications instance for that environment and launch the Script Author applet from the corresponding Scripting Administration console.

Connecting to the database from the Script Author client application (prior to Interaction Center Family Pack P or Oracle Scripting release 11.5.9) requires specification of the database host machine, port, and SID, plus use of the APPS level user name and password.

See Also

- Section 2.4.1.1, "Script Author Graphical Layout"

- Section 2.4.1.2, "Graphical Script Object and Branch Properties"

- Section 2.4.1.3, "Panel Layout Editor"

2.4.2 Scripting Engine Features

The Scripting Engine is a set of base Java classes that process a Script Author script in runtime. The script provides a customized set of pre-programmed actions to take based in part on responses provided by the end user. At runtime, the Scripting engine interprets:

- Instructions in the script,

- The end user's responses to questions and answers, and

- Custom code referenced in the script.

Interpretation of Custom Code

Custom code is referenced in the script by associating commands to objects in the script (or to the script itself) before it is deployed to the database from the Script Author. Based on the type of command, the code may be stored in the meta-data of the script itself (e.g., constant commands or PL/SQL commands), in the Scripting session (Blackboard commands), in custom code deployed to the applications server (Java commands), or in the database (commands referencing stored PL/SQL packages).

Two Runtime Interfaces

There are two runtime interfaces, both of which receive information from the base Java methods that together comprise the Scripting Engine.

The first is the **interaction center agent user interface** (agent interface). This is a Java-based UI launched from an Oracle Applications session when an agent starts a script – typically from within one of three Oracle business applications: Oracle TeleSales, Oracle Collections, or the Customer Support module of Oracle TeleService. Agents can also launch a script in "stand-alone" mode from within an Oracle Applications session, although this is recommended primarily for script testing. In current releases, scripts viewed in this model present as a Java bean wrapped in an Oracle form. HTML interpretation and rendering is provided by Java classes in the SWING library of JavaSoft's Java Development Kit (JDK) 1.1.8.

The second interface is the **Survey runtime user interface** (Web interface), which is essentially HTML running Java Server Pages (JSP) in an Oracle Applications 11*i*-certified Web browser. This runtime interface is intended for two models: for users of self-service Web applications to run a script in a Web browser, or for individuals to view a script to participate in a survey. Both sets of users run the script using their Web browser to access a particular URL. The URL specifies a set of parameters that identify an enterprise's Oracle Applications-specific Apache Web server, invoke a JSP page for survey operations, specify a deployment identification code (or dID), and (for targeted or list-based survey deployments), a unique respondent identification code or rID). When taking a survey by visiting a survey URL sent in an invitation or e-mail reminder, or when coming across the survey URL on an enterprise's Web site, the survey respondent logs into an Oracle Applications session established for a guest user. This user has no privileges other than the ability to participate in a survey. For individuals taking a survey from an integrated self-service Web application such as Oracle iSupport, the Oracle

Applications login information for the current valid Oracle Applications session is used, and the individual selects a link from iSupport to execute the script in a Web browser.

In this model, the Scripting Engine interprets the respondent's responses, the script, and custom code, passing metadata and instructions to the Web browser to be interpreted on the respondent's client computer. In addition to a script, survey execution of scripts requires a survey campaign to be established and deployed, and requires the presence of additional JSP resources (to provide a header and footer section for each survey page, an error page to display if required, and a final page after completion of the survey).

What Information Is Displayed?

Of the objects created in the Script Author, only the contents of a panel display in runtime in the Scripting Engine (the other objects control processing). Each panel contains, at minimum, one answer definition, which displays the appropriate answer control at runtime. Panels may include any number of answer definitions. At the script developer's discretion, the WYSIWYG (What You See Is What You Get) interface allows liberally customized display of formatted text and images to appear in a panel, interspersed (if desired) between answer controls. At runtime, each panel displays a button at the bottom of the panel. When the end user clicks the button, this registers the end user's responses to any other answer controls in that panel (radio buttons, checkboxes, drop-down or multi-select lists, or text in a text field, text area, or password field), and progresses the script to the next panel.

See Also

- Section 2.4.1, "Script Author Features"

- Section 2.4.3, "Scripting Administration Console Features"

- Section 2.4.4, "Survey Administration Console Features"

- Section 2.4.5, "Accessibility"

2.4.3 Scripting Administration Console Features

The Scripting Administration console is an HTML administration user interface for script developers and administrators. It has three primary functions: to launch the Script Author Java applet, to provide administration of Oracle Scripting files, and to provide access to agent application reports.

See Also

- Section 2.4.1, "Script Author Features"

- Section 2.4.2, "Scripting Engine Features"

- Section 2.4.4, "Survey Administration Console Features"

- Section 2.4.5, "Accessibility"

2.4.3.1 Script Author Applet

From the Home tab, logged-in users of the Scripting Administration console can launch the Script Author as a Java applet. No additional login information is required to launch the applet, connect to the database, access the command library, or deploy scripts.

2.4.3.2 Oracle Scripting File Administration

From the Administration tab, administrators can administer deployed scripts and Java archive files used by Oracle Scripting. Specifically, you can perform the following:

- View and delete deployed scripts.

- View, upload, update, and remove custom Java archives in support of Scripting operations.

- Set and remove the Global property for uploaded JAR files. The global property enables a specified set of code to automatically be loaded and available to all active scripts.

- Map Java archives to specified scripts to enable that code to be explicitly loaded and available to the script.

This console is accessed by logging into Oracle HTML-based applications using a user account with the Scripting Administrator responsibility.

2.4.3.3 Oracle Scripting Agent Interface Reports

From the Reports tab, you can generate and view panel footprint reports for a specified script. This report is instrumental in tuning a script, and indicates for each session or interaction of a script which panels were visited and the duration of the visit (in milliseconds). This is currently the only agent interface report for Oracle Scripting.

2.4.4 Survey Administration Console Features

The survey component of Scripting was previously known as iSurvey.

The Survey Administration console is an HTML administration user interface for survey campaign administrators and manager. This console integrates functionality from various Oracle applications for the benefit of administering and monitoring survey campaigns. Oracle Scripting, Oracle One-to-One Fulfillment, and Oracle Marketing functionality are all accessible from this single user interface. The Survey Administration console has the following primary functions:

1. To administer survey campaigns, cycles, and deployments in support of a survey campaign. This is performed from the Survey Campaigns tab of the Survey Administration console. Additionally, you can view responses received from existing survey campaigns by viewing deployment detail in the response view.

2. To administer survey resources, which are JSP files that appear as header or footer sections, error pages, or final pages for scripts executed in a Web browser. This is performed from the Survey Resources tab of the Survey Administration console.

3. To perform list management supporting targeted survey deployments. This accesses Oracle Marketing functionality. This is performed from the Audience tab of the Survey Administration console.

4. To perform management of Oracle One-to-One Fulfillment master documents, queries, and templates, which provide the ability to send through e-mail invitations and reminders for targeted survey deployments. This is performed from the Invitations tab of the Survey Administration console.

See Also

- Section 2.4.1, "Script Author Features"
- Section 2.4.2, "Scripting Engine Features"
- Section 2.4.3, "Scripting Administration Console Features"
- Section 2.4.5, "Accessibility"

2.4.5 Accessibility

It is our goal to make Oracle products, services, and supporting documentation accessible to the disabled community. Oracle Scripting Release 11.5.9 supports accessibility features.

To make the best use of our accessibility features, Oracle Corporation recommends the following software configuration:

- Windows NT 4.0 (with Service Pack 6) or Windows 2000

- Sun Java Access Bridge 1.0.2

- JAWS 3.70.87

- Microsoft Internet Explorer 5.5

- JDK 1.3.1

For additional information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Documentation

Our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers.

User Interfaces

Oracle Applications user interfaces required for administration of Oracle Scripting are rendered in HTML, making them available to users of assistive technology.

Script Development

New with this release, the Script Wizard feature of the Script Author component extends script development capabilities to people who are blind or visually impaired. The Script Wizard provides an alternative to the script development tools using visual layout tools, and can be used in combination with appropriate assistive technology (AT) such as screen readers. For example, the JAWS screen reader works with Java applications. Other AT products may apply as appropriate.

See Also

- Section 2.4.1, "Script Author Features"

- Section 2.4.2, "Scripting Engine Features"

- Section 2.4.3, "Scripting Administration Console Features"

- Section 2.4.4, "Survey Administration Console Features"

2.5 Architecture

Oracle Scripting 11.5.6 and later features the Apache mid-tier architecture. This architecture improves scalability over the caching architecture introduced in release 11.5.4, and follows the Oracle Applications philosophy of a distributed architecture using the Apache Web server as the middle tier.

This section includes the following topics:

- Section 2.5.1, "Oracle Scripting and the Session Object"

- Section 2.5.2, "Apache Mid-Tier Architecture"

- Section 2.5.3, "Caching Architecture"

2.5.1 Oracle Scripting and the Session Object

Oracle Scripting is a stateful application. The application incorporates the concept of the script "transaction," and monitors the Scripting Engine user's state throughout the session.

A script transaction begins when a script is launched in the Scripting Engine (agent or Web interface), and completes when the script runs to completion. Completion is indicated by reaching a termination node on the root graph of a graphical script, or by reaching a path in a panel whose next destination is "End script." In the agent interface, clicking the Disconnect button results in directing the session to the termination node on the root graph; clicking the Suspend button, for the purposes of this discussion, can be considered the temporary completion of a script.

All script transactions require a valid Oracle Applications session.

Oracle Scripting and the Session Object

Session is the representation of state for a particular user. At script runtime, a Scripting Engine user is connected with the scripting servlet. For each individual executing a script, there is a session object running in the Java Virtual Machine (JVM) of the Apache JServ (for Apache mid-tier architecture implementations) or in the JVM of Oracle JInitiator (for caching architecture implementations).

Agents using the Scripting Engine agent interface maintain the session for the duration of their Oracle Applications session. Within one session, users can run one or more subsequent script transactions. Only one script can be executed at a time.

Monitoring State

Oracle Scripting is a stateful application, and relies strongly on the tracking of the user's state via the session object. Some examples of the session's function include:

- Loads the script from the applications database

- Processes branching logic and answers provided by the client

- Executes custom code

By tracking state for each user, Oracle Scripting tracks where in the flow of a script a user is and for how long (footprinting), provides for data collection associated with the transaction footprint, and provides the functionality of the Scripting blackboard.

Number of Transactions Per Session May Differ By Interface

Users of the Scripting Engine agent interface (typically interaction center agents) tend to run multiple script transactions in one session. The session is retained as long as the agent is logged into Oracle Applications.

Each Oracle Scripting transaction executed in the Scripting Engine agent interface is typically associated with one customer (whether the party is identified or not). The Oracle Scripting end user (usually an interaction center agent) initiates an Oracle Scripting transaction, typically in response to reaching a customer on the telephone (inbound or outbound calls). The agent then uses the script to control the flow of the conversation, and capture information required for the intended business purpose.

Oracle Corporation recommends Oracle Scripting in the agent interface to be used in combination with an integrated business application. Agents either specifically launch a script, or a designated script is launched by the integrated business application, to address the needs of a customer. The agent navigates through the script as required, and exits the script when the script business purpose is complete, ending the transaction. Each invocation of a script is a separate Oracle Scripting transaction.

Users of the Scripting Engine Web interface access scripts either as survey questionnaires, or as Web scripts. Executing scripts as a Web-based survey involves only a single Oracle Scripting transaction (and precludes multiple transactions). Web script use cases support multiple Oracle Scripting transactions if required. The number of Oracle interactions allowed, as well as the authentication scheme and method of initiating an Oracle Applications session, is governed by the use of a specific Java server page as determined from the hosting option selected on the Create Survey Deployment page of the Survey Administration console. For example:

- IESSVYMAIN.JSP is used when you select the Standalone option from the Hosting Options field when creating a survey deployment. Select this option for executing Web-based surveys.

- IESSVYMENUBASED.JSP is used when you select the Menu based option from the Hosting Options field when creating a survey deployment. Select this option for Web scripts.

For survey questionnaires, when the survey respondent navigates to a designated survey URL in a Web browser, an Oracle Applications session starts (typically using a guest username and password established for this purpose). The Oracle Scripting transaction begins, and the first panel of the script appears in the respondent's Web browser. The respondent answers the questions or otherwise navigates through the script. When the question in the final panel of the script has been answered, the final page survey resource is displayed, and the script transaction is completed. The user cannot navigate backwards in the Web browser after reaching the final page, because the transaction has ended. Thus, each survey questionnaire session will include only a single invocation of a script (a single Oracle Scripting transaction).

For Web scripts, a self-service Web application user logs into the appropriate application, beginning an Oracle Applications session. When a survey URL or link is accessed, the authentication used for the calling Web application is passed to the database, and the user is authenticated to begin an Oracle Scripting transaction, resulting in the appearance in the user's Web browser of the first panel in the script. Like a survey questionnaire, the Web script user answers the questions or otherwise navigates through the script. When the question in the final panel of the script has been answered, the final page survey resource is displayed, and the script transaction is completed. The user cannot navigate backwards in the Web browser after reaching the final page, because the transaction has ended. However, the self-service Web application (and the Oracle Applications session) are still valid. If supported by the business rules of the governing survey campaign, a second Web script can be launched from the Web application. Each invocation of a script is a separate transaction.

Post-Transaction Processing

When a script transaction ends, the Scripting Engine must execute several tasks. These include:

- Update record in IES_TRANSACTIONS table with script end time

- Write answers collected during transaction into appropriate table (IES_QUESTION_DATA), assuming answer collection is enabled for the script

- ■ Write footprinting data into appropriate table (IES_PANEL_DATA), assuming answer collection or footprinting is enabled for the script

End-of-script processing occurs in separate thread using a different database connection. This allows the next script transaction to be started (in the existing thread) before the data from the prior session is completely written to the database. The data from the previous transaction is concurrently written to the database using the newly instantiated thread. The net end user effect is the ability of agent interface users to start a new Scripting transaction much sooner.

2.5.2 Apache Mid-Tier Architecture

Oracle Scripting Runtime Architecture
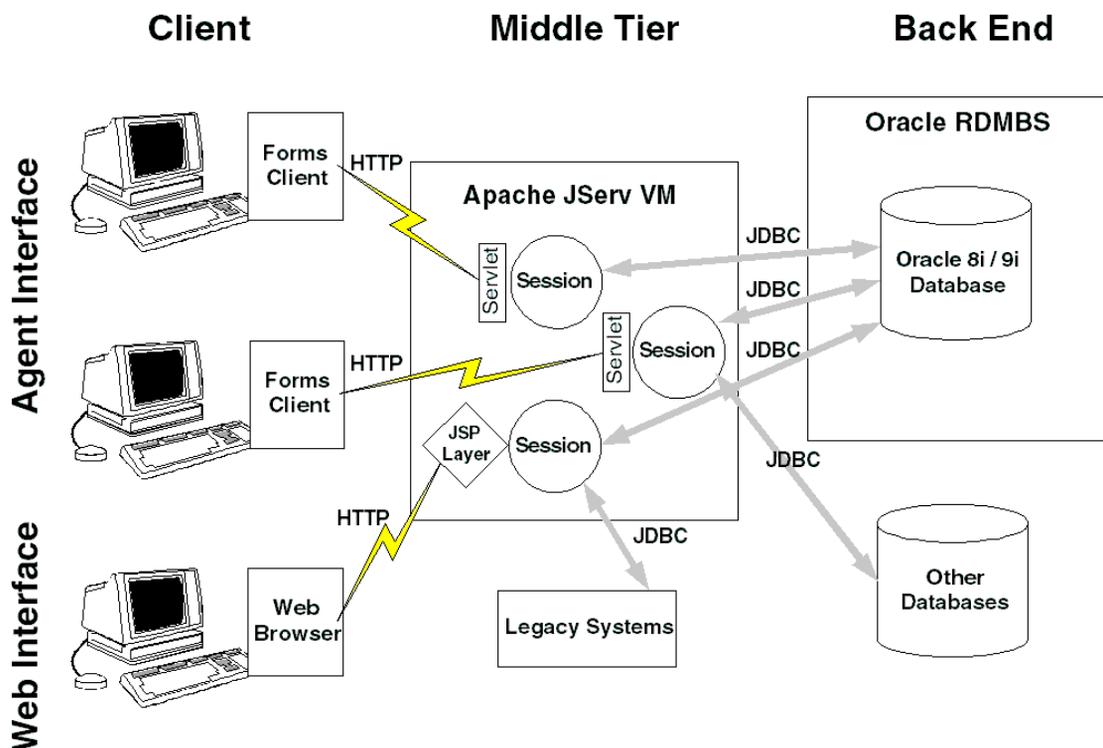
 Using the Apache mid-tier architecture, business logic is executed on the enterprise Web server. The Apache JServ provides the Java Virtual Machine (JVM), executing the script at runtime as a servlet. Scripts are executed by the Scripting Engine using two different interfaces.

All communications with the client (for either Scripting Engine interface) are through the hypertext transfer protocol. Scripts can be executed over a firewall with appropriately configured secure HTTP (HTTPS). From the middle tier, the session

itself communicates with the data tier (the Oracle Applications database, as well as any other database that may be used) using Java database connectivity (JDBC).

Agent Interface

Customer service or interaction center agents typically use Oracle Scripting through a Forms-based business application such as Oracle TeleSales, Oracle Collections, or Oracle TeleService. Regardless of how agents are using the Scripting Engine agent interface, when an agent requests a script (or launches an application that automatically requests a script), a session is started using HTTP between the Forms client (the agent workstation) and the Scripting servlet on the Apache server. Back-end communication is handled with JDBC.

Web Interface

Individuals using the Web interface of the Scripting Engine are executing a script either as a survey for from self-service applications such as Oracle iSupport. The script end user accesses a specified URL from a supported Web browser, sending the message via HTTP to launch a script associated with a particular set of requirements. These requirements are defined in the survey campaign, and include cycle and deployment parameters. From a Web interface perspective, the Scripting Engine is a set of Java classes on the middle tier that receives requests from the JSP layer (including the message to launch a script), and processes them. All business logic as determined by the survey questionnaire script, custom code or database calls, and respondent actions are evaluated by the Scripting Engine, which passes processing information to the JSP layer to interpret, which in turn sends information to the Web browser to display.

Configuration File Customization No Longer Required

Using the latest functionality supporting this architecture, custom Java code can be deployed to the Oracle Applications database using the Scripting Administration console. This precludes the need to manually deploy custom code to the applications server, and eliminates the need to specify the custom Java code's class path in a configuration file. Code loaded from the Scripting Administration console can be made available globally to all scripts, or can be mapped to specific scripts. Either way, the Scripting Engine obtains the appropriate code from the database dynamically using JDBC so the code is available to the script at runtime.

Using the Apache mid-tier architecture, appropriately compiled and packaged custom Java code that was manually deployed to the applications server should now be deployed to the database using the Scripting Administration console and

mapped accordingly. No customization or updates to the JSERV.PROPERTIES file will be required thereafter.

Additionally, individuals upgrading to the Apache mid-tier architecture from the caching architecture will no longer need to customize or update their APPSWEB.CFG files to reference custom Java. All Java archive code loaded to the database using the Scripting Administration console will be made available to the Scripting Engine based on its script mapping or global status.

2.5.3 Caching Architecture

Oracle Corporation also provides limited support for the caching architecture. The caching architecture is not supported for new users of Oracle Scripting. Only users upgrading from previous releases of Oracle Scripting and who are already using the caching architecture will continue to be supported in this architecture by Oracle Support Services. Oracle Corporation recommends that caching architecture users upgrade to the Apache mid-tier architecture.

Using the caching architecture, business logic is executed using the Java Virtual Machine (JVM) provided on the client by Oracle JInitiator. Scalability over the previous architecture is much improved, at the expense of executing the JVM on client workstations (fat client model). This may necessitate increased processing power on agent workstations and is inconsistent with the Oracle Applications release 11*i* philosophy. Use of the caching architecture also requires customization of APPSWEB.CFG for specification of the class path for any custom Java to be used. Corresponding Java archives must then be deployed to the applications server in the same class path referenced in the configuration file, typically in a directory such as <JAVA_TOP>/ies_custom. For more information, see Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File".

2.6 Responsibilities

Various responsibilities may be required for using Oracle Scripting, based on which component you are using and which integrated applications you want to use with Scripting. Additionally, many more responsibilities are required for implementing this product. For more detailed information on the various users required to implement and use Oracle Scripting, see Section 4.7.1.4, "Implementation Responsibility Matrix".

This section includes the following topics:

- Section 2.6.1, "Scripting Responsibilities"

- Section 2.6.2, "Survey Responsibilities"

2.6.1 Scripting Responsibilities

User Type	Responsibility	Function	For Component(s)
Agent	Scripting User Scripting Agent	Launch scripts in "stand-alone mode" from Oracle Forms, typically to test deployed scripts.	Scripting Engine for agent interface
Agent	Customer Support	For agents accessing Oracle Scripting from within the integrated Oracle TeleService application.	Scripting Engine for agent application
Agent	TeleSales Agent	For agents accessing Oracle Scripting from within the integrated Oracle TeleSales application.	Scripting Engine for agent application
Agent	Collections Agent	For agents accessing Oracle Scripting from within the integrated Oracle Collections application.	Scripting Engine for agent application
Administrator	Scripting Administrator	Access the Scripting Administration console. This is required to: ■ Launch the Script Author as a Java applet, accessible from the Home tab. ■ Administer Oracle Scripting files, including viewing and removing deployed scripts, viewing, uploading, updating or removing custom Java archive files. ■ Generate, view and analyze the Panel Footprint agent interface report.	Scripting Administration console
Administrator	Interaction Center Server Log Viewer, Interaction Center Server Manager, Call Center HTML Administration, or CRM HTML Administration	To access message logs for access to server logs and informational messages. Based on which responsibility is used, this may appear under various tabs. Messages tab for access to server logs and informational messages.	Scripting Engine for agent interface or Web interface

2.6.2 Survey Responsibilities

User Type	Responsibility	Function	For Component(s)
Administrator	Survey Administrator	Access the Survey Administration console. This is required to: ■ Define survey resources (headers, final pages, error pages in JSP format) ■ Create survey campaigns and cycles. ■ Define or activate survey deployments ■ View specific responses provided by survey respondents ■ Administer list information for targeted deployments ■ Administer invitation or reminder master documents, queries and templates for targeted deployments	Survey Administration console
Administrator	iSurvey User	Schedule or run concurrent programs.	Survey Administration console
Administrator	Fulfillment Administrator	■ Access the Fulfillment Administration console. ■ Set up or change fulfillment groups ■ Set up or change mail servers associated with fulfillment groups ■ View failed fulfillment requests after concurrent programs succeed in passing requests to the fulfillment server.	Survey Administration console These Fulfillment Administrator tasks are required for implementations using targeted survey deployments

User Type	Responsibility	Function	For Component(s)
Administrator	Oracle Marketing Super User	■ Access the Marketing Administration console. ■ Establish campaigns, assign agents to campaigns, assign scripts to agents or scripts to campaigns ■ Create, import, modify lists	Scripting Engine for agent application If Oracle TeleSales or Oracle Collections agents must launch a specific script, the script must be associated with a campaign or campaign schedule. Survey For targeted survey deployments, lists must be created and administered.
Administrator	Interaction Center Server Log Viewer, Interaction Center Server Manager, Call Center HTML Administration, or CRM HTML Administration	To access message logs for access to server logs and informational messages. Based on which responsibility is used, this may appear under various tabs. Messages tab for access to server logs and informational messages.	Scripting Engine for agent interface or Web interface

2.7 Script Author Concepts

This section includes the following topics:

- Scripts
- Global Script Properties
- Oracle Scripting Users
- Custom Code
- Using Custom Java

See Also

- Scripting Engine Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

2.7.1 Scripts

A script is a miniature program built using the Script Author by trained functional users known as script developers.

A script facilitates the flow of information based on customized business requirements. Each script enforces business rules programmed into it by script developers.

At runtime, scripts can display text, prompts, and images to its end users. The information displayed, and the order in which that information is presented, is based on:

1. A predetermined set of requirements (business rules, custom code, and specific scripted text) that have been programmed into a script.

2. Responses provided by the end user. Responses come in the form of mouse clicks or keyboard actions.

Script developers can create two types of script from the Script Author. Using the standard graphical tools, script developers create graphical scripts by selecting Script Author objects from one of two toolbars in the user interface, and placing the selected object on the graphical workspace known as the canvas. The developer then associates properties to each object. Objects are connected to each other, similar to a flow chart. Flow of the script at runtime is determined in part by the choice of branch used to connect objects. For graphical scripts, flow can also be determined dynamically for each script runtime session, using commands that evaluate data received into the script.

Decision-enabling data for graphical scripts includes previous end user responses for the current session, information received from external database tables or applications through forms commands, blackboard commands or PL/SQL commands, and events or conditions such as the time of day.

Using the Script Wizard, script developers can provide script information in a sequence of windows called a wizard. The resulting wizard script can be saved when syntactically correct, and can be deployed to the database from the Script Wizard user interface.

Decision-enabling data for wizard scripts includes previous end user responses for the current session, and validation of text responses provided if answer validation was specified by the script developer.

See Also

■ Global Script Properties

- Oracle Scripting Users
- Custom Code
- Using Custom Java

2.7.2 Global Script Properties

Global script properties are those properties that apply to an entire script. This is in contrast to properties of a specific object in a script such as a panel, group, block, or branch. Global script properties are assigned default values when a script is created.

The first property, Script Type, is determined when you select Graphical script or Wizard script in the Script Author (**File** > **New**). This property is not modifiable.

From the Script Author visual layout, you can view and update all modifiable global script properties from the File menu (**File** > **Script Properties**) or by right-clicking on an empty portion of the canvas (**Edit** > **Edit Blob Properties**).

From the Script Wizard, you can view and update all modifiable global script properties in the first wizard window (Define Script Properties).

For more information, including a table of global properties and a description of each, see *Oracle Scripting User Guide* > Understanding the Script Author > Understanding Global Script Properties.

See Also

- Scripts
- Oracle Scripting Users
- Custom Code
- Using Custom Java

2.7.3 Oracle Scripting Users

This section includes the following topics:

- Script Author Users
- Scripting Administrative Users
- Scripting Engine Users
- Survey Administrative Users

See Also

- Scripts

- Global Script Properties

- Custom Code

- Using Custom Java

2.7.3.1 Script Author Users

The Script Author development environment is the component of Oracle Scripting which provides the sole means of creating scripts for execution in any Oracle Scripting runtime Scripting Engine interface.

As of Oracle Scripting release 11.5.9 or later (or Interaction Center Family Pack P or later), the Script Author is a Java applet accessed through Oracle Applications by a user with the Script Administrator responsibility. In previous versions, the Script Author was a stand-alone Windows Java application that required separate implementation, installation, and setup. Regardless, users that access the Script Author are referred to as script developers.

The Script Author provides a graphical user interface intended for the functional user with some technical knowledge.

The graphical development environment provides for reuse of defined commands and existing Scripting components. The script developer will define these commands. Hooks in the Script Author UI reference technical components (for example, Oracle Forms, custom Java methods, and PL/SQL packages stored in the database) which are primarily developed externally. These components provide sophisticated functions to accomplish the functionality most enterprises want. They must be developed by individuals certified and knowledgeable in the relevant technologies. The script developer must work with these highly technical resources to ensure the custom components are appropriately integrated into the script. The script developer must also ensure the code is appropriately loaded in the database or applications server (based on approach) and properly referenced in the script. This will ensure the code is available to the base Java classes that provide Scripting runtime functionality for the Scripting Engine agent or Web interface at runtime.

As of Oracle Scripting release 11.5.9 or later (or Interaction Center Family Pack Q or later), the Script Author includes a Script Wizard component accessed through the menu or tool bar from the Script Author. With this new feature, less technical users can quickly and easily create simple scripts or surveys by providing script information in a series of windows known as a wizard.

Script Wizard users have limited access to the more technical features available in the graphical development environment. This provides the opportunity to divide script development amongst, for example, business process engineers with full knowledge of the business and flow requirements of a script, after which they can turn the script over to more experienced developers to add hooks, commands, reference Java or Forms, and so forth.

Users of the Script Author include campaign administrators, Java developers or database programmers, business process engineers, and experienced interaction center agents with technical aptitude. The keys to successful script development are: (1) familiarity with how Oracle Scripting captures and processes data, (2) knowledge of associated technologies (including access to experts in these technologies), and (3) adequate training and familiarity with existing documentation.

Using the stand-alone Script Author Java application, no responsibility was required to launch the Script Author on a Windows client. For current and future releases, only the Script Author Java applet, accessed through a validated Oracle Applications session, is supported.

Since script developers can deploy, delete, and otherwise affect scripts in production, and can manipulate information in the applications or other enterprise database, Oracle Corporation strongly recommends that only trusted users be provided with the Scripting Administrator responsibility. For more information, see Section 4.2, "Creating an Administrator for Oracle Scripting".

See Also

- Scripting Administrative Users
- Scripting Engine Users
- Survey Administrative Users

2.7.3.2 Scripting Administrative Users

Users of the JSP/HTML-based Scripting Administration console include script developers (see Script Author Users), who launch the Script Author applet from the Home tab, and administer deployed scripts and custom Java archive files from the Administration tab. Interaction center campaign administrators or system administrators (as well as script developers) will also typically access this console to run panel footprint reports to help tune a script's structure, increase agent performance and reduce average talk time. To access the Scripting Administration console user interface from Oracle HTML-based applications, these users must have the Scripting Administrator responsibility.

See Also

- Script Author Users

- Scripting Engine Users

- Survey Administrative Users

2.7.3.3 Scripting Engine Users

End users of the Scripting Engine agent interface are trained interaction center agents ("agents") or customer service representatives. These are non-technical users who have received simple but thorough training in the Java-based Scripting Engine interface. These individuals include call center agents taking or presenting information over the telephone, as well as interaction center agents taking advantage of other media such as intranets, enterprise portals over the World-Wide Web, and so forth.

Scripting Engine agent interface users typically launch scripts from a business application such as Oracle TeleSales, Oracle Collections, or the Customer Support component of Oracle TeleService. For testing purposes, the agent interface can also be launched in "stand-alone" mode. Agent interface users must have access to the appropriate Oracle Applications responsibility to launch the integrated business application from which Oracle Scripting is integrated.

End users of the Scripting Engine Web interface include users of targeted (list-based) or standard (non-list-based) survey campaign deployments, or self-service Web application users. Survey respondents may have been invited to participate in a survey through an e-mail message or through navigation to a survey-enabled site. Self-service Web application users access a script or survey in a Web browser through a self-service Web application scenario such as Oracle iSupport.

To access a survey using a self-service application, the appropriate responsibility to access that application is required. No Oracle Applications responsibilities are required of the end user to execute a script as a standard or targeted survey deployment. The user simply accesses the given survey URL in any Oracle Applications 11*i*-certified Web browser.

See Also

- Script Author Users

- Scripting Administrative Users

- Survey Administrative Users

2.7.3.4 Survey Administrative Users

Users of the JSP/HTML-based survey campaign administrative console are non-technical users with access to detailed project requirements. These individuals are typically interaction center survey campaign administrators or system administrators.

To access the Survey Campaign administrative interface from Oracle Personal Homepage (PHP) login, these users must have the Survey Administrator responsibility.

See Also

- Script Author Users

- Scripting Administrative Users

- Scripting Engine Users

2.7.4 Custom Code

Custom code is used to obtain information needed during the execution of the script, or to enforce specific business rules. Custom code supported by Oracle Scripting includes Java commands, PL/SQL commands, SQL calls to the database, Scripting blackboard commands, forms commands, and constant commands.

Interpretation of Custom Code

Custom code is referenced in the script by associating commands to objects in the script (or to the script itself) before it is deployed to the database from the Script Author. Based on the type of command, the code may be stored in the meta-data of the script itself (e.g., constant commands or PL/SQL commands), in the Scripting session (blackboard commands), on the applications server (custom Java commands exposed to the application by defining the class path), or in the database (custom Java commands deployed to the database from the Scripting Administration console, and PL/SQL packages stored in the database).

See Also

- Scripts

- Global Script Properties

- Oracle Scripting Users

- Using Custom Java

2.7.5 Using Custom Java

Custom Java can be used with Oracle Scripting in two ways: executing commands at runtime, and replacing agent interface panels with Java beans.

Executing Script Author Commands

Scripts that reference custom Java methods associated with a Script Author command can execute each method as specified by the command at runtime.

This applies to all scripts executed in the Scripting Engine, using either the agent interface or the Web interface.

Replacing Panels with User Interface Java Beans

Scripts can also use custom Java beans to replace an entire panel in the runtime session on the agent client workstation. This provides agent users with extended functionality as customized in the Java bean. This bean is executed by Oracle JInitiator on the client.

This applies only to scripts executed in the Scripting Engine agent interface.

> **Note:** Replacing a single answer in a panel with a user interface Java bean is no longer supported functionality.

Java Compilation and Oracle JInitiator Dependencies

> **Note:** This section includes information about Java Development Kit (JDK) and Java Runtime Engine (JRE) compatibility with Oracle JInitiator and with Oracle Applications. The information provided includes version numbers certified at the time of publication. Certification and compatibility information frequently changes. For the latest information, consult Oracle*MetaLink* or Oracle iSupport.

When using custom Java in support of Oracle Scripting, there are compilation dependencies based on specific circumstances. These are based primarily on the purpose of the Java archive in question (to provide Script Author commands or to replace panels with Java beans), and include the following environmental factors:

- The level of Java Runtime Environment (JRE) used on the Apache Web server for your Oracle Applications environment

- The level of Java Development Kit (JDK) used to compile your custom Java code

- The Oracle JInitiator version used on the client

- The Oracle Scripting architecture type used at the enterprise.

Custom Java for use as a Script Author command is executed by the Java Virtual Machine in the Scripting session. Using the Apache mid-tier architecture of Oracle Scripting, custom Java code is executed on the Apache Web server. Thus, this code must be compiled using a version of JDK that is compatible with the JRE used on the Apache Web server (this must be the same level or lower). At this time, appropriate JDK versions may include JDK 1.3, 1.2, or 1.1.8.

Using the caching architecture of Oracle Scripting, custom Java code is executed by Oracle JInitiator on the client workstation (for the agent interface). Thus, this code must be compiled using a version of JDK that is compatible with Oracle JInitiator on the agent client (this must be the same level or lower). At this time, appropriate versions may include 1.3 or 1.1.8.

All scripts executed as surveys using the Scripting Engine Web interface use the JVM of the Apache Web server. Thus, this code must be compiled using a version of JDK that is compatible with the JRE used on the Apache Web server (this must be the same level or lower). At this time, appropriate JDK versions may include JDK 1.3, 1.2, or 1.1.8.

Custom Java that replaces a script panel with a user interface Java bean at runtime executes on the agent client. Thus, this code must be compiled using a version of JDK that is compatible with Oracle JInitiator on the agent client (this must be the same level or lower). At this time, appropriate versions may include 1.3 or 1.1.8.

Java Archive File Format Requirements

The source code for Java methods or Java beans must be compiled into executable class files and packaged into Java archives in one of two file formats, as described in the table below:

Format	Description	Example
JAR	Java Archive (JAR)	SOURCE.JAR
ZIP	WinZip archive	SOURCE.ZIP

> **Note:** Oracle Corporation recommends using JAR file formats, although both JAR and ZIP file formats are supported. For the purposes of this document, the term "Java archive" applies to both file formats, assuming the archive contains appropriately compiled and packaged code.

> **Note:** When the Scripting Administration console references Jar files or Jars, appropriately packaged ZIP files are included in this definition.

See Also

- Scripts
- Global Script Properties
- Oracle Scripting Users
- Custom Code

2.8 Scripting Engine Concepts

This section includes the following topics:

- Scripting Engine Function
- Scripting Engine Agent Interface
- Scripting Engine Web Interface
- Footprinting and Answer Collection

See Also

- Script Author Concepts
- Oracle Scripting Administration Concepts
- Oracle Scripting Survey Concepts

2.8.1 Scripting Engine Function

The Scripting Engine process a script in runtime, including displaying the text, images, questions and prompts in the script, interpreting the flow based on end user responses and custom code.

Scripts developed with Script Author can be executed in one of two Scripting Engine interfaces: the agent interface, a Java application running in an Oracle form, and the Web interface, a sequential interpretation of a script with each panel represented by one JSP page.

See Also

- Scripting Engine Agent Interface

- Scripting Engine Web Interface

- Footprinting and Answer Collection

2.8.2 Scripting Engine Agent Interface

The agent interface is used by customer service or interaction center agents, typically from within Oracle TeleSales, Oracle Collections, or the Customer Support module of Oracle TeleService. Scripts can also be run in the agent interface in "stand-alone" mode (still from an active Oracle Applications session). However, this is recommended primarily for script testing.

In addition to rendering panels that may contain text and images and that must contain at least one "question" or answer control per panel, this user interface involves a progress panel that displays the flow of the current script and answers provided for the active session; a programmable script information area; a programmable button bar that can display script navigation shortcuts; a status bar; and a programmable Disconnect button.

See Also

- Scripting Engine Function

- Scripting Engine Web Interface

- Footprinting and Answer Collection

2.8.3 Scripting Engine Web Interface

The Web interface supports obtaining survey data, feedback or opinions from customers who respond to a survey questionnaire. The survey questionnaire is a

Script Author script executed as a series of Java server pages in a Web browser. This user interface includes only panel display. There is no progress area, script information area, button bar, or Disconnect button. End users can use the browser's Back and Forward controls to aid with navigation.

Executing a script in the Web interface first requires administration, in the form of establishing a set of survey campaign requirements using the Survey Administration console. These requirements include creating a survey campaign and defining an associated cycle and deployment, identifying JSP components known as survey resources, and identifying a deployed script as the questionnaire.

- Survey data can be solicited by sending out e-mail invitations and reminders, leveraging Oracle Marketing's list management capabilities and Oracle One-to-One Fulfillment's e-mail template, data field merging, and delivery capabilities. These are known as targeted or list-based survey deployments.

- Survey data can also be solicited by links on an enterprise Web site.

- Finally, survey data can be solicited by customized links on self-service Oracle Application user interfaces such as Oracle iSupport.

Each of these models relies on a survey URL that defines the Oracle Applications instance, and identifies (at minimum) survey campaign and deployment identification code (dID) parameters. Other potential parameters in the URL include a respondent identification code (rID, which associates an individual with a list record), and any custom parameters to pass information into the Scripting session for evaluation and use during the interaction. Accessing this URL launches the script in the Scripting Engine Web interface, regardless of end user entry.

The Scripting Engine interprets the respondent's responses, the script, and custom code, passing metadata and instructions to the Web browser to be interpreted on the respondent's client computer.

See Also

- Scripting Engine Function

- Scripting Engine Agent Interface

- Footprinting and Answer Collection

2.8.4 Footprinting and Answer Collection

Footprinting is the recording in the Oracle Applications database of which panels in a script transaction were visited by a script end user, in sequence, and the duration

of time (in milliseconds) before the next panel is requested. Footprint data is stored in the IES_PANEL_DATA table in the Oracle Applications schema.

Answer collection is the recording in the Oracle Applications database of end user responses ("answers") to all answer controls ("questions") that are marked in the script as collectable. Answer collection data is stored in the IES_QUESTION_DATA table in the Oracle Applications schema.

Footprinting and answer collection are both global script properties. For any given script, these features are either on or off. These options can be viewed and set in the Script Author by selecting **File > Script Properties**.

If enabled, footprinting data and answers are collected for each transaction or session of the script running in the Scripting Engine, in either interface.

In order for footprint data to be collected, either the Footprinting option or the Answer Collection option must be selected at the global script level. Otherwise, regardless of which specific answers are marked as collectable, information for each script end user will be discarded at the end of each script session.

Footprinting and Answer Collection Dependencies

- Answers are only collected for questions designated as collectable. The Collectable option is a boolean property of a Script Author question (answer definition), represented by a checkbox. *This option is selected by default for all questions defined in the Script Author.* You can modify any single question so that answers provided at runtime are not collected, by clearing the Collectable? option selection in the data dictionary for that answer definition.

- Answers are only collected for scripts for which the Answer Collection option is selected. To prevent answers designated as collectable from being collected for all questions in a script, you can clear the Answer Collection option at the global script level. Doing so will prevent survey summary reporting capabilities for scripts with answer collection disabled.

- Answer collection requires footprinting. Therefore, regardless of whether the Footprinting option is explicitly selected, footprinting data will be collected for any script with the answer collection option selected.

  If the Footprinting option is selected but the Answer Collection option is not, only footprinting data for each session or transaction of that script will be saved to IES_PANEL_DATA. If neither option is selected for a specific script, no footprinting or answer collection data is saved.

- Saving footprint data whenever answer collection is enabled ensures, for each script session, a link between each response provided at runtime, and between

the specific panel instance from which that response was provided. In the answer collection table (IES_QUESTION_DATA), column PANEL_DATA_ID) contains the foreign key reference to IES_PANEL_DATA (the footprinting table).

References

Section 2.2.7, "Answer Collection Property Now Forces Footprinting Data Collection"

See Also

This section includes the following topics:

- Scripting Engine Function
- Scripting Engine Agent Interface
- Scripting Engine Web Interface

2.9 Oracle Scripting Administration Concepts

Oracle Scripting provides the ability to create, modify, and deploy scripts (using the Script Author component) that can be executed in the Scripting Engine component. The Scripting Engine has two interfaces (the agent interface and the Web interface), both of which display the script at runtime for their intended audience. Each runtime interface interprets the script, end user input, and any custom code associated with the script. If using the Web interface, you must use the Survey component to create and administer guidelines for the script to be executed within a Web browser.

Scripts are deployed to the applications database using the Script Author. Scripts may rely on custom Java code, compiled and deployed as Java archive (JAR) or zipped archive (ZIP) files and referenced in the script. Scripts can also reference PL/SQL procedures stored in the applications database.

In support of Scripting operations, you can use the Scripting Administration console to access the Script Author as a Java applet; administer scripts and script Java archive files; and monitor Scripting Engine agent interface reports.

This section includes the following topics:

- Scripting Administration Console
- Scripting Administration Console View List

- Agent Interface Reports

See Also

- Script Author Concepts

- Scripting Engine Concepts

- Oracle Scripting Survey Concepts

2.9.1 Scripting Administration Console

The Scripting Administration console provides script administrators the interface to launch the Script Author as a Java applet, to view and delete deployed scripts, to view and administer Java archive files, to map Java archive files to specific scripts, and to view agent interface reports. This console is accessed by logging into Oracle HTML-based applications using a user account with the Scripting Administrator responsibility.

2.9.2 Scripting Administration Console View List

When you view any page displaying a summary list in the Scripting Administration console, the set of records which displays in the list is filtered by the parameter selected in the View list. By default, only items created by the Oracle Applications user account with which you are currently logged in display in each summary list. To view items created by all users, change the value in the View list. When you select a filter option from the View list, the page refreshes. The summary view list displays, listing only the objects that meet the selected criteria.

If displaying a list of Java archive files on the Jar Listings or Jar Mapping pages, for example, the value **My Jars** is the default, resulting in the display of all JAR and ZIP files uploaded using the Scripting Administration console with your user name. To display a list of *all* Java archive files deployed from the Scripting Administration console (including those uploaded by other users in this environment), select **All Jars** from the View list.

If using the View menu to display a list of deployed scripts, additional filtering criteria is available based on a script's active or inactive status. Scripts are deployed to the IES_DEPLOYED_SCRIPTS table of the applications database from the Script Author. Active scripts, which can be executed by any Scripting Engine of a compatible code level, contain a value of "1" for the ACTIVE_STATUS field within that table. Inactive scripts contain a value of "0" in this table field. Deployed scripts with ACTIVE_STATUS set to "0" are retained in IES_DEPLOYED_SCRIPTS so that existing footprinting and answer collection data can maintain valid references.

If displaying a list of deployed scripts on the Deployed Scripts page, the value **My Active Scripts** is the default, resulting in the display of all active scripts created with your user name. To display a list of all scripts created with your user name regardless of active status, select **My Scripts**. To display a list of all active scripts created with any user name, select **All Active Scripts**. To display a list of all scripts created with any user name, regardless of active status, select **All Scripts**.

Note that if no objects meeting the filter parameter for a certain category are found, then the list headings will appear for the summary table, with no records listed. As soon as an object is created meeting that criteria, it will appear in a refreshed list.

2.9.3 Agent Interface Reports

The ability to report and analyze scripts executed in the Scripting Engine agent interface is critical; enterprises can use information collected in the applications database for various purposes. Reports from information collected in scripting-specific tables of the applications database as a result of Scripting operations, and other data collected from customized scripts and stored in custom tables, can be generated using any analytical tool such as Oracle Discoverer or Crystal Reports. Additionally, the Scripting Administration console provides access to panel footprint reports compiling footprinting data.

The panel footprint report, and additional reports, are also available from the Survey Administration console. These additional reports require the use of Scripting-specific concurrent processes and summary tables specific to survey operations. For additional information, see Oracle Scripting Survey Concepts.

This section includes the following topics:

- Reports and Data

- Analysis and Tuning with the Panel Footprint Summary Report

- Required Report Parameters

2.9.3.1 Reports and Data

At this time, the only report available through the Scripting Administration console is a panel footprinting summary report. There are two requirements for receiving and reporting data in the Scripting Administration console:

1. In order to appropriately view reports, two script-level parameters should be enabled. These parameters, Footprinting and Data Collection, are established in the global script properties from the Script Author prior to deploying a script.

> **Note:** Technically, footprinting is enabled when the data collection option is selected. However, Oracle Corporation recommends explicitly enabling the Footprinting option if footprinting information is desired for the purposes of reporting.

2. To generate a meaningful report, data to be displayed in the report must already be generated. Therefore, scripts must be executed in order to tabulate data displayed in the report. Each time a script is executed (assuming the appropriate parameters are enabled), data regarding the paths taken in the script (footprinting) and the answers selected during the script session (answer collection) are collected in IES tables in the Oracle Applications database.

   For a panel footprint report, this signifies a script with footprinting specified has been executed at least once, either in the interaction center interface by an agent running through a script, or in the Web interface as a respondent participates in a survey using a Web browser.

2.9.3.2 Analysis and Tuning with the Panel Footprint Summary Report

The panel footprint report may be used either for analysis of surveys, or of scripts in use in the interaction center.

Cost Savings in the Interaction Center

This report is overtly useful to enterprises using scripts in the interaction center, as footprint analysis can lead directly to reducing average talk time for an interaction center agent. Doing so results in measurable reduction in costs and increased agent efficiency.

Disabling Footprinting

Footprinting (the act of recording which panels in a script were visited and for how long) can provide useful script tuning data. However, it also consumes system resources. If an enterprise does not need to view individual responses or generate reports, then footprinting should be disabled at the script level to conserve system resources. Note that even if footprinting is not specifically enabled, but data collection is enabled at the script level, then footprinting data will be collected in order to maintain the integrity of data collected.

2.9.3.3 Required Report Parameters

The reports take the respective parameters listed below.

2.10 Oracle Scripting Survey Concepts

The survey component of Oracle Scripting allows enterprises to use Script Author scripts as Web-based survey questionnaires that can be executed in a Web browser (over the Internet or on an intranet). The survey component of Oracle Scripting was previously known as iSurvey.

A survey respondent participates in a survey questionnaire by accessing a specific survey deployment URL either from an enterprise's Web site, from a self-service Oracle Application such as Oracle iSupport, or from an invitation e-mail message inviting survey participation. Using the survey component of Oracle Scripting, enterprises can create, manage, and report on surveys to evaluate customer satisfaction, gain customer input on new initiatives, and gain other feedback from survey respondents. The return data can then be used to improve product lines, target new or improved services, or otherwise improve responsiveness.

The collection of requirements for conducting such an effort is referred to as a survey campaign. Survey campaigns are set up and managed from the Survey Administration console.

This section includes the following topics:

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes

See Also

- Script Author Concepts
- Scripting Engine Concepts
- Oracle Scripting Administration Concepts

2.10.1 Survey Administration Console

In order to execute a Script Author script in a Web browser using the Scripting Engine Web interface, survey administrators must define survey campaign requirements in an HTML administration console. This is true regardless of whether the script will be executed as a standard or targeted survey, or from a self-service Web application.

The Survey Administration console is an HTML user interface accessed from the Oracle Applications Personal Homepage (PHP) login by a user with the Survey Administrator responsibility. In this interface, a survey campaign is created and its dependencies (JSP resources, Script Author script designated as the questionnaire, a survey campaign, cycle, and deployment) are defined.

Additionally, survey administrators can view individual results from a survey (ongoing or completed) from the response view of the Deployment Details page. Answers provided by respondents are stored in the Oracle Applications database schema.

Essentially, the Survey Administration console provides survey administrators the means to set up, execute, and monitor survey campaigns.

See Also

- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes

2.10.2 The Survey Questionnaire

The survey questionnaire is a Script Author script created to obtain specific information. End users of the Survey component are customers or prospects viewing a survey questionnaire using any Oracle Applications 11*i*-certified Web browser, or individuals executing a Web script from an Oracle self-service

application such as Oracle iSupport. As the individual participating in a survey ("survey respondent") or using the Web script moves through each HTML page (corresponding to a single script panel in the Script Author), the objects embedded in the script control complicated processing.

On the end user's desktop, the user is guided through the script either through a rigidly prescribed order, or through a dynamically determined path (as determined by the custom script built based on the needs of the enterprise). The survey questionnaire script displays as a series of Java Server Pages (JSP) files in an HTML-based interface (the Web browser).

On the Apache Web server associated with the Oracle Applications instance for the enterprise conducting the survey or hosting the Web script, all business logic is executed dynamically. This includes the business rules embedded in the survey questionnaire script (such as rules-based branching, data integration, and commands associated with specific objects and events), the end user's answers, and any custom Java or PL/SQL code.

See Also

- Survey Administration Console
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Deployments
- Survey Resources
- Prototypes

2.10.3 Survey Administration Console View List

When you view survey campaigns or survey resources in the Survey Administration console, the set of records which displays is filtered by the value in the View list. By default, only items created by the logged in user displays. To view items created by all users, change the value in the View list accordingly and click **Go**.

For example, when viewing survey resources, the default selection in the View list is "My Survey Resources." Correspondingly, the records that display on the page (if any) consist only of survey resources defined in the Survey Administration console by the Oracle Applications user account with which you are currently logged in. To display all survey resources (including those created by other users in this environment), select **All Survey Resources** from the View list and click **Go**. The Survey Resources page refreshes. The summary view table now lists survey resources (if any) created by all users for this environment. This is applicable to survey campaigns as well.

Note that if no objects of that category have been created, then the list headings will appear, with no entries. As soon as you create an object, it will appear in the refreshed list.

See Also

- Survey Administration Console

- The Survey Questionnaire

- Survey Reports

- Survey Hierarchy, Levels and Objects

- Survey Campaigns

- Cycles

- Deployments

- Survey Resources

- Prototypes

2.10.4 Survey Reports

After summarizing survey data for optimum performance using Concurrent Manager, reports on scripts executed in a Web browser can also be generated using Oracle Business Intelligence.

Custom reports can also be generated from tables in the Oracle Applications schema as desired, using tools such as Oracle Discoverer.

See Also

- Survey Administration Console

- The Survey Questionnaire

- Survey Administration Console View List

- Survey Hierarchy, Levels and Objects

- Survey Campaigns

- Cycles

- Deployments

- Survey Resources

- Prototypes

2.10.5 Survey Hierarchy, Levels and Objects

The Survey component of Oracle Scripting includes three hierarchical levels: survey campaigns, cycles, and deployments. Each level is a set of requirements to ultimately execute survey campaigns.

The collection of requirements for each level is stored as an object in the database. For this reason, survey campaigns, cycles and deployments are described as objects. Each existing survey object can be listed in the Survey Administration console.

When creating a survey campaign in the Create Survey Campaigns page, you also define a child object, the first cycle. You can subsequently modify the cycle name or create additional cycles for an open or active survey campaign from the Survey Campaign Details page.

Deployments are children objects to cycles. Deployments are created from the Create Survey Deployment page, and subsequently viewed and modified from the Deployment Details page.

You can also view responses to individual surveys taken for a deployment from the Deployment Details page by selecting the Response View option.

This section includes the following topics:

- Survey Object Dependencies

- The Parent-Child Survey Objects Relationship

See Also

- Survey Administration Console

- The Survey Questionnaire

- Survey Administration Console View List

- Survey Reports

- Survey Hierarchy, Levels and Objects

- Survey Campaigns

- Cycles

- Deployments

- Survey Resources

- Prototypes

2.10.5.1 Survey Object Dependencies

General Survey object dependencies and rules include the following:

- Each survey campaign must have at least one cycle. It may have as many as required.

- Each cycle must have at least one deployment. It may have as many as required.

- Child objects cannot be shared, reused or inherited.

- Child objects can be added to existing parent objects. For example, for an active or open survey campaign, you may add a second cycle, or additional deployments.

- Before being executed, each deployment must be deployed (set to active status).

- The collection of requirements that is actually executed is the deployment, the lowest and most granular collection of execution parameters. Thus:

  - Deployments can be grouped into cycles to execute the same questionnaire and parameters over a different timeline for comparison purposes.

  - Cancelling a deployment does not affect the parent objects.

2.10.5.2 Parent-Child Survey Objects Relationship

In order to execute a survey campaign, you must have at minimum three survey objects, directly associated in a parent-child relationship.

- You must define one survey campaign.

- You must create one cycle that belongs to (or is a "child" of) the survey campaign.

- You must create one deployment that belongs to (or is a "child" of) the cycle.

Each parent object must have at least one child in order to create executable survey campaigns. Each child object can have a "one-to-many" relationship with its parent object. Thus, a survey campaign must have one cycle and may have more than one, and each cycle must have one deployment and may have more than one. There is no limit to how many cycles or deployments can be created.

Why Allow One-to-Many Relationships?

The primary reason to create more than one deployment is to use different lists, since lists are associated at the deployment level. Thus, a cycle with several deployments may be executed over the same time period but with different lists. These deployments would then be executed by different audiences over the same period of time.

The practical reason to create more than one cycle for a single survey campaign is to execute the same deployment (or set of deployments) over a different period of time.

As an example of these principles, company ABC decides to conduct a survey campaign to measure customer satisfaction with its products. It wishes to send out the same survey questionnaire, six months apart: once before it introduces a new line of products, and once afterwards. ABC customers are identified by three lists: direct mail customers, Internet customers, and retail customers.

Since the same survey will be executed in two separate time periods, company ABC survey administrators can choose to define two separate cycles (cycle 1 and cycle 2) as child objects of survey campaign ABC. In this model, each cycle represents a given time period for survey execution. Define three deployments (one for each list) for each cycle. Each deployment for cycle 1 has identical start and end dates (for example, January through March), and each deployment for cycle 2 also has identical start and end dates, for the second cycle of time (for example, July through September). All other parameters are identical.

Alternatively, one cycle can be created to contain all deployments. In this model, the first three deployments (one for each list) contain identical start and end dates for the same period (for example, January through March). The last three deployments for this cycle (again, one for each list) use identical start and end dates for the second period of time, for example, July through September.

No Sharing of Survey Objects

You cannot share or reuse existing survey campaign, cycle, or deployment objects. Nor can you copy an existing object and make modifications to one or more parameters. Even if you require a survey object (for example, a deployment) with

the same set of parameters, you must redefine the object, associating it with its parent object (for cycles and deployments).

The only objects that can be shared and used are survey resources.

Therefore, in the above example, Cycle 1 and Cycle 2 will need to be created separately. Also, each of the three deployments for the three ABC lists will need to be created each twice, once for each cycle.

2.10.6 Survey Campaigns

The Survey component relies on the concept of a *campaign*: a focused effort to achieve a particular goal from a targeted population over a specific period of time for a particular business purpose. The goal of a *survey campaign* is typically for an enterprise to obtain specific data by polling a target market segment or population for subsequent analysis and subsequent action. Typical actions might include the offering of a new product or service, or a change in business processes to improve satisfaction. As part of a self-service application, a survey campaign typically measures satisfaction regarding the customer experience for that self-service application.

Survey campaigns have requirements that must be defined based on the campaign goals (for example: target population, purpose for obtaining the information, which information will be gathered, for what purpose, how long the campaign will be conducted, and in how many separate deployments of the same requirements).

The campaign goals are achieved by two processes: creating a survey questionnaire built with the campaign goals in mind, and in administering the campaign from the enterprise. The Survey Administration console supports these functions.

The survey campaign is the top-level object for creating and executing survey questionnaire campaigns. Survey campaigns are created using the Survey Administration console, the primary user interface for the Survey component of Oracle Scripting.

This section includes the following topics:

- Survey Campaign Dependencies
- Survey Campaign Status

See Also
- Survey Administration Console
- The Survey Questionnaire

- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Cycles
- Deployments
- Survey Resources
- Prototypes

Survey Campaign Dependencies

Each survey campaign requires:

- A script

  A script serving as the survey questionnaire must already be created and deployed to the applications database using the Script Author component of Oracle Scripting.

- A name to identify the survey campaign

- Header section, footer section, final page, and error page survey resources

  Three survey resources must be defined before creating a survey campaign. Footer section resources are optional. Each is a JSP page that must be defined in the Survey Administration console, and physically loaded to the $OA_HTML directory on the applications server.

- At least one cycle

Survey Campaign Status

Status	Description
Open	When it is initially created, a survey campaign is open and remains in this state until the first deployment is activated.
Active	When any deployment is activated, the parent survey campaign changes from open to active status. Once active, a survey campaign status may change to idle only. Its status never returns to open, nor can it be closed prior to changing to idle status. Active is only valid from the open or idle status. A survey campaign cannot be manually set to active by the survey administrator; this status is established by the system based on these business rules.

Status	Description
Idle	A survey campaign status is idle if at least one of its deployments was activated in the past (the survey campaign was previously active and the deployment was previously either pending or active), but the survey campaign currently has no active or pending deployments. Idle is only valid from the active status. A survey campaign cannot be manually set to idle by the survey administrator; this status is established by the system based on these business rules.
Cancelled	If a survey campaign status is open (after it is created), but has either had no deployments defined or has had no deployments activated, the survey campaign can be set to a status of cancelled, indicating that its purpose is no longer relevant. Once cancelled, no updates can be made to the survey campaign or any of its children objects (cycles or deployments). Cancelled is only valid from the open status. Only a survey administrator can change the status of an open survey campaign to cancelled.
Closed	If a survey campaign is idle (if it currently has no active deployments), and its deployments have been successfully run for the intended duration, the survey campaign can be set to a status of closed, indicating that its purpose was fulfilled. Once closed, no updates can be made to the survey campaign or any of its children objects (cycles or deployments). Closed is only valid from the idle status. Only a survey administrator can change the status of an idle survey campaign to closed.

Once created, and prior to defining a deployment, the status of a survey campaign is Open. You can view the status from the Survey Campaigns page. For open survey campaigns, you can change the status from the Status list by selecting **Cancelled** from the list. This invalidates the survey campaign definition, ensuring it can no longer be used.

Unless you explicitly change the status of a survey campaign after creation, it remains open until a child deployment is activated. Once a deployment is activated, the parent survey campaign status changes to active. A survey campaign remains active as long as at least one of its deployments is active or pending. From active, a survey campaign can become idle, and from idle to cancelled or closed. Cancelled and closed statuses only result from manual intervention by the survey administrator. An active survey campaign can automatically change to idle based on events related to its deployments. For more information, see the section on deployment status.

Once all the deployments for a survey campaign have executed successfully, and the survey campaign is in idle status, you can close the survey campaign from the Status list by selecting **Closed** from the list. This indicates that the survey campaign has served its business purpose. Once a survey campaign is closed, no properties of the closed survey campaign, its cycles, or deployments, can be changed. Return data remains available for viewing responses, or from Oracle Business Intelligence, survey reports can be executed for analysis of survey returned data.

2.10.7 Cycles

A cycle is a child object of the survey campaign. It is the smallest set of requirements for survey campaigns, including only a cycle name, and is defined at the same time its parent object (the survey campaign) is created. Each survey campaign must have at least one cycle defined, and may have many.

The purpose for the cycle object is to provide the capability of grouping its children objects (deployments) for reporting purposes. When analyzing data from executed survey deployments, an enterprise can compare the results of one cycle (one grouping of deployments) to the results of another cycle executed at a different time. This comparative analysis provides the ability to quantify over time effectiveness, satisfaction, or other metrics related to the purpose for a survey campaign.

Cycles are defined from the Survey Campaign tab of the Survey Administration console for any open, active, or idle survey campaign. As with all information entered into the Survey Administration console, the cycle name should follow any existing predefined survey campaign requirements provided to the survey administrator.

Cycle Dependencies

Each cycle requires a cycle name. Oracle Corporation recommends that each cycle be uniquely named within a single survey campaign.

See Also

- Survey Administration Console

- The Survey Questionnaire

- Survey Administration Console View List

- Survey Reports

- Survey Hierarchy, Levels and Objects

- Survey Campaigns

- Deployments

- Survey Resources

- Prototypes

2.10.8 Deployments

A deployment is the most detailed set of execution parameters. There are two types of deployments: targeted (previously referred to as list-based) and standard (previously referred to as non-list-based).

Targeted deployments use Oracle Marketing lists to send e-mail invitations (through Oracle One-to-One Fulfillment) to each member of the list, inviting them to participate in answering a survey questionnaire. These list members can be tracked by individual respondent.

Standard deployments are anonymous and have fewer requirements, as detailed below.

This section includes the following topics:

- Deployment Dependencies
- Targeted Deployment Dependencies
- Deployment Status

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles
- Survey Resources
- Prototypes

Deployment Dependencies

Each deployment requires:

- A parent survey campaign and cycle
- A deployment name
- A media type (currently Web)

- A status

- A deployment start date and time (in the future)

- A response end date and time (in the future)

- A deployment type (standard or targeted)

Targeted Deployment Dependencies

For list-based deployments only, the following are also required:

- List name

- Maximum number of responses per person

- Target response percentage

- Hosting option (stand-alone or menu-based)

- Survey URL (automatically generated from logged-in Apache Web server)

- Invitation template name (identifying invitation master document)

- Invitation e-mail message subject heading

If using reminders in addition to invitations, then the following are also required:

- Reminder template name (reminder master document)

- Reminder e-mail message subject heading

- Number of reminders

- Reminder interval (in days)

Deployment Status

Status	Description
Open	A deployment is open when the deployment is initially created and remains in this state until activated by the survey administrator.
Pending	When a targeted deployment is activated, if the deployment start date and time are in the future, its status changes from open to pending. The deployment remains in this state until the deployment start date and time equal SYSDATE. At this time, the SUBMIT GROUP FM REQUEST FROM IES concurrent program executes, causing the fulfillment request to be submitted to the fulfillment server. Upon a successful submission of the fulfillment request, the deployment status changes to Active. Pending is only valid from the open status, and applies only to targeted deployments

Status	Description
Error	Error status indicates that the concurrent program generated an error while attempting to submit the fulfillment request to the fulfillment server. This status does not include errors that occur after the fulfillment request is successfully sent to the server. If problems occur that prevent invitation or reminder e-mail messages from being sent, the deployment status remains active, and fulfillment debugging should commence by a fulfillment administrator from the Oracle One-to-One Fulfillment administration console. Error is only valid from the pending status, and applies only to targeted deployments.
Active	A standard deployment is active immediately after it is activated by the survey administrator. A targeted deployment is active when the deployment start date and time equal SYSDATE, and the fulfillment request is successfully sent to the fulfillment server by the concurrent program. Note that active status for targeted deployments does not indicate successful delivery of invitation or reminder e-mail messages.
Cancelled	If a deployment has a status of open, its status can be manually set to cancelled, indicating there is no more need to execute this set of requirements. After being cancelled, a deployment cannot be executed. Cancelled is only valid from the open deployment status.
Incomplete	If a deployment has an active status but there is a no interest on the part of the surveying enterprise to view the results, it can be set to incomplete status.
Closed	If a deployment is in active status and has run for its intended duration or has resulted in sufficient responses, it can be set to a status of closed, indicating that its purpose was fulfilled. Deployments in a pending or error status can also be set to closed.

Once a deployment is created, and before it is activated, its status is Open. You can view the deployment status from the Survey Campaign Details page or from the Deployment Details page. From either of these pages you can also change the status of an open deployment by selecting **Cancelled** from the status list. This invalidates the deployment definition, ensuring it can no longer be used.

The only way to change the status of a standard deployment to active is to activate it from the Deployment Details page. The Deployment Details page will refresh, with the status set to Active and a valid survey hyperlink displayed. This is true whether the Deploy Date is in the past or the future.

The only way to change the status of a targeted deployment to pending is to activate it from the Deployment Details page. If the deployment start date and time are in the future, the status will change to pending until they equal SYSDATE. If the deployment start date and time are in the past, the concurrent program will attempt to submit the fulfillment request to the fulfillment server immediately. Upon successful submission of the fulfillment request to the server, the deployment status will change to active. For targeted deployments, the deployment details will not display a survey URL, since respondent identification numbers for each list member are required.

Once a deployment is activated, and before the deployment end date is reached, its status can change from active to closed or incomplete. Do this from the deployment details page.

- Use incomplete to indicate that there is no interest in executing this deployment and viewing its results.

- Use closed when the deployment goals have been achieved. For example, after the deployment start and end date have passed and the deployment was successfully executed, change active status to closed. As another example, close the deployment if you determine that a sufficient number of responses for a deployment have been received to perform the required analysis or make the appropriate business decisions for which the survey campaign is intended.

For targeted deployments, after the deployment is activated, it may contain either a status of pending (indicating that it is activated and the concurrent request is in the queue or its start date is in the future), active (indicating that the concurrent request completed successfully), or error (indicating that the concurrent program generated an error while creating and sending the fulfillment request). Deployments with a status of error can be modified so that the error can be corrected, and when reactivated will again display a pending status. Targeted deployments can also contain a status of closed or incomplete, following the same guidelines described above for standard deployments.

Error status cannot be set manually. This status, for targeted deployments only, indicates that the concurrent program generated an error while attempting to submit the fulfillment request to the fulfillment server.

Additional Requirements for Targeted Deployments

For targeted survey deployments, once a survey deployment is activated, you must wait until the fulfillment engine completes its activity (completes the fulfillment request and succeeds in sending the invitation master documents to the outgoing mail server specified in Oracle One-to-One Fulfillment) before you will have respondents. The concurrent request ID is listed on the deployment details page when displayed in the deployment view, and can be used to track the concurrent request and its status.

Survey administrators who have also been assigned the JTF role JTF_FM_ADMIN are able to view the status and history of fulfillment requests from the Invitations tab of the Survey Administration console.

> **Note:** Granting JTF roles requires the grantor to be assigned the JTF system administrator role, JTF_SYSTEM_ADMIN_ROLE, in addition to whichever other JTF role you want to assign. For this purpose, if required, you can use the seeded sysadmin Oracle Applications user account. For more information, see Section 4.7, "Creating and Administering Oracle Scripting Users".

When viewing status, a request status of **Submitted** indicates that the list was successfully sent to the fulfillment engine. An outcome code of **Success** indicates that the fulfillment server was successful in sending the invitation (or reminder) master document. For any other outcome code (e.g., **Partially Successful** or **Failure**), you will need to log into the Fulfillment Administration console to resolve. In this scenario, list members must receive and respond to an e-mailed invitation before you can expect activity for this deployment.

To access the Fulfillment Administration console, a user must be assigned the JTF_ FM_ADMIN role and the Fulfillment Administrator responsibility. Consult with an Oracle One-to-One Fulfillment administrator if required.

Effects of Setting Deployment to Closed

Immediately upon changing a deployment's status from active to closed, the status for its parent survey campaign changes from active back to idle, *if there are no other active or pending deployments.* The survey campaign will remain active if other deployments have been activated and have not encountered an error.

With survey campaigns for which the last deployment has been changed to closed, you can close the survey campaign, or create another cycle or deployment if you wish to receive more responses for this survey campaign.

Closing a survey campaign or deployment signifies that the requirements or objectives for that object have been met. No new information can be received from a closed survey campaign or deployment.

From the Survey Administration console, responses can still be viewed for closed deployments.

Survey reports can still be viewed for closed deployments.

Effects of Setting Deployment to Incomplete

Once you designate a deployment as incomplete, if there are no other active or pending deployments, the parent survey campaign changes to idle, and can be

closed unless you wish to create new cycles or deployments to obtain additional responses. The incomplete deployment status indicates that the requirements or objectives for the deployment are no longer relevant. No new information can be received from a cancelled deployment. While responses and reports can be viewed for this deployment, the information generated by reports is likely to be incomplete.

2.10.9 Survey Resources

In order to use the Survey component of Oracle Scripting, you must use survey resources, including a header section, footer section, error page, and final page resource. Survey resources are Java Server Page (JSP) files, or HTML files saved with a .JSP file extension.

The header section resource appears at the top portion of each Web browser page when a respondent takes a survey. This resource is required for all survey campaigns. Similarly, the footer section resource appears at the bottom of each Web page for a survey, if defined. Footer section resources are the only optional survey resource. The error page only results in the event of certain errors, and provide access to stack trace information. The final page displays after a respondent completes a survey, and typically has hypertext links to the enterprise home page. Error page and final page resources do not display the header or footer section resources.

Survey resources must be created, uploaded to the applications server, defined in the Survey Administration console, and specified for each survey campaign.

This section includes the following topics:

- Creating Survey Resource JSP Files

- Seeded Survey Resources

See Also
- Survey Administration Console

- The Survey Questionnaire

- Survey Administration Console View List

- Survey Reports

- Survey Hierarchy, Levels and Objects

- Survey Campaigns

- Cycles

- Deployments

- Prototypes

2.10.9.1 Creating Survey Resource JSP Files

Survey resources are JSP files, or HTML files saved with a .JSP file extension.

You must have the requisite knowledge and resources to create or modify HTML and JSP files to use these appropriately with the Survey component of Oracle Scripting.

No provisions are made to create HTML or JSP files within the context of the Oracle Scripting product. However, as described in the section Defining Survey Resources, four test resource JSP files ship with Oracle Applications, seeded in the appropriate directory ($OA_HTML) on the applications server. You can use the four test resources to test your implementation and to serve as building blocks, modifying copies of these files for your own use as appropriate.

Sample Code for Test Header Section Resource

Following is the source code for the sample test header section. *This sample is provided for informational purposes only.* Oracle Corporation is not responsible for the correct implementation of JSP in your environment, and does not provide support for customized survey resources. Consult appropriate HTML and JSP experts for more information.

```
<!-- $Header: iessvytestheader.jsp $ -->
<table align=center border=0>
    <tbody>
     <tr>
        <td> </td>
        <td class=pageTitletcolspan=4 noWrap>Test Header Section</td>
        <td> </td>
     </tr>
    </tbody>
  </table>
```

2.10.9.2 Seeded Survey Resources

Survey resources must be defined in the Survey Administration console and must map to existing HTML or JSP documents residing on the server (as described in Uploading Survey Resources).

For survey campaigns in production, you will most likely want to create your own resources. For testing and implementation verification, you may want to use the survey resources seeded with Oracle Applications.

Test Survey Resources

Test survey resources, identified in the table below, ship with Oracle Applications beginning with IES MiniPack G, and available with any Rapid Install from 11.5.5 and later. These resources are physically located in $OA_HTML on the applications server.

Note that there is no seeded footer section resource. For testing purposes, to ensure a footer will appear as expected in a production environment, you can use another JSP page (for example, the seeded header).

Use Test Survey Resources for Implementation Testing

Utilizing these test survey resources provides a method to verify that survey resources are appropriately displayed upon execution of the survey campaign deployment in an HTML user interface. At the same time, you are provided with a layer of abstraction in regard to the need to ensure the product is performing as expected without needing to test the creation of tailored JSP or HTML pages. For this purpose, it is recommended that you first test execution of scripts in a Web browser using seeded test survey resources listed above, to ensure successful implementation. Subsequent to successful execution of a script in a Web browser with all appropriate test resources displaying, you can change the survey resources assigned to a survey campaign to use customized survey resources.

Additional Error Page Resource for Hosted Surveys

Header section and final page seeded survey resources are intended for use with stand-alone and menu-based survey deployments. The IESSVYTESTERROR.JSP error page is specifically intended for use with stand-alone deployments. For hosted scripting operations (self-service Web applications such as Oracle iSupport, or other self-service Web applications customized to provide access to scripts executed in a Web browser), the IESSVYTESTERROR.JSP error page should be used.

To test a hosted scenario, you can use any JSP header section (or HTML page saved in JSP file format). Again, you can use the seeded test header section to test footer section functionality as well. Error and final pages must also be in JSP file format. The error page must adhere to the hosted template. There are two ways to handle the final page:

1. Define as the final survey resource page the JSP page which displays the list of URLs to take the survey. In this way, when the survey is completed, the respondent will be returned to the starting page.

2. Define another JSP page as the final page survey resource, but include in that page a JSP forward command to point the user back to the page that displays the list of URLs to take the survey.

For more information, refer to the discussion in the section Non-List-Based Entry Points in *Oracle Scripting User's Guide*.

2.10.10 Prototypes

Prototype is a boolean characteristic of survey campaigns. The default for this characteristic is null (survey campaigns are not designated as prototypes unless you select this option when creating or editing a survey campaign).

Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. The purpose is to allow survey campaign administrators more freedom to refine requirements for survey campaigns, including modification to the script for the designated survey campaign.

The Survey Campaigns tab includes an option to exclude prototypes. Selecting this option filters prototypes out of the lists of survey campaigns displayed. This option is also null by default (prototype survey campaigns are included in survey campaign lists unless you select this box and click Go).

Although you can also put a prototype survey campaign into production, the script will not be locked. The prototype option can be selected or cleared while a survey campaign status is Open. Thereafter, you cannot change this option.

See Also

- Survey Administration Console
- The Survey Questionnaire
- Survey Administration Console View List
- Survey Reports
- Survey Hierarchy, Levels and Objects
- Survey Campaigns
- Cycles

- Deployments
- Survey Resources
- Prototypes

3

Before You Begin

This section includes the following topics:

- Section 3.1, "Related Product Documentation"

- Section 3.2, "Dependencies"

- Section 3.3, "Minimum Software Requirements"

- Section 3.4, "Minimum Hardware Requirements"

- Section 3.5, "Scalability and Performance Guidelines"

- Section 3.6, "Installing Oracle Scripting"

- Section 3.7, "Accessing Oracle Scripting"

3.1 Related Product Documentation

You can choose from many sources of information, including online documentation, training, and support services, to increase your knowledge and understanding of Oracle Scripting.

If this guide refers you to other Oracle Applications documentation, use only the Release 11*i* versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF). Some online help patches are available on Oracle*MetaLink*. Oracle Scripting online help is only available with installation of Oracle Scripting release 11.5.6 or later.

Related Documentation

Oracle Scripting shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other product documentation when you set up and use Oracle Scripting.

You can read the documents online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at http://oraclestore.oracle.com.

Documents Related to All Products

Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI) available with this release of Oracle Scripting (and any other Oracle Applications products). This guide also includes information on setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Documents Related to This Product

Oracle Scripting Implementation Guide 11*i*

This guide describes how to implement Oracle Scripting components and test the implementation appropriately. Administration information includes one-time implementation-specific tasks, script-dependent file administration, survey campaign and survey resource administration. Other administration topics include Oracle Marketing Online campaign and list administration in support of Scripting, and Oracle One-to-One Fulfillment administration in support of list-based survey campaigns.

Oracle Scripting User Guide

This guide includes information on understanding each Oracle Scripting component and planning Oracle Scripting projects. This guide also includes task-based steps for using the Script Author, and describes use of the Scripting Engine.

Oracle Scripting Developer's Guide

This guide is for developers of Script Author scripts and developers of code integrating with scripts. It includes a section explaining how and where to use Script Author commands. Also detailed is information on customizing Oracle Scripting with reusable seeded commands, best practice Java methods and survey scripts, and building block components supporting Oracle Scripting. Specific guidance for customizing panel layouts is also detailed, along with troubleshooting information.

Installation and System Administration

Oracle Applications Concepts

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

Installing Oracle Applications

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications, the Oracle8 technology stack, and the Oracle8*i* Server technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you need to perform to finish your installation. You should use this guide in conjunction with individual product user's guides and implementation guides.

Oracle Applications Supplemental CRM Installation Steps

This guide contains specific steps needed to complete installation of a few of the CRM products. The steps should be done immediately following that tasks given in the Installing Oracle Applications guide.

Upgrading Oracle Applications

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7 (NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

Maintaining Oracle Applications

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you need to run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

Oracle Applications System Administrator's Guide

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

Oracle Alert User's Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Applications Developer's Guide

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. It also provides information to help you build your custom Oracle Forms Developer *6i* forms so that they integrate with Oracle Applications.

Oracle Applications User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built by using Oracle Forms.

Other Implementation Documentation

Multiple Reporting Currencies in Oracle Applications

If you use the Multiple Reporting Currencies feature to record transactions in more than one currency, use this manual before implementing Oracle Scripting. This manual details additional steps and setup considerations for implementing Oracle Scripting with this feature.

Multiple Organizations in Oracle Applications

This guide describes how to set up and use Oracle Scripting with Oracle Applications' Multiple Organization support feature, so you can define and support different organization structures when running a single installation of Oracle Scripting.

Oracle Workflow Guide

This guide explains how to define new workflow business processes as well as customize existing Oracle Applications-embedded workflow processes.You also use this guide to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Scripting implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This manual also provides information on creating custom reports on flexfields data.

Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle*MetaLink*

Oracle Manufacturing APIs and Open Interfaces Manual

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes APIs and open interfaces found in Oracle Manufacturing.

Oracle Order Management Suite APIs and Open Interfaces Manual

This manual contains up-to-date information about integrating with other Oracle Manufacturing applications and with your other systems. This documentation includes APIs and open interfaces found in Oracle Order Management Suite.

Oracle Applications Message Reference Manual

This manual describes Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11i.

Oracle CRM Application Foundation Implementation Guide

Many CRM products use components from CRM Application Foundation. Use this guide to correctly implement CRM Application Foundation.

Training and Support

Training

Oracle offers training courses to help you and your staff master Oracle Scripting and reach full productivity quickly. You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many Education Centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization structure, terminology, and data as examples in a customized training session delivered at your own facility.

Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Scripting working for you. This team includes your Technical Representative, Account Manager, and Oracle's large staff of consultants and support specialists with expertise in your business area, managing an Oracle8*i* server, and your hardware and software environment.

Oracle*MetaLink*

Oracle*MetaLink* is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With Oracle*MetaLink*, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use Oracle*MetaLink*, register at (http://metalink.oracle.com).

Alerts: You should check Oracle*MetaLink* alerts before you begin to install or upgrade any of your Oracle Applications. Navigate to the Alerts page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade/Alerts.

Self-Service Toolkit: You may also find information by navigating to the Self-Service Toolkit page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade.

See Also

-

-

-

-

-

-

3.2 Dependencies

This section includes the following topics:

-

-

See Also

-

-

-

-

-

-

3.2.1 Mandatory Dependencies

- Oracle Applications must first be installed using Oracle Applications release 11*i* Rapid Install or release 11*i* Platinum, followed by appropriate upgrades to release 11.5.9 or later.

- In order to execute scripts using Oracle Scripting 11.5.6 or later, scripts must first be compiled and deployed using Script Author 11.5.6 or later.

  - All deployed scripts will display in the Script Chooser, including any scripts deployed using previous versions of Oracle Scripting. However, scripts compiled in versions of the Script Author prior to 11.5.6 will not

execute. Selecting a pre-11.5.6 script to run will result in a blank Scripting window with error messages (see troubleshooting).

- If a script from previous versions of the Script Author must be used, open the script using Script Author 11.5.6 or later, recompile, save the updated script, and redeploy to the appropriate applications database.

- For new implementations of Oracle Scripting, you must specify the value Apache Mid Tier/Servlet Architecture in the system profile IES : ARCHITECTURE TYPE in order to use the Scripting Engine agent application.

- You must use an Oracle Applications 11*i*-certified Web browser to launch Oracle Applications. For software certifications, see Oracle*MetaLink*.

- You must have a certified version of Oracle JInitiator to launch Oracle Applications. If not present, you should be able to download the appropriate version when you attempt to access an Oracle Applications URL. For certifications, see Oracle*MetaLink*.

- If using a UNIX server, you must identify an X Server as a display server to dynamically generate graphics using JFC/SWING and AWT, in order to view panel footprint reports accessible from the Reporting tab on the Scripting Administration console.

  See Section 4.6.5, "Setting Display Server for UNIX Environments" for information on the setting a display server.

  This requirement no longer applies to UNIX server implementations in order for respondents to participate in a survey, or to use the Survey Administration console.

See Also
- Section 3.2.2, "Conditional Dependencies"

3.2.2 Conditional Dependencies

The following conditional dependencies may apply, as appropriate:

- Section 3.2.2.1, "Script Author Dependencies"

- Section 3.2.2.2, "Scripting Engine Agent User Interface Dependencies"

- Section 3.2.2.3, "Survey Component Dependencies"

- Section 3.2.2.4, "Java Development Kit Dependencies"

See Also

- Section 3.2.1, "Mandatory Dependencies"

3.2.2.1 Script Author Dependencies

If using custom Java code to write commands supporting the Scripting Engine in the agent interface, you must compile custom Java classes using a version of JDK that is compatible with the JRE used at runtime. For implementations using the Apache mid-tier architecture, this is the version of JDK used on the Apache Web server. For implementations using the caching architecture (supported for upgrades from release 11.5.4 only), this implies a version of JDK that is compatible with Oracle JInitiator on the agent client (at the same level or lower).

References

- Section 2.7.5, "Using Custom Java"

- Section 3.2.2.4, "Java Development Kit Dependencies"

See Also

- Section 3.2.2.2, "Scripting Engine Agent User Interface Dependencies"

- Section 3.2.2.3, "Survey Component Dependencies"

- Section 3.2.2.4, "Java Development Kit Dependencies"

3.2.2.2 Scripting Engine Agent User Interface Dependencies

If referencing custom Java code in a script:

- The custom JAR must be appropriately packaged.

- You must compile custom Java classes using JDK 1.1.8.

- You must appropriately reference any custom Java methods in the script, and ensure the compiled and packaged code is available to the Scripting Engine at runtime.

  - When deploying custom Java using the Scripting Administration console, you must deploy scripts using the Script Author Java applet.

  - If storing your custom Java code on the applications server, you must reference the class path of any custom Java in the Apache Web server JSERV.PROPERTIES file.

References

- For information on packaging JAR files, see Section 3.2.2.4.2, "Packaging Java Bean Code Into a Java Archive File".

- For information on JDK 11.18 requirements, see Script Author release 11.5.6, see Section 3.2.2.4.1, "Compiling Custom Code Using JDK 1.1.8".

- For information on referencing the class path, see Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File".

See Also

- Section 3.2.2.1, "Script Author Dependencies"

- Section 3.2.2.3, "Survey Component Dependencies"

- Section 3.2.2.4, "Java Development Kit Dependencies"

3.2.2.3 Survey Component Dependencies

List-Based Survey Dependencies

If an implementation of the Survey component of Oracle Scripting requires the use of list-based survey campaigns, the following are dependencies:

- You must have Oracle One-to-One Fulfillment installed and implemented.

- You must have an appropriately configured fulfillment server.

- You must have Oracle Marketing Online installed and implemented.

- You must have access to a user with the Survey Administrator responsibility.

- In order to use HTTPS, you must have proxy servers appropriately configured, and you must set the appropriate system profiles (IES: Proxy Server Name, IES: Proxy Server Port and Apps Servlet Agent).

Self-Service Web Application Customization Dependencies

Implementing the Survey component of Oracle Scripting for use from a self-service Web application such as Oracle iSupport, the following are dependencies:

- You must customize the self-service Web application to list survey URLs from existing valid non-list-based survey deployments.

- The customized page listing survey URL inks is recommended to be designated as the final page survey resource; at the completion of script execution in an

Oracle Applications 11*i*-certified web browser, the user is returned to the page from which they launched the survey script.

- You must use a hosted survey error page such as the seeded sample JSP for this purpose entitled IESSVYMENUBASEDTESTERROR.JSP.

References

- For information on Oracle One-to-One Fulfillment requirements, use, and implementation, see *Oracle One-to-One Fulfillment Implementation Guide*.

- For information on Oracle Marketing Online requirements and implementation, see *Oracle Marketing Online Implementation Guide*.

- For information on using and administering Oracle Marketing Online, see *Oracle Marketing Online Concepts and Procedures*.

- For information on the appropriate system profiles for using HTTPS over a proxy server, see Section 4.4, "Implementing the Scripting Engine Agent Interface".

- For information on customizing self-service Web applications for use with Oracle Scripting, see Section 10.2.1.1, "Oracle iSupport".

See Also

- Section 3.2.2.1, "Script Author Dependencies"
- Section 3.2.2.2, "Scripting Engine Agent User Interface Dependencies"
- Section 3.2.2.4, "Java Development Kit Dependencies"

3.2.2.4 Java Development Kit Dependencies

This section includes the following topics:

- Section 3.2.2.4.1, "Compiling Custom Code Using JDK 1.1.8"
- Section 3.2.2.4.2, "Packaging Java Bean Code Into a Java Archive File"

See Also

- Section 3.2.2.1, "Script Author Dependencies"
- Section 3.2.2.2, "Scripting Engine Agent User Interface Dependencies"
- Section 3.2.2.3, "Survey Component Dependencies"

3.2.2.4.1 Compiling Custom Code Using JDK 1.1.8 Oracle Scripting information renders in HTML using different HTML rendering engines. For scripts executed in HTML as a survey, the Oracle Applications 11*i*-certified Web browser provides HTML rendering and interpretation. Various browsers, and various versions of the same browser, may look substantially different from release to release.

For all Oracle Applications 11*i* releases to date, including release 11.5.9, scripts running in the Scripting Engine agent interface use the Java classes in the SWING library of JDK 1.1.8 to render the HTML.

For this reason, Oracle Corporation recommends:

1. Testing scripts in the lowest common denominator (the Scripting agent interface) to ensure expected interpretation of HTML elements.

2. Compiling custom Java code using JDK 1.1.8 to ensure compatibility for custom code when executing scripts in the Scripting Engine agent interface.

3. For Scripting implementations using the Caching Architecture, the Java Virtual Machine for executing Scripting is provided by Oracle JInitiator, which at the time of this writing is not supported with Oracle Scripting at versions above JDK 1.1.8.x. Thus, *all custom code* to be executed in this architecture *must be compiled using JDK 1.1.8* to ensure compatibility. This requirement will relax if JInitiator versions compiled against higher JDK versions than 1.1.8 are released and certified by Oracle Corporation.

    For this and other certification information, refer to Oracle*MetaLink* or contact your Oracle Support Services representative.

See Also

■ Section 3.2.2.4.2, "Packaging Java Bean Code Into a Java Archive File"

3.2.2.4.2   Packaging Java Bean Code Into a Java Archive File

For more information, see Packaging Java Bean or Custom Java Code Into a JAR File in *Oracle Scripting User's Guide*.

Oracle Scripting 11*i* interprets custom Java code when appropriately referenced in a script using a Script Author command. The two common uses of custom Java are to substitute a custom Java Bean user interface in place of a standard Scripting panel (in the agent interface only), or (for any Scripting Engine interface) to execute commands in the script at runtime.

In order for a script referencing custom Java (a Java bean or a Java method) to supply that code to the Scripting Engine user interface at runtime, the Java code

must first be written, tested, compiled, and finally packaged into a Java Archive (JAR) or WinZip (ZIP) file. Oracle Corporation recommends using JAR file format.

Due to a limitation in JDK 1.1.8, there are only two supported methods of packaging a JAR file appropriately. Use either of the following methods to appropriately package custom Java code for execution at runtime.

The two methods are:

1. Use the command-line "jar" utility from JDK 1.1.8 and specify no compression. For example:

    ```
    jar -cf0 TestBean.jar ...).
    ```

    For more information, refer to JDK documentation.

2. Create the custom JAR file in standard .ZIP or .JAR format (with any PC-standard compression utility) with or without compression. If that utility saves the resulting archive in .ZIP format, then simply rename the file extension from .ZIP to .JAR.

    This method of packaging a JAR file may differ based on the compression utility used. For specifics, see the compression utility manufacturer's documentation.

See Also
- Section 3.2.2.4.1, "Compiling Custom Code Using JDK 1.1.8"

3.3 Minimum Software Requirements

The following are the minimum software requirements for using Oracle Scripting release 11.5.9:

- You must have Oracle Applications release 11*i* installed or updated to the appropriate patch level.

- You must have Oracle Scripting 11.5.6 or higher (IES MiniPack L or later) to execute scripts in Scripting Engine releases 11.5.6 and higher.

- You must have an Oracle Applications 11*i*-certified Web browser.

- You must have Oracle Scripting release 11.5.8 (Interaction Center Family Pack P) or later to deploy custom Java to the database using the Scripting Administration console. This is a requirement for replacing a script panel with a

user interface Java bean at runtime and can be used to load Java for all custom Java methods.

- You must have custom Java compiled using a version of JDK that is compatible with the JRE used on the Apache Web server (the same level or lower) to execute custom Java using the Apache mid-tier architecture of Oracle Scripting. At this time, appropriate JDK versions may include JDK 1.3, 1.2, or 1.1.8.

- You must have custom Java compiled using a version of JDK that is compatible with Oracle JInitiator (the same level or lower) on the agent client to execute custom Java using the caching architecture of Oracle Scripting. At this time, appropriate versions may include 1.3 or 1.1.8.

- You must have custom Java compiled using a version of JDK that is compatible with Oracle JInitiator (the same level or lower) on the agent client to replace a script panel with a user interface Java bean at runtime. At this time, appropriate versions may include 1.3 or 1.1.8.

Certification information is constantly changing. For the latest information on software certifications or compatibility, refer to Oracle*MetaLink* or Oracle iSupport.

See Also
- Section 3.1, "Related Product Documentation"
- Section 3.2, "Dependencies"
- Section 3.3, "Minimum Software Requirements"
- Section 3.4, "Minimum Hardware Requirements"
- Section 3.5, "Scalability and Performance Guidelines"
- Section 3.6, "Installing Oracle Scripting"
- Section 3.7, "Accessing Oracle Scripting"

3.4 Minimum Hardware Requirements

Oracle Applications must be installed (using a Rapid Install) on an appropriate server-class UNIX or Windows NT server. The server must have suitable memory (RAM) and storage (hard drive space) for enterprise-specific tasks, as determined by enterprise system administrators. Guidance can be found for minimum requirements in *Installing Oracle Applications*.

See Also

- Section 3.1, "Related Product Documentation"

- Section 3.2, "Dependencies"

- Section 3.3, "Minimum Software Requirements"

- Section 3.5, "Scalability and Performance Guidelines"

- Section 3.6, "Installing Oracle Scripting"

- Section 3.7, "Accessing Oracle Scripting"

3.5 Scalability and Performance Guidelines

Oracle Scripting has the same scalability and performance guidelines as other Oracle Applications CRM release 11*i* products. For other scalability and performance guidelines, refer to *Installing Oracle Applications*.

Some implementations install the database tier of Oracle Applications on a separate physical host machine to improve performance. Technically, each tier in a Rapid Install may be installed on a separate host machine for additional performance gains, although this is a rare occurrence. Detectable performance gains for multi-server implementations are likely if load on the applications database is heavy. For medium to light load there should be little appreciable gain. Regardless of whether you have a single or multiple configuration, Rapid Install is a prerequisite. Consult with your systems administrator for more information.

See Also

- Section 3.1, "Related Product Documentation"

- Section 3.2, "Dependencies"

- Section 3.3, "Minimum Software Requirements"

- Section 3.4, "Minimum Hardware Requirements"

- Section 3.6, "Installing Oracle Scripting"

- Section 3.7, "Accessing Oracle Scripting"

3.5.1 Installation and Dependency Verification

You must verify an implementation using the procedures described in the section Section 11, "Implementation Verification". This includes:

For All Implementations

- Obtaining the regression test script from Oracle*MetaLink* to perform Scripting Engine testing in stand-alone mode.

- Opening, compiling, and deploying the regression test script to your applications database.

- Uploading custom JAR file to the database using the Scripting Administration console.

- For testing scripts in the agent interface, assigning the Scripting User or Scripting Agent responsibility to an existing or new Oracle Applications user.

- Successfully launching and executing the regression test script in the Scripting Engine.

For Survey Implementations

- Assigning the Survey Administrator responsibility to an existing or new Oracle Applications user.

- Creating or modifying survey resources and ensuring they are present in the $OA_HTML directory on the server.

- Defining survey resources, a survey campaign, cycle, and deployment in the Survey Administration console.

- Activating a deployment.

- Ensuring you can participate in a survey using a Web browser.

- If viewing responses, ensure answer collection is enabled in the appropriate script to ensure survey response data is collected for viewing.

- Assigning the iSurvey User responsibility to an existing or new Oracle Applications user, to administer concurrent programs.

- Ensuring you can schedule or perform concurrent programs to update deployment status and move survey data to survey summary tables.

- If using reporting, perform the Oracle Discoverer implementation steps described in Appendix C, "Oracle Discoverer Workbooks".

3.6 Installing Oracle Scripting

You have the following options for installing Oracle Scripting:

> **Note:** Prior to installation or upgrade, *always* review the readme and patch list for the Oracle Interaction Center Family Pack. The readme contains installation instructions, a summary of product changes, and a list of known issues for a family pack. The patch list contains a list of patches that are required or recommended for use with a family pack. The patch list for each Oracle Interaction Center Family Pack is available on Oracle*MetaLink* at http://metalink.oracle.com. Each patch list contains a hyperlink to the readme. (Perform an advanced search for Document ID 219238.1. Enter the document ID in the Search Field and select the Doc ID option.)

- Oracle Applications Rapid Install

  The Rapid Install is intended for customers who are installing Oracle Applications for the first time or upgrading to Release 11*i* from to Release 11.0 or Release 10.7. It contains the family packs or product minipacks for all products in Oracle Applications.

  The Rapid Install is provided on CD-ROMs and is available from Oracle Store at http://oraclestore.oracle.com. For information about installing Oracle Applications using Rapid Install, see *Installing Oracle Applications*. For information about upgrading Oracle Applications using Rapid Install, see *Upgrading Oracle Applications*.

- Oracle Applications Maintenance Pack

  The Maintenance Pack is intended for customers who have already installed Oracle Applications Release 11*i*. It contains the family packs or product minipacks for all products in Oracle Applications.

  The Maintenance Pack is provided as a patch and is available on Oracle*MetaLink* at http://metalink.oracle.com. For information about upgrading Oracle Applications Release 11i using the Maintenance Pack, see Maintenance Pack Release Instructions on Oracle*MetaLink* at http://metalink.oracle.com. (Perform an advanced search for Document ID 232834.1. Enter the document ID in the Search Field and select the Doc ID option.)

  When installing an Oracle Applications Maintenance Pack, you must also check the patch list for the corresponding Oracle Interaction Center Family Pack, for any new issues that have emerged since the Maintenance Pack was released.

The patch list for each Oracle Interaction Center Family Pack is available on Oracle*MetaLink* at http://metalink.oracle.com. (Perform an advanced search for Document ID 219238.1. Enter the document ID in the Search Field and select the Doc ID option.)

- Oracle Interaction Center Family Pack

  The Oracle Interaction Center Family Packs are intended for customers who have already installed or upgraded to Oracle Applications Release 11*i* and wish to upgrade Oracle Interaction Center, or are upgrading another Oracle Applications product family, such as Oracle Service, which uses Oracle Interaction Center components, such as Universal Work Queue. The family pack is cumulative and contains only the minipacks for products in the Oracle Interaction Center family.

  > **Note:** Oracle Interaction Center products integrate with other products in Oracle Applications. Therefore, you may have to install family packs, product minipacks, or individual product patches for *other* products before installing the Oracle Interaction Center Family Pack.

The Oracle Interaction Center Family Pack is provided as a patch and is available on OracleMetaLink at http://metalink.oracle.com. The patch list for each Oracle Interaction Center Family Pack is available on OracleMetaLink at http://metalink.oracle.com. (Perform an advanced search for Document ID 232834.1. Enter the document ID in the Search Field and select the Doc ID option.)

See Also

- Section 3.1, "Related Product Documentation"

- Section 3.2, "Dependencies"

- Section 3.3, "Minimum Software Requirements"

- Section 3.4, "Minimum Hardware Requirements"

- Section 3.5, "Scalability and Performance Guidelines"

- Section 3.7, "Accessing Oracle Scripting"

3.7 Accessing Oracle Scripting

Oracle Applications uses two different technology stacks:

- Oracle Application Object Library (AOL)

- Java Technology Foundation (JTF)

AOL applications are Oracle Forms developed using Oracle Developer and are usually referred to as Forms-based application. JTF applications are Java Server Pages (JSPs) developed using Oracle JDeveloper and are usually referred to as HTML-based applications. Each type of application accesses the same database and can share information with the other.

The product interfaces are accessed by providing the Uniform Resource Locator (URL) for the environment in an Oracle Applications 11*i*-compatible Web browser and navigating to the hyperlink for the login page for the specific technology stack. You can also provide the URL for the specific login page. This URL is referred to as your login URL.

Oracle Applications URL   Use this URL to navigate to the Personal Home Page URL or the CRM Home page URL.

```
http://<host>:<port>/
```

- To navigate to the Personal Home Page URL, choose **Apps Logon Links** > **Personal Home Page**.

- To navigate to the CRM Home Page URL, choose **Apps Logon Links** > **CRM Home Page**.

CRM Home Page URL   This URL is sometimes referred to as the Apache or JTF login URL. Use this URL to open the login page for HTML-based applications.

```
http://<host>:<port>/OA_HTML/jtflogin.jsp
```

Personal Home Page URL: This URL is sometimes referred to as the Self-Service Web Applications or SSWA login URL. Use this URL to open the login window for Oracle Applications via the Personal Home Page. You can access Forms-based or HTML-based applications from the Personal Home Page.

```
http://<host>:<port>/OA_HTML/US/ICXINDEX.htm
```

Forms URL   Use this URL to open the login page for Forms-based applications. This login URL is typically used by system administrators, not end users.

```
http://<host>:<port>/dev60cgi/f60cgi
```

3.7.1 User Accounts

An application user is an authorized user of Oracle Applications and is uniquely identified by a username. After the user account has been defined, the application user can sign on to Oracle Applications at the CRM Home Page, Personal Home Page, or Forms login.

> **Note:** Oracle Applications is installed with a system defined username and password.
>
> ■ Username: sysadmin
>
> ■ Password: sysadmin

An application user enters a username along with a password to sign on to Oracle Applications. The password assigned by the system administrator is temporary. When signing on for the first time, the application user will be prompted to change the password. Access to specific functionality and data will be determined by the responsibilities assigned to your user account.

3.7.2 Responsibilities

A system administrator assigns one or more responsibilities to an application user. A responsibility is a level of authority that allows a user to access specific functionality and data in Oracle Applications. Oracle Applications is installed with predefined responsibilities. A system administrator can modify a predefined responsibility or create custom responsibilities.

Various predefined responsibilities are required to implement Oracle Scripting, based on which components are required. These responsibilities are clearly defined in each relevant implementation section of this document.

In the Forms interface, if an application user has only one responsibility, then the related menu or application (if there is only one function in the menu) appears after the user signs on. If an application user has more than one responsibility, then a list of available responsibilities appears after the user signs on. To switch responsibilities, choose Switch Responsibility from the File menu.

In the HTML interface, an application user must select a default responsibility (even if the user has only one responsibility). The next time the application user signs on, the tabs related to the default responsibility appear. To switch responsibilities, go to Navigation Preferences in your profile (Profile icon). In the Switch Responsibilities section, select another responsibility from the Current Responsibility list.

See Also

- Section 3.1, "Related Product Documentation"
- Section 3.2, "Dependencies"
- Section 3.3, "Minimum Software Requirements"
- Section 3.4, "Minimum Hardware Requirements"
- Section 3.5, "Scalability and Performance Guidelines"
- Section 3.6, "Installing Oracle Scripting"

4

Implementation Tasks

This section describes all tasks that may be required to implement Oracle Scripting. This section includes the following topics:

- Section 4.1, "Implementation Task Sequence"

- Section 4.2, "Creating an Administrator for Oracle Scripting"

- Section 4.3, "Creating an Agent"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.6, "Implementing Administrative Interfaces"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.1 Implementation Task Sequence

You must have an understanding of which Oracle Scripting components and interfaces are required at the enterprise in order to implement successfully. Scripting components include the Script Author, the Scripting Engine (which includes a Web interface and an agent interface), and two HTML user interfaces for administration: the Scripting Administration console and the Survey Administration console.

You must also have a clear definition of the users required for your implementation, based on the functions of Oracle Scripting required at the enterprise. All implementations require one or more administrative users which may have varying attributes. Required end users also vary; the Scripting Engine agent interface end user is an interaction center agent. The Scripting Engine Web interface end user may be customers accessing scripts through a Web browser, or enterprise employees or self-service Web application customers accessing scripts through

integrated Web applications such as Oracle iSupport or Oracle iStore. For detailed information, see Section 4.7.1.1, "Types of Users Required"). Implementation requirements for each varies accordingly.

The table below indicates the tasks and sequence required to implement Oracle Scripting to meet your requirements.

Task	Task Description	Required for	Consult documented tasks...
1	Create and administer Oracle Scripting administrators.	All	Section 4.2, "Creating an Administrator for Oracle Scripting"
2	Create and administer Oracle Scripting end users.	All	Section 4.3, "Creating an Agent" Section 4.5.3, "Administering the Guest User"
3	Implement the Scripting Engine agent interface.	Agent UI	Section 4.4, "Implementing the Scripting Engine Agent Interface"
4	Implement the Scripting Engine Web interface.	Web UI	Section 4.5, "Implementing the Scripting Engine Web Interface"
5	Implement the Oracle Scripting administrative interface.	All	Section 4.6, "Implementing Administrative Interfaces"
6	Implement the Survey administrative interface.	Web UI	Section 4.6, "Implementing Administrative Interfaces"

See Also

- Section 4.2, "Creating an Administrator for Oracle Scripting"

- Section 4.3, "Creating an Agent"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.6, "Implementing Administrative Interfaces"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.2 Creating an Administrator for Oracle Scripting

To create an administrator for Oracle Scripting, you must know all tasks the administrator will perform, and how Oracle Scripting will be used at the enterprise.

For implementation purposes, administrators typically require only an Oracle Applications user account and the appropriate responsibilities. In some cases, as

indicated below, administrator users require specific roles or group membership requirements. If you want administrator users to be super users with agent and administrator privileges (to perform work in CRM applications or in business applications for which campaign administration is required), you must also perform steps for creating an agent for this Oracle Applications user. See Section 4.3, "Creating an Agent".

Review the information below to determine which requirements apply to your administrator user(s). Then define users, assign responsibilities, and otherwise perform all steps as applicable.

> **Note:** Some tasks (such as granting roles for list and fulfillment administration, or adding JTF system-level properties) require existing JTF roles such as JTF_SYSTEM_ADMIN_ROLE and JTF_FM_ADMIN role. Your Oracle Applications login must already be granted JTF roles in order to assign them or to administer advanced system-level properties. For this purpose, if required, you can use the seeded sysadmin Oracle Applications user account. For more information, see Section 4.7, "Creating and Administering Oracle Scripting Users".

Assign Administrative Responsibilities to Trusted Users Only

The Scripting Administrator and Survey Administrator responsibilities are required to implement Oracle Scripting and administer Oracle Scripting for use at an enterprise, as described below.

Survey administrators have the ability to modify the properties of active survey campaigns, including changing active survey campaign status (cancelling or closing the campaign). If performed inadvertently, this could destroy the integrity of survey data.

In the same manner, Scripting administrators can modify, deploy, and delete deployed scripts, delete custom Java, and change specific Java mappings. If performed inadvertently, any of these tasks could cause loss of data and script functionality. Additionally, scripts with SQL code can access sensitive database tables. Resulting information, if misused, can introduce liability issues for the enterprise.

For these reasons, Oracle Corporation strongly recommends that only trusted users be provided with Scripting Administrator or Survey Administrator responsibilities.

Scripting Administrators

Since Oracle Scripting implementations typically require a customized script, and since the new Script Author Java applet and custom Java archive deployment and mapping capabilities are only accessible from the Scripting Administration console, all new or upgraded implementations require a Scripting administrator. Therefore, all administrators should be assigned the Scripting Administrator responsibility. See Section 4.3.1, "Assigning Scripting Administrator Responsibilities to a User".

Survey Administrators

Implementations requiring execution of a script in a Web browser must have an administrator user who is assigned the Survey Administrator responsibility. See Section 4.3.2, "Assigning Survey-Related Responsibilities to a User".

Fulfillment and List Management Administration

Implementations requiring execution of list-based surveys require additional roles, responsibilities, and must meet or bypass sales group membership requirements. See the following sections:

- Section 4.6.3.1, "Assigning Fulfillment Administrator Role to Survey Administrator"

- Section 5.3, "Granting JTF Roles"

- Section 5.2, "Bypassing the Sales Group Membership Requirement"

- Section 9.3, "Administering Lists in Oracle Marketing"

Consult the table below to determine which administrator user administration steps are required for your implementation. If your administrator needs agent functions, also consult Section 4.3, "Creating an Agent". Then perform all steps as applicable.

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
Create or modify Oracle Applications users	- Assign System Administrator responsibility - Access Forms-based applications as a system administrator	Section 4.7.2.4, "Defining Oracle Applications Users"

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
Create or modify an employee in the enterprise database	■ Assign appropriate HRMS responsibility (based on country) or CRM Resource Manager responsibility (if HRMS not installed) ■ Access HRMS or CRM Resource Manager as appropriate	Section 4.7.2.1, "Creating an Employee in the Enterprise Database"
Create a resource group in CRM Resource Manager	■ Assign CRM Resource Manager responsibility ■ Access CRM Resource Manager to create group and assign membership roles and members	Section 4.7.2.2, "Creating a Resource Group"
Import an employee as a CRM resource	■ Assign CRM Resource Manager responsibility ■ Access CRM Resource Manager to import employee	Section 4.7.2.3, "Importing a CRM Resource"
Access the Scripting Administration console to perform any of the following: ■ Launch the Script Author Java applet. ■ Create, modify, or deploy custom scripts using Script Author Java applet. ■ View or remove deployed scripts. ■ Deploy, update, or remove custom Java archives to the Applications database. ■ Define uploaded Java archives as global to apply to all deployed scripts. ■ Map uploaded Java archives to one or more specific scripts. ■ View and generate panel footprint reports.	■ Assign Scripting Administrator responsibility	Section 4.3.1, "Assigning Scripting Administrator Responsibilities to a User"

If administrator needs to do this...	Perform these steps...	Consult these documented tasks...
Access the Survey Administration console to administer survey resources	■ Assign Survey Administrator responsibility ■ Provide FTP access to applications server to deploy JSP resources to $OA_HTML and images to $OA_MEDIA as appropriate	Section 4.3.2, "Assigning Survey-Related Responsibilities to a User"
Access the Survey Administration console to perform any of the following: ■ Create survey campaigns and cycles ■ View or modify survey campaign details ■ Create additional cycles ■ Rename cycles ■ Define deployments ■ View or modify deployment details ■ Change or view deployment status	■ Assign Survey Administrator responsibility	Section 4.3.2, "Assigning Survey-Related Responsibilities to a User"
Manage survey campaign responses, including: ■ View individual Web interface responses ■ Send reminders manually to list members	■ Assign Survey Administrator responsibility	Section 4.3.2, "Assigning Survey-Related Responsibilities to a User"
Generate summarized Web interface reports or panel footprint reports	■ Assigning iSurvey User responsibility to access Concurrent Manager	Assigning iSurvey User Responsibility for Administering Concurrent Programs for Survey Execution

If administrator needs to do this...	Perform these steps...		Consult these documented tasks...
View, generate, or modify lists for list-based survey campaigns	■	Assign Oracle Marketing Super User responsibility	■ Assigning Oracle Marketing Responsibilities to a User
	■	Assign sales group membership and roles *or* bypass group membership requirements	■ Assigning Sales Group Membership and Roles to a User *or* Section 5.2, "Bypassing the Sales Group Membership Requirement"
			■ Section 9.3, "Administering Lists in Oracle Marketing"
Access the Invitations tab from the Survey Administration console to administer fulfillment documents for list-based survey campaigns	■	Assign JTF_SYSTEM_ADMIN_ ROLE role	■ Section 4.6.3.1, "Assigning Fulfillment Administrator Role to Survey Administrator"
	■	Assign JTF_FM_ADMIN role	■ Section 5.3, "Granting JTF Roles"

See Also

- Section 4.1, "Implementation Task Sequence"

- Section 4.3, "Creating an Agent"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.6, "Implementing Administrative Interfaces"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.3 Creating an Agent

Many CRM applications require users to be associated with CRM resource identification codes. This requires users to be defined as employees in the enterprise database, and to be imported as CRM resources. Oracle Corporation recommends that these steps be performed for all agent users as the first steps in agent creation.

You must define agent users that can log into an Oracle Applications session. Each agent user must be assigned the appropriate responsibilities to perform a particular set of functions. To define an agent for Oracle Scripting, you must know all tasks the agent will perform, and how Oracle Scripting will be used at the enterprise.

Additionally, applications that rely on campaign information (such as Oracle TeleSales, Oracle Collections, and Oracle Marketing) may require agents to have specific group membership and roles assigned to them.

Execution-level requirements also exist. For example, in order to launch a script from Oracle Collections, the script must be associated with a campaign. Thus, campaign management and administration is a prerequisite of using Oracle Scripting with Oracle Collections.

Review the information below to determine which requirements apply to your agent user(s). Then define users, assign responsibilities, and otherwise perform all steps as applicable.

The process of defining and administering users requires you to use an administrator user. For more information, see Section 4.2, "Creating an Administrator for Oracle Scripting".

Consult the table below to determine which agent user administration steps are required for your implementation. Then perform all steps as applicable.

If agent needs to do this...	Perform these steps...	Consult these tasks...
Use the Agent interface to execute scripts in stand-alone mode	■ Assign Scripting Agent or Scripting User responsibility	■ Section 4.7.2.4, "Defining Oracle Applications Users"
Use the Agent interface to execute scripts from Customer Support component of Oracle TeleService	■ Assign Customer Support responsibility ■ Optionally, create a relationship plan	■ Section 4.7.2.4, "Defining Oracle Applications Users"

If agent needs to do this...	Perform these steps...	Consult these tasks...
Use the Agent interface to execute scripts from business applications requiring campaign information (Oracle TeleSales, Oracle Collections)	■ Assign Scripting Administrator responsibility ■ Access CRM Resource Manager to provide agent with Sales and TeleSales roles ■ Access CRM Resource Manager to add agent to sales group ■ Access Oracle Marketing to create a campaign and a campaign schedule, associate a script to a campaign schedule, assign agents to a campaign schedule or vice versa	■ Section 4.3.1, "Assigning Scripting Administrator Responsibilities to a User" ■ Section 9.1.1, "Creating a Campaign" ■ Section 9.1.2, "Creating a Campaign Schedule" ■ Section 9.1.3, "Associating a Script to a Campaign Schedule" ■ Section 9.1.4, "Associating a Script to a Campaign" ■ Section 9.2.1, "Assigning Agents to a Campaign Schedule" ■ Section 9.2.2, "Assigning Campaign Schedules to Agents"
Use the Web interface to execute scripts from a self-service Web application in a Web browser	■ Assign self-service Web application responsibility	■ Section 4.3.2, "Assigning Survey-Related Responsibilities to a User"
Use the Web interface to execute surveys in a Web browser	■ Access an active survey URL	■ Scripting Engine Users
Use the Web interface to execute list-based surveys in a Web browser	■ Access an active survey URL from an e-mail message invitation or reminder	■ Scripting Engine Users

Oracle Scripting has varying requirements for administration, based on the components required for your implementation.

■ Since Oracle Scripting implementations typically require a customized script, and since the new Script Author Java applet and custom Java archive deployment and mapping capabilities are only accessible from the Scripting Administration console, all new or upgraded implementations require a Scripting administrator.

■ Implementations requiring execution of a script in a Web browser require a survey administrator.

- Implementations requiring execution of list-based surveys require additional roles, responsibilities, and group membership requirements.

- Implementations requiring customization of self-service Web applications need administrators of the appropriate application to customize the JSP pages to include survey URL links.

- Implementations using Oracle Applications on the UNIX platform have additional configuration steps based on platform and patch level.

- Implementations requiring multiple Apache listeners require Apache Web server administrators intimately familiar with Oracle Applications administration.

This section includes the following topics:

- Section 4.3.1, "Assigning Scripting Administrator Responsibilities to a User"

- Section 4.3.2, "Assigning Survey-Related Responsibilities to a User"

See Also

- Section 4.1, "Implementation Task Sequence"

- Section 4.2, "Creating an Administrator for Oracle Scripting"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.6, "Implementing Administrative Interfaces"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.3.1 Assigning Scripting Administrator Responsibilities to a User

The **Scripting Administrator** responsibility is required for a user to log into the Scripting Administration console, from which the Script Author Java applet can be launched, deployed scripts can be viewed and deleted, Java archives can be viewed, deleted, or mapped to deployed scripts, and agent interface reports can be generated and analyzed.

Use this procedure to create a Scripting administrator:

Assign the **Scripting Administrator** responsibility to users of the Scripting Administration console, as described in Section 4.7.2.4, "Defining Oracle Applications Users".

For testing purposes, you may want to use an agent user who can execute scripts after they are created and deployed. In this case, you will also want the **Scripting Agent, Vision Enterprises**, or **Scripting User** responsibility as well. Alternatively, you can provide the Scripting administrator with a responsibility to launch a business application such as Oracle TeleSales, Oracle Collections or the Customer Care component of Oracle TeleService.

Alternatively, if executing scripts in the Scripting Engine Web interface, you may want the same administrator user to have access to survey administration. In this case, follow all additional recommendations in Section 4.3.2, "Assigning Survey-Related Responsibilities to a User".

For a full list of responsibilities that may be required for Oracle Scripting, see Section 4.7.1.4, "Implementation Responsibility Matrix".

See Also

Section 4.3.2, "Assigning Survey-Related Responsibilities to a User"

4.3.2 Assigning Survey-Related Responsibilities to a User

Assign the appropriate Survey-related responsibilities to appropriate administrator users as described in Section 4.7.2.4, "Defining Oracle Applications Users".

Following is a summary of Oracle Applications responsibilities for using Survey:

- The **Survey Administrator** responsibility is required for a user to log into the Survey Administration console, from which survey campaigns are created, deployed, and managed.

- The **iSurvey User** responsibility is required to run Concurrent Manager for processing and scheduling of concurrent programs required to support survey operations.

- Optionally, you can assign the **Fulfillment Administrator** responsibility, required to access Oracle One-to-One Fulfillment, to an Oracle Applications user with the Survey Administrator responsibility. In this way, the same user would have the ability to administer all aspects of Oracle One-to-One Fulfillment required for list-based survey operations.

> **Note:** While assigning the Fulfillment administrator to a survey administrator user is not required, logging into Oracle Applications as a Fulfillment administrator *is* required in order to perform the implementation step described in Section 4.6.3.2, "Adding Survey Administrator User to a Fulfillment Group".

- Optionally, you can , required to access Oracle Marketing, to an Oracle Applications user with the Survey Administrator responsibility. In this way, the same user would have the ability to administer lists in Oracle Marketing for list-based survey operations.

For a full list of responsibilities that may be required for Oracle Scripting, see Section 4.7.1.4, "Implementation Responsibility Matrix".

See Also

Section 4.3.1, "Assigning Scripting Administrator Responsibilities to a User"

4.4 Implementing the Scripting Engine Agent Interface

The implementation tasks detailed in this section are required only for Scripting Engine operations for the Java-based interaction center agent application.

In addition to these tasks, you will also need to set up the Scripting Administration console as described in Section 4.6.2, "Setting Up the Scripting Administrative Interface".

This section includes the following topics:

- Section 4.4.1, "Assigning Scripting Engine-Related Responsibilities to a User"
- Section 4.4.2, "Establishing Profile Settings for the Agent Interface"
- Section 4.4.3, "Setting Session Idle Timeout Values for Apache JServ"

See Also

- Section 4.1, "Implementation Task Sequence"
- Section 4.2, "Creating an Administrator for Oracle Scripting"
- Section 4.3, "Creating an Agent"
- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.6, "Implementing Administrative Interfaces"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.4.1 Assigning Scripting Engine-Related Responsibilities to a User

Assign the appropriate Scripting Engine-related responsibilities to appropriate agent and administrator users as described in Section 4.7.2.4, "Defining Oracle Applications Users".

The agent interface for the Scripting Engine is a Java bean that runs in an Oracle form. A script can be executed in the agent application by any Oracle Applications user with the appropriate responsibility.

Scripts can be launched in the agent application either in stand-alone mode (recommended for testing), or from an integrated application. In either case, you must have the appropriate responsibility for that application assigned to your Oracle Applications login.

Following is a summary of Oracle Applications responsibilities for using the Scripting Engine in the agent application:

- The **Scripting User** or **Scripting Agent** responsibility is required for a user to launch scripts in stand-alone mode.

- The **Customer Support** responsibility is required to run the Oracle Support component of Oracle TeleService, from which a user can launch a script in the agent interface.

- The **TeleSales Agent** responsibility is required to run Oracle TeleSales, from which a user can launch a script in the agent interface.

- The **Collections Agent** responsibility is required for a user to launch Oracle Collections, from which a user can launch a script in the agent interface.

For a full list of responsibilities that may be required for Oracle Scripting, see Section 4.7.1.4, "Implementation Responsibility Matrix".

See Also

- Section 4.4.2, "Establishing Profile Settings for the Agent Interface"

- Section 4.4.3, "Setting Session Idle Timeout Values for Apache JServ"

4.4.2 Establishing Profile Settings for the Agent Interface

Setting system profiles determines the behavior of Oracle Applications. To implement the Scripting Engine agent interface, follow the procedure described in Section 5.4, "Setting System Profile Values" for the following system profile options.

See the Guidelines section below for information on these settings.

Step	Option	Level	Value
1	IES : IES ARCHITECTURE TYPE	Site	Apache Mid Tier/Servlet Architecture
2	IES : Scripting Panel Display Mode	Site	Display Multiple Panels at a time in Scripting window *or* Display Single Panel at a time in Scripting window
3	IES: PROXY SERVER NAME	Site or Application	<Name of Proxy server if used for Scripting Engine agent UI>
4	IES: PROXY SERVER PORT	Site or Application	<Port of Proxy server if used for Scripting Engine agent UI>
5	IES : Debug Mode	Site, Application, Responsibility, or User	Debug off *or* Debug on
6	Apps Servlet Agent	Site or Application[1]	<URL for the Servlet zone of the enterprise Apache Web server>
7	IES : Display Suspend Button on Script Frame	Site, Application, Responsibility, or User	False *or* True
8	IES : Initial Script Frame Size	Site, Application, Responsibility, or User	<Desired frame size, based on agent desktop resolution or overall requirements>

[1] If your implementation requires a separate Apache JServ listener established for Oracle Scripting, select the **Application** option. This may be required to avoid session timeout errors, as described in Section 4.4.3, "Setting Session Idle Timeout Values for Apache JServ".

Guidelines

Architecture Type: The only supported architecture type for new Scripting Engine 11.5.8 implementations is the Apache Mid-Tier/Servlet Architecture.

> **Note:** If Apache Mid Tier/ Servlet Architecture value is not in the list of values, and instead you see Two Tier Mode and Three Tier Mode, stop your work and see your systems administrator. This means your environment has not been upgraded beyond Oracle Applications 11.5.5. *Do not select Three Tier Mode.*

> **Note:** New Scripting Engine implementations with this profile option configured for the Caching Architecture *will not be supported.*

Scripting Panel Display Mode: There are two lookup values from which to choose: Display Multiple Panels at a time in Scripting window and Display Single Panel at a time in Scripting window. Using single-panel mode, only the current (active) panel will appear on the screen at any given time in the agent UI. Under multi-panel display mode, the active panel will have focus, but panels previously visited will appear in the UI, scrolling out of site. Panels that are not active are de-emphasized (they appear smaller and gray). Active panels display unformatted, spoken, instructional, and custom text in black, blue, magenta, and custom colors as designed. For more information, see the section "The Panel Display Area" in the Understanding Oracle Scripting > Understanding the Scripting Engine section of the User's Guide and online documentation.

Proxy Server Name: This references the name of any proxy server required for using Oracle Scripting, and enables the use of HTTPS. This is only required for Scripting Engine agent UI implementations using a Proxy server.

Proxy Server Port: This references the port number of the appropriate proxy server required for using Oracle Scripting, and enables the use of HTTPS. This is only required for Scripting Engine agent UI implementations using a Proxy server.

Debug Mode Type: Turning Debug mode on will log additional informational messages regarding Oracle Scripting and other operations. Scripting error message logs (at any time) and informational message logs (when debug mode is on) can be viewed by the CRM HTML-based server log viewer. This requires logging into CRM HTML applications as a user with the Interaction Center Server Log Viewer or Interaction Center Server Manager responsibility.

Apps Servlet Agent: Type the URL for the Servlet zone of the Apache JServ listener intended for Oracle Scripting in this field.

Syntax:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>
```

Example:

```
http://server1.yourcompany.com:7777/oa_servlets
```

This setting is required for Apache mid-tier architecture implementations of Oracle Scripting. This profile is populated with the servlet zone of the Apache Web server. If your implementation requires only one Apache Web server listener, this profile is set at the site level and is used for all applications. Some environments require multiple Apache Web server listeners. For example, if some applications require short session idle timeout settings and others (like Oracle Scripting) require longer idle timeout settings, additional listeners may be configured. In such cases, the URL of the alternate listener is set at the application level for this profile. The specified applications will then reference the alternate Apache JServ instance.

IES : Display Suspend Button on Script Frame: Deployed scripts which have the global Suspendable script property option checked (set to True) can suspend Oracle Scripting transactions when this option is set to True. If enabled, a Suspend button displays on the bottom of the script frame (next to the Disconnect button). This button is hidden if the profile is set to False (or remains null). Oracle Corporation recommends setting this option to True only (a) if the agent interface is used in stand-alone mode, or (b) when applications integrated with Oracle Scripting are updated to include the ability to resume the script from the business application. This can be set at the site, application, responsibility, or user level, as required.

IES : Initial Script Frame Size: This profile option controls the initial size of the Scripting Engine frame window in the Forms/Java agent user interface. Default size is 800 pixels in width and 600 pixels in height. Administrators can now select other options including 1024 X 768, 1280 X 1024, or Maximized. This can be set at the site, application, responsibility, or user level, as required.

Reference

- Section B, "Oracle Scripting Profile Options"
- Section 5.4, "Setting System Profile Values"

See Also

- Section 4.4.1, "Assigning Scripting Engine-Related Responsibilities to a User"
- Section 4.4.3, "Setting Session Idle Timeout Values for Apache JServ"

4.4.3 Setting Session Idle Timeout Values for Apache JServ

In the Oracle Scripting Apache mid-tier architecture, the Scripting servlet runs in an Apache JServ. Every Apache JServ has a session idle timeout setting that controls how long a servlet session can be idle before it is automatically timed out (expired and terminated) by the Apache JServ. This behavior ensures that abandoned sessions in the servlet are not left to consume resources unnecessarily. Sessions in the Apache JServ can be abandoned due to a variety of causes (end user walked away from the computer without logging out, network failure, etc.).

When a servlet session is active in the JServ, but is idle (has received no client requests) for longer than the specified idle timeout setting, the JServ will automatically time out or expire the session. In Scripting, the servlet session stores all of the state information of a running script (panels accessed, answers given, etc.). Therefore, a timeout of a Scripting servlet session while an agent is running a script is a fatal error, causing the agent to lose all data that was collected in that session of the script. When this occurs, the agent must close the Scripting window and start a new session of the script, recapturing all information. To avoid this problem, you should verify that the ZONE.PROPERTIES file for the appropriate Apache JServ instance configured for Oracle Scripting has an appropriate timeout setting, and modify this setting if required.

Oracle Scripting Timeout Requirements

Oracle Scripting requires longer Apache JServ session idle timeout settings than other types of applications using Apache JServ. Interaction centers generally use a relatively small number of dedicated Oracle Scripting agents, who will typically remain logged in throughout the course of the day. In between script transactions, Scripting agents may be idle (from an application perspective) for long periods of time (possibly hours). During an active script transaction, when the agent is conversing on the telephone with customers, agents will be generally more active, but will still have some idle periods, possibly lasting several minutes. *It is critical to ensure the Scripting session does not expire during a script transaction,* as all data and history collected during the script transaction is lost when a session times out. Due to the usage pattern for Scripting and the importance of keeping the session intact throughout an active script transaction, session timeout values for an Apache JServ instance that is running Scripting sessions should be verified and may need to be extended.

The session idle timeout setting for Apache JServ must be set to a large enough value that it will not cause session time-outs during a script transaction. Naturally, this value will vary based on your specific business needs, as well as agent behavior

and training, and the use of other applications relying on Apache JServ. In general, a value of 30 minutes is reasonable for typical Oracle Scripting configurations.

Oracle iStore Timeout Requirements

Oracle applications using Apache JServ that have many users and who log in for short periods of time throughout the day have different idle timeout requirements. For applications that follow this model (for example, Oracle iStore), the JServ should time-out idle sessions quickly (after a few minutes) so that resources are freed and available for other users.

Single Apache JServ Listeners Serving Multiple Application Types

Generally, it is not recommended to have the Apache JServ session idle timeout value set longer than 30 minutes if you have an application using the Apache JServ in which many users log in for short periods of time throughout the day, as well as other users who log in for extensive periods of time and have long idle periods.

Multiple Apache JServ Listeners

If your Apache is currently serving both types of applications, Oracle Corporation recommends configuring one Apache listener for each application type. Each Apache port/listener will have its own respective ZONE.PROPERTIES file, and can therefore support different session idle timeout settings per Apache listener. Thus, for example, if using Oracle iStore and Oracle Scripting, the ZONE.PROPERTIES file for the Apache listener configured for Oracle iStore should be set for a short session idle timeout, whereas the other should be set for a longer timeout for Scripting operations.

Using multiple Apache listeners may require changing the Apps Servlet Agent profile to specify a different Apache servlet zone at the Application level specifically for Oracle Scripting.

Use the following guidance to verify or set the session idle timeout to a value that will prevent any Scripting servlet session from timing out while an agent is in the middle of a script transaction.

Prerequisites

- This procedure is only required for Oracle Scripting release 11.5.6 or later implementations using the Scripting Engine agent interface in the Apache mid-tier architecture. This problem does not affect the Survey component of Oracle Scripting.

- At least one ZONE.PROPERTIES file must exist from appropriate installation of Apache Web server software installed during Oracle Applications implementation.

- As with all customization, you should back up the ZONE.PROPERTIES file before modification in the event of data failure.

- Modifications should occur at low-traffic times so the Web server can be taken offline momentarily.

- Stop the Apache Web server prior to modification.

Steps

1. Change to the directory where your Apache Web server is installed.

   Refer to Apache documentation or consult your Apache Web server administrator if you are not sure where to find this.

2. Change to the directory in which the ZONE.PROPERTIES file resides.

   This file is generally in the location <ORAHTTP_TOP>/Jserv/etc/ but may vary, depending on your specific Apache JServ configuration.

3. Open the ZONE.PROPERTIES file and locate the section of the file containing the session.timeout parameter.

   This is preceded by several commented lines (prepended with "#"). The section of the ZONE.PROPERTIES file should appear similar to the following:

```
# Set the number of milliseconds to wait before
# invalidating an unused session.
# Syntax: session.timeout=(long)>0
# Default: 1800000 (30 mins)
session.timeout=1800000
```

4. Verify that the value (in milliseconds) meets requirements for this Apache listener. If not, change the value after "session.timeout=" (adjust it higher or lower accordingly).

   > **Note:** There are 60,000 milliseconds in one minute.

5. Save your work.

6. Restart the Web server.

See Also

- Section 4.4.1, "Assigning Scripting Engine-Related Responsibilities to a User"

- Section 4.4.2, "Establishing Profile Settings for the Agent Interface"

4.5 Implementing the Scripting Engine Web Interface

Using the Scripting Engine Web interface, you can execute scripts in an Oracle Applications 11*i*-certified Web browser, either from a self-service Web application, or as an information-gathering survey.

This section includes the following topics:

Section 4.5.1, "Implementation Considerations"

Section 4.5.2, "Assigning Web Interface Responsibilities to a User"

Section 4.5.3, "Administering the Guest User"

Section 4.6.5, "Setting Display Server for UNIX Environments"

See Also

- Section 4.1, "Implementation Task Sequence"

- Section 4.2, "Creating an Administrator for Oracle Scripting"

- Section 4.3, "Creating an Agent"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.6, "Implementing Administrative Interfaces"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.5.1 Implementation Considerations

The following requirements are all factors in implementing the Scripting Engine Web interface:

- Active survey campaign requirements corresponding to a survey URL

- Appropriate survey administrator responsibilities, roles and requirements

- A valid Oracle Applications session

- Appropriate script end-user responsibility

- Oracle Applications platform and patch level

Active Survey Campaign and Survey URL Requirements

Executing a script using the Scripting Engine Web interface requires appropriate survey campaign information, which must be administered in the Survey Administration console.

To execute a script in a Web browser or to participate in a Web-based survey, the Scripting Engine Web interface end user typically clicks a hyperlink to access a survey URL. The survey URL references a specific active survey deployment (identified by the dID parameter) on the surveying enterprise's Apache Web server listener port. This may be the same Apache listener port established for all Oracle Applications at the enterprise, or it may reference an Apache JServ instance specifically set up for Oracle Scripting operations.

Survey Administration Responsibility, Roles and Requirements

The survey URL accessed by the script end user in the Web interface is generated when a survey administrator completes all requirements for a survey campaign in the Survey Administration console and makes the deployment active.

Administering survey campaign information is performed by an Oracle Applications user with the Survey Administrator responsibility. This administrator must have access to existing survey resources, survey campaign requirements (including start and end dates), and a valid, tested deployed script.

This administrator must have access to Oracle One-to-One Fulfillment master documents and lists created with Oracle Marketing. To access the Invitations tab from the Survey Administration console, this administrator user must be added to the appropriate fulfillment group, and requires two JTF roles (JTF_SYSTEM_ADMIN_ROLE and JTF_FM_ADMIN). This user will also require the Fulfillment Administrator responsibility if the same Oracle Applications user will also be the administrator for Oracle One-to-One Fulfillment.

Valid Oracle Applications Sessions

Accessing the script or survey in a Web browser associated with the survey URL requires a valid, authenticated Oracle Applications session by an Oracle Applications user typically associated with one or more responsibilities.

For individuals executing a script in a Web browser from an integrated application such as Oracle iSupport, the Oracle Applications login information for the current valid Oracle Applications session is used. The individual selects a link from the customized self-service Web application interface (e.g., Oracle iSupport) to take the survey, which starts immediately using current login authentication.

For individuals executing a script in a Web browser as a survey, the Oracle Applications session is initiated behind the scenes when the survey respondent clicks on the survey URL.

Script End User Responsibilities

Individuals executing a script in a Web browser from a self-service Web application such as Oracle iSupport will use the appropriate responsibility for that application.

Individuals executing a script in a Web browser as a survey use a guest user login. No responsibilities are typically associated with a guest user, although other setup steps are required. See Section 4.5.3, "Administering the Guest User" for more information.

Oracle Applications Platform and Patch-Level Requirements

■ Display server setup for UNIX servers

■ Manual configuration of XML SQL utility

References

■ Section 4.5.2, "Assigning Web Interface Responsibilities to a User"

■ Section 4.6.5, "Setting Display Server for UNIX Environments"

See Also

Section 4.5.2, "Assigning Web Interface Responsibilities to a User"

Section 4.5.3, "Administering the Guest User"

4.5.2 Assigning Web Interface Responsibilities to a User

Accessing the script or survey in a Web browser associated with the survey URL requires a valid, authenticated Oracle Applications session by an Oracle Applications user typically associated with one or more responsibilities.

For individuals executing a script in a Web browser from an integrated application such as Oracle iSupport, the Oracle Applications login information for the current valid Oracle Applications session is used, including appropriate iSupport responsibility. The individual selects a link from iSupport to take the survey, which starts immediately using current login authentication.

For individuals executing a script in a Web browser as a survey, an Oracle Applications session is initiated using a guest user login. No responsibilities are typically associated with a guest user, although other setup steps are required.

See Also

Section 4.5.1, "Implementation Considerations"

Section 4.5.3, "Administering the Guest User"

4.5.3 Administering the Guest User

When a script is executed in a Web browser as a survey (either by visiting a survey URL sent in an invitation or e-mail reminder, or when coming across the survey URL on an enterprise's Web site), an Oracle Applications session is initiated using a guest user login. This Oracle Applications login allows the respondent to participate in an online survey only, granting no other privileges in the Oracle Applications session. No responsibilities are typically associated with the guest user.

To enable the guest user, you must create a guest login (Oracle Applications user name and password) for the enterprise. Additionally, this login must be associated with guest user and guest password JTF system-level properties.

This section includes the following topics:

- Section 4.5.3.1, "Verifying or Creating a Guest User Login"
- Section 4.5.3.2, "Verifying or Configuring the Guest User"

See Also

Section 4.5.1, "Implementation Considerations"

Section 4.5.2, "Assigning Web Interface Responsibilities to a User"

4.5.3.1 Verifying or Creating a Guest User Login

Some enterprises may already have a guest user login (user name and password) established. Thus, before creating the Oracle Applications guest user and assigning a guest password, you should verify whether these requirements are already implemented.

Although this document uses IESGUEST or GUEST, there are no firm requirements for the guest user name. Likely user names for this guest user include IESGUEST, GUEST, and IBEGUEST. You can perform a wildcard search using a string such as %GUEST%. Case is not a factor for user names, as all entries are converted to upper case by the application (during administration, upon entry; at login, before submitting the request).

For more information, see Section 4.7.2.4, "Defining Oracle Applications Users".

Use this procedure to verify if the guest user login exists, or to create this account if required.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > User > Define** and click **Open**.

    The Users window appears.

2. From the **View** menu, select **Query By Example > Enter** or press **F11** on your keyboard.

3. In the User Name field, enter your search criteria.

    You can use the % wildcard character in your search.

    For example, type %guest%.

4. Execute the query by selecting **View > Query By Example > Run** or pressing **Ctrl-F11** on your keyboard.

    A successful query results in one or more found records.

5. Optionally, if the query is successful, review each record to determine if a guest user is defined. You can navigate between records by pressing the **Page Up**, **Page Down**, or up and down arrow keys on your keyboard.

6. If a guest user is found, ensure the following attributes are set for this user:

    a. In the Password Expiration area, ensure a suitable option is selected. The **None** option is recommended to avoid future issues with the guest user.

    b. In the Effective Dates area, the start date must be earlier than or equal to the SYSDATE, and the end date must be later than the SYSDATE or null. The null option is recommended to avoid future issues with the guest user.

    c. No responsibilities should be assigned to this user.

7. If the query is unsuccessful, change your search criteria and search again. If no guest user is found, create one now by performing the following:

   a. In the User Name field, enter an appropriate user name for the guest user.

      For example, type **IESGUEST** or **GUEST**.

   b. In the Password field, enter an appropriate password, and press the tab key.

      For example, type **guest**, **iesguest**, or **welcome**.

   c. In the Password field, retype the same password for confirmation, and press the tab key.

   d. In the Password Expiration area, select a suitable option. The **None** option is recommended to avoid future issues with the guest user.

   e. In the Effective Dates area, ensure the start date is earlier than or equal to an the SYSDATE, and the end date is later than the SYSDATE or null.

8. Optionally, in the Description field, type a description of this user.

9. Select **File > Save and Proceed** to save your work.

10. Continue your work or log out.

See Also

- Section 4.5.3.2, "Verifying or Configuring the Guest User"

4.5.3.2 Verifying or Configuring the Guest User

To configure a guest user, you must establish guest user name and guest password system-level properties for product code JTF (Java Technology Foundation). These properties contain the values of an Oracle Applications user name and password for a guest user.

System-level properties are stored in key value pairs. The key is the property name (for example, guest_username and guest_password), and the value refers to the Oracle Applications property value (the actual user name and password created for the guest user in Oracle Applications Forms-based applications).

Some enterprises may already have a JTF advanced system-level properties for a guest user name and password. Thus, before creating these properties, verify that they do not already exist.

Although this document uses guest_username and guest_password as key names, there are no firm requirements for these JTF property key names. Likely key names for the guest user and password include guest, guest_user, guest_username, and

guest_account. Bear in mind when viewing lists in the CRM Applications Administration console that upper-case letters are higher in display order than lower-case letters, and are considered to be different entries. It is recommended that you search for likely JTF system-level property keys in upper, lower, and mixed-case letters before assuming the user name and password keys do not exist. Otherwise you may end up with duplicate entries (for example, both guest_username and GUEST_USERNAME).

Use this procedure to verify if JTF advanced system-level properties for a guest user login exist, or to create such properties if required.

Prerequisites

- A login (user name and password) for the guest user must already exist, and you must know the password.

- To perform this procedure, your Oracle Applications user must have the appropriate responsibility assigned to provide administrative access to CRM HTML-based applications.

- To administer Oracle CRM HTML applications, including performing this task, your Oracle Applications user must be granted the JTF_SYSTEM_ADMIN_ROLE. If required, you can use the seeded sysadmin user name. For more information, see Section 5.3, "Granting JTF Roles".

Login

Log into Oracle HTML-based applications.

Responsibility

CRM HTML Administration or CRM ETF Administration

Steps

1. From the CRM Applications Administration console, click the Settings tab.

2. Click the System subtab.

3. From the side panel, under Properties, click **Advanced**.

   The Advanced - Properties page appears.

> **Note:** If this page does not appear, your login does not have the JTF_SYSTEM_ADMIN_ROLE role. Log out and log in with an appropriate user.

4. From the View list, select JTF.

   The Advanced - Properties page refreshes, listing all advanced properties set for JTF, if any.

5. Examine all existing JTF properties in the list to determine whether the guest user name and guest password JTF properties have already been created.

> **Note:** It is likely that not all JTF properties will appear on the first display page. To navigate back and forth through multiple pages of records, click the **First**, **Previous**, **Next**, or **Last** links as appropriate.

6. If you see a guest user name and password established as JTF properties in the list, no new administrative steps are required, and you can sign out of this administrative console by selecting **Sign Out**.

   If these properties do not exist, continue the following steps to create them.

7. If there are no JTF advanced system properties created for the guest user and guest password, click **Create**.

   The Create Key page appears. The values stored in this table are stored in key value pairs. Sequence is not relevant to this procedure.

8. In the **Key** text field, type an appropriate key for the guest user name.

   For example, type **guest_username**.

9. In the **Value** text field, type the value of a User ID that is designated for a guest login.

   For example, type **IESGUEST** or **GUEST**.

   CRM HTML-based applications use the term User ID for an Oracle Applications User Name. This is an Oracle Applications account that must already exist (administration steps performed using Forms-based applications) and must have *no responsibilities* associated with it.

10. When you are satisfied that the guest user which you have defined meets your implementation specifications, click **Update**.

The Advanced - Properties page refreshes.

11. Ensure the properties displayed are for JTF, and click the **Create** button.

   The Key Details page appears.

12. In the **Key** text field, type an appropriate key for the guest password.

   For example, type **guest_password**.

13. In the **Value** field, type the value of the corresponding password for the user identified by the User ID you defined in the previous step.

   For example, type **iesguest** or **welcome**.

14. When you are satisfied that the guest password you have defined meets your implementation specifications, click **Update**.

15. Continue your work or click **Sign Out** to exit Oracle Applications.

See Also

- Section 4.5.3.1, "Verifying or Creating a Guest User Login"

4.6 Implementing Administrative Interfaces

Oracle Scripting has two administrative interfaces: the Scripting Administration console and the Survey Administration console. Both are JSP-based HTML consoles accessed by administrators to control different aspects of Oracle Scripting administration. Some setup steps common to both administrator interfaces, some are specific to one interface, and others relate to the installation, patch level, or platform for Oracle Applications.

The Scripting Administration console provides the user interface with which script developers can launch the Script Author as a Java applet, and script administrators can administer Oracle Scripting files (deployed scripts and Java archives), as well as generate, view and analyze agent interface reports.

The Survey Administration console provides the user interface with which survey campaign administrators can administer survey resources; create and administer list- and non-list-based survey campaign requirements; and monitor individual responses.

Which Interfaces Do You Need?

- As of Oracle Scripting release 11.5.8 (Interaction Center Family Pack P and later), all implementations require the Scripting Administration console.

- Any implementation making use of the Scripting Engine Web interface to execute a script in a Web browser will also require the Survey Administration console.

System Profile Settings

Users of either administrative interface must have Oracle Applications profile settings appropriately established. The required profiles are described in each section below.

Reporting Implementation Requirements

To generate and view reports for analysis of survey information using Oracle Discoverer, you must perform certain post-installation implementation and administration steps, as outlined in Appendix C, **Oracle Discoverer Workbooks**.

Additionally, to generate and view the panel footprint report accessible from the Oracle Scripting Administration console, the following apply:

- Users of Oracle Applications installed between releases 11.5.1 and 11.5.4 must configure the Oracle XML SQL utility if this is not already configured.

- Users of Oracle Applications installed on a UNIX platform must set up a display X server.

Follow the procedures below to implement one or more administrative interfaces to meet the requirements of your implementation.

This section includes the following topics:

- Section 4.6.1, "Establishing Profile Settings"

- Section 4.6.2, "Setting Up the Scripting Administrative Interface"

- Section 4.6.3, "Setting Up the Survey Administrative Interface"

- Section 4.6.4, "Configuring the Oracle XML SQL Utility"

- Section 4.6.5, "Setting Display Server for UNIX Environments"

See Also

- Section 4.1, "Implementation Task Sequence"

- Section 4.2, "Creating an Administrator for Oracle Scripting"

- Section 4.3, "Creating an Agent"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.7, "Creating and Administering Oracle Scripting Users"

4.6.1 Establishing Profile Settings

Setting system profiles determines the behavior of Oracle Applications. When implementing Oracle Scripting, follow the procedure described in Section 5.4, "Setting System Profile Values" for the following system profile options. These include six JTF profiles establishing default behavior of JTF HTML-based applications, one JTF profile that enables searches in the new Survey Administration console, and one ICX profile establishing the default language.

See the Guidelines section below for information on these settings. Set the following profiles as indicated. Order below is recommended only; you can set these options in any sequence.

Step	Option	Level	Value	Required For
1	JTF_PROFILE_DEFAULT_APPLICATION	Site	519 (see note below)	All
2	JTF_PROFILE_DEFAULT_BLANK_ROWS	Site	3	All
3	JTF_PROFILE_DEFAULT_CSS	Site	JTFUCSS.CSS	All
4	JTF_PROFILE_DEFAULT_CURRENCY	Site	<Applicable currency. For example, USD for U.S. Dollars>	All
5	JTF_PROFILE_DEFAULT_NUM_ROWS	Site	10	All
6	JTF_PROFILE_DEFAULT_RESPONSIBILITY	Site or User	(see note below)	All
7	ICX: Language	User	As required. Example: American English	All

Guidelines

General: The settings above (such as default number of blank rows or default number of rows) are provided as recommendations, typically for the site level. If there are business rules or requirements in your environment for other settings, let those requirements take precedence, or set profile settings at lower levels for the specified users.

Default Application: This setting assumes this Survey administrator user does not already have other functions in Oracle CRM technology foundation (JTT) HTML-based applications. Should other responsibilities exist for this user, confer

with your systems administrator or the specific user to determine whether to modify the existing setting, or set this at the user level. 519 as indicated above is the product code for Oracle Scripting. *This setting is not environment-specific.*

Default Responsibility: This setting determines the responsibility that is displayed when a user logs into CRM HTML-based applications. If no other responsibilities existed for this user, select Survey Administrator or Scripting Administrator, based on the requirements of your administrative users.If this user already has other responsibilities, it may be a requirement to leave this setting, or set this at the user level. The value for this setting may be specific to each environment. For more information, refer to Section B.3, "Oracle CRM Technology Foundation (JTT) Profile Options".

ICX Language: This setting establishes language for the Internet Cartridge Exchange for Web-based transaction settings. This value is selected from the appropriate language from the FND_LANGUAGES table, and is typically set at the user level.

References

- Section B.3, "Oracle CRM Technology Foundation (JTT) Profile Options"
- Section 5.4, "Setting System Profile Values"

See Also

- Section 4.6.2, "Setting Up the Scripting Administrative Interface"
- Section 4.6.3, "Setting Up the Survey Administrative Interface"
- Section 4.6.4, "Configuring the Oracle XML SQL Utility"
- Section 4.6.5, "Setting Display Server for UNIX Environments"

4.6.2 Setting Up the Scripting Administrative Interface

The Scripting Administration console provides users who are assigned the Scripting Administration responsibility with the user interface from which the Script Author can be launched as a Java applet. Additionally, from this user interface, you can view and remove scripts deployed to the database; you can view, upload, and map custom Java archives to any script deployed to the database; and you can generate, view and analyze panel footprint reports.

Prior to this release, the Survey Administration console included a Quick Find menu feature to locate survey campaigns, cycles, or deployments. For users upgrading Oracle Applications, this menu continues to display in any CRM

Technology Foundation (JTT technical stack) application associated with product code IES (Oracle Scripting), even though it is obsolete. (This menu will not appear in new implementations.) Thus, for upgrading implementations prior to Interaction Center Family Pack Q or release 11.5.9, this menu will appear in the Scripting Administration console. You can remove the Quick Find menu from Scripting applications manually, after which it will not appear. See Removing the Quick Find Menu.

Other than performing installation, patch level, or platform-specific setup steps, no other tasks are required to implement the Scripting Administrative Interface.

See Also

- Section 4.6.1, "Establishing Profile Settings"
- Section 4.6.3, "Setting Up the Survey Administrative Interface"
- Section 4.6.4, "Configuring the Oracle XML SQL Utility"
- Section 4.6.5, "Setting Display Server for UNIX Environments"

4.6.3 Setting Up the Survey Administrative Interface

The Survey Administration console provides users assigned the Survey Administration responsibility with the user interface from which survey campaigns are created and administered. Active survey campaigns are required in order to execute a script in a Web browser using the Scripting Engine Web interface.

From this user interface, you can administer survey resources, define or change survey campaign information and status, access Oracle Marketing and Oracle One-to-One Fulfillment functionality required for targeted (list-based) survey deployments, and view responses received from users who executed the survey campaign script in a Web browser.

Other than performing installation, patch level, or platform-specific setup steps, no other tasks are required to implement the Survey Administrative interface for use with standard (non-list-based) survey campaigns.

Implementations using targeted deployments require the fulfillment administrator role to be assigned to the survey administrator, and require the survey administrator to be a member of a fulfillment group.

Tasks

Tasks include:

See Also

-
-
-
-

4.6.3.1 Assigning Fulfillment Administrator Role to Survey Administrator

For list-based survey implementations, a user with the Fulfillment Administrator and Survey Administrator responsibilities must also be assigned the Fulfillment Administrative role (JTF_FM_ADMIN). This role provides access to Oracle One-to-One Fulfillment's user interface from within the Survey Administration console.

To perform some Oracle HTML-based applications administration, including this task, your Oracle Applications user must be granted the JTF_SYSTEM_ADMIN_ ROLE. If required, you can use the seeded sysadmin user name. For more information, see Section 5.3, "Granting JTF Roles".

This step is only required for implementations requiring targeted (list-based) survey deployments, to allow access to the Invitations tab. If you only wish to access Oracle One-to-One Fulfillment from within that application's user interface (and not from the Invitations tab in the Survey Administration console), then perform this step for the Oracle Applications user assigned the Fulfillment Administrator responsibility.

> **Note:** This step is not required for standard (non-list-based) survey campaign implementations, which will not be leveraging Oracle One-to-One Fulfillment capabilities.

Guidelines

- The user with the Survey Administrator and Fulfillment Administrator responsibilities must be an employee in the enterprise database, must be available as a CRM resource, and must be associated with an Oracle Applications login.

- The Oracle Applications login must have the CRM HTML Administrator or CRM ETF Administration responsibility assigned to provide access to CRM HTML-based applications as an administrator with full privileges. This includes establishing the appropriate set of system profiles for site and user levels as discussed in previous implementation tasks.

- Oracle One-to-One Fulfillment does not need to be implemented at the enterprise to perform this step. Note, however, that Oracle One-to-One Fulfillment must be implemented in order to *use* the survey administration component of Oracle Scripting for list-based implementations, *including for testing*.

See Also

- Section 4.6.3.2, "Adding Survey Administrator User to a Fulfillment Group"

4.6.3.2 Adding Survey Administrator User to a Fulfillment Group

This task is only required for list-based implementations, and should be performed once per Survey administrator user. However, this step is not required for non-list-based survey campaign implementations, which will not be leveraging Oracle One-to-One Fulfillment capabilities.

Use this procedure to add the Survey Administrator user to a Fulfillment group.

Prerequisites

- An Oracle HRMS employee must already exist in the enterprise, must be available as a CRM resource, and must be associated with an Oracle Applications login.

- An Oracle Applications user account must already be assigned both the Survey Administrator responsibility and the Fulfillment Administrator responsibility.

- An Oracle One-to-One Fulfillment group must exist in order to perform this step.

- Oracle One-to-One Fulfillment does not need to be fully implemented at the enterprise to perform this step, as long as a Fulfillment group has been created. Note, however, that Oracle One-to-One Fulfillment must be implemented in order to *use* the survey component of Oracle Scripting for list-based implementations, *including for testing*.

Login

Log into Oracle HTML-based applications.

Responsibility

Fulfillment Administrator

Steps

1. From the Fulfillment Administration console, click the Group tab.

   The Groups page appears. A hyperlinked list of defined group names (if any) appear in a table.

   > **Note:** If no group names appear in the list, you must first create a valid Fulfillment group as documented in *Oracle Oracle One-to-One Fulfillment Implementation Guide*.

2. If you want to add the survey administrator user to an existing fulfillment group, do the following:

   a. Click on the appropriate Group Name hyperlink.

      The Group Detail page appears.

   b. In the Name text field of the Agents table, add the Oracle Applications user name of the agent (survey administrator) whom you wish to add to the Fulfillment group, and click **Go**.

      The Select Agent page appears.

   c. Click the appropriate hyperlinked user name in the Agent User Name column.

      If necessary, refine your search for the appropriate user name in the Select Agent text field and click **Search**. You can narrow your criteria using the % wildcard character.

      The Group Detail page refreshes. The user name you selected now appears in the Agents list.

   d. Click **Update** to confirm your selection.

      The Groups page refreshes.

      Perform the remaining steps to verify that this implementation task was successful.

   e. Select the same Group Name hyperlink from the Groups page.

      The Group Detail page appears.

f. Verify that the survey administrator user name appears in the list of agents.

3. If you want to add the survey administrator user to a new fulfillment group, do the following:

a. Click **Create**.

The Create Group page appears.

b. In the Group Name text field, enter an appropriate unique group name.

c. Optionally, in the Description text area, enter a description of this group.

d. From the Server Name list, select an appropriate server on which this Fulfillment group will be created.

e. In the Name field under the Agents area, enter search criteria for the first agent whom you wish to add to this Fulfillment group and click **Go**.

■ The name criteria should specify the Oracle Applications user name.

■ Search criteria is case-sensitive.

■ You can use the % wildcard character.

The Select Agent page appears.

f. Click the appropriate hyperlinked user name in the Agent User Name column.

If necessary, refine your search for the appropriate user name in the Select Agent text field and click **Search**. You can narrow your criteria using the % wildcard character.

The Create Group page refreshes. The user name you selected now appears in the Agents list.

g. Repeat steps e and f until you have added all agents whom you wish to add to this Fulfillment group.

h. Click **Create** to save this group definition with the group members specified.

Perform the remaining steps to verify that this implementation task was successful.

i. Select the same Group Name hyperlink from the Groups page.

The Group Detail page appears.

j. Verify that the specified user names appear in the list of agents.

4. Continue your work, or click **Sign Out** to exit Oracle Applications.

See Also

- Section 4.6.3.1, "Assigning Fulfillment Administrator Role to Survey Administrator"

4.6.4 Configuring the Oracle XML SQL Utility

Enterprises that implemented Oracle Applications using Rapid Install 11.5.1 through 11.5.6 that wish to use reporting in the Scripting or Survey Administration consoles must manually configure the Oracle XML SQL utility. Oracle Applications implementations using Rapid Install 11.5.7 or later include automatic configuration of the Oracle XML SQL utility and do not require this task to be performed.

Use this procedure to obtain the appropriate files from your Web server, locate them appropriately, and modify the classpath in the JSERV.PROPERTIES file used by the servlet engine in Oracle HTTP Server.

> **Note:** There is more than one way to specify a class path. *If the method described below is not supported by your enterprise implementation,* consult with your Apache Web server administrator to determine the appropriate method for your environment. This may include identifying the classpath in a control file or another configuration.

Login and Responsibility

You must have physical or Telnet access to the Web server, and have applmgr or sysadmin privileges. Use the appropriate login for these privileges in your environment.

Prerequisites

- Your web server must be configured to use JDK 1.2.2 or above. If you are using a JDK version below JDK 1.2.2, you can upgrade to the most recently certified JDK version with Oracle Applications using Oracle*MetaLink* Note 130091.1, titled *Upgrading to JDK 1.3 with Oracle Applications 11i.*

- You must retrieve the necessary Java libraries from the $COMMON_TOP/util directory of your web server to configure your web server properly. Obtain Xsu12.zip (for 11.5.1 installations) or XSU12_ver1_2_1.zip (for 11.5.2 through 11.5.5 installations).

Steps

1. Connect using Telnet to the enterprise system.

2. Change to the directory where your Oracle HTTP Server is installed. Refer to the Appendix of your Oracle Applications Installation manual if you are not sure where to find this.

3. Create a directory called **xsu**.

4. If the Java library you obtained is xsu12.zip, extract this file using the command:

   ```
   unzip $COMMON_TOP/util/Xsu12.zip OracleXSU/lib/oraclexmlsql.jar
   ```

   If the Java library you obtained is XSU12_ver1_2_1.zip, extract this file using the command:

   ```
   unzip $COMMON_TOP/XSU12_ver1_2_1.zip OracleXSU12/lib/xsu12.jar
   ```

5. Copy the file you just extracted to the new xsu directory.

   For example, on UNIX, you would execute one of the following commands:

   ```
   cp OracleXSU/lib/oraclexmlsql.jar xsu/
   ```
   or

   ```
   cp OracleXSU12/lib/xsu12.jar xsu/
   ```

6. Remove the OracleXSU or OracleXSU12 directory created by the unzip command.

7. Change to the directory in which the JSERV.PROPERTIES file resides.

   > **Note:** The actual name of this file may include an underscore and the Apache JServ port (for example, JSERV.PROPERTIES_9404). For the purposes of this document, the Apache Web server configuration file will be referred to generically as the JSERV.PROPERTIES file. If you have any difficulties locating the file, consult with your Apache Web server administrator.

   The location of this file is environmentally dependent. It *may* be in one of the following paths:

   <ORAHTTP_TOP>/Jserv/etc/jserv.properties

   <ORAHTTP_TOP>/apache39/conf

The ORAHTTP_TOP variable may not be set for your environment, in which case the path will be fully qualified to identify the entire physical path in the file system. If your configuration is different, consult with your Apache Web administrator to locate this directory.

8. Load the JSERV.PROPERTIES file into a text editor.

9. Look in the JSERV.PROPERTIES file for several lines beginning with **wrapper.classpath=**.

   This is the section in the file that sets the CLASSPATH used by the web server.

10. If the Java library you obtained is **xsu12.zip**, add the following line to the JSERV.PROPERTIES file to include the XML SQL Utility libraries in your CLASSPATH:

```
wrapper.classpath=[ORAHTTP_TOP]/xsu/oraclexmlsql.jar
```

   If the Java library you obtained is **XSU12_ver1_2_1.zip**, add the following line to the JSERV.PROPERTIES file to include the XML SQL Utility libraries in your CLASSPATH:

```
wrapper.classpath=[ORAHTTP_TOP]/xsu/xsu12.jar
```

   Note: Replace [ORAHTTP_TOP] with the physical path where your Oracle HTTP Server is installed.

11. Save the JSERV.PROPERTIES file, and exit the text editor.

   Your web server is now configured to use the Oracle XML SQL Utility.

Guidelines

- Back up your JSERV.PROPERTIES file prior to customization. After configuring and testing, also back up your customized JSERV.PROPERTIES file in a separate location.

   Note: When you apply a patch, any customizations you have made to the JSERV.PROPERTIES file *may be lost*. In this case you will need to restore customizations to this file after patching.

See Also

- Section 4.6.1, "Establishing Profile Settings"

- Section 4.6.2, "Setting Up the Scripting Administrative Interface"

- Section 4.6.3, "Setting Up the Survey Administrative Interface"

- Section 4.6.5, "Setting Display Server for UNIX Environments"

4.6.5 Setting Display Server for UNIX Environments

Oracle Applications can dynamically generate and cache images with embedded text at runtime. These dynamic images are used, for example, to create buttons and tab menu bars displayed throughout various HTML-based transaction pages.

JFC/SWING and AWT (graphics libraries that are standard components within Java) require access to the native library of the operating system to process graphics (e.g., to display a window, paint a picture, generate a GIF image, and so on).

The reporting functionality for Oracle Scripting (the panel footprint report available from the Reports tab of the Scripting Administration console) employs JFC/SWING and AWT on the middle tier, where GIF images are processed in support of dynamically generated reports with graphics, and to render a script in a Web browser using HTTP.

Java AWT on Windows servers (Windows NT, Windows 2000) does not need a display server to render the graphics dynamically. However, in order for Java AWT to perform dynamic image generation on UNIX, you must specify an X Server that will be used as a display server to generate dynamic images. Images are generated and cached on the file system. Once these images have been cached, they need not be generated again. For this reason, the display server does not need to be a high-powered machine, nor is it required to be a dedicated server. However, it must be accessible by the Apache server that will call it.

To ensure graphics display functions in all UNIX environments, there are two processes you should perform in a UNIX environment:

- Modify JSERV.PROPERTIES to implement the Display Server identifier

- Reference the X Server in the apachectl or jservctl file (whichever is used in your environment to start the JServ).

Prerequisites

You must have an X Server up and running, on the network and accessible from the Apache JServ machine.

Steps

1. Access the JSERV.PROPERTIES file.

   This is typically located at <ORAHTTP_TOP>/Jserv/etc/jserv.properties on the Web server.

2. Make the following changes to implement the Display Server identifier.

   This can be done through xhost +, or through a more secure xauth UNIX command.

   k. Add the following DISPLAY parameter immediately below the wrapper.bin variable.

   ```
   wrapper.env=DISPLAY=<xserver-hostname>
   <xserver-displayport>
   ```

   l. Replace <xserver-hostname> and <xserver-displayport> with the machine name and port number where the X Server is running.

   Example:

   ```
   wrapper.env=DISPLAY=myxserver.mycorp.com:0.0
   ```

   Note: You can also reference a Windows NT machine that is running a UNIX emulator such as Exceed. See the man (UNIX manual) pages for xhost and xauth for more information.

3. Reference the X Server in the apachectl or jservctl file:

   Example:

   ```
   export DISPLAY=myxserver.mycorp.com:0.0
   ```

   Note: Whichever file is used to start the JServ for your environment (apachectl or jservctl) should be modified to reference the X server.

Guidelines

- Establishing an X Server as a display server is no longer required for UNIX systems in order for survey respondents or self-service Web application users to participate in a survey or script using a Web browser.

- Establishing an X Server as a display server is no longer required for UNIX systems using the Survey Administration console.

- This step is still required in order to view reports from the Scripting Administration console.

References

- Section 11.4, "Testing Display Server for UNIX Environments"

- Section 11.3.3.2, "Guest User Verification Steps"

See Also

- Section 4.6.1, "Establishing Profile Settings"

- Section 4.6.2, "Setting Up the Scripting Administrative Interface"

- Section 4.6.3, "Setting Up the Survey Administrative Interface"

- Section 4.6.4, "Configuring the Oracle XML SQL Utility"

4.7 Creating and Administering Oracle Scripting Users

This section provides information on the users created for Oracle Scripting and on suggested approaches for creating and administering Oracle Applications users.

This section includes the following topics:

- Section 4.7.1, "Understanding Oracle Scripting Users"

- Section 4.7.2, "Creating and Administering Users"

See Also

- Section 4.1, "Implementation Task Sequence"

- Section 4.2, "Creating an Administrator for Oracle Scripting"

- Section 4.3, "Creating an Agent"

- Section 4.4, "Implementing the Scripting Engine Agent Interface"

- Section 4.5, "Implementing the Scripting Engine Web Interface"

- Section 4.6, "Implementing Administrative Interfaces"

4.7.1 Understanding Oracle Scripting Users

Prior to creating and administering users for Oracle Scripting, you must understand the types of users required, the applicable responsibilities to assign to the appropriate Oracle Applications login, and the specific tasks to accomplish.

This section includes the following topics:

- Section 4.7.1.1, "Types of Users Required"

- Section 4.7.1.2, "Administrator Users"

- Section 4.7.1.3, "Script End Users"

- Section 4.7.1.4, "Implementation Responsibility Matrix"

- Section 4.7.1.5, "Road Map to User Creation and Administration"

See Also

Section 4.7.2, "Creating and Administering Users"

4.7.1.1 Types of Users Required

From an Oracle Applications perspective, you must define users that can log into an Oracle Applications session, and assign each user with the appropriate responsibility or responsibilities to perform a particular set of functions.

At minimum, for each implementation, you will need to define two types of users:

- An administrator, and

- A script end user.

  Script end users will use one of two Scripting Engine interfaces: the agent interface (a Java-based user interface for interaction center agents) or the Web interface (running a script in a Web browser, based on established survey campaign requirements).

The responsibilities required to be associated with each user type depends on how Oracle Scripting will be implemented at the enterprise, and which Oracle Applications functions will be required.

There are three specific tasks associated with creating and administering users:

1. Create employees in the enterprise database.

2. Create (import) employees as a CRM resource.

3. Define Oracle Applications users (selecting user names and passwords, assigning responsibilities, and associating the user with an employee in the enterprise database).

Script end users who are survey respondents have the fewest setup requirements.

See Also

- Section 4.7.1.2, "Administrator Users"

- Section 4.7.1.3, "Script End Users"

- Section 4.7.1.4, "Implementation Responsibility Matrix"

- Section 4.7.1.5, "Road Map to User Creation and Administration"

4.7.1.2 Administrator Users

For the purposes of Oracle Scripting implementation and use, administrators fall into three categories: system administrators, script administrators, and survey campaign administrators. The first two categories are required for all implementations, whereas survey administrators are required for any implementation in which scripts are intended to be executed in a Web browser. You can establish the same individual with all administration privileges or use various administrative users.

This section includes the following topics:

- Section 4.7.1.2.1, "System Administrators"

- Section 4.7.1.2.2, "Script Administrators"

- Section 4.7.1.2.3, "Survey Administrators"

See Also

- Section 4.7.1.1, "Types of Users Required"

- Section 4.7.1.3, "Script End Users"

- Section 4.7.1.4, "Implementation Responsibility Matrix"

- Section 4.7.1.5, "Road Map to User Creation and Administration"

4.7.1.2.1 System Administrators System administrators must have access to functions that the typical interaction center agent will not, and administers users and applications across the broad spectrum. For example, one function of a system

administrator is to create agent users. This necessitates the creation of an administrative user as the first step.

4.7.1.2.2   Script Administrators Script administrators will access the Scripting Administration console, an HTML user interface that provides access to the Script Author Java applet, allows administration of Oracle Scripting files, and provides access to agent interface report generation and analysis. Thus, individuals who develop scripts using the Script Author component require access to this administrative UI.

4.7.1.2.3   Survey Administrators Survey administrators will access the Survey Administration console, an HTML user interface that provides the ability to establish survey campaigns and to administer list-based survey campaigns using Oracle One-to-One Fulfillment and Oracle Marketing functionality. This console also provides access to survey response and agent interface summary report generation and analysis. Thus, individuals who monitor campaign status for ongoing active survey campaigns or who analyze past campaigns require access to this administrative UI.

To successfully implement Oracle Scripting, you must create at least one administrator user. If you do not want one user account to have access to multiple Oracle applications, you can create several administrator users and assign each the appropriate responsibility.

Administrator users can be created by any Oracle Applications user account with the System Administrator responsibility, with one exception. If implementing the survey component, the administrator user must be granted the JTF_SYSTEM_ ADMIN_ROLE and may require the JTF_FM_ADMIN role. Your Oracle Applications login must already be granted JTF roles in order to assign them. For this purpose, you can use the seeded sysadmin Oracle Applications user account.

For implementation purposes, the administrator may require only an Oracle Applications user account. However, if the administrator user is an enterprise super user that will perform work in CRM applications in addition to performing administrative tasks with the same login, or if the administrator user must access business applications for which campaign administration is required, you must perform all three user creation and administration tasks for the administrator.

4.7.1.3 Script End Users

This section includes the following topics:

■   Section 4.7.1.3.1, "Agent Users"

- Section 4.7.1.3.2, "Self-Service Web Application Users"

- Section 4.7.1.3.3, "\Survey Respondents"

See Also

- Section 4.7.1.1, "Types of Users Required"

- Section 4.7.1.2, "Administrator Users"

- Section 4.7.1.4, "Implementation Responsibility Matrix"

- Section 4.7.1.5, "Road Map to User Creation and Administration"

4.7.1.3.1 Agent Users For implementation purposes, the Oracle Scripting agent may require only an Oracle Applications user account if using Scripting in stand-alone mode. However, agent users typically have access to at least one business application, requiring these users to be members of the enterprise database. Additionally, some business applications require group membership, necessitating users to be imported into CRM Resource Manager. Therefore, it is recommended that you perform all three user creation and administration tasks for agent users.

4.7.1.3.2 Self-Service Web Application Users Self-Service Web application users typically execute a script by selecting a survey URL link added to the customized Web application. Thus, they gain access to an Oracle Applications session prior to executing the script (when logging into Oracle Applications to use the integrated self-service Web application).

4.7.1.3.3 \Survey Respondents Survey respondents who are not using self-service Web applications typically gain access to an Oracle Applications session by accessing a survey URL. This URL provides access to a guest user with script execution privileges only.

See Also

Section 4.7.1.4, "Implementation Responsibility Matrix"

Section 4.7.1.5, "Road Map to User Creation and Administration"

4.7.1.4 Implementation Responsibility Matrix

A matrix of responsibilities that may be required for various users is included below. The matrix includes user type (administrator or agent), responsibility name, the function provided by the responsibility, and the specific component that may require the responsibility.

Not all responsibilities listed in the matrix for each user type is necessarily required. For example, if Oracle Human Resources Management System is not installed, the HRMS Manager responsibility is not required. Similarly, for survey implementations, Oracle Marketing and Oracle One-to-One Fulfillment administration responsibilities are only required for list-based survey implementations.

Review the specific requirements for your implementation. Each implementation or administration task in this document lists the responsibility required to perform that task.

User Type	Responsibility	Function	For Component(s)
Administrator	System Administrator	■ Create and administer Oracle Applications users ■ Establish or modify system profile settings	All
Administrator	HRMS Manager	■ Create employees in the enterprise human resources database, if Oracle Human Resources Management System (HRMS) is licensed and installed at the enterprise.	All
Administrator	CRM Resource Manager	■ Create employees in the enterprise human resources database, if HRMS is not installed at the enterprise. ■ Import employees from the enterprise human resources database as CRM resources	All
Administrator	Scripting Administrator	Access the Scripting Administration console. This is required to: ■ Launch the Script Author Java applet ■ Deploy scripts associated with an Oracle Applications user name, so the script can be mapped to specific custom Java archives ■ View or remove scripts deployed using the Script Author Java applet ■ View, upload, update, or remove custom Java archives to the database ■ Map custom Java archives to specific scripts ■ Categorize custom Java archives as global to apply to all deployed scripts ■ Generate and view panel footprint reports	All

User Type	Responsibility	Function	For Component(s)
Administrator	Survey Administrator	Access the Survey Administration console. This is required to: ■ Define survey resources (headers, final pages, error pages in JSP format) ■ Create survey campaigns ■ Define survey cycles ■ Define or deploy survey deployments ■ View specific responses provided by survey respondents to a previously or currently active deployment	Survey
Administrator	iSurvey User	Schedule or run concurrent programs.	Survey
Administrator	Fulfillment Administrator	■ Access the Fulfillment Administration console. ■ Set up or change Fulfillment groups ■ Set up or change mail servers associated with Fulfillment groups ■ View failed Fulfillment requests after concurrent programs succeed in passing requests to the Fulfillment server.	Survey These Fulfillment Administrator tasks are required for implementations using list-based survey campaigns
Administrator	Oracle Marketing Super User	■ Access the Marketing Administration console. ■ Establish campaigns, assign agents to campaigns, assign scripts to agents or scripts to campaigns ■ Create, import, modify lists	Scripting Engine for agent application If Oracle TeleSales or Oracle Collections agents must launch a specific script, the script must be associated with a campaign or campaign schedule. Survey For list-based survey campaigns, lists must be created and administered.
Administrator	Interaction Center Server Log Viewer	Access the Call Center Technologies Administration console strictly to access the Log Messages tab for access to server logs and informational messages.	Scripting Engine for agent interface or Web interface

User Type	Responsibility	Function	For Component(s)
Administrator	Interaction Center Server Manager	Access the Call Center Technologies Administration console to access server groups, nodes, logs, and to obtain and set up ICSM server process for interaction center administration.	Scripting Engine for agent interface or Web interface
Administrator	Call Center HTML Administration	Access the Call Center Technologies Administration console to access server logs, set up interaction center servers and server groups, route points, classifications, and UWQ media actions.	Scripting Engine for agent interface or Web interface
Administrator	CRM HTML Administration or CRM ETF Administration	Administer Oracle CRM HTML applications, including verifying or adding Guest User JTF system-level properties so that survey respondents can start an Oracle applications session.	Survey
Agent	Scripting User or Scripting Agent	Launch scripts in "stand-alone mode" from Oracle Forms, typically to test deployed scripts.	Scripting Engine for agent interface
Agent	Customer Support	For agents accessing Oracle Scripting from within the integrated Oracle TeleService application.	Scripting Engine for agent application
Agent	TeleSales Agent	For agents accessing Oracle Scripting from within the integrated Oracle TeleSales application.	Scripting Engine for agent application
Agent	Collections Agent	For agents accessing Oracle Scripting from within the integrated Oracle Collections application.	Scripting Engine for agent application

See Also

- Section 4.7.1.1, "Types of Users Required"

- Section 4.7.1.2, "Administrator Users"

- Section 4.7.1.3, "Script End Users"

- Section 4.7.1.5, "Road Map to User Creation and Administration"

4.7.1.5 Road Map to User Creation and Administration

Order and Timing of User Type Creation

1. Create and administer one or more administrator user with the appropriate responsibilities for your implementation.

Administrator users must be created first, and are used to accomplish all other implementation and administration steps. Use the Implementation Responsibility Matrix above as a guideline.

2. Optionally, *c*reate and administer one or more agent user with the appropriate responsibilities for your implementation.

The order in which agent users are created is at your discretion. You may choose to create Oracle Applications users for each agent immediately following the creation of an administrative user, assuming you have all information required. Alternatively, agent users can be created immediately prior to use of an agent login for administration or testing purposes.

Use the Implementation Responsibility Matrix above as a guideline. Specific steps are detailed in *Oracle Interaction Center Server Manager Implementation Guide Release 11i*.

Method of Creating and Administering Users

1. The first task for user creation and administration is to create employees in the employee database.

   a. If HRMS is installed, you must use Oracle HRMS to create and administer employees. This requires the HRMS Manager responsibility.

   b. If HRMS is *not* installed, you must use Oracle CRM Resource Manager to create and administer employees. This requires the CRM Resource Manager responsibility.

2. After creating employees in the database, import each employees as a CRM resource. This requires the CRM Resource Manager responsibility.

   a. If group membership is required for the user, create and administer groups first. Ensure you provide the appropriate role types, roles, and usages to the group. If the appropriate group already exists and has the appropriate attributes, skip this step.

   b. After group creation (if applicable), import the employee as a CRM resource. Ensure you provide the appropriate role type and role for the employee. Employees do not have usages.

   c. If required, query the appropriate group and associate the employee from within the group. Verify the person has the appropriate role types and roles.

3. Define Oracle Applications users. This requires the System Administrator responsibility.

    a. First select the user name and password. Verify the password.

    b. Assign the appropriate responsibilities for this user. Use the Implementation Responsibility Matrix above as a guideline.

    c. If an employee in the database, associate this Oracle Applications user account with the employee in the enterprise database.

See Also

- Section 4.7.1.1, "Types of Users Required"

- Section 4.7.1.2, "Administrator Users"

- Section 4.7.1.3, "Script End Users"

- Section 4.7.1.4, "Implementation Responsibility Matrix"

4.7.2 Creating and Administering Users

This section includes the following topics:

- Section 4.7.2.1, "Creating an Employee in the Enterprise Database"

- Section 4.7.2.2, "Creating a Resource Group"

- Section 4.7.2.3, "Importing a CRM Resource"

- Section 4.7.2.4, "Defining Oracle Applications Users"

See Also

Section 4.7.1, "Understanding Oracle Scripting Users"

4.7.2.1 Creating an Employee in the Enterprise Database

Enterprises using Oracle Human Resources Management Systems (HRMS) must create employees with this application. For enterprises not using HRMS as their human resources application, employees must be created using CRM Resource Manager.

Use this procedure to create an employee in Oracle Human Resource Management

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

HRMS Manager or CRM Resource Manager

Steps

1. If using HRMS, from the Navigator, select **People > Enter and Maintain** and click **Open**.

1. The Find Person window appears.

2. If using CRM Resource Manager, from the Navigator, select **Maintain Employee > Employee > Enter Person** and click **Open**.

3. Click **New**.

   The People window appears.

4. Enter the information for the new person.

   The following fields are required:

   - Last (Name)

   - Gender

   - Type (select **Employee**)

   - Date of Birth (if using Payroll)

   - Social Security Number (if using US HRMS)

   - Employee (Number) - If the Employee field is inactive, then Oracle HRMS is set up to automatically generate the employee number when the record is saved.

   Other fields may be required depending on how your enterprise has set up Oracle HRMS or CRM Resource Manager. In addition, when you save the record, you may receive one or more messages that explain the consequences of leaving certain fields blank.

5. From the File menu, click **Save**.

   In the status line, confirmation appears that a record has been applied and saved.

6. Continue your work or log out.

Guidelines

In a test environment, the Survey Administrator responsibility can be assigned to any Oracle Applications user. In a production environment, however, this user must be an existing HRMS employee. This employee must also be designated as a CRM Resource. Steps to create employees in the enterprise database, import them as CRM resources, define Oracle Applications users, assign them responsibilities, and associate them with employees in the enterprise database are discussed in Section 4.7.2, "Creating and Administering Users".

HRMS Employee

The HRMS employee is typically created in Oracle Human Resources Management System (HRMS), if licensed by an enterprise. For enterprises that have *not* licensed HRMS, the ability to create employees is *also* a scaled-down function of Oracle CRM Resource Manager.

If HRMS *is* installed, Oracle Applications enforces the business rule of *requiring* the user to create a person and designate them as the person type "employee" within HRMS. If you attempt to create an employee using CRM Resource Manager while HRMS is installed, you will experience an error.

Creating an employee in HRMS is accomplished by a user that has been assigned the appropriate HRMS responsibility (in the United States, for example, the US HRMS Manager responsibility).

If HRMS is not installed and licensed by an enterprise, the employee must be created using the CRM Resource Manager responsibility.

CRM Resource

After an employee is created, additional information (not required by HRMS) is required in order for CRM applications at the enterprise to appropriately reference this individual. If the employee is created using CRM Resource Manager, this information is part of the definition process.

If the employee is created using HRMS, employee information must subsequently be identified as a CRM resource. (By "resource," the concept of a person in the enterprise is inferred.) To do so, import the HRMS employee into CRM Resource Manager using the CRM Resource Manager responsibility. This process includes creation of a Resource Number and a corresponding Resource ID for this individual which is required by CRM applications.

Regardless of which method is used to create the employee (using HRMS or CRM Resource Manager), this document refers to this employee as an HRMS employee.

The other assumption that is a prerequisite for all Survey implementation steps is that this employee is identified as a CRM resource.

See Also

Section 4.7.2.2, "Creating a Resource Group"

Section 4.7.2.3, "Importing a CRM Resource"

Section 4.7.2.4, "Defining Oracle Applications Users"

4.7.2.2 Creating a Resource Group

Use this procedure to set up resource groups in CRM Resource Manager. Resource groups allow the assignment of role types, roles, and usages to one group that may contain one or many members.

An employee in a group with the role of manager automatically becomes the manager of the other employees in that group and of the employees in the groups below in the hierarchy.

> **Note:**   You must not assign more than one employee with the role of Manager per group. Doing so will impact reporting accuracy.

Use this procedure to create a resource group.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

CRM Resource Manager

Steps

1. From the Navigator, select **Maintain Resources > Groups** and click **Open**.

   The Define Groups page appears.

2. In the Group Name field, type a name for your resource group.

3. Click the Roles tab.

4. From the list in the Role Type field, select the appropriate role type.

   For example, select **TeleSales** in the first row and **Collections** in the second row if this is a resource group for Oracle Collections agents, or select **TeleSales** if this is a resource group for Oracle TeleSales agents.

5. From the list in the Role field, select the appropriate role for the selected role type.

   For example, select **TeleSales Agent** and **Collections Agent** for use with Oracle Collections, or select **TeleSales Agent** for use with Oracle TeleSales.

6. Click the Usages tab.

7. From the list in the Usage field, select the appropriate usages for the group.

   For example, select **Sales and TeleSales** and **Collections** if this is a resource group for Oracle Collections agents, or select **Sales and TeleSales** if this is a resource group for Oracle TeleSales agents.

8. Click the **Save** icon.

9. Continue your work or log out.

See Also

Section 4.7.2.1, "Creating an Employee in the Enterprise Database"

Section 4.7.2.3, "Importing a CRM Resource"

Section 4.7.2.4, "Defining Oracle Applications Users"

4.7.2.3 Importing a CRM Resource

Use the following procedure to import a CRM resource.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

CRM Resource Manager

Steps

1. From the Navigator, select **Maintain Resources > Import Resources** and click **Open**.

   The Selection Criteria window appears.

2. In the Resource Category field, select **Employee**.

3. Enter any additional selection criteria.

   For example, in the Name field, select the name of an employee.

4. Click **Search**.

   Employees that meet the search criteria are listed in the Search Results area.

5. Refine your search if necessary. Select the appropriate employee and click **OK**.

   The Selection Criteria window refreshes with the criteria you confirmed.

6. When satisfied with your selection criteria, click **Search**.

   The Selection Criteria window refreshes. Employees that meet the specified search criteria are listed in the Search Results area. The Select checkbox for the matching employees are automatically selected.

7. Clear the Select checkbox for each employee for whom you do not want to create a CRM resource.

8. Click **Create Resource**.

   The Default Values window appears.

   > **Note:** Do not make any changes or add roles from this window. You can add or modify this information in the resource details later.

9. Click **OK** to accept the defaults.

   The Selected Resources window appears. The Comments field indicates whether the resource is a new record, a duplicate record, or a duplicate record with a new role definition. The Select checkbox is automatically selected.

10. Clear the Select checkbox for each employee for whom you do not want to save as a resource.

11. To save the resources, click **Save Resource**.

A transaction number appears in the Transaction Number field. The transaction number is associated with each resource created during this transaction. More than one resource can have the same transaction number.

12. Click **Details**.

The Resource window appears. Verify the name of the resource (Name), the name of the employee (Source Name), and the user name (User Name) and note the resource number (Number).

13. From the Resource window, click the Roles tab (if it is not already displayed).

14. In the Role Type field, select the appropriate roles for this imported resource.

For example, if you want this user to access the Administration tab in Oracle Marketing, select **Sales** from the list of values. If you want this user to access Oracle TeleSales, select **Telesales** from the list of values. If you want this user to access Oracle Collections, select **Collections** in the first row and **Telesales** in the second row.

15. In the Role field, select the appropriate role for each role type.

For example, if you want this user to access the Administration tab in Oracle Marketing, select **Sales Manager** from the list of values. If you want this user to access Oracle TeleSales, select **Telesales Agent** from the list of values. If you want this user to access Oracle Collections, select **Collections Agent** in the first row and **Telesales Agent** in the second row.

16. Click the **Save** icon.

17. From the Resource window, click the Groups tab.

18. In the Name field under Groups, select the group name you created in the Creating a Resource Group topic.

19. In the Group Member Roles area, click in the Name field.

This field auto-populates with the appropriate role.

20. Click the **Save** icon.

21. Continue your work or log out.

See Also

Section 4.7.2.1, "Creating an Employee in the Enterprise Database"

Section 4.7.2.2, "Creating a Resource Group"

Section 4.7.2.4, "Defining Oracle Applications Users"

4.7.2.4 Defining Oracle Applications Users

Use the following procedure to define Oracle Applications user accounts (user name and password), assign responsibilities to users, and associate an Oracle Applications user with an employee in the enterprise database.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > User > Define** and click **Open**.

    The Users window appears.

2. In the User Name field, type the user name you want to define and press the tab key.

    User names should be between 6 and 64 characters in length and should not contain spaces. Avoid using special characters such as %, slash, or backslash. Hyphen and underscore characters are supported.

3.   Optionally, in the Description field, type a description of this user.

4. In the Password field, type the password for this user and press the tab key.

    The first time the user logs in, this password must be changed.

    Passwords should be between 5 and 64 characters in length and should not contain spaces. Avoid using special characters such as %, slash, or backslash. Hyphen and underscore characters are supported.

5. In the Password field, retype the same password for confirmation, and press the tab key.

6. Optionally, in the Password Expiration area, select criteria for the expiration of this password.

7. In the Responsibilities tab, click in the first row of the Responsibility column.

8. Type the appropriate responsibility name and press the Tab key. Optionally, you can search for the appropriate responsibility name by selecting the ellipsis (...).

   ■ You can enter the full responsibility name, or you can enter partial criteria followed by the percent wildcard. Oracle Applications will attempt to automatically validate and autofill based on the criteria you entered. If an exact match results, the remaining columns will be populated.

   ■ If an exact match cannot be determined, a Responsibilities window appears on top of the Users window. Your search criteria appears in the Find tab field, with all results that meet your criteria appearing in a list below.

   ■ If many choices appear in the list, you may scroll through the list to find the appropriate responsibility. You may also refine your search criteria in the Find field and click **Find**. Highlight the appropriate choice in the list, and click **OK**. The Find window closes, and the appropriate criteria for all columns populates in the current row.

   Responsibility information appears for the selected responsibility. By default, the Effective Date (From field) is populated with the SYSDATE.

9. Change the date in the Effective Date From field to a prior date (yesterday's date, the beginning of the current month, or other appropriate choice).

   While an effective date starting with the SYSDATE may be effective immediately in some environments, best practices dictate that you ensure that the effective date you designate will take effect immediately in all environments.

10. Optionally, enter a date beyond which the responsibility will no longer be effective in the To field. This date must be later than the SYSDATE.

11. Select **File > Save and Proceed** to save your work.

12. In a new row in the Responsibility table, repeat steps 8 through 11 for each responsibility you want to assign to this user.

13. In the Person field, search for the name of the person (employee in the enterprise database) for whom you want to associate this user account.

   ■ Enter full or partial criteria, in syntax Last Name, Title, First Name.

   ■ You can use the % wildcard character.

   If an exact match is found, the employee name will populate in this field. Otherwise, the Person Names window results.

14. If required, refine your search in the Person Names window and click **OK**.

15. Select **File > Save and Proceed** to save your work.

16. Continue your work or log out.

See Also

Section 4.7.2.1, "Creating an Employee in the Enterprise Database"

Section 4.7.2.2, "Creating a Resource Group"

Section 4.7.2.3, "Importing a CRM Resource"

5

Implementation Administration Tasks

Tasks in this section address administration that may be required for implementation of Oracle Scripting.

Tasks

You can perform the following tasks:

- Setting Up Server Logging

- Bypassing the Sales Group Membership Requirement

- Granting JTF Roles

- Setting System Profile Values

- Finding Application and Responsibility ID Values

- Removing the Quick Find Menu

5.1 Setting Up Server Logging

When agents execute scripts in the Scripting Engine Java-based UI using the Apache Mid-Tier architecture, or when survey respondents execute scripts in a Web browser, error messages are written to server logs. When the Debug mode is on, additional informational messages are also written to info logs.

> **Note:** Server-side error and informational logs when executing the agent application in the Caching Architecture (pre-11.5.6 implementations) are written to the JInitiator Java console. Also, client-side errors for Scripting Engine agent operations will continue to display in the JInitiator Java console.

Implementation of server logging is outside the scope of this documentation. From an Oracle Scripting perspective, agents or administrators must have access to the Interaction Center Server Log Viewer responsibility or the Interaction Center Server Manager responsibility to access log files for viewing error and informational messages. This procedure is described below.

Prerequisites

Server logging for interaction center technologies must already be implemented at the site to view server logs. However, the following procedure may be accomplished at any time.

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > User > Define** and click **Open**.

    The Users window appears. From this window, begin selecting the user for whom you wish to assign the appropriate responsibility to view server logs.

2. With your cursor in the first tab field (User Name), from the **View** menu, select **Query By Example > Enter** to begin query mode for a particular user.

    > **Note:** You can also enter query mode to enter your search criteria by pressing F11 on your keyboard.

    The selected tab field (User Name) changes from yellow to blue, and the cursor blinks, indicating that you can now enter your query information.

3. Enter the Oracle Applications account user name for which you are querying. You can use the % wildcard character as part of your search. Upon entering your search criteria, from the **View** menu, select **Query By Example > Run**.

    > **Note:** You can also run the query with the criteria you have entered by pressing Ctrl-F11 on your keyboard.

- For a successful query in which a single user meets the designated search criteria, the status line on the bottom left of the screen will display "Record 1/1." Fields in the window populate with information pertaining to this user.

- If multiple users meet the designated search criteria, the status line on the bottom left of the screen will display "Record 1/?," indicating more records are available. Information pertaining to the first user in the found set will display in the window. Navigate through the remaining records meeting your search criteria by using the Up and Down arrow keys on your keyboard, until you find the appropriate record.

In the lower half of the window, a gray region appears displaying information for the selected tab (Responsibilities or Securing Attributes).

4. If not already active, select the Responsibilities tab.

All of the responsibilities currently assigned to the queried user are displayed.

5. Place your cursor anywhere in the Responsibilities table and click.

This will result in blue highlighting in that area of the table.

> **Caution:** Even if highlighting is already evident, perform this step. If you do not click here, you will create a new *user* instead of a new *responsibility row* for the *current* user if you select the New icon from the button menu in the next step. It does not matter where in the Responsibility table you click.

6. If a blank row is visible in the responsibility table, click in it and proceed to step 8.

7. If no blank rows are visible in the responsibility table, click on any populated row in the table, then create a new row for the responsibility you will add. To add a new responsibility row, click the first icon in the toolbar (a green plus sign icon which displays a tool tip of New).

A new row appears immediately below the portion of the Responsibilities table that was highlighted.

> **Caution:** Even if the responsibility table seems to display
> highlighting, ensure that you click anywhere in the table before
> clicking the New icon from the button menu. If you do not click
> here, you will create a new *user* instead of a new *responsibility row*
> for the *current* user.

8. In the Responsibility column of the blank row, search for or enter the
    appropriate responsibility name. Press the Tab key or select the ellipsis (...)
    towards the right of the Responsibility tab field.

    ■   You can enter the full responsibility name, or you can enter partial criteria
        followed by the percent wildcard. Oracle Applications will attempt to
        automatically validate and autofill based on the criteria you entered. If an
        exact match results, the remaining columns will be populated.

    ■   If an exact match cannot be determined, a Responsibilities window appears
        on top of the Users window. Your search criteria appears in the Find tab
        field, with all results that meet your criteria appearing in a list below.

    ■   If many choices appear in the list, you may scroll through the list to find the
        appropriate responsibility. You may also refine your search criteria in the
        Find field and click **Find**. Highlight the appropriate choice in the list, and
        click **OK**. The Find window closes, and the appropriate criteria for all
        columns populates in the current row.

    Responsibility information should appear in the five columns displayed in the
    Responsibility table. By default, the Effective Date (From field) is populated
    with the SYSDATE.

9. Change the date in the Effective Date From field to a prior date (yesterday's
    date, the beginning of the current month, or other appropriate choice).

    While an effective date starting with the SYSDATE may be effective
    immediately in some environments, best practices dictate that you ensure that
    the effective date you designate will take effect immediately in all
    environments.

10. If desired, enter a date beyond which the responsibility will no longer be
    effective in the To field. This date must be later than the SYSDATE.

11. When satisfied that the values for the Interaction Center Server Log Viewer
    responsibility meet your implementation requirements, save the record by
    selecting **Save and Proceed** from the **File** menu.

> **Note:** You can also save the record by clicking the disk icon in the toolbar (fifth icon from the left, with a tool tip value of "Save").

12. If no other tasks are required, log out of Oracle Applications by selecting **Exit Oracle Applications** from the **File** menu.

Guidelines

- Use Interaction Center Server Log Viewer to provide access to the Log Messages tab only. This is appropriate for most agents.

- Use Interaction Center Server Manager or Call Center HTML Administration for interaction center administrators that may need to access other tabs in the Call Center Technologies administration console.

Assuming the server logging function is appropriately implemented, this user can now access the server logs for Oracle Scripting message logs.

See Also

- Bypassing the Sales Group Membership Requirement

- Granting JTF Roles

- Setting System Profile Values

- Finding Application and Responsibility ID Values

- Removing the Quick Find Menu

5.2 Bypassing the Sales Group Membership Requirement

To access the Audience tab in the Survey Administration Console, the Oracle Applications user account must have a sales group ID or a sales role. Administration of resource groups is accomplished in Oracle CRM Resource Manager.

By setting a profile option, you can bypass this requirement for the Survey Administrator responsibility as described below. This profile can also be set at the application level.

Sales Groups, Roles and Usages

Sales groups, roles and usages are administered in Oracle CRM Resource Manager. These are used by various applications, including Oracle Marketing, Oracle TeleSales and Oracle Collections.

You must define a sales group and assign the group roles and usages prior to assigning any members to the group. If a sales group already exists, you can add members to the group. Each group member must have a CRM resource ID, which is created upon importing an employee into CRM Resource Manager.

Some applications (such as Oracle Marketing) require steps to be performed in a particular order. Thus, the recommended order is as follows:

- Creating an employee using Oracle HRMS (if installed) or CRM Resource Manager (if HRMS is not installed).

- Creating an Oracle Applications user account with the appropriate responsibilities.

- Associating the Oracle Applications user account with the employee.

- Creating a resource group and assigning roles and usages to the group.

- Importing a CRM resource and assigning roles and usages to the resource.

- Adding a defined resource as a member of an existing group.

Prerequisites

None

Responsibility

System Administrator

Login

Log into Oracle Forms-based applications.

Steps

1. From the Navigator, select **Profile > System** and click **Open**.

   The Find System Profile Values window appears above the unpopulated System Profile Values window. In the Display area, the Site and Profile with No Values options are selected.

2. In the Display area, select **Responsibility**.

The field changes color, and an ellipses appears.

In the Responsibility field, type the appropriate responsibility for which you want to bypass the requirement of group membership, and press the tab key.

For example, type **Survey Administrator** and press the tab key.

3. Leave the **Profiles with No Values** option selected.

   This allows the query to return profiles that meet your search criteria but have no values associated with them yet (blank profiles).

4. In the Profile field, enter **Bypass Group Validation**.

5. Click **Find**.

   The System Profile Values window appears.

6. In the Bypass Group Validation row, click once in the Responsibility column.

   An ellipses appears.

7. Click the ellipses.

   The Bypass Group Validation window appears.

8. From the list of values, select **Yes** and click **OK**.

9. Save your work and exit.

See Also

- Setting Up Server Logging

- Granting JTF Roles

- Setting System Profile Values

- Finding Application and Responsibility ID Values

- Removing the Quick Find Menu

5.3 Granting JTF Roles

To perform certain JTF tasks, a user must be assigned a specific JTF role. For example, the JTF_SYSTEM_ADMIN_ROLE role provides certain administrative capabilities to a user. From an Oracle Scripting perspective, this is required in order to create a guest user for survey operations. Additionally, Oracle Scripting survey administrators using list-based survey campaigns must have the JTF_FM_ADMIN role to access the Invitations tab from the Survey Administration console.

Use the procedure below to grant JTF roles.

Prerequisites

- To perform some Oracle HTML-based applications administration, including this task, your Oracle Applications user must be granted the JTF_SYSTEM_ADMIN_ROLE. If required, you can use the seeded sysadmin user name.

- Your Oracle Applications user must be granted a JTF role before you can assign that JTF role to another user. Thus, this user must be granted the JTF_FM_ADMIN role. This role is available to the seeded sysadmin user.

Login

Log into Oracle HTML-based applications.

Responsibility

CRM HTML Administration or CRM ETF Administration

Steps

1. From the CRM Applications Administration console, click the Users tab.

2. Click the Registration subtab.

   The Introduction page appears.

3. From the side panel, click **User Maintenance**.

   The Users page appears.

4. In the search criteria field, enter the appropriate search criteria to find the appropriate user name, and click **Go**.

   The Users page refreshes.

5. In the search criteria field, enter the appropriate search criteria to find the appropriate user name, and click **Go**.

   > **Note:** You must enter at least three characters. You can use the % wildcard search character in your criteria.

   The Users page refreshes, displaying a summary list table containing the set of records matching your criteria.

6. If necessary, refine your search criteria and click **Go**.

7. From the resulting list of users, click the appropriate entry in the Username column.

   The User Details page appears.

8. Under User Details, click **Roles**.

   The User-Roles Mapping page appears. Two lists are displayed (Available Roles and Assigned Roles).

9. From the Available Roles list, select each role you want to assign and click **> Move** to move it to the Assigned Roles list.

   For example, select JTF_SYSTEM_ADMIN_ROLE and JTF_FM_ADMIN and move them to the Assigned Roles list.

10. When satisfied, click **Update**.

    The User-Roles Mapping page refreshes. Two lists are displayed (Available Roles and Assigned Roles).

11. Continue your work or sign out of Oracle Applications.

See Also

- Setting Up Server Logging
- Bypassing the Sales Group Membership Requirement
- Setting System Profile Values
- Finding Application and Responsibility ID Values
- Removing the Quick Find Menu

5.4 Setting System Profile Values

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Profile > System** and click **Open**.

   The System Profile Values window appears. Above this, the Find System Profile Values window appears and has focus.

2. If you want to display and set profile options by site, in the Display area of the Find System Profile Values window, ensure **Site** is selected. If you do not wish to view or set profiles at the site level, clear the **Site** option.

3. You can display and set profile options by Application, Responsibility, or User. For each selection:

   a. Mark the appropriate option.

   b. In the text field next to the selected Application, Responsibility, or User option, type search criteria and click tab.

      If you typed full or partial search criteria with one match, the appropriate value populates the field. Otherwise, a window appears where you can narrow your search criteria.

   c. Refine your search if required, select the appropriate option, and click **OK**.

   Note: Site is the highest and broadest level of Oracle Applications profile settings, and affects all Oracle Applications users unless a specific lower level is specified. By changing the search criteria from Site to any of the other values (Application, Responsibility, or User), you can query and set profiles for lower and more granular levels. A lower level profile setting, if populated, always takes precedence and supersedes any value set at a higher level.

   ---

4. Ensure the **Profiles with No Values** option is selected.

   This allows the query to return profiles that meet your search criteria but have no values currently associated with them (blank profiles).

5. In the Profile field, enter your search criteria. You may include the % wildcard character.

   For example, enter IES%. IES is the product code for Oracle Scripting. Profile values associated with the survey component are included in this category.

6. Click **Find**.

The System Profile Values window populates, displaying all profile values for the requested search criteria at the requested levels.

7. Set each profile option as required by your implementation task. You can set these options in any sequence.

8. Select **File > Save and Proceed** to save your work.

   The status line will indicate that records has been saved.

9. If you do not need to set additional profile options at this time, then proceed to step 13.

10. If you want to set additional profile options with different search criteria, then from the **View** menu, select **Find**.

    The Find System Profile Values window appears.

11. Repeat steps 2 through 8 above.

12. When satisfied, select **File > Save and Proceed** to save your work.

    The status line will indicate that records have been saved.

13. Select **File > Close Form** to close this window.

14. Continue your work or log out of Oracle Applications by selecting **File > Exit Oracle Applications**.

See Also

- Setting Up Server Logging
- Bypassing the Sales Group Membership Requirement
- Granting JTF Roles
- Finding Application and Responsibility ID Values
- Removing the Quick Find Menu

5.5 Finding Application and Responsibility ID Values

You must know the APPLICATION_ID value for your application and the RESPONSIBILITY_ID value for a specified responsibility before you can set the required Oracle CRM Technology Foundation (JTT) profile options appropriately.

Based on how Oracle Applications was installed in your environment, a responsibility may have different names. For example, in a Vision Database

installation, the responsibility to access the Scripting Engine agent interface in stand-alone mode is Scripting Agent, Vision Enterprises. In a fresh installation, this responsibility is referred to as Scripting User. Nevertheless, both will have a single responsibility identification.

> **Note:** The APPLICATION_ID value for Oracle Scripting is typically 519. The RESPONSIBILITY_ID value for Survey Administrator is typically 21685. The RESPONSIBILITY_ID for Scripting Administrator is typically 23723. However, in some scenarios, the responsibility identification may differ, based on environmental factors.

Use the following procedure to find the APPLICATION_ID value and RESPONSIBILITY_ID value of a responsibility.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, select **Security > Responsibility > Define** and click **Open**.

   The Responsibilities form appears.

2. Choose **View > Find**.

   The Responsibilities search window appears.

3. Search for the appropriate responsibility, highlight it, and click **OK** in the search window.

   The Responsibilities form is populated with the record for the selected responsibility.

4. With your cursor in any field of the record, choose **Help > Diagnostics > Examine.**

If the Enable Diagnostics window appears, enter the apps level database password and click **OK**.

The Examine Field and Variable Values window appears.

5. If you want to determine the Application ID, in the Examine Field and Variable Values window, from the Field list, select **APPLICATION_ID** and click **OK**.

The Value field in the Examine Field and Variable Values window is populated with the value of APPLICATION_ID.

6. If you want to determine the Responsibility ID, in the Examine Field and Variable Values window, from the Field list, select **RESPONSIBILITY_ID** and click **OK**.

The Value field in the Examine Field and Variable Values window is populated with the value of RESPONSIBILITY_ID.

7. If you want to determine the Responsibility Key, check the value of the selected responsibility in the **Responsibility Key** field.

See Also

- Setting Up Server Logging

- Bypassing the Sales Group Membership Requirement

- Granting JTF Roles

- Setting System Profile Values

- Removing the Quick Find Menu

5.6 Removing the Quick Find Menu

When upgrading Oracle Applications, the obsolete Quick Find menu remains associated with product code IES, causing this menu to appear in the Scripting Administration console, where it has no function. This menu will not appear in the Scripting administrative interface for new implementations. For upgrade implementations, you must modify the JTT properties for product IES (Oracle Scripting) to remove the association between this menu and the Scripting product.

Use the procedure below to remove the Quick Find menu from its association with Oracle Scripting (product code IES).

Prerequisites

- You must log in as a user with JTF_SYSTEM_ADMIN_ROLE. For example, you can use the seeded sysadmin Oracle Applications account.

Login

Log into Oracle HTML-based applications.

Responsibility

CRM HTML Administration or CRM ETF Administration

Steps

1. From the CRM Applications Administration console, click the Settings tab.

2. Click the System subtab.

   The Introduction page appears.

3. From the side panel, select **Properties > Advanced.**

   The Advanced - Properties page appears.

   > **Note:** If this page does not appear, your login does not have the JTF_SYSTEM_ADMIN_ROLE role. Log out and log in with an appropriate user.

4. From the View menu, select **IES**.

   The Advanced - Properties page refreshes. The records displayed show advanced properties for product code IES (Oracle Scripting).

   The following keys for IES advanced properties must be modified:

   a. search.factories

   b. service.oracle.apps.ies.survey.search.server.IESSurveySearchFactory.categories

   c. service.oracle.apps.ies.survey.search.server.IESSurveySearchFactory.desc

5. For each of the keys listed above, append a period and the responsibility ID for the Survey Administrator responsibility, as indicated below:

> **Note:** The responsibility ID for the Survey Administrator is typically 21685, but this may differ in your environment. To determine the appropriate ID for your environment, see Section 5.5, "Finding Application and Responsibility ID Values".

a. In the Advanced - Properties table, from the Key column, click on the appropriate property.

   The Key Details page appears.

b. In the Key field, append a period and the responsibility ID.

   For example, for the search.factories key, modify the entry to read:

   ```
   search.factories.21685
   ```

c. Click **Update**.

   The Advanced - Properties page refreshes. The modified key is displayed.

d. Repeat steps a through c for the remaining keys to be modified.

e. In the Key field, append a period and the responsibility ID.

6. If the changes do not immediately result in the removal of the Quick Find menu in the Scripting Administration console, stop and restart the Apache Web server to see the changes take effect.

7. Continue your work or log out of Oracle Applications by selecting **Sign Out**.

See Also

- Setting Up Server Logging

- Bypassing the Sales Group Membership Requirement

- Granting JTF Roles

- Setting System Profile Values

- Finding Application and Responsibility ID Values

6

Survey Resources Administration Tasks

6.1 Administering Survey Resources

Survey resources are HTML files that display to users of the Scripting Engine Web interface at runtime. The header and footer section survey resources display, respectively, at the top and bottom of each page of a script executed in a Web browser. The center of the page is where each panel of the script appears. The error page and final page survey resources are full HTML pages. When an error condition occurs while the end user is executing a script, the error page is displayed. After the last panel in a script, the final page is displayed. These full-page resources do not display the header or footer section resources.

All scripts executed in the Scripting Engine Web interface must have header, error page, and final page survey resources associated with them. Footer section resources are optional.

The physical survey resource files must be saved in Java Server Page (JSP) format, even if they contain no dynamic content. Survey resources, like any other HTML or JSP page, may include images and hyperlinks. Survey resources must be located in the $OA_HTML directory on the applications server to use at runtime. Any objects (such as GIF or JPG images) referenced in a resource page must also be available to the application server at runtime.

Each survey resource must be defined in the Survey Administration console to provide a logical pointer to its corresponding physical JSP file (in $OA_HTML) which displays at runtime. Once defined, survey resource definitions persist; the same resources can be used by any number of survey campaigns without limit.

> **Note:** Prior to activating a survey campaign deployment, you must ensure the files are located in the $OA_HTML directory on the applications server to use at runtime. Four test JSP files are seeded in the appropriate directory and may be used to test survey functionality. Although these test survey resources exist in the appropriate directory, they must still be defined prior to use, as described below.

When you define a survey resource in the Survey Administration console, the values you assign are stored in the IES_SVY_RESOURCES table of the applications database. Once you create a survey campaign and identify resources to be associated with that survey campaign, these resources and the survey campaigns with which they are associated are populated in the IES_SVY_SVYRESOURCES table of the applications database.

Although survey resources are not required to be uploaded or defined in order to *create* a survey campaign, both of these tasks are required in order to *execute* a script in the Scripting Engine Web interface.

Creating, modifying, and uploading the physical JSP files that serve as survey resources are not accomplished from within the Survey Administration console. Nor can existing survey resource definitions be modified or deleted from within the application.

Use the Survey Resources subtab to define survey resources or to view a list of defined survey resources.

Tasks

You can perform the following tasks:

- Creating Survey Resources

- Defining Survey Resources

- Uploading Survey Resources

6.1.1 Creating Survey Resources

No provisions are made to create HTML or JSP files within the context of the Oracle Scripting product. Nonetheless, the physical survey resource JSP files must be created and uploaded, and subsequently defined in the Survey Administration

console, in order to execute a script in the Web interface. Creation of survey resources is required customization.

Test survey resource JSP files ship with Oracle Applications, seeded in the appropriate directory ($OA_HTML) on the applications server. You can use the provided test resources to test your implementation and to serve as building blocks for creating your own resources. Ensure you retain the seeded resources for testing. For more information, see Oracle Scripting Survey Concepts > Survey Resources.

> **Note:** Files with a JSP file extension must be tested on an existing Web server to view appropriately.

You must have the requisite knowledge and resources to create or modify HTML and JSP files in order to administer survey resources for use with Oracle Scripting to execute scripts in a Web browser.

See Also
- Defining Survey Resources
- Uploading Survey Resources

6.1.2 Defining Survey Resources

Survey resources must be defined in the Survey Administration console and must correspond to HTML or JSP documents physically residing on the server at runtime (as described in Uploading Survey Resources). After defining all appropriate survey resources using the Survey Resources tab, you can associate defined resources with the header section, footer section (optional), error page, final page of a survey campaign. Resources can be used for any number of survey campaigns.

> **Note:** The physical JSP survey resource files corresponding to files you define in this step are *not* required to exist on the server *at the time of definition* in the Survey Administration console. However, these files *must* physically exist in the appropriate location on the server when a script is executed in a Web browser, or a 'file not found' exception will result.

To execute any survey campaign, you must have a header, error page, and final page survey resource defined. Footer resources are optional.

Use this procedure to define survey resources.

Prerequisites
None

Login
Log into Oracle applications using the Personal Homepage login.

Responsibility
Survey Administrator

Steps

1. Click the Survey Resources tab.

   The Survey Resources page appears, displaying a list of any existing survey resources that match the View list criteria.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   - To view survey resources defined by all users, select **All Survey Resources** from the View list and click **Go**.

   - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. Click **Create** to begin defining a survey resource.

   The Define Survey Resources page appears.

4. In the File Name field, type the name of the JSP file that will correspond to this survey resource definition. Include the .JSP file extension.

   For example, **iessvytestheader.jsp**

   This name appears in the File Name column of any table listing survey resources.

> **Note:** JSP files identifying physical survey resources are *not*
> required to physically exist on the server at the time you define a
> resource. However, you must ensure they are uploaded prior to
> executing the survey. If a corresponding JSP resource (with a file
> name that precisely matches this File Name field) is not physically
> present on the server, then a "file not found" exception will result in
> the Web browser of any respondent who attempts to participate in
> this survey or self-service Web application user who attempts to
> view a script.

5. In the Logical Name field, type a name to serve as a pointer to the file.

   This name appears in the Logical Name column of any table listing survey
   resources.

   You can use a logical name identical to the JSP file name, or a meaningful
   descriptive name that follows your enterprise or campaign requirements or
   naming conventions.

6. Optionally, in the **Description** text area, type a meaningful description.

7. From the Language list, select a language.

   This field reads and displays available languages from the Oracle RDBMS
   FND_LANGUAGES table. Select the appropriate language for your
   environment, if it is not already displayed.

   For example, for English in the United States, select **AMERICAN.**

8. When satisfied with the values for this resource, click **Apply**.

   The Survey Resources page refreshes, listing the resource you just defined.

9. If required, then repeat steps 3 through 8 above as appropriate.

   For any survey campaign, three resources must be identified: header section,
   error page, and final page. Footer section resources are optional.

See Also

- Creating Survey Resources
- Uploading Survey Resources

6.1.3 Uploading Survey Resources

Once JSP files are created to use as survey resources, you must upload them to make them accessible to the Web server.

> **Note:**   These files *must* be physically stored in the $OA_HTML directory on the applications server. Test resources are already seeded in this directory.

If you employ graphics in custom JSP resources, the appropriate physical image files (typically in GIF or JPG file format) must be uploaded and stored to a server directory to any location accessible to the Apache Web server at runtime. Since images supporting Oracle Applications are stored in the $OA_MEDIA directory on the server, this location is suggested. Correspondingly, the custom code for the resource must reference the image in its path, as in the following example:

```
<img src="/OA_MEDIA/[image_name].gif" ...>
```

Consult with your system administrator to determine if there are different requirements for your environment.

Prerequisites

- Physical JSP files to serve as survey resources must exist prior to this step.

- To upload physical survey resource files, you need access to the $OA_HTML directory on the applications server and may need access to $OA_MEDIA. These variables correspond to physical file locations on the Apache Web server, which may differ for each environment. Consult your Apache Web server administrator or system administrator for access.

Login

Not applicable

Responsibility

Not applicable

Steps

1. On your local computer (or over a network), locate the <filename>.JSP files created for use as survey resources.

2. Connect to the applications server through an appropriate method such as TELNET.

   If you do not have permissions to access the appropriate environment, consult your system administrator.

3. Navigate to the $OA_HTML directory.

   $OA_HTML is a variable to point to a physical directory. JSP resources for Oracle Scripting must physically exist in this directory path in order to execute a script in a Web browser.

4. Using an appropriate protocol such as FTP, "put" or copy the target JSP files to the $OA_HTML directory.

   You may choose to specify ASCII format for file transfer.

5. If any survey resource file includes references to graphics, navigate to the appropriate directory specified in the survey resource (for example, $OA_ MEDIA).

   $OA_MEDIA is a variable to point to a physical directory in which media items for Oracle Applications are stored. The physical directory path may differ based on your specific environment or enterprise requirements for storing customized images in support of Oracle Applications.

6. Using an appropriate protocol such as FTP, "put" or copy the target graphic files to the appropriate directory (for example, $OA_MEDIA).

   The survey resources (and any supporting image files) are now available to Survey administrators through the Survey Administration console. Before these files can be used, resources must be defined in the Survey Administration console.

See Also

- Creating Survey Resources
- Defining Survey Resources

7

Scripting Administration Tasks

The Scripting Administration console provides the user interface with which script developers can launch the Script Author as a Java applet, and script administrators can administer Oracle Scripting files, as well as generate, view and analyze agent interface reports.

This console is accessed by logging into Oracle HTML-based applications using an Oracle Applications user account with the Scripting Administrator responsibility.

This section includes the following topics:

- Scripting Home
- Administering Oracle Scripting Files
- Generating and Viewing Panel Footprint Reports

7.1 Scripting Home

Upon logging into the Scripting Administration console, the Home tab appears.

Tasks

You can perform the following tasks:

- Launching the Script Author Java Applet

See Also

- Administering Oracle Scripting Files
- Generating and Viewing Panel Footprint Reports

7.1.1 Launching the Script Author Java Applet

Using the Script Author, you can create, modify, and deploy scripts to the applications database.

Use this procedure to launch the Script Author as a Java applet.

Prerequisites

None

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration console, click the Home tab.

2. Click **Launch Script Author**.

   A separate browser window appears, in which Oracle JInitiator launches, using authentication information from the current Oracle Applications session. Presently, a separate Java applet window appears with the Script Author user interface.

7.2 Administering Oracle Scripting Files

From the Administration tab of the Scripting Administration console, you can monitor and administer files supporting Oracle Scripting, including scripts deployed with the Script Author, and custom Java archive files.

Tasks

You can perform the following tasks:

- Viewing and Removing Deployed Scripts

- Viewing and Administering Custom Java Archive Files

See Also

- Scripting Home

- Generating and Viewing Panel Footprint Reports

7.2.1 Viewing and Removing Deployed Scripts

Use the Deployed Scripts subtab to view or remove scripts deployed to the applications database from the Script Author. Removing a script deletes the script and all related metadata from the database tables.

Use this procedure to view or remove deployed scripts.

Prerequisites

At least one script must be deployed to the applications database from the Script Author.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Deployed Scripts subtab.

   The Deployed Scripts page appears. A summary list of all deployed scripts meeting the filter criteria in the View list appears in the Deployed Scripts table.

3. Optionally, to change the set of records displayed in this list, perform any of the following:

   - From the View list, select a different filter option to display more or fewer deployed scripts based on active status and which user created the script.

   - To display subsequent records in the found set, click **Next** or the right arrow. To view previous records in the found set, click **Previous** or the left arrow.

   - To navigate through the found set of records in groups of ten, from the records range list between the Previous and Next links, select the appropriate range of records (for example, select **11 to 20 of _n_**, where n is the number of records in the found set).

After any selection above, the page refreshes, displaying the requested records.

4. Optionally, to remove scripts from the applications database, in the Select column, mark the **Select** option for each record you want to delete, and click **Remove**.

Note: Oracle Corporation recommends that you back up any script you may want to reuse to a local or network file system before deleting it from the database.

Guidelines

- The value in the Locked column is a read-only property indicating whether the listed script can be modified and deployed from the Script Author (if value is **No**), or whether changes to the script metadata are prohibited (if value is **Yes**).

- Deployed scripts with ACTIVE_STATUS set to "0" are retained in IES_DEPLOYED_SCRIPTS so that existing footprinting and answer collection data can maintain valid references. Only active deployed scripts (those scripts with an ACTIVE_STATUS set to "1") can be executed.

See Also

- Viewing and Administering Custom Java Archive Files

7.2.2 Viewing and Administering Custom Java Archive Files

At runtime, scripts that contain commands referencing custom Java methods can execute each method as specified by the Script Author command parameters. Customized scripts can also use custom user interface Java beans to provide functionality at runtime. The source code for custom Java supporting Oracle Scripting must be compiled into executable class files and packaged into Java archives as appropriate.

Using the Scripting Administration console, you can upload custom Java archive files to the IES_JARFILES table in the applications database. Management of these files is accomplished from the Jar Listings subtab of the Administration tab.

Custom code loaded from the Scripting Administration console to the applications database need not be added to the class path of any configuration file.

Conversely, for custom code that is deployed manually to the APPL_TOP (the model available before this console was available), the corresponding class path must still be specified (in the JSERV.PROPERTIES file for enterprises using the

Apache mid-tier architecture, or in the APPSWEB.CFG file for enterprises using the caching architecture). Manually deployed Java archives will not appear in the list of Java archive files on the **Administration > Jar Listings** page, nor can they be removed using this user interface.

Tasks

You can perform the following tasks:

- Viewing and Removing Uploaded Java Archive Files

- Uploading New Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives

- Changing the Global Property of Uploaded Java Archive Files

- Creating, Removing, and Viewing Java Archive Mapping Definitions

- Specifying Load Order of Java Archives Files

See Also

- Viewing and Removing Deployed Scripts

7.2.2.1 Viewing and Removing Uploaded Java Archive Files

When a Java archive is deployed to the database using the Scripting Administration console, a corresponding record is created in the IES_JARFILES table.

Use this procedure to view and remove custom Java code archives (JAR and ZIP files) uploaded to the applications database from the Scripting Administration console. Removing a Java archive deletes the JAR or ZIP file and all related metadata from the database tables.

Prerequisites

- At least one Java archive file must be uploaded to the applications database using the Scripting Administration Console.

- If a Java archive you want to remove is mapped to a script, delete the mapping prior to performing this task. You cannot remove a Java archive deployed to the database until any mappings associated with the archive are also removed.

> **Note:** Manually deployed Java archives will not appear in the list of JAR files on the Administration > Jar Listings page.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Jar Listings subtab.

The Jar Files page appears. A summary list of all uploaded Java code archive files meeting the filter criteria in the View list appears in the Jar Files table.

3. If required, change the set of records displayed in the Jar Files table list as appropriate to display the desired code archives by performing any of the following:

- From the View list, select a different filter option to display more or fewer Java archive files based on the Oracle Applications login used when the files were uploaded to the database.

- To display subsequent records in the found set, click **Next** or the right arrow. To view previous records in the found set, click **Previous** or the left arrow.

- To navigate through the found set of records in groups of ten, from the records range list between the Previous and Next links, select the appropriate range of records (for example, select **11 to 20 of *n***, where n is the number of records in the found set).

After any selection above, the page refreshes, displaying the requested records.

4. Optionally, to remove uploaded Java archive files from the server, in the Select column, mark the **Select** option for each Java archive you want to delete, and click **Remove**.

Guidelines

You cannot remove a Java archive deployed to the database until any mappings associated with the archive are also removed.

See Also

- [Uploading New Java Archive Files](#)

- Overwriting Existing Java Archive Files with Updated Archives

- Changing the Global Property of Uploaded Java Archive Files

- Creating, Removing, and Viewing Java Archive Mapping Definitions

- Specifying Load Order of Java Archives Files

7.2.2.2 Uploading New Java Archive Files

As discussed in Using Custom Java, custom Java archives can be used with Oracle Scripting to support the execution of custom commands in any script, or the use of Java beans in the agent interface. Using the Scripting Administration console, you can upload Java archive files from the Jar Listings subtab of the Administration tab.

When a Java archive is deployed to the database using the Scripting Administration console, a corresponding record is created in the IES_JARFILES table.

Custom code loaded from the Scripting Administration console to the applications database need not be added to the class path of any configuration file.

Use this procedure to upload new Java archive files for execution in a script at runtime.

Prerequisites

- Write the Java bean or Java method, compile into a class file, and package into a Java archive file.

- Java beans deployed to the database must be for use with the Scripting Engine agent interface only. The Web interface does not support the use of Java beans.

- To execute a method, a script must appropriately reference the method using a Script Author command.

- The script must be deployed to the applications database from the Script Author.

- You must compile your custom code using a JDK level compatible with the Java Runtime Environment (JRE) used for your Scripting implementation.

- For all Java archive files deployed to the applications database from the Scripting Administration console for execution in scripts, the following limitations apply:

    1. Java archive files must be created using the "jar" utility of the JDK and specifying no compression (e.g., "jar -cf0...."), or by creating a ZIP or JAR file

using a compression and archiving utility such as WinZip (with or without compression).

2. Java archive files must be smaller than 1 megabyte.

3. JAR file format is recommended over ZIP file format.

4. You must meet the compilation requirements discussed in Using Custom Java > Java Compilation and Oracle JInitiator Dependencies.

Login
Log into Oracle HTML-based applications.

Responsibility
Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Jar Listings subtab.

   The Jar Files page appears.

3. To begin uploading a Java archive file, click **Upload New Jar**.

   The Upload File page appears.

4. In the **Name** field, type the name you want to associate with this Java archive file in the database.

   You can use any unique name to associate with this file. If you leave this field blank, the file name of the specified JAR or ZIP file will be used. You cannot use a name already assigned to another Java archive.

   If the Java archive is a Java bean, this name will be referenced by the Jar File Name field of the Script Author panel properties window.

5. Optionally, if you want this Java archive to be available globally (to all scripts deployed to this environment), select the **Global** flag option.

6. In the File field, click **Browse...** to begin locating and uploading the Java archive file you want to upload.

   The Choose File window appears.

7. Navigate to the appropriate location on your local file system or network, highlight the file you want to upload, and click **Open**.

> **Note:** You can only upload ZIP or JAR files. Other files will not execute appropriately using either Scripting Engine interface, and are therefore not allowed to be uploaded in this user interface.

The Choose File window closes, and the File window is populated with the path and filename for the specified Java archive file.

8. To clear this selection and specify another file, click **Reset**.

9. To complete uploading the selected file, click **Upload**.

The Jar Files page refreshes. An informational message indicates the logical name of the file you successfully uploaded.

Guidelines

- You must compile your custom code using a JDK level compatible with the Java Runtime Environment (JRE) used for your Scripting implementation. For more information, see Using Custom Java.

- JAR or ZIP files loaded from the Scripting Administration Console must be smaller than 1 megabyte.

- You cannot upload files of any type other than JAR or ZIP.

- When specifying the name of the Java archive file to upload, if you leave the Name field blank, the file name of the specified file will be used.

- Once uploaded, Java archives intended to provide custom commands to scripts must either be designated as global or must be mapped to one or more scripts to be accessible to the Scripting Engine at runtime.

- Java archives uploaded to replace panels with Java beans in the Scripting Engine agent interface do not require JAR mappings.

See Also

- Viewing and Removing Uploaded Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives

- Changing the Global Property of Uploaded Java Archive Files

- Creating, Removing, and Viewing Java Archive Mapping Definitions

- Specifying Load Order of Java Archives Files

7.2.2.3 Overwriting Existing Java Archive Files with Updated Archives

When custom code supporting the use of Oracle Scripting is updated, you must recompile the code, package it appropriately as a JAR or ZIP file, and upload the modified code to the database. This overwrites the previously loaded Java archive file.

Use this procedure to overwrite an existing Java archive file.

Prerequisites

At least one Java archive file must be uploaded to the applications database using the Scripting Administration Console.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Jar Listings subtab.

   The Jar Files page appears. A summary list of all uploaded JAR files meeting the filter criteria in the View list appears in the Jar Files table.

3. If required, change the set of records displayed in the Jar Files table list as appropriate to display the desired code archives.

4. Locate the JAR file in the Jar Files table that you wish to overwrite.

5. In the Select column, mark the **Select** option for the Java archive file you want to overwrite, and click **Overwrite Existing Jar**.

   The Upload File page appears.

6. Click **Browse...** to identify the Java archive you wish to upload in place of the existing file.

   The File Upload window appears.

7. In the File Upload window:

   a. Locate the appropriate JAR or ZIP file from your local file system or network.

   b. Highlight the file name.

   c. Click **Open**.

   The Upload File page refreshes. The FILE field contains the path and file
   name of the designated file.

8. To clear this selection and specify another file, click **Reset**.

9. To complete uploading the selected file, click **Upload**.

The Jar Files page appears. An informational message indicates the logical
name of the file you successfully uploaded.

See Also

- Viewing and Removing Uploaded Java Archive Files

- Uploading New Java Archive Files

- Changing the Global Property of Uploaded Java Archive Files

- Creating, Removing, and Viewing Java Archive Mapping Definitions

- Specifying Load Order of Java Archives Files

7.2.2.4 Changing the Global Property of Uploaded Java Archive Files

Java archive files uploaded using the Scripting Administration console have a
global attribute. The data type for this attribute is Boolean, and defaults to false or
null. To enable or disable this attribute for a selected code archive, you must
explicitly select or clear the global attribute. Code archives specified as global are
loaded into the class loader for all scripts executed in the Scripting Engine.

Java code archives designated as global are loaded into the class loader for all
scripts executed in the Scripting Engine in the specified environment, regardless of
whether the scripts were deployed from the Script Author applet or from a
stand-alone version of the Script Author application (use release Script Author
11.5.6 or later for compatibility with an 11.5.6 or later Scripting Engine).

Methods referenced in a global code archive need not be mapped to a specific script
in order to execute at runtime. Each method in a global code archive is
automatically accessible for execution in the Scripting Engine in any interface. You
can change the global property at any time from the Scripting Administration
console.

Use this procedure to select or clear the global property for one or more code
archives.

Prerequisites

■ Each Java archive file for which you want to change the global property must be uploaded to the applications database using the Scripting Administration Console.

■ Each global Java archive file for which you want to clear the global property must be set as global using the Scripting Administration Console.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Jar Listings subtab.

   The Jar Files page appears. A summary list of all uploaded JAR files meeting the filter criteria in the View list appears in the Jar Files table.

3. If required, change the set of records displayed in the Jar Files table list as appropriate to display the desired code archives.

4. If you want to set the global property:

   a. For each Java code archive file displayed on the page that you want to set as global, in the Select column, click the Select option.

   b. For each Java code archive file displayed on the page that you want to set as global, in the Global column, select the **Global** option.

      A check mark indicates that the option is selected.

5. If you want to clear the global property:

   a. For each global Java code archive file displayed on the page for which you want to clear the global property, in the Select column, click the Select option.

   b. For each appropriate global Java code archive file displayed on the page, in the Global column, clear the **Global** option.

      An empty check box indicates that the option is cleared.

6. To complete the change to the global property of each selected code archive, click **Update**.

The Jar Files page refreshes. For each code archive you selected (and any code archive previously set as global), the Global option is selected. For each code archive previously marked as global that you cleared, the Global option is cleared.

7. To change the global property for other archives, repeat steps 3 through 6 above.

See Also

- Viewing and Removing Uploaded Java Archive Files

- Uploading New Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives

- Creating, Removing, and Viewing Java Archive Mapping Definitions

- Specifying Load Order of Java Archives Files

7.2.2.5 Creating, Removing, and Viewing Java Archive Mapping Definitions

Java archive files supporting custom Java methods can be mapped to specific scripts, or specified as applying globally to all Oracle Scripting scripts. Methods in mapped files are only loaded into the class loader by the Scripting Engine if it identifies an appropriate script mapping. Code archives specified as global are loaded into the class loader for all scripts.

An uploaded Java archive can be mapped to zero, one, or many deployed scripts. When a Java archive is mapped to one or more deployed script, a corresponding record is created in the IES_SCRIPT_JARFILES table tracking this relationship. When a mapping definition is removed, the mapping record is deleted from this database table and no longer appears in the mapping list. An existing mapping definition must be deleted before the mapped Java archive can be removed.

Java archive files supporting custom Java beans do not need to be mapped to any specific script. The Java beans will be available to run as required in each script containing Java beans.

Use the Jar Mapping subtab to map uploaded Java archive files to apply to specific scripts, to remove mapping definitions, to view defined mappings, and to specify load order.

Prerequisites

- At least one Java archive file must be uploaded to the applications database using the Scripting Administration Console.

- At least one script must be deployed to the applications database from the Script Author.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Jar Mapping subtab.

   The Script Jar Files Mapping page appears, containing a Script Jar Files Mapping area and a Create Mapping area. A summary list of all existing mappings meeting the filter criteria in the View list appears in the Script Jar Files Mapping table.

3. Optionally, request additional Java archive mappings to be listed by changing the View By selection or by clicking the First, Previous, Next, or Last navigation hyperlinks at the bottom of the page.

4. To create a new mapping, in the Create Mapping area, do the following:

   a. From the **Select Script** list, select a deployed script.

   b. From the **Select Jar File** list, select the Java archive you want to map to the designated script.

   c. In the Load Order field, enter an integer representing the load order of this Java archive.

   d. Click **Create Mapping**.

      The Script Jar Files Mapping page refreshes, displaying the new mapping definition in the table.

5. To remove one or more existing mappings, in the Script Jar Files Mapping area, do the following:

a. In the Select column, select each record representing a mapping you want to delete.

A checkmark in the Select column indicates the record is selected.

b. Click **Remove**.

The Script Jar Files Mapping page refreshes. The records you selected for deletion are no longer listed in the table, and the corresponding information has been deleted from the database.

See Also

- Viewing and Removing Uploaded Java Archive Files

- Uploading New Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives

- Changing the Global Property of Uploaded Java Archive Files

- Specifying Load Order of Java Archives Files

7.2.2.6 Specifying Load Order of Java Archives Files

Load order refers to the order in which the Scripting Engine loads mapped Java archive files. If a Java method of the same name is included in two or more Java archives, and you want to control which method is used, specify the desired archive as having a higher load order. This is chiefly relevant only when methods of the same name have differences in the code.

There is currently no verification to ensure that specified load order of Java archives do not overlap. Thus, the Scripting Engine will load Java archives of the same load order level in the order they are returned to the Scripting Engine from its query of the database.

Use this procedure to create or modify the load order for uploaded Java archives mapped to specific scripts.

Prerequisites

- At least one Java archive file must be uploaded to the applications database using the Scripting Administration console.

- At least one script must be deployed to the applications database from the Script Author.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration Console, click the Administration tab.

2. From the Administration tab, click the Jar Mapping subtab.

   The Script Jar Files Mapping page appears, containing a Script Jar Files Mapping area and a Create Mapping area.

3. To modify an existing load order, from the Script Jar Files Mapping list, change the value in the Load Order field to a different integer and click **Update**.

   The Script Jar Files Mapping page refreshes. The modified load order is listed accordingly.

4. To create a load order, you must first specify mapping information. In the Create Mapping area, from the **Select Script** list, select the Java archive file for which you want to specify load order.

5. In the **Load Order** field, type a whole number indicating the intended load order, where the lowest number indicates earlier load order.

6. To save this mapping, including load order, click **Create Mapping**.

   The Script Jar Files Mapping page refreshes. Any new mappings are included in the Script Jar Files Mapping list. The load order is a modifiable field.

Guidelines

- Load order is only relevant for mapped Java methods.

- Load order is only relevant when Java methods of the same name may contain different code.

See Also

- Viewing and Removing Uploaded Java Archive Files

- Uploading New Java Archive Files

- Overwriting Existing Java Archive Files with Updated Archives

- Changing the Global Property of Uploaded Java Archive Files
- Creating, Removing, and Viewing Java Archive Mapping Definitions

7.3 Generating and Viewing Panel Footprint Reports

Use the Reports tab of the Scripting Administration console to generate, view and analyze the panel footprint report.

Other reports supporting Oracle Scripting are available through Oracle Business Intelligence.

7.3.1 Generating a Panel Footprint Report

The panel footprint report indicates, by panel per script, what panels were visited and the duration of time (in milliseconds) spent in each panel. The business objective of running this report is to see how effective a script is at leading an agent through a dialog with a customer (for the interaction center) or determining which questions may have been the most confusing (for survey respondents). Essentially, this report helps in tuning a script for maximum clarity and streamlined flow. It is typically of most use in the interaction center to reduce talk time and improve a script's efficiency. Note that if this report is not used, then footprinting should be disabled at the script level to conserve system resources.

Use this procedure to generate a panel footprint report.

Prerequisites

- You must know the script name, reporting period start date, and reporting period end date to run this report.

Login

Log into Oracle HTML-based applications.

Responsibility

Scripting Administrator

Steps

1. From the Scripting Administration console, click the Reports tab.

The Panel Footprint Summary page appears.

2. In the Select a Script field, type search criteria for the specific script for which you want to run the report and click **Go**.

   - If required, then you can further refine your search and click **Go**.

   - You can use partial search criteria and the % wildcard character if desired. To see a list of every available script, just type % and click **Go**.

   A hyperlinked list of records which meet the specified search criteria, if any, will appear.

3. Click the appropriate hyperlinked script name.

   The Panel Footprint Summary page refreshes. The selected script is listed in the Select a Script field.

4. In the Start Date field, type the start date for reporting data, using the appropriate date format. Optionally, you can populate this field by selecting a start date using the calendar control.

5. In the End Date field, type the end date for reporting data, using the appropriate date format. Optionally, you can populate this field by selecting an end date using the calendar control.

6. Click **Run Report**.

   The report generates, appearing on the refreshed Panel Footprint Summary page. The report lists which panels were visited and the duration of time (in milliseconds) spent in each panel.

7. Exit the window or run another report as desired.

   Reports are dynamically generated. There is no need to save your work.

See Also

- Scripting Home
- Administering Oracle Scripting Files

8

Survey Campaign Administration Tasks

Administration tasks for Oracle Scripting survey campaigns that can be accomplished using the Survey Administration console include creating, viewing, modifying and deleting survey campaigns, cycles, and deployments. From this user interface, deployments must also be activated. Survey administrators perform these tasks by logging into the Oracle Personal Homepage login as an Oracle Applications user with the Survey Administrator responsibility.

When active survey campaigns receive responses through the execution of the questionnaire script in a Web browser, you can also view responses received from the Survey Administration console.

Targeted deployments, which leverage the abilities of Oracle Marketing for list management and Oracle One-to-One Fulfillment for the delivery of invitations and reminders, also require setting up lists, and administering invitations and reminders and associating them with templates. The Audience and Invitations tabs, hosted in the Survey Administration console, provide these capabilities, respectively.

In order to support targeted deployments and ensure survey administrators have the most recent information available, administration of concurrent programs is also required. These steps, performed in Oracle Forms-based applications as an Oracle Applications user with the iSurvey User responsibility, are described here.

This section includes the following topics:

- Section 8.1, "Survey Home"

- Section 8.2, "Administering Survey Campaigns"

- Section 8.4, "Administering Concurrent Programs for Survey Execution"

8.1 Survey Home

Upon logging into the Survey Administration console, the Home tab appears. At this time, the Survey home page contains only identifying information for the page. There are no navigable links or destinations for this category, and no functions associated with this page.

Tasks

You can perform the following tasks:

- Navigating the Survey Administration Console

- Using the View List Filter

- Setting Survey Administration General Preferences

See Also
- Administering Survey Campaigns

- Administering Concurrent Programs for Survey Execution

8.1.1 Navigating the Survey Administration Console

Tab Navigation

There are five tabs in the Survey Administration console user interface (UI). These tabs provide access to the following functions:

Tab	Scripting Function	Tasks You Can Accomplish
Home	Identify user interface	None
Survey Campaigns	Administer survey campaigns	- Create survey campaigns and cycles - View and modify survey campaign details - Add additional cycles to existing survey campaigns or modify cycle details - Delete survey campaigns and cycles - Define deployments - View and modify deployment details - Delete deployments - View responses for completed surveys

Tab	Scripting Function	Tasks You Can Accomplish
Survey Resources	Administer survey resources	■ Create survey resource definitions ■ View and modify survey resource definitions ■ Delete survey resource definitions
Audience	Administer Oracle Marketing lists for targeted survey deployments	■ Create, import, or segment a list ■ Use Oracle Discoverer to build a list using the Discoverer workbook ■ Search lists ■ Administer the organization or person for list members
Invitations	Administer Oracle One-to-One Fulfillment invitations and reminders for targeted survey deployments	■ List Oracle One-to-One Fulfillment invitation and reminder master document details ■ Create and upload new master documents ■ Upload existing master documents ■ Create queries for master documents ■ Create a template identifying master documents ■ Verify the status of a fulfillment request ■ Check fulfillment request history.

The Audience and Invitations tabs provide access to user interfaces owned by other Oracle applications: Oracle Marketing and Oracle One-to-One Fulfillment, respectively. Oracle Marketing provides list management and campaign management functionality. Oracle One-to-One Fulfillment is used for creation, management, deployment and delivery of invitations and reminders. For survey administration, these two tabs are only required for targeted (list-based) survey deployments. As these products are documented elsewhere in appropriate product-specific documentation, only their functionality specific to using Oracle Scripting is addressed in this document.

See Also

■ Using the View List Filter

■ Setting Survey Administration General Preferences

8.1.2 Using the View List Filter

In the Survey Campaigns and Survey Resources pages of the Survey
Administration console, there is a **View** list filter. This control is a
context-sensitive drop-down list which serves as a filter for the current page.

The active filter criteria in the View list determines whether the items shown on
the page for the current category (survey campaigns or survey resources)
display only the set of those items created by an Oracle Applications user with
your login, or display *all* items in that category.

For example, on the Survey Campaigns page, the default option selected in the
View list is **My Survey Campaigns**. The records displayed in this list (if any)
are those survey campaigns created with the current Oracle Applications user
account (login). If you select the other option (**All Survey Campaigns**) from the
View list, the page refreshes, listing all survey campaigns (if any) created by all
users in your environment.

This topic describes how to change the **View** drop-down selection and apply
this change to the current page in the Survey Administration console. Although
the procedure below uses My Survey Campaigns as an example, the same
procedure applies to survey resources.

Prerequisites

None

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, check the status of the View list.

   > **Note:**   The View list only appears on pages containing summary
   > lists, and is context-sensitive to the type of objects displayed in the
   > list.

If the current value reads **My Survey Campaigns**, the resulting list will include only survey campaigns created by the user name you used to log into Oracle Applications.

For example, My Survey Campaigns results in a summary list containing survey campaigns, cycles, deployments, or survey resources, respectively, *exclusively created using your login*.

If the current value reads **All Survey Campaigns**, the resulting list will include *all* items of that category.

2. Click on the **View** list.

   All options appear in the drop-down list of values.

3. If using the mouse, select a value from the list and release the mouse button.

   If using the keyboard, press the tab key until the View list is selected, and use the up or down arrow keys to select a value from the list.

4. When the appropriate choice is selected in the list, click **Go**.

   The page refreshes, displaying a summary list using the new criteria you selected.

See Also

- Navigating the Survey Administration Console
- Setting Survey Administration General Preferences

8.1.3 Setting Survey Administration General Preferences

From the General Preferences page of the Survey Administration console, you can change your current or default responsibility for HTML applications, or affect the way HTML applications display.

The following preferences can be changed on this page:

Profile Category	Profile Option	Description	Example
General Preferences	Known As	First and last name of logged-in user, derived from Known As field from HRMS database.	For user Michael Smith, Mike Smith.
General Preferences	Language	Default language	American English

Profile Category	Profile Option	Description	Example
General Preferences	Territory	Country in which Oracle Applications is in operation.	United States
General Preferences	Timezone	Timezone, based in hours before or after Greenwich Mean Time (GMT)	(GMT -08:00) Pacific Time
General Preferences	Client Character Encoding	Appropriate character encoding for the client computer workstation of logged-in user.	Western European (Windows)
General Preferences	Accessibility Features	To provide additional accessibility to users of assisted technologies.	No
Notification	Notification Style	Electronic mail settings, enabled if logged-in user has default e-mail address defined.	Do not send me mail.
Formatting	Date Format	Defines date format for table listings, database queries, etc.	20-APR-2003
Formatting	Number Format	Determines placeholder usage for thousands place based on standard country usage.	10,000
Password Settings	Old Password	Password used to log into current session.	OLDPASSWORD
Password Settings	New Password	New password intended to govern the remainder of this session and all future sessions until changed again.	NEWPASSWORD
Password Settings	Repeat Password	Repeat the new password entered in the previous field.	NEWPASSWORD

Use the procedure below to set or change general preferences for the logged-in user in the Survey Administration console.

Prerequisites
- To change display preferences, the new selections must be supported in your environment.

Login
Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click **Preferences**.

    The **Preferences** page appears.

2. If you want to change any general, notification, formatting, or password preferences, enter the new setting in the appropriate field using the following guidelines:

    If you want to edit a value appearing in a text field, replace the existing contents of the text field with the desired value.

    If you want to change a value appearing in any list, select the new value from the list using your up and down arrow keys or by clicking the drop-down list control, scrolling to highlight the appropriate value, and releasing the mouse.

    If you want to change your current password, do the following:

    a. In the Old Password field, enter the exact value of the password you used to log into the current Oracle Applications session.

    b. In the New Password field, enter the exact value of the new password you want to use.

        Passwords should be between 5 and 64 characters in length and should not contain spaces. Avoid using special characters such as %, slash, or backslash. Hyphen and underscore characters are supported.

    c. In the Repeat Password field, enter again the exact value of the new password you want to use.

3. Review any modifications.

4. When satisfied, click **Apply**. To exit the application without saving any changes you have made to the fields on this page, click **Cancel**.

    When you apply changes, the page refreshes, and a confirmation appears indicating that your modifications were successfully completed.

    When you click **Cancel**, the page closes and the Navigator appears.

    When viewing subsequent HTML pages in this application, the appropriate settings will now take effect.

5. Continue your work or log out, as applicable.

See Also

- Navigating the Survey Administration Console
- Using the View List Filter

8.2 Administering Survey Campaigns

From an administration perspective, the survey campaign is a collection of top-level requirements for executing an information-gathering survey campaign, or for executing a script in a Web browser. These requirements are stored as a set of objects in the database. To achieve the goals of the campaign, you must create a survey campaign, and define for each survey campaign at least one cycle and at least one deployment. The first cycle for any survey campaign is created at the time survey campaign creation. Additional cycles can be defined later (while the survey campaign status is open or active).

Defining deployments is a separate process that can be performed either immediately after survey campaign and cycle creation, or later (while the survey campaign status is open or active).

Use the Survey Campaigns tab to create, modify, or view survey campaigns, cycles, deployments, or to view responses for active or closed survey campaigns.

Tasks

You can perform the following tasks:

- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign Details
- Deleting Survey Campaigns and Cycles
- Add or Modify a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received

See Also

- Survey Home
- Administering Concurrent Programs for Survey Execution

8.2.1 Creating Survey Campaigns and Cycles

Use this procedure to create a survey campaign and cycle in the Survey Administration console.

Prerequisites

In order to create a survey campaign in the Survey Administration console:

- An appropriate script serving as the survey questionnaire must be deployed to the applications database from the Script Author.

- Survey Resources must already be defined, as described in the section Administering Survey Resources.

  While the corresponding physical JSP files for defined survey resources need not exist on the server in order to *create* a survey campaign, they must reside in the appropriate directory in order to *execute* the survey questionnaire in a Web browser.

- You must know in advance all required parameters of a survey campaign and a cycle before you begin this step.

  Parameters may follow specific naming conventions or otherwise adhere to enterprise or campaign requirements. If in doubt, then consult with the individual responsible for determining survey campaign requirements.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Click **Create** to begin defining a survey campaign.

   The Create Survey Campaign page appears.

3. In the Survey Campaign Name field, type the name you want to use for this survey campaign.

This is the name by which this survey campaign will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

4. Optionally, in the Description field, type a description of this survey campaign.

   For example, enter meaningful information by which survey administrators can quickly discern the purpose of the campaign, its status, group ownership within the enterprise, or other identifying information.

5. From the Status list, verify that the survey campaign status is **Open**.

6. If you want this survey campaign to be defined as a prototype, select the **Prototype** box.

   Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. The purpose is to allow survey campaign administrators more freedom to refine requirements for survey campaigns, including modification to the script for the designated survey campaign.

7. In the Activation Requirements area, from the Questionnaire Script field, click the search icon to begin locating the specific deployed script to designate as the survey questionnaire script for this survey campaign.

   The Search and Select: Questionnaire Script page appears in a new window. This page contains a Search area and a Results area.

8. In the Search area, type specific search criteria in the Search field.

   The only criteria by which to search is the global name of the deployed script (DscriptName). If you do not know the name of the script you want to use, you can use the % wildcard character.

9. When you are satisfied with the search criteria you have typed, click **Go**.

   The Search and Select: Questionnaire Script page refreshes, showing all records (if any) matching your search criteria.

   ■ If the desired deployed survey is not listed in the Results area, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

   ■ If necessary, use the **Previous** and **Next** navigation buttons to view additional records returned by the query.

10. In the Results list, choose the Select option to indicate the appropriate script.

11. Click **Select** to confirm your selection and proceed, or **Cancel** to clear the selected list item.

When you click Select, the Search and Select page closes.

12. Verify that the Questionnaire Script field on the Create Survey Campaign page contains the name of the appropriate script.

13. In the Survey Resources area, perform the following steps for each resource type.

> **Note:** The desired survey resources must be defined before you can associate them with a survey campaign. For more information, see Administering Survey Resources.

Survey resources are files saved in JSP format that provide functionality to a script executing in the Scripting Engine Web interface. Survey resource types include header page, thank you page, error page, and footer page resources.

a. From the text field for the appropriate resource, click **Search**.

The Search and Select page for the appropriate resource type appears in a new window. This page contains a Search area and a Results area.

b. In the Search area, from the Search By list, select the appropriate search criteria.

Logical name is an optional name assigned to the JSP resource when uploading.

Physical name is the name of the JSP resource.

c. In the Search field, type specific search criteria for this resource and click **Go**. You can use the % wildcard character to assist in your search.

The Search and Select page for the appropriate resource type refreshes, showing all records (if any) matching your search criteria.

d. If the desired JSP survey resource is not listed in the Results area, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

If necessary, use the **Previous** and **Next** navigation buttons to view additional records returned by the query.

e. In the Results list, choose the Select option to indicate the appropriate survey resource.

f. Click **Select** to confirm your selection and proceed, or **Cancel** to clear the selected list item.

When you click Select, the Search and Select page closes.

g. Verify that the Questionnaire Script field on the Create Survey Campaign page contains the name of the appropriate script.

h. Verify that the resource field on the Create Survey Campaign page contains the name of the appropriate resource.

14. In the Create Cycle area, in the Create Cycle field, type an appropriate cycle name for this survey campaign.

This is the name by which this cycle will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

15. Click **Create**.

The new survey campaign and cycle information are created, and all defined fields are saved to the database.

The Survey Campaign Details page appears, similar to the Create Survey Campaign page but with a Deployments area, from which you can define a deployment for the cycle you defined in step 14 above.

Guidelines

- The fields required to *create* and *save* a survey campaign and cycle include: Survey Campaign Name, Survey status (required to be Open), Questionnaire Script, and Cycle Name.

- In order to *execute* a survey campaign, the following additional fields are required: Header Section, Error Page, Final Page, and Footer Page.

- The Prototype option is recommended only for test survey campaigns, since its script does not lock and since an active Prototype survey campaign cannot be converted into a standard locking survey campaign type.

See Also

- Viewing or Modifying Survey Campaign Details

- Deleting Survey Campaigns and Cycles

- Add or Modify a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received

8.2.2 Viewing or Modifying Survey Campaign Details

Once a survey campaign is created and before it is active, you can view or modify its properties at any time.

Use this procedure to view or modify survey campaign details.

Prerequisites

A survey campaign must already be created and saved.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

    The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

    - To view survey campaigns created by all users, select **All Survey Campaigns** from the View list and click **Go**.

    - To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

    - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

The Survey Campaign Details page appears.

4. View survey campaign details or make changes as desired.

5. Click **Update** to save your work.

See Also

- Creating Survey Campaigns and Cycles
- Deleting Survey Campaigns and Cycles
- Add or Modify a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details
- Deleting Deployments
- Viewing Responses Received

8.2.3 Deleting Survey Campaigns and Cycles

Use this procedure to delete one or more survey campaigns and all its child objects (cycles and deployments) from the Survey Administration console and from the applications database.

Prerequisites

- A survey campaign and cycle must exist in order to delete it.
- You must know the name of the survey campaign to delete.
- Survey campaigns cannot contain cycles with active deployments.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click the Survey Campaigns tab.

The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   ■ To view survey campaigns created by all users, select **All Survey Campaigns** from the View list and click **Go**.

   ■ To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

   ■ To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. In the Select column of the Survey Campaigns table, check each record that you wish to delete.

   > **Note:** To delete multiple survey campaigns in one step, all survey campaigns you want to delete must appear on the page. If you cannot adjust search criteria to view all relevant survey campaigns, you must search for and delete survey campaigns separately.

4. Click **Delete**.

   The Survey Campaigns page refreshes.

5. Verify that the survey campaign you deleted no longer appears in the Survey Campaigns list.

   When a survey campaign is deleted, its child objects are also deleted.

See Also

■ Creating Survey Campaigns and Cycles

■ Viewing or Modifying Survey Campaign Details

■ Add or Modify a Cycle for an Existing Survey Campaign

■ Defining Deployments

■ Viewing or Modifying Deployment Details

■ Deleting Deployments

- Viewing Responses Received

8.2.4 Add or Modify a Cycle for an Existing Survey Campaign

One cycle is defined for each survey campaign upon creating the survey campaign. Survey campaigns can contain any number of cycles.

Use this procedure to modify the cycle name, or to add a cycle to an existing survey campaign.

Prerequisites

A survey campaign and cycle must already be created and saved.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   - To view survey campaigns created by all users, select All Survey Campaigns from the View list and click **Go**.

   - To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

   - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

   The Survey Campaign Details page appears.

4. To modify the name of an existing cycle, in the Cycles and Deployments area, do the following:

    a. If the name of the cycle you want to modify for this survey campaign is not already displayed, then from the Cycle Name list, select the appropriate cycle name.

    b. Click **Update Name**.

        The Rename Cycle page appears.

    c. In the Cycle Name field, enter the new name you want to assign to this cycle and click **Update**.

        The Rename Cycle page closes. The Survey Campaign Details page appears.

    d. Verify that the Cycle Name field is updated with the new name. If this survey campaign has more than one cycle, you can see other cycle names by viewing the Cycle Name list.

5. To add a cycle to this survey campaign, in the Cycles and Deployments area, do the following:

    a.   Click **Add Cycle**.

        The Create a Cycle page appears. The parent survey campaign is listed.

    b. In the Cycle Name field, type an appropriate cycle name and click **Create**.

        The Create a Cycle page closes. The Survey Campaign Details page appears.

    c. Verify that the Cycle Name field is updated with the new name. If this survey campaign has more than one cycle, you can see other cycle names by viewing the Cycle Name list.

See Also

- Creating Survey Campaigns and Cycles

- Viewing or Modifying Survey Campaign Details

- Deleting Survey Campaigns and Cycles

- Defining Deployments

- Viewing or Modifying Deployment Details

- Deleting Deployments

■   Viewing Responses Received

8.2.5 Defining Deployments

A deployment is a detailed set of execution requirements for conducting a survey campaign or establishing a Web script. At least one deployment must be defined for each cycle for a survey campaign.

Cycles can contain any number of deployments.

After a deployment is created, it must be activated (deployed) before a script can be executed in the Scripting Engine Web interface.

Use this procedure to define a deployment for a cycle.

Prerequisites

A survey campaign and cycle must already be created.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   ■   To view survey campaigns created by all users, select All Survey Campaigns from the View list and click **Go**.

   ■   To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

   ■   To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

   The Survey Campaign Details page appears.

4. In the Deployments area, click **Create**.

   The Create Survey Deployment page appears.

5. Verify that the parent survey campaign is listed in the Survey Campaign Name field.

6. Verify that the parent survey cycle is listed in the Survey Cycle Name field.

7. In the Deployment Name field, type the name of the deployment you want to create.

   This is the name by which this deployment will be referenced. This name should be specified by the campaign administrator in advance, and must follow enterprise requirements or project naming conventions, if applicable.

8. Verify that the media type listed is **WEB**.

   At this time, no other media type is supported.

9. In the Status field, verify that the **Open** option is selected.

10. In the Start Date field, click the calendar control widget.

    The Pick a Date page opens in a separate window.

11. If necessary, change the calendar settings in the Pick a Date page to display the month and year on which you want this deployment to start.

    To navigate to the correct month and year, click **Previous month** or **Next month**, as appropriate, or select the appropriate month and year from the drop-down lists at the top of the calendar control.

12. Click on the date for which you want this deployment to start.

> **Note:** The deployment start date and time must be in the future, as determined by SYSDATE. If setting the deployment start date for the same day, ensure the time specified is later than the time it will be when you save this deployment.

The Pick a Date page closes. The Create Survey Deployment page appears. The deployment date you selected, and the time 00:00:00 (signifying 12 AM), appear in the Start Date field.

13. Optionally, if you want the deployment time to begin at a time other than 12 AM, modify the time in this field.

Note: Changing the time is required for deployments for which the start date equals the current date.

---

14. Repeat steps 10 through 13 to set the response end date and time for this deployment.

15. In the Deployment Type field:

    ■ For standard deployments only, verify that the deployment type specified is **Standard**. If so, skip to step 42. If not, select **Standard**, click **Go**, and proceed to step 42.

      Standard deployments require no other deployment details. Standard deployments are sometimes referred to as "non-list-based" deployments.

    ■ For targeted deployments only, select **Targeted**, click **Go**, and proceed to the next step.

      The Create Survey Deployment page refreshes, displaying additional fields applicable to targeted deployments.

      Targeted deployments require detailed information to specify a targeted a list of possible survey respondents (functionality provided by Oracle Marketing) and details regarding invitations and reminders sent to the targeted list members (functionality provided by Oracle One-to-One Fulfillment). Targeted deployments are sometimes referred to as "list-based" deployments.

16. In the List Parameters area, from the List Name field, click the search icon to begin locating the Oracle Marketing list that identifies the target audience.

    The Search and Select: List Name page appears in a new window. This page contains a Search area and a Results area.

17. In the Search area, type specific search criteria in the Search field.

The only criteria by which to search is the Oracle Marketing list name (List). If you do not know the name of the list you want to use, you can use the % wildcard character.

18. When you are satisfied with the search criteria you have typed, click **Go**.

The Search and Select: List Name page refreshes, showing all records (if any) matching your search criteria.

- If the desired list is not included in the Results list, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

- If necessary, click **Previous** and **Next**, as appropriate, to view additional records returned by the query.

19. In the Results list, choose the Select option to indicate the appropriate list.

20. Click **Select** to confirm your selection and proceed, or **Cancel** to clear the selected list item.

After clicking **Select**, the Search and Select: List Name page closes. The Create Survey Deployment page appears.

21. In the Create Survey Deployment page, verify that the list you selected appears in the List Name field.

22. In the Maximum Responses Per Person field, if survey campaign requirements specify that list members can participate in a survey or execute a Web script for an established number of times, enter the appropriate integer. If no maximum applies, type **0** in this field.

For example, if each list member must be restricted from executing the questionnaire script in a Web browser more than once, enter 1 in this field.

This value determines how many times an individual (list member) can participate in a survey or execute a Web script. This restriction is based on the unique survey URL for a list member, which includes both the deployment identification code and the unique respondent identification code.

23. Optionally, in the Target Response Percent field, type an integer between 1 and 100.

This value is an administrator-defined target percentage, used in survey reports accessible by users of Oracle Business Intelligence. Reports compare the target percentage indicated against the total number of responses received for a deployment. Thus, the value is provided as a report metric, and is not used to enforce any survey campaign-level business rules.

24. In the Hosting Options area, select a hosting option based on your survey campaign requirements.

- For survey questionnaire campaigns, select **Standalone**.

- For Web scripts hosted from self-service Web applications such as Oracle iSupport, select **Menu Based**.

This setting determines which of the two main Oracle Scripting Java server pages (IESSVYMAIN.JSP or IESSVYMENUBASED.JSP, respectively) will be used to execute a script at runtime.

In the Survey Web URL field, verify that the URL of the current Apache Web server session (used to log into the Survey Administration console) populates by default. If it does not, or if you need to point this to a second Apache Web server based on predetermined requirements, then type the name of the Web server hostname including domain, port, and reference to HTML bin, ending with a slash. For example, enter:

```
http://server2.company.com:8888/OA_HTML/
```

25. In the Invitations area, from the Invitation Template Name field, click the search icon to begin locating the Oracle One-to-One Fulfillment template associated with the desired invitation master document.

The Search and Select: Invitation Template Name page appears in a new window. This page contains a Search area and a Results area.

26. In the Search area, type specific search criteria in the Search field.

The only criteria by which to search is the Oracle One-to-One Fulfillment template name (Template). A template identifies a master document name and any associated queries. In this case, you want to specify a template associated with the appropriate invitation master document. If you do not know the name of the master document you want to use, you can use the % wildcard character.

27. When you are satisfied with the search criteria you have typed, click **Go**.

The Search and Select: Invitation Template Name page refreshes, showing all records (if any) matching your search criteria.

- If the desired invitation template name is not included in the Results list, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

- If necessary, click **Previous** and **Next**, as appropriate, to view additional records returned by the query.

28. In the Results list, choose the Select option to indicate the appropriate invitation template name.

29. Click **Select** to confirm your selection and proceed, or **Cancel** to clear the selected list item.

After clicking **Select**, the Search and Select: Invitation Template Name page closes. The Create Survey Deployment page is visible.

30. In the Create Survey Deployment page, in the Invitation Template Name field, verify that the correct template name appears.

31. In the Invitation E-mail Subject Heading field, type the name you want to appear as the subject line of the outgoing message created using the invitation master document

For example, type **Please participate in our survey.**

32. If no reminders are required, skip to step 42.

33. In the Reminders area, from the Reminder Template Name field, click the search icon to begin locating the Oracle One-to-One Fulfillment reminder master document.

The Search and Select: Reminder Template Name page appears in a new window. This page contains a Search area and a Results area.

34. In the Search area, type specific search criteria in the Search field.

The only criteria by which to search is the Oracle One-to-One Fulfillment template name (Template). A template identifies a master document name and any associated queries. In this case, you want to specify a template associated with the appropriate reminder master document. If survey campaign requirements dictate, this may be the same template used to specify the invitation. If you do not know the name of the master document you want to use, you can use the % wildcard character.

35. When you are satisfied with the search criteria you have typed, click **Go**.

The Search and Select: Reminder Template Name page refreshes, showing all records (if any) matching your search criteria.

- If the desired reminder template name is not included in the Results list, or if a large number of records were returned, refine your search criteria in the Search field and click **Go**.

- If necessary, click **Previous** and **Next**, as appropriate, to view additional records returned by the query.

36. In the Results list, choose the Select option to indicate the appropriate reminder template name.

37. Click **Select** to confirm your selection and proceed, or **Cancel** to clear the selected list item.

After clicking **Select**, the Search and Select: Reminder Template Name page closes. The Create Survey Deployment page is visible.

38. In the Create Survey Deployment page, in the Reminder Template Name field, verify that the correct template name appears.

39. In the Reminder E-mail Subject Heading field, type the name you want to appear as the subject line of the outgoing message created using the reminder master document

For example, type **Reminder: survey closes last day of the month!**

40. In the Number of Reminders field, type an integer based on survey campaign requirements

This number determines how many reminders will be automatically sent to all list members. Upon request, additional reminders can be sent manually regardless of the value entered in this field.

41. In the Reminder Interval In Days field, type an integer.

This number represents, in days, the span of time that will pass between reminders to be sent automatically to all list members.

To determine this value, divide the number of reminders by the duration of the deployment (in days).

42. Review the deployment detail selections you entered for this deployment. When satisfied, click **Create**.

The new deployment information is created, and all defined fields are saved to the database. The Deployment Details page appears.

To modify deployment details or activate a deployment, proceed to step 6 in the task Viewing or Modifying Deployment Details.

See Also

- Creating Survey Campaigns and Cycles

- Viewing or Modifying Survey Campaign Details

- Deleting Survey Campaigns and Cycles

8.2.6 Viewing or Modifying Deployment Details

Use this procedure to view or modify details for an existing deployment, or to activate a deployment.

If performing this procedure immediately after creating or updating a deployment, begin with step 6.

Prerequisites

A survey campaign, cycle, and deployment must already be created.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   - To view survey campaigns created by all users, select All Survey Campaigns from the View list and click **Go**.

   - To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

   - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

The Survey Campaign Details page appears.

4. In the Cycles and Deployments area, if the cycle with which the desired deployment is associated is not displayed, then from the Cycle Name list, select the appropriate cycle and click **Go**.

The Survey Campaign Details page refreshes.

5. In the Deployments area, from the Deployment Name column, click the appropriate deployment name.

The Deployment Details page appears.

6. In the Deployment Details page, select **Deployment View**.

The Deployment Details page refreshes. This page has two views. In the deployment view, you can modify deployment information or activate this deployment. In the response view, you can see the results of any responses received for this deployment.

7. If you want to view or modify deployment details, then do the following:

a. Optionally, to view deployment details specific to targeted deployments, then from the Deployment Type field, select **Targeted** and click **Go**.

The Deployment Details page refreshes, displaying additional fields applicable to targeted deployments.

b. Make any changes as appropriate.

c. When satisfied, click **Update**.

8. If you want to activate this deployment, click **Deploy**.

The Deployment Details page refreshes. For an active deployment, the only editable fields include Status and Response End Date.

For standard deployments, the Survey URL field contains an active hyperlink to the active deployment, including the message **Please click here to take the survey.**

Note: If you cannot see the hyperlink, click **Deploy** a second time.

For targeted deployments, the Survey URL field contains the URL of the current Apache Web server session.

9. If you want to participate in this survey:

For standard deployments, access the URL in the Survey URL field.

> **Note:** If you access this link using the same Web browser (for example, to test this deployment), you will invalidate your current Oracle Applications session. This will require a subsequent login to view details in the Survey Administration console.

For targeted deployments, access an invitation e-mail message sent from the fulfillment server and access the unique URL it contains.

10. If you want to obtain the deployment ID for a standard deployment, position the mouse over the link and view the deployment identification number (dID) in the Web browser status bar.

See Also

- Creating Survey Campaigns and Cycles

- Viewing or Modifying Survey Campaign Details

- Deleting Survey Campaigns and Cycles

- Add or Modify a Cycle for an Existing Survey Campaign

- Defining Deployments

- Deleting Deployments

- Viewing Responses Received

8.2.7 Deleting Deployments

Use this procedure to delete one or more deployments from the Survey Administration console and from the applications database. Deleting a deployment does not otherwise affect the parent cycle, or its parent object, the survey campaign.

Prerequisites

- A survey campaign and cycle must exist in order to delete it.

- You must know the name of the survey campaign, cycle, and deployment in order to delete the deployment.

- The deployment must not be active.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. From the Survey Administration console, click the Survey Campaigns tab.

    The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

    - To view survey campaigns created by all users, select **All Survey Campaigns** from the View list and click **Go**.

    - To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

    - To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

    The Survey Campaign Details page appears.

4. In the Cycles and Deployments area, if the cycle with which the desired deployment is associated is not displayed, then from the Cycle Name list, select the appropriate cycle and click **Go**.

    The Survey Campaign Details page refreshes.

5. In the Cycles and Deployments area, in the Select column, check each deployment that you wish to delete.

> **Note:** To delete multiple deployments in one step, all deployments you want to delete must appear on the page. If you cannot adjust search criteria to view all relevant deployments, you must search for and delete deployments separately.

6. Click **Delete**.

   The Survey Campaign Details page refreshes.

7. Verify that the deployments you deleted no longer appear in the Deployments list.

See Also

- Creating Survey Campaigns and Cycles

- Viewing or Modifying Survey Campaign Details

- Deleting Survey Campaigns and Cycles

- Add or Modify a Cycle for an Existing Survey Campaign

- Defining Deployments

- Viewing or Modifying Deployment Details

- Viewing Responses Received

8.2.8 Viewing Responses Received

Use this procedure to view individual responses to a script executed in a Web browser.

Prerequisites

- A survey campaign, cycle, and deployment must already exist.

- At least one deployment must be active.

- At least one individual must already have executed a script in a Web browser. For targeted survey campaign deployments, at least one list member must have responded by taking a survey.

- You must know the specific survey campaign, cycle and deployment for which you want to review individual responses.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

    The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

    ■   To view survey campaigns created by all users, select All Survey Campaigns from the View list and click **Go**.

    ■   To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

    ■   To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

    The Survey Campaign Details page appears.

4. In the Cycles and Deployments area, if the cycle with which the desired deployment is associated is not displayed, then from the Cycle Name list, select the appropriate cycle and click **Go**.

    The Survey Campaign Details page refreshes.

5. In the Cycles and Deployments area, click **Response View**.

    The Survey Campaign Details page refreshes. In the Deployments area, deployments for which responses have been received (if any) are listed in the deployments table.

6. In the Deployments area, from the Deployment Name column, click the appropriate deployment name.

    The Deployment Details page appears. Individual responses received for this deployment (if any) are listed in the Responses table.

7. From the Answer column, click the **View Answers** icon.

   The Survey Responses page appears for this respondent. A table appears with an expandable navigation tree for the respondent. Subsequent navigation trees appear for each panel and question.

8. To expand or collapse all navigation trees for each panel or question at any time, click **Expand Al**l or **Collapse All**, as appropriate.

9. For any panel or question, from the Focus column, click the focus icon to focus in on that aspect of the response.

   An uncluttered view of the panel or question appears. Navigation links at the top of the table allow you to navigate back to the parent question or panel.

10. From the middle column, click the navigation tree.

    Additional rows appear, one for each panel visited by the respondent, and each containing its own expandable navigation tree.

11. To view all the questions in a panel, click the appropriate panel's navigation tree.

    Additional rows appear, one for each question in the panel. Each row contains its own expandable navigation tree to view any answer choices or lookup values included in the question.

12. To view the answer choices available and see which choice the respondent made, click the appropriate question's navigation tree.

    The answer choice selected by this respondent has an entry in the Answer column of the table.

    Additional rows appear, one for each question in the panel. For each answer choice or lookup value provided in this question, another row appears. For questions with only one possible answer choice, only one row appears.

See Also

- Creating Survey Campaigns and Cycles
- Viewing or Modifying Survey Campaign Details
- Deleting Survey Campaigns and Cycles
- Add or Modify a Cycle for an Existing Survey Campaign
- Defining Deployments
- Viewing or Modifying Deployment Details

■ Deleting Deployments

8.2.9 Sending Reminders Manually to List Members

Use this procedure to manually send a reminder message to a specified list member as part of a survey campaign deployment. This procedure results in initiating a fulfillment request to send out the reminder, just as if the reminder were scheduled. This will not interfere with the schedule for any currently scheduled reminders.

Prerequisites

■ A survey campaign, cycle, and deployment must already exist.

■ At least one deployment must be active.

■ The active deployment must be targeted (include valid list criteria).

■ The active deployment must include reminder information in the deployment definition.

■ You must know, or be able to determine, the list member for whom you want to send reminders.

Login

Log into Oracle applications using the Personal Homepage login.

Responsibility

Survey Administrator

Steps

1. Click the Survey Campaigns tab.

   The Survey Campaigns page appears.

2. Optionally, change the set of records displayed by modifying filter or display criteria using the appropriate method:

   ■ To view survey campaigns created by all users, select All Survey Campaigns from the View list and click **Go**.

   ■ To filter survey campaigns designated as prototypes out of the list of displayed survey campaigns, check the Exclude Prototypes box and click **Go**.

- To navigate through the set of records returned by the active list criteria, click **Previous** or **Next**. These navigation links appear when a specified minimum number of records (typically ten) is displayed.

3. From the Campaign Name column, click the appropriate survey campaign name.

   The Survey Campaign Details page appears.

4. In the Cycles and Deployments area, if the cycle with which the desired deployment is associated is not displayed, then from the Cycle Name list, select the appropriate cycle and click **Go**.

   The Survey Campaign Details page refreshes.

5. In the Cycles and Deployments area, click **Response View**.

   The Survey Campaign Details page refreshes. In the Deployments area, deployments for which responses have been received (if any) are listed in the deployments table.

6. In the Deployments area, from the Deployment Name column, click the appropriate deployment name.

   The Deployment Details page appears. Individual responses received for this deployment (if any) are listed in the Responses table.

7. From the Send Reminder column, click the **Send Reminder** icon for the appropriate list member.

   The Deployment Details page refreshes. If successful, a confirmation message appears.

   > **Note:** Reminders can only be sent manually for active targeted deployments for which reminder details are established.

Guidelines

- A confirmation message when manually sending a reminder does not indicate that the reminder was successfully sent to the list member. Instead, this indicates that the fulfillment request is successfully sent to the fulfillment server.

- To confirm delivery, sign into Oracle One-to-One Fulfillment as a fulfillment administrator.

See Also

- Creating Survey Campaigns and Cycles

- Viewing or Modifying Survey Campaign Details

- Deleting Survey Campaigns and Cycles

- Add or Modify a Cycle for an Existing Survey Campaign

- Defining Deployments

- Viewing or Modifying Deployment Details

- Deleting Deployments

8.3 Setting Up Invitations and Reminders

For targeted deployment survey operations, list members are sent an *invitation* to participate in a survey. This invitation is an HTML-formatted e-mail message sent from the fulfillment engine (a component of Oracle One-to-One Fulfillment) based on a list designated from Oracle Marketing. If desired, a second e-mail message can be sent as a *reminder*. Reminders can be sent to all list members on a scheduled basis, or manually for one or more list members upon request.

Invitations and reminders are fulfillment master documents. From a Survey perspective, they are only required for *targeted (list-based)* survey campaigns. Administration of master documents is typically a function of fulfillment administrators. This section includes a discussion of the theory and suggested Oracle One-to-One Fulfillment administration steps, most of which may be performed from the Survey Administration console. For more information on any of the following topics, consult with the fulfillment administrator for your environment, or refer to *Oracle One-to-One Fulfillment Implementation Guide.*

This section includes the following topics:

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

See Also

- Survey Home

- Administering Survey Campaigns

- Administering Concurrent Programs for Survey Execution

8.3.1 Oracle One-to-One Fulfillment Terminology

A **template** is a high-level definition of one or more Oracle One-to-One Fulfillment master documents, associated queries, and any related collateral. Templates may also include campaign specifications. From an Oracle Scripting perspective, targeted survey deployments call upon a template that identifies an invitation master document (and its associated query). If reminders are also required for the survey deployment, and the text of the reminder document differs from the invitation, a second template defining the reminder master document and its associated query is required. No collateral is typically associated with a survey campaign.

A **master document** is a message sent from Oracle One-to-One Fulfillment. These typically include merge fields populated by an associated SQL query. From a survey perspective, master documents are invitation or reminder messages in HTML format that are sent by e-mail from the fulfillment engine to members of a defined Oracle Marketing list.

Merge fields are data elements in an Oracle One-to-One Fulfillment master document surrounded by merge field delimiters. In print, merge field delimiters appear using open and close guillemet characters, as in this sample: «merge_field». These are represented in HTML by the ASCII codes **«** (for open or left delimiter) and **»** (for right or close delimiter). When a fulfillment request is executed, a query is performed to retrieve the merge data from the table columns or views specified in the query. For each merge field used in a master document, there must be one corresponding request in the query to retrieve the appropriate data. Invitation or reminder master documents contain at least one merge field, to retrieve the survey URL that the recipient must access to participate in the survey. Contact information for each list member (title, first name and last name) and deployment end date are other merge fields frequently included in invitation and reminder master documents.

Queries are database requests for information constructed in structured query language (SQL). Each Oracle One-to-One Fulfillment master document must have an associated query which obtains information populated into the resulting merged document in placeholders called merge fields. A query must contain a request for each data element included as a merge field in the master document. The query may contain requests for additional data elements as well (providing the requests are valid), regardless of whether the data elements requested in the query are used. Thus, for example, you can construct a single query that requests all information used to populate an invitation master document and a reminder master document, even if they include different merge fields. A single query can be associated with any number of master documents. Return data from a successfully executed query is merged with the master document. In this process, the master document is replicated once for each row returned. For every replicated copy, each merge field placeholder is replaced with its corresponding data element from the query. The amount of merged master documents will match the number of list records in the Oracle Marketing list, one per list member. Each merged document is then sent via e-mail to the e-mail address of each list member by the fulfillment server.

See Also

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.2 Fulfillment Activities for Targeted Survey Deployments

For ease of administration, survey administrators have access to an Invitations tab in the Survey Administration console. This tab gives access to the Oracle One-to-One Fulfillment functions required for performing survey administration for targeted deployments.

From the Invitations tab, survey administrators can administer fulfillment templates and queries and verify fulfillment request status and history.

Some Functions Not Available to Survey Administrators

Administration of fulfillment servers or fulfillment groups is intended to be performed by a knowledgeable fulfillment administrator. As such, these functions (available in the Fulfillment Administration console using the Server and Group tabs) are excluded from the user interface in the Survey Administration console. Additionally, if fulfillment requests have successfully been sent to the fulfillment server but have failed to execute, a fulfillment administrator is required to determine the cause and to correct the error. A targeted deployment with a status of Active indicates only that the fulfillment request was successfully submitted to the server, and does not indicate successful transmission of the invitation or master document e-mail messages.

Since Oracle One-to-One Fulfillment implementation and functionality is a prerequisite of survey operations, a fulfillment administrator should already exist in the enterprise who has access to the Fulfillment Administration console to perform these administration steps.

Required Tasks

From an Oracle One-to-One Fulfillment perspective, you need to accomplish the following tasks to create and administer invitations and reminders:

- Create and format a master document in HTML

- Insert merge field references into the master document

- Upload the master document to MES

- Create a SQL query to populate merge fields into master document

- Associate the query to a specific master document

- Create a fulfillment template that identifies a master document and its associated query

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.3 Recommended Order of Steps

The recommended order of steps for administering invitations and reminders using Oracle One-to-One Fulfillment functionality is as follows:

1. Identify merge fields to be populated in the master document, and identify locations of tables or views in the Oracle Applications database schema from which the merge fields will be populated.

2. Create SQL query, including one merge field for each table field queried.

   A single query can be used for any number of master documents, providing that the query retrieves the data for each required merge field in each master document associated with the query.

3. Create invitation and reminder master documents in HTML using a text editor or HTML tool of your choice, or the Oracle One-to-One Fulfillment master document editor.

4. Upload master documents to MES.

   If using the master document editor feature of Oracle One-to-One Fulfillment, this step is performed automatically.

5. Associate each required master document with its query.

   If defining the query using the master document editor feature of Oracle One-to-One Fulfillment, this step is performed automatically.

6. Define templates.

   One template is required for each master document. If a survey campaign includes invitation and reminder documents with the same text, only one template need be created. If a survey campaign includes invitation and reminder documents with different content, create one template for each master document.

7. Associate template with the required master document.

Two Approaches

If you have already created an HTML document to serve as your invitation or reminder master document, you can upload this to MES from the Survey Administration console (Invitations tab > Master Document subtab > **Upload**). Then you can associate the master document with a query you have created or identified, and then create a template that includes that master document (the query is associated at the master document level).

Additionally, Oracle One-to-One Fulfillment now has a master document editor. Using this feature, you can quickly and easily perform most tasks associated with the creation of invitation or reminder master documents and their requisite queries.

In sequence, you can create a SQL query, validate the query, create and format a customized master document, insert merge fields identified by the query, and automatically associate the query with the master document. Following this, you simply create a template, associating it with your master document.

Reference

For more information, refer to *Oracle One-to-One Fulfillment Implementation Guide.*

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.4 Invitations

Typically, an invitation is an HTML message inviting list members to take a survey. The invitation is referred to in Oracle One-to-One Fulfillment as a master document. Using Oracle One-to-One Fulfillment APIs, invitations can be customized by populating data from RDBMS tables.

If personalized, the HTML document contains placeholders called **merge fields**, in which information from a query is placed by the fulfillment server upon execution. These might typically include a list member's title, first name, and last name. While such personalizations are not required, a Survey URL merge field *is* required. In list-based scenarios, a unique URL is generated for each list member (using dID and rID as parameters in the URL string). In this way, if a survey campaign has a business rule to allow list members to only take a survey one time, this rule can be enforced.

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.5 Reminders

If required by a survey campaign, reminders can also be sent to list members. Like the invitation, a reminder is an e-mail message identified in Oracle One-to-One Fulfillment as a master document. It can be the same e-mail message initially sent to list members (in which case, only a single template is required), or it can be a different message (requiring a second template to identify the master document and its associated query). If using the same template and master document, you have the option of using a different e-mail message subject. If using a separate e-mail message as the reminder, it can be worded as differently as you need. Since queries can be defined once and used many times, you can use the same query that the invitation uses (unless you add other merge fields). Reminders typically emphasize the response end date of the specific deployment for that survey campaign. Like invitations, each reminder must include the survey URL as a merge field, and must have a corresponding request for that data field in its associated query. Unlike invitations, reminders are an optional feature. In addition to being sent

automatically based on reminder requirements defined at the deployment level, reminders can be sent by a survey administrator manually upon request.

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.6 Sample Invitation Master Document

The following is a sample invitation master document. Master documents for your enterprise may appear substantially different. For example, this sample includes three merge fields (signified by open and close guillemet characters). Upon delivery, for each list member, one e-mail message is generated, containing the first and last name of the list member and the unique survey URL for this deployment. Only the survey URL merge field is required for any invitation or reminder master document.

Sample Invitation Text

Following is the sample text of the e-mail message:

Hello «FIRST_NAME» «LAST_NAME»,

Please take a moment to participate in this customer satisfaction survey, using your Web browser. Click «SURVEY_URL» to begin.

Sincerely,

Vision Customer Service

Sample Invitation Code

Following is the HTML code of the above e-mail message master document. Note that the open and close guillemet characters which surround merge fields are designated in the HTML sample using ASCII references.

```
<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html">
<META NAME="Author" CONTENT="Anony Mouse">
<META NAME="GENERATOR" CONTENT="Mozilla/4.76 [en] (WinNT; U)
[Netscape]">
<TITLE>Sample Invitation Master Document</TITLE>
</HEAD>
<BODY>
Hello &laquo;FIRST_NAME&raquo; &laquo;LAST_NAME&raquo;,
<p>Please take a moment to participate in this customer satisfaction
survey, using your Web browser. Click &laquo;SURVEY_URL&raquo; to
begin.
<P>Sincerely,</P>
<BR>Vision Customer Service
</BODY>
</HTML>
```

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.7, "Sample Reminder Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.7 Sample Reminder Master Document

The following is a sample reminder master document. Master documents for your enterprise may appear substantially different. In this sample, the reminder uses the same three merge fields as the original invitation, and is associated with the same query. Note, however, that a reminder can include different merge fields, as long as the associated query provides that information.

Sample Reminder Text

Following is the sample text of the e-mail message:

Dear «FIRST_NAME» «LAST_NAME»,

Several weeks ago, you were sent an invitation to participate in our customer satisfaction survey. We will soon be completing the survey process, and collating the results to see how to best improve customer satisfaction.

If you would like your opinions to be considered, please click «SURVEY_ URL» to participate in this online survey using your Web browser.

As always, we appreciate your business and look forward to your opinions to enable us to improve our responsiveness.

Sincerely,

Vision Customer Service

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

- Section 8.3.8, "Sample Query"

- Section 8.3.9, "Using the Master Document Editor"

- Section 8.3.10, "Verifying Status of a Fulfillment Request"

8.3.8 Sample Query

The following is a sample query. The function of this query is to populate the «FIRST_NAME», «LAST_NAME», and «SURVEY_URL» merge fields in the above invitation and reminder master document samples. This query is only a sample. Based on the merge fields included in your master documents, your query may be substantially different.

Each master document may have its own query. Conversely, the same query can be used for numerous fulfillment functions, including multiple invitation and reminder master documents. To be successful, the query must contain a structured query language (SQL) request for each merge field in each associated master document. Note, however, that one query may contain multiple SQL requests, including requests for data that are not used in your master document. The SQL in the query must be properly constructed to return the requested data from tables or views, regardless of whether the data is required for your master document.

The following sample query will populate the «FIRST_NAME», «LAST_NAME», and «SURVEY_URL» merge fields in the above invitation and reminder samples:

```
select
    ams_list_entries.first_name FIRST_NAME,
    ams_list_entries.last_name LAST_NAME,
    '<a href='||IES_SVY_HOSTING_PROF_V.SURVEY_URL||'?dID='||IES_SVY_DEPLYMENTS_
    V.survey_deployment_id||'&rID='||IES_SVY_LIST_ENTRIES_V.respondent_id||'>
    here </a>' SURVEY_URL
from
    IES_SVY_LIST_ENTRIES_V,
    ams_list_entries,
    IES_SVY_DEPLYMENTS_V,
    IES_SVY_HOSTING_PROF_V
where
    ams_list_entries.list_entry_id = IES_SVY_LIST_ENTRIES_V.list_entry_id
and
    IES_SVY_LIST_ENTRIES_V.survey_deployment_id =
    IES_SVY_DEPLYMENTS_V.survey_deployment_id
and
    IES_SVY_HOSTING_PROF_V.survey_deployment_id =
    IES_SVY_DEPLYMENTS_V.survey_deployment_id
and
    IES_SVY_DEPLYMENTS_V.survey_deployment_id = :deployment_id
and
    ams_list_entries.list_entry_id = :party_id
```

See Also

-

-

-

-

-

-

-

-

-

8.3.9 Using the Master Document Editor

Use the following procedure to create a SQL query, and to create and format a master document using merge fields from the query.

Login

Log into Oracle HTML-based applications.

Responsibility

Survey Administrator

Prerequisites

A fulfillment server must already exist.

Steps

1. From the Survey Administration console, click the Invitations tab.

2. Click the Master Document subtab.

   The Master Document page appears.

3. Click **Create and Upload**.

   The Master Document Editor page appears.

4. In the Master Document Name field, type a unique name for the master document.

5. If you want to define a SQL statement and use merge fields identified in the query in your master document, do the following:

   a. In the SQL Merge Field field, click Create Query.

      The SQL Editor page appears.

   b. In the Query Name field, type a unique name for this query.

   c. Optionally, in the Query Description field, type a description for this query.

   d. In the SQL Query field, using well-formed structured query language, type or paste the contents for your query. It must include, at minimum, SQL code to obtain a valid survey URL.

      For example, to create a query for first name, last name, and valid survey URL, copy the contents of the sample query from Section 8.3.8, "Sample Query" and paste in this field.

   e. Click **Validate SQL**.

      The SQL Editor page refreshes. For any merge field that requires a constraint to be defined, you will be directed to enter the constraint value or range.

   f. If prompted, enter constraints for each merge field as appropriate. When finished, click **Validate SQL**.

      The Master Document Editor page refreshes. All merge fields available based on the query you entered are now accessible in the SQL Merge Field list.

6. If you want the master document to appear in plain text, click **Change to Plain text mode**.

   For example, if your master document contains no images or character-level formatting, or if the recipients of your master document will not be receiving the merged document in e-mail capable of interpreting HTML format, select plain text mode.

7. If you want the master document to contain formatting you can apply using the master document editor's rich text format (RTF) formatting tools, keep the default option. To switch modes from plain text mode, select **Change to Rich text mode**.

8. In the Master Document Body field, type the text for your master document.

9. At each appropriate location in the body of your master document, add merge fields to your master document as appropriate.

> **Note:** At minimum, your invitation or reminder master document must contain at least one merge field to generate a survey URL.

10. If entering merge fields based on a SQL query you created on this page, then do the following:

    a. From the SQL Merger Field list, select the appropriate merge field.

    b. Click **Insert**.

    c. Fix any spacing errors in the master document resulting from the inclusion of this merge field.

11. If entering custom merge fields, then do the following:

    a. In the Custom Merge Field field, type the precise merge field name for a value returned from a query.

       For example, type **SURVEY_URL**.

    b. Click **Insert**.

    c. Fix any spacing errors in the master document resulting from the inclusion of this merge field.

       Each merge field appears in the master document body. Merge fields are the fields available from a SQL query.

12. Verify the content and accuracy of your master document.

13. Optionally, if using rich text mode, apply formatting to your master document by selecting text and clicking the appropriate formatting tools.

14. When you have finished, click **Save**.

See Also

- Section 8.3.1, "Oracle One-to-One Fulfillment Terminology"

- Section 8.3.2, "Fulfillment Activities for Targeted Survey Deployments"

- Section 8.3.3, "Recommended Order of Steps"

- Section 8.3.4, "Invitations"

- Section 8.3.5, "Reminders"

- Section 8.3.6, "Sample Invitation Master Document"

8.3.10 Verifying Status of a Fulfillment Request

The Oracle One-to-One Fulfillment Request Status page (accessed from the Survey Administration console by clicking Invitations tab > Status subtab) indicates the status of fulfillment requests. A request status of **Submitted** indicates the fulfillment request has been successfully submitted to the fulfillment engine.

> **Note:** A successfully submitted request does not necessarily signify that the engine succeeded in delivering the master document in question. This status only indicates that the request was sent to the fulfillment engine as appropriate.

If any status other than **Submitted** displays in the Request Status page, consult with a fulfillment administrator.

To verify that the fulfillment engine successfully completed distribution of a fulfillment request, check for an outcome code of **Success** in the Request History page. This can be accessed from the Survey Administration console by clicking Invitations tab > History subtab.

If any status other than **Success** displays in the Request History page, consult with a fulfillment administrator.

References

For specific information on implementing, using, or administering Oracle One-to-One Fulfillment, refer to *Oracle One-to-One Fulfillment Implementation Guide.*

See Also

8.4 Administering Concurrent Programs for Survey Execution

Concurrent programs are executed using Oracle Concurrent Manager, a Forms-based application built on Oracle Application Object Library (AOL) classes. Concurrent Manager is included in Oracle Applications 11*i* with an appropriate Rapid Install. Seeded concurrent programs can be scheduled to execute at certain times, or can be administered to execute in near-real time. To schedule or execute concurrent programs supporting Oracle Scripting's survey component, you must access Oracle Forms-based applications with a user assigned the iSurvey User responsibility.

This section includes the following topics:

- Seeded Concurrent Programs for Survey Execution

- Scheduling or Executing Concurrent Programs

See Also
- Survey Home

- Administering Survey Campaigns

8.4.1 Seeded Concurrent Programs for Survey Execution

Oracle Scripting uses three seeded concurrent programs to support survey component operations. The table below briefly describes these programs:

Concurrent Program	Survey function supported	Recommended Execution Method
SUBMIT GROUP FM REQUEST FROM IES Concurrent Program	Targeted survey deployments	■ Automatically (typical) ■ Manually (when Fulfillment Request ID fails to generate)

Concurrent Program	Survey function supported	Recommended Execution Method
Summarize Survey Data Concurrent Program	Reports generated in Survey Administration console's Analysis tab	■ Scheduled (typical - set up to execute on a periodic basis or on specific days) ■ Manually (on demand when fresh reporting tabulation is desired)
Update Deployment Status Concurrent Program	Survey Administration console table updates	Scheduled (set up to execute on a periodic basis or on specific days, generally once daily)

This section includes the following topics:

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program

- Summarize Survey Data Concurrent Program

- Update Deployment Status Concurrent Program

See Also

- Scheduling or Executing Concurrent Programs

8.4.1.1 SUBMIT GROUP FM REQUEST FROM IES Concurrent Program

When you activate a targeted deployment, a concurrent request is generated which is scheduled to run at the deployment start date and time (specified when defining a deployment). If the deploy start date and time are in the past, the concurrent program executes immediately. The purpose of this concurrent program is to make a request to Oracle One-to-One Fulfillment to send out e-mail invitations to the target population specified by the list.

> **Note:** Immediately upon activating a deployment, when the deployment details page refreshes the corresponding Concurrent Request ID is visible at the bottom of the page. You can use this Concurrent Request ID to monitor the Fulfillment Request submission through the Oracle Forms-based interface associated with the iSurvey User responsibility.

If reminders are also established for this deployment, concurrent requests are also generated (one per reminder) to execute at the specified interval (in days) as defined for the deployment. These requests will execute at the appropriate interval.

When the concurrent manager runs the request, it attempts to submit a fulfillment request for this survey deployment.

If the request is successful, a fulfillment request ID will be generated. This is visible from the Survey Administration console in the Deployment Detail page, immediately above the Concurrent Request ID.

If the request is not successful, you can attempt to manually run this concurrent program using the steps below.

Once a fulfillment request has been successfully submitted to Oracle One-to-One Fulfillment, there is no communication with the Survey Administration console regarding any possible problems in executing the Fulfillment request. Thus, if necessary, obtain the fulfillment request ID from the Deployment Detail page of the Survey Administration console and attempt to remedy using Oracle One-to-One Fulfillment debugging techniques.

Fulfillment request status is visible in the Fulfillment Administration console. In order to access this console, you must log into Oracle HTML-based applications as a user with the Fulfillment Administrator responsibility.

Parameters

The set of parameters for the SUBMIT GROUP FM REQUEST FROM IES concurrent program are listed in the table below.

There are two cases in which this concurrent program needs to submitted manually:

1. If the original deployment fails.

2. If the reminder concurrent program fails.

Parameters 7 and 9 below require specific values, as indicated, if the purpose for manually executing this request is the failure of a previously scheduled concurrent program for reminders.

No.	Parameter	Meaning	Value
1	p_api_version	PL/SQL API Constant required by Apps.	1
2	p_init_msg_list	Another parameter required by Apps	Leave parameter null
3	p_commit	Tells the API whether or not to commit changes. If set to false the concurrent manager commits it instead of the API.	FND_API.G_FALSE
4	p_validation_level	Parameter required by Apps	40

No.	Parameter	Meaning	Value
5	p_deployment_id	Deployment ID for survey campaign deployment	Determine the deployment ID and use it here
6	p_template_id	Template ID for the reminder template used in survey campaign deployment.	Determine the template ID and use it here.
7	p_reminder_type		If deployment concurrent program fails, leave parameter null.
			If reminder concurrent program fails, use REMINDER.
8	p_user_id	User ID of the user submitting the concurrent request	Determine the user ID and provide it here.
9	p_reminder_hst_id	Reminder History ID created by the system when the original deployment is submitted	If deployment concurrent program fails, leave parameter null.
			If reminder concurrent program fails, select the p_reminder_hst_id using the following SQL statement:
| | | | ```
select survey_reminder_hst_id
from ies_svy_reminder_hst_v ,
ies_svy_reminders_v where ies_
svy_reminders_v.deployment_id =
<p_deployment_id> and ies_svy_
reminder_hst_v.survey_reminder_
id = ies_svy_reminders_
v.survey_reminder_id.
``` |

Guidance

There are no prerequisites for executing this concurrent program. However, in order for this program to have any useful effect:

- The Survey component of Oracle Scripting must be fully implemented.

- Oracle One-to-One Fulfillment must be fully implemented and configured, and a Fulfillment server must be functional.

- A survey campaign and cycle must already exist, and a targeted deployment already defined.

- The associated list must be valid and accessible to the system.

- A targeted deployment must be active.

- The deployment start date and time, as compared to SYSDATE, must be in the past.

- Fulfillment request status is visible in the Survey Administration console in the deployment detail. However, if you need to log into the Fulfillment Administration console for detailed troubleshooting, you must log into Oracle HTML-based applications with a user that has the Fulfillment Administrator responsibility.

References

- For more information on implementing, administering, or using Oracle One-to-One Fulfillment, refer to *Oracle One-to-One Fulfillment Implementation Guide*.

- For specific details on executing or scheduling concurrent programs, please refer to Chapter 5 of *Oracle System Administrator's Guide*.

See Also

- Summarize Survey Data Concurrent Program

- Update Deployment Status Concurrent Program

8.4.1.2 Summarize Survey Data Concurrent Program

When interaction center agents execute a script, or when survey respondents participate in a survey, their individual answers are collected in the Oracle Scripting schema of the applications database.

For generating reports on survey data, information must be moved into summary tables that make the data accessible to reporting tools (Oracle Discoverer workbooks) as part of the Interaction Center Intelligence product family.

Run this concurrent program prior to viewing any reports for the most current data, or schedule this program to execute on a regularly scheduled basis (for example, once daily at an off-peak time, when load on the system is not high).

Parameters

There is a single parameter for the Summarize Survey Data concurrent program. The parameter details are listed in the table below.

No.	Parameter	Meaning	Value
1	p_cycle_id	Cycle ID of the specified cycle for which you want to execute this concurrent program.	Locate the appropriate cycle ID from the p_cycle_id list of values and use it here.

Guidance

There are no prerequisites for executing this concurrent program. However, in order for this program to have any useful effect:

- The Survey component of Oracle Scripting must be fully implemented.

- A survey campaign must already have been created and its children objects (survey cycle and survey deployment) defined.

- At least one deployment must be active.

- For survey reports, respondents must have participated in the survey. In this way, data is available in the IES schema of the Applications database to summarize for reporting purposes.

- It is not necessary to run this concurrent program to obtain current data for footprinting reports.

See Also

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program

- Update Deployment Status Concurrent Program

- Appendix C, "Oracle Discoverer Workbooks"

8.4.1.3 Update Deployment Status Concurrent Program

When you define a survey deployment, one set of criteria includes the range of time for which the deployment is valid. This set includes the deployment start date, and the response end date and time.

A concurrent program defined in Oracle Applications for survey operations must be executed on a regular basis to check this response end date and time. This process occurs from the Concurrent Manager and is performed by an individual with the iSurvey User responsibility (typically an administrator or supervisor).

The Update Deployment Status concurrent program compares the response end date for each deployment against SYSDATE, and changes the status of all deployments from active to closed if SYSDATE is past the response end date and

time. Additionally, this concurrent program changes the higher level survey campaign status to idle if the survey campaign contains no active or pending deployments.

Oracle Corporation recommends executing this concurrent program once daily at an off-peak time (a time when load on the system is not high). This concurrent program can be manually executed. For example, executing this concurrent program could be one of the closing operations an interaction center supervisor performs at the end of the day. Alternatively, this program can be scheduled to run automatically, for example, daily at 2 AM.

Parameters

There are no configurable parameters for the Update Deployment Status concurrent program.

Guidance

There are no prerequisites for executing this concurrent program. However, in order for this program to have any useful effect:

- The Survey component of Oracle Scripting must be fully implemented.

- A survey campaign must already exist and its children objects (survey cycle and survey deployment) already defined.

- At least one deployment must be active.

- The deployment end date and time, as defined in the deployment detail and compared to SYSDATE, must be in the past.

See Also

- SUBMIT GROUP FM REQUEST FROM IES Concurrent Program

- Summarize Survey Data Concurrent Program

8.4.2 Scheduling or Executing Concurrent Programs

Use the following procedure to schedule or execute concurrent programs supporting survey operations for Oracle Scripting implementations.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

iSurvey User

Steps

1. From the Navigator, select **Survey Administratin > Requests > Run** and click Open.

    The Submit a New Request window appears.

2. From the Submit a New Request window, select **Single Request** and click **OK**.

    The Submit a New Request window closes, and the Submit Request window appears.

3. In the Run this Request... area, in the Name field, click the ellipses.

    The Reports window appears, listing the concurrent programs accessible to this responsibility.

4. To execute the Fulfillment concurrent program, from the Reports window, select SUBMIT GROUP FM REQUEST FROM IES and click **OK**.

    To execute the Summarize Survey Data concurrent program, from the Reports window, select **Summarize Survey Data** and click **OK**.

    To execute the Update Deployment Status concurrent program, from the Reports window, select **Update Deployment Status** and click **OK**.

    ■   The Reports window closes.

    ■   The Name field is populated with the selected report type in the Submit Request window.

    ■   The Parameters window appears and has focus.

5. If scheduling or executing the Update Deployment Status concurrent program, no parameters are required. Proceed to 7.

6. For other concurrent programs, enter the required parameters as prompted.

    > **Note:**   You must enter parameter information using the appropriate data type. You cannot use the % wildcard character in this field.

See Section 8.4.1, "Seeded Concurrent Programs for Survey Execution" for parameter information.

Optionally, to complete this field, you can search existing information in the database for each parameter. To do this:

a. Click the ellipses next to the parameter field.

The Longlist window appears, indicating that the displayed list is lengthy.

b. In the Longlist window, in the Partial Value field, enter search criteria.

Note: You can use the % wildcard character in this field.

---

c. If you entered valid criteria, the Longlist window closes and the parameter field in the Parameters window is populated with your selection. The corresponding description field of your selection appears next to the field. To re-query, return to a.

d. If you entered partial search criteria, refine your search in the resulting window and click **OK**.

The Parameters window refreshes, populated with your selection.

e. After entering all required parameters, click **OK**.

The Parameters window closes. The Submit Request window refreshes, populated with your parameter selections.

7. If a default choice to execute the concurrent program is selected in the At these Times... area and meets your scheduling or execution requirements, proceed to 14.

8. If you want to schedule the concurrent program to execute, in the At these Times... area, click **Schedule**.

The Schedule window appears.

9. If there is a saved schedule you want to use:

a. Click **Apply a Saved Schedule.**

The Pre-defined Schedules window appears.

b. From the Pre-defined Schedules window, select the appropriate schedule.

c. Click **OK**.

The Schedule window refreshes, with the Advanced option selected in the Run the Job... area.

d. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

e. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

f. To save and apply schedule details, from the Schedule window, click **OK**.

10. If you want to execute the concurrent program as soon as possible, from the Run the Job... area, select **As Soon as Possible** and click **OK**.

11. If you want to execute the concurrent program one time at a specified date:

a. In the Run the Job... area, select **Once**.

The Schedule window refreshes, displaying a Run At field.

b. Click the ellipses in the Run At field.

The calendar window appears.

c. From the calendar window, specify the date and time to execute the concurrent program, and click **OK**.

d. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

e. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

f. To save and apply schedule details, from the Schedule window, click **OK**.

12. If you want to execute the concurrent program several times periodically:

a. In the Run the Job... area, select **Periodically**.

The Schedule window refreshes, displaying new fields.

b. In the Start At field, click the ellipses.

The calendar window appears.

c. From the calendar window, specify the date and time to start executing the concurrent program, and click **OK**.

d. If you want the concurrent process to execute periodically for an indefinite period, do not enter a value in the End At field.

e. To specify an end date, in the End At field, click the ellipses.

The calendar window appears.

f. From the calendar window, specify the date and time to stop executing the concurrent program, and click **OK**.

g. In the Re-run fields, specify the value and interval for the period of time between executions.

h. In the Apply the Interval... area, determine the criteria for applying the interval.

i. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

j. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

k. To save and apply schedule details, from the Schedule window, click **OK**.

13. If you want to execute the concurrent program on specific days:

a. In the Run the Job... area, select **On Specific Days**.

The Schedule window refreshes, displaying new fields.

b. In the Start At field, click the ellipses.

The calendar window appears.

c. From the calendar window, specify the date and time to start executing the concurrent program, and click **OK**.

d. To specify an end date, in the End At field, click the ellipses.

The calendar window appears.

e. From the calendar window, specify the date and time to stop executing the concurrent program, and click **OK**.

f. In the Dates of Every Month area, click once on each day of the month in the calendar on which you want to execute the program. To deselect a calendar date, click it again.

g. In the Days of Every Week area, click once on each day of the week on which you want to execute the program. To deselect a week day, click it again.

h. Optionally, to increment date parameters for each execution of the concurrent program, from the Schedule window, select **Increment date parameters each run**.

i. Optionally, to save the schedule, from the Schedule window, select **Save this schedule**.

j. To save and apply schedule details, from the Schedule window, click **OK**.

14. In the Submit Request window, in the Upon Completion... area, check the **Save all Output Files** box.

15. Optionally, if you want a party to be notified each time the concurrent program is executed, then do the following:

a. In the Upon Completion... area, click **Options**.

The Upon Completion window appears.

b. Verify that the **Save all Output Files** box is checked.

c. In the Name column, for each party whom you want to notify upon completion, click the ellipses and select the individual from the database.

d. Optionally, to print output files, select a networked printer from the Printer field, and specify the number of copies and default language.

e. To save Upon Completion information, click **OK**.

The Upon Completion window closes. The Submit Request window refreshes, displaying the selections you made.

16. To save and submit the concurrent request, in the Submit Request window, click **Submit**.

The Decision window appears, providing the request ID of the concurrent request. Note this request number for subsequent tracking.

17. To submit another request, in the Decision window, click **Yes**, and then enter criteria for the next request in the Submit Request window.

Otherwise, click No. The Decision window and the Submit Request window close. The Navigator appears.

18. Submit another request for concurrent processing by repeating this procedure, or log out of Oracle Applications by selecting **File > Exit Oracle Applications**.

References

For specific details on executing or scheduling concurrent programs, please refer to Chapter 5 of *Oracle System Administrator's Guide*.

See Also

- Seeded Concurrent Programs for Survey Execution

9

Campaign and List Administration Tasks

Oracle Scripting interacts with various components in the Oracle Applications suite. Although Scripting can be executed alone, it is typically used in combination with various other applications. For example, Scripting is typically launched from an Oracle business application. If using Oracle Scripting from Oracle TeleSales or Oracle Collections, you must first administer campaigns in Oracle Marketing and Oracle Sales Online.

Additionally, in order to execute survey campaigns using the survey component of Oracle Scripting, Oracle Marketing lists are required.

The following Oracle Marketing and Oracle Sales Online administration tasks are therefore included in this document to provide a single location for campaign administration and list administration information required for Oracle Scripting.

Functionality of Oracle applications is constantly improving and, thus, changing. Oracle Marketing and Oracle Sales Online functionality, user interfaces, requirements, and dependencies may change at any time. Therefore, campaign and list administration procedures or information contained in the latest relevant Oracle Marketing and Oracle Sales Online product documentation, if different, should take precedence over the steps provided below.

This section includes the following topics:

- Section 9.1, "Administering Campaigns in Oracle Marketing"

- Section 9.2, "Administering Campaigns in Oracle Sales Online"

- Section 9.3, "Administering Lists in Oracle Marketing"

9.1 Administering Campaigns in Oracle Marketing

In order to launch a specified script from Oracle TeleSales or Oracle Collections, campaign administration steps are required to be performed in Oracle Marketing.

Tasks

You can perform the following tasks:

- Section 9.1.1, "Creating a Campaign"

- Section 9.1.2, "Creating a Campaign Schedule"

- Section 9.1.3, "Associating a Script to a Campaign Schedule"

- Section 9.1.4, "Associating a Script to a Campaign"

See Also

- Section 9.2, "Administering Campaigns in Oracle Sales Online"

- Section 9.3, "Administering Lists in Oracle Marketing"

9.1.1 Creating a Campaign

Use this procedure to create a campaign in Oracle Marketing.

Prerequisites

None

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, displaying a summary view table of any existing campaigns.

2. Click **Create**.

   The Create Campaign page appears.

3. In the Setup Type field, perform the following:

    a. Type your search criteria for a setup type and click the search icon.

> **Note:** You can use the % wildcard for database searches.

    For example, to create a campaign, type **Campaigns** and click the search icon.

    Setup type is determined by the business application with which the campaign is used. Select a setup type of **Campaigns** if using Oracle TeleSales, or **Collections** if using Oracle Collections.

    The Campaign Selector page appears, listing results matching your search criteria.

    b. Optionally, refine your search. You can use the % wildcard character anywhere on this page.

    c. In the Setup Name list, click the appropriate hyperlink.

    For example, click **Campaigns**.

    The Campaign Selector page closes, and the Create Campaign page refreshes, listing the item you selected.

4. In the Name field, type a name for this new campaign.

For example, type **TestCampaign**.

5. Optionally, in the Program field, perform the following:

    a. Type your search criteria for a setup type and click the search icon.

    The Program Selector page appears, listing results matching your search criteria.

    b. Optionally, refine your search. You can use the % wildcard character anywhere on this page.

    c. In the table of programs, click the appropriate hyperlink.

    The Program Selector page closes, and the Create Campaign page refreshes, listing the item you selected.

6. If creating a campaign for use with Oracle Collections, in the Source Code field, type a unique source code.

If using this campaign for other applications, this field is optional.

7. Optionally, from the Business Unit list, select a business unit.

8. In the version field, type a version number.

   When creating a campaign with the same name as an existing campaign, only one version is active at a time. This field allows you to designate the new version as active.

9. Optionally, from the Purpose list, select a purpose for the campaign.

10. Optionally, in the Start Period field, perform the following:

   a. Type your search criteria and click the search icon.

      The Period Selector page appears, listing results matching your search criteria.

   b. Optionally, refine your search. You can use the % wildcard character anywhere on this page.

   c. In the period list, click the appropriate hyperlink.

      The Period Selector page closes, and the Create Campaign page refreshes, listing the item you selected.

11. Optionally, to enter an end period, in the End Period field, repeat step 10.

12. In the Start Date field, perform the following:

   a. Click the Date Picker widget.

      The Select a Date window appears.

   b. Using the controls, navigate to the appropriate month and year if required. Then, click on the appropriate date.

      The Select a Date window closes, and the Create Campaign page refreshes, listing the item you selected.

13. To enter an end date, in the End Date field, repeat step 12.

14. From the Currency list, ensure the appropriate currency is displayed. If not, select it from the list.

15. Optionally, from the Priority list, select a priority for the campaign.

16. In the Owner field, ensure the appropriate owner name is displayed. If not, type the appropriate search criteria and select the appropriate owner from the list.

   The current username populates this field by default.

17. Optionally, if a global campaign, select Global.

18. In the Country field, ensure the appropriate country is displayed. If not, type the appropriate search criteria and select the appropriate country from the list.

19. Optionally, if a confidential campaign, select Confidential.

A confidential Campaign cannot be viewed by people outside of the team.

20. From the Language list, ensure the appropriate language is displayed. If not, type the appropriate search criteria and select the appropriate language from the list.

21. Optionally, if you intend to use this campaign as a template on which to build other campaigns, select Template.

22. Optionally, in the description text area, type a description of this campaign.

23. Review your campaign parameters and modify as required. When satisfied, click **Create**.

The Create Campaign page refreshes, becoming the Campaign Details page. A confirmation that a new item has been created appears on the top of the page.

24. If required, make any changes at this time and click **Update** to save your work.

See Also

- Section 9.1.2, "Creating a Campaign Schedule"

- Section 9.1.3, "Associating a Script to a Campaign Schedule"

- Section 9.1.4, "Associating a Script to a Campaign"

9.1.2 Creating a Campaign Schedule

Use this procedure to create a campaign schedule in Oracle Marketing.

Prerequisites

A campaign must already exist to which you want to associate a schedule.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

    The Campaigns page appears, displaying a summary view table of any existing campaigns.

2. Click the campaign for which you want to create a campaign schedule.

    For example, click **TestCampaign**.

    The Campaign Details page appears.

3. From the side panel, click the **Execution** hyperlink.

    The Campaign Details – Campaign Schedule page appears.

4. Click **Create**.

    The Create Campaign Schedule page appears.

5. From the Setup Type field, select the appropriate setup type.

    -   For example, for campaign schedules to be used with Oracle Advanced Outbound, select **Direct Marketing – Telemarketing**.

    -   If using a custom setup type to avoid the budget approval step, select the custom setup type.

    The Activity Type and Activity fields will auto-populate.

6. In the Name field, type a name for this new campaign schedule.

    For example, type **TestCampaignSchedule**.

7. From the Language list, ensure the appropriate language is displayed. If not, select the appropriate language from the list.

8. In the Coordinator field, ensure the appropriate campaign schedule coordinator name is displayed. If not, enter the appropriate search criteria and select the appropriate coordinator from the list.

    The current username populates this field by default.

9. Optionally, in the Source Code field, type a unique source code if not already populated. To use the source code of the parent campaign, select **Use Parent Source Code**.

10. In the Activity and Activity Type lists, ensure the values specified by the campaign schedule setup type are displayed.

11. Optionally, in the Marketing Medium field, enter the appropriate search criteria and select the appropriate marketing medium from the list.

12. From the Currency list, ensure the appropriate currency is displayed. If not, select it from the list.

13. From the Time Zone list, ensure the appropriate time zone is displayed. If not, select it from the list.

This time zone setting should match the time zone for the geographic location where the campaign schedule activity takes place.

14. In the Start Date field, perform the following:

a. Click the Date Picker widget.

The Select a Date window appears.

b. Using the controls, navigate to the appropriate month and year if required. Then, click on the appropriate date.

The Select a Date window closes, and the Create Campaign Schedule page refreshes, listing the item you selected.

15. Optionally, to enter an end date, in the End Date field, repeat step 14.

16. Optionally, in the Start Time fields, select the appropriate hour, minute, and AM/PM designation from the list.

17. Optionally, in the End Time fields, select the appropriate hour, minute, and AM/PM designation from the list.

18. Optionally, from the Objective list, select the appropriate objective for this campaign schedule.

Objectives may not appear in the list of values, based on previous selections.

19. Optionally, from the Priority list, select the appropriate priority for this campaign schedule.

20. Optionally, in the description text area, type a description of this campaign schedule.

21. Ensure that the **Do Not Use Trigger** option is selected. When this option is selected, no values appear in the Trigger Condition list.

22. Optionally, to enter approval conditions, in the Execute After Approval by field, search for and select an appropriate approving authority.

23. Optionally, in the Send Notification to field, search for and select the users whom you want to be notified of the approval.

24. If using notifications, in the Subject field, enter an e-mail message subject for the approval notification.

25. If you want to use an e-mail template for the notification, in Template field, search for and select the appropriate template.

26. Review your campaign schedule parameters and modify as required. When satisfied, click **Create**.

27. The Create Campaign Schedule page refreshes, becoming the Campaign Schedule Details page.

28. If required, make any changes at this time and click **Update** to save your work.

See Also

- Section 9.1.1, "Creating a Campaign"

- Section 9.1.3, "Associating a Script to a Campaign Schedule"

- Section 9.1.4, "Associating a Script to a Campaign"

- Section 9.2.1, "Assigning Agents to a Campaign Schedule"

- Section 9.2.2, "Assigning Campaign Schedules to Agents"

9.1.3 Associating a Script to a Campaign Schedule

Use this procedure to associate a particular deployed Script Author script to a campaign schedule for use in a business application. Thereafter, this association is applied to all users of the campaign schedule.

Prerequisites

- A valid campaign and campaign schedule must be defined.

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- You must know whether the script is an inbound or outbound script.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, listing a summary view table containing all existing campaigns created by the current user.

2. From the Name field in the table, select the appropriate campaign.

   > **Note:** This campaign must contain the campaign schedule under which your target group resides.

   The Campaign Details page appears.

3. From the side panel, click **Execution**.

   The Campaign Details – Campaign Schedule page appears, displaying a summary view table of all campaign schedules associated with the selected campaign.

4. From the Name field in the table, click the campaign schedule to which you want to associate a script.

   The Campaign Schedule Details page appears.

5. From the side panel, click **Collaboration**.

   The Campaign Schedule Details – Contact Points page appears, displaying the contact points table, which lists any existing Oracle Scripting script associated to the selected campaign schedule.

6. From the Type list, select **Inbound Script** or **Outbound Script**, as appropriate.

7. In the Name field, search for and select the name of the specific script you want to associate.

   The Campaign Schedule Details – Contact Point page refreshes, populated with the selected values.

   c. To save the selected values, click **Update**.

The Campaign Schedule Details – Contact Point page refreshes, indicating that the selected script is now associated with the campaign schedule.

8. If you wish to associate any other script with this campaign schedule, repeat steps 6 and 7 above.

See Also

- Section 9.1.1, "Creating a Campaign"

- Section 9.1.2, "Creating a Campaign Schedule"

- Section 9.1.4, "Associating a Script to a Campaign"

9.1.4 Associating a Script to a Campaign

Use this procedure to associate a particular deployed Script Author script to a campaign for use in a business application. Thereafter, this association is applied to all users of the campaign.

Prerequisites

- A valid campaign must be defined.

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- You must know whether the script is an inbound or outbound script.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Campaign tab.

   The Campaigns page appears, listing a summary view table containing all existing campaigns created by the current user.

2. From the Name field in the table, select the appropriate campaign.

The Campaign Details page appears.

3. From the side panel, click **Collaboration**.

   The Campaign Details – Contact Point page appears, displaying a table listing any existing Oracle Scripting scripts associated to the selected campaign.

4. From the Type list, select **Inbound Script** or **Outbound Script**, as appropriate.

5. In the Name field, search for and select the name of the specific script you want to associate.

   The Campaign Schedule Details – Contact Point page refreshes, populated with the selected values.

6. To save the selected values, click **Update**.

   The Campaign Details – Contact Point page refreshes, indicating that the selected script is now associated with the designated campaign.

7. If you wish to associate any other script with this campaign, repeat steps 5 and 6 above.

See Also

- Section 9.1.1, "Creating a Campaign"

- Section 9.1.2, "Creating a Campaign Schedule"

- Section 9.1.3, "Associating a Script to a Campaign Schedule"

9.2 Administering Campaigns in Oracle Sales Online

In order to have agents associated with campaign schedules or vice versa, campaign administration steps are required to be performed in Oracle Sales Online.

Tasks

You can perform the following tasks:

- Section 9.2.1, "Assigning Agents to a Campaign Schedule"

- Section 9.2.2, "Assigning Campaign Schedules to Agents"

See Also

- Section 9.1, "Administering Campaigns in Oracle Marketing"

- Section 9.3, "Administering Lists in Oracle Marketing"

9.2.1 Assigning Agents to a Campaign Schedule

Use this procedure to assign one or more agents to a campaign schedule in Oracle Sales Online.

Prerequisites

None

Login

Log into Oracle HTML-based applications.

Responsibility

Sales Online Super User

Steps

1. From the Oracle Sales Online administration console, click the Administration tab.

    > **Note:** In order to access this tab, your Oracle Applications account must be a member of a valid sales group. If the required group does not yet exist, it should be created prior to importing the administrator as a CRM resource. Groups and group membership are administered in CRM Resource Manager.

2. Click the Sales subtab.

3. From the side panel, under Call Center, click **Agent Assignments.**

    The Resource Assignment page appears.

4. In the Schedule Name area, click **Go**.

    The Valid Campaign Schedules page appears.

5. If the campaign schedule you want to assign does not appear, then in the Campaign Schedule Name field, search for and select the appropriate campaign schedule.

    For example, type **TestCampaignSchedule** and click **Search**.

    The Valid Campaign Schedules page refreshes, showing a list of valid campaign schedules that meet your search criteria.

6. In the Valid Campaign Schedules page, in the Select column, select one or more appropriate valid campaign schedules and click **Ok**.

   The Resource Assignment page refreshes, listing the valid campaign schedules you selected.

7. Under the Assign To heading, select **Resource**.

8. From the Current Resource Assignments area, click **Add Resource**.

   The Resource Assignment page refreshes. The Current Resource Assignments table includes one or more rows, with text fields under the Resources column.

9. In the Resource field, perform the following:

   a. Type search criteria for an agent (by agent's last name) and click **Go**.

      The Resources Selector page appears, displaying results of your search.

   b. Optionally, refine your search.

      You can use the % wildcard character anywhere on this page.

   c. In the resources list, click the appropriate hyperlink.

      The Resources Selector page closes, and the Current Resource Assignments list on the Resource Assignment page refreshes, listing the specified agent name you selected.

10. Repeat the previous step as required for all agents. When satisfied, click **Update**.

11. If necessary, repeat steps 5 through 10 for each campaign schedule you want to assign.

See Also
- Section 9.2.2, "Assigning Campaign Schedules to Agents"

9.2.2 Assigning Campaign Schedules to Agents

Use this procedure to assign one or more campaign schedules to agents in Oracle Sales Online.

Prerequisites
None

Login

Log into Oracle HTML-based applications.

Responsibility

Sales Online Super User

Steps

1. From the Oracle Sales Online administration console, click the Administration tab.

   > **Note:** In order to access this tab, your Oracle Applications account must be a member of a valid sales group. If the required group does not yet exist, it should be created prior to importing the administrator as a CRM resource. Groups and group membership are administered in CRM Resource Manager.

2. Click the Sales subtab.

3. From the side navigation bar, under Call Center, click **Campaign Assignments.**

   The Campaign Assignment page appears.

4. In the Assign To area, select **Resource** and click **Go**.

   The Resources page appears.

5. If the resource you want to assign is not in the Resources list, then in the Resource Name field, perform the following:

   a. Type search criteria for an agent (by agent's last name) and click **Search**.

      The Resources page refreshes, displaying results of your search.

   b. Optionally, refine your search.

      You can use the % wildcard character anywhere on this page.

   c. In the Select column, select one or more appropriate valid resources and click **Ok**.

      The Campaign Assignment page refreshes, listing the valid resources you selected.

6. Click **Apply**.

The Campaign Assignment page refreshes.

7. Under the Current Campaign Assignments heading, click **Add Campaign**.

The Campaign Assignment page refreshes. The Current Campaign Assignments table includes one or more rows, with text fields under the Name column.

8. In the **Name** field, perform the following:

   a. Type search criteria for a campaign schedule and click **Go**.

   The Valid Campaign Schedules selector page appears, displaying results of your search.

   b. Optionally, refine your search.

   You can use the % wildcard character anywhere on this page.

   c. In the campaign schedules list, click the appropriate hyperlink.

   The Valid Campaign Schedules selector page closes, and the Current Campaign Assignments list on the Campaign Assignment page refreshes, listing the schedules you selected.

9. Repeat the previous step as required for all campaign schedules. When satisfied, click **Update**.

Successful assignment of a campaign schedule to an agent is indicated by a confirmation message at the top of the page.

See Also

■ Section 9.2.1, "Assigning Agents to a Campaign Schedule"

9.3 Administering Lists in Oracle Marketing

Using the Survey component of Oracle Scripting, lists created with the list management features of Oracle Marketing are used as the contact sources for executing the survey campaign using e-mail as the media channel. The list must already exist at the time you define a deployment for a survey campaign. The list can be created from the Audience tab of the Survey Administration Console, or the Audience tab of the Marketing Administration Console. These lists are passed to Oracle One-to-One Fulfillment. Oracle One-to-One Fulfillment uses a master document (an electronic mail message in HTML format), and an associated query, to merge variables into the message at time of execution. The combination of e-mail master document and the query are referred to in Oracle One-to-One Fulfillment as

a template. Upon execution of a concurrent request, the Fulfillment engine executes the query. For each list member, the data retrieved from the query is merged into a copy of the master document, personalizing the HTML invitation or reminder with unique data.

At minimum, an invitation or reminder master document template includes as merge field variables each list member's e-mail address, and the survey URL. For list-based surveys, the URL contains a parameter to uniquely identify each respondent. This respondent identification code (rID) ties each response to its originating list member.

Typically, some other data from the Oracle Marketing list (for example, title of address, first name, last name, organization, organization address information, country, and so forth) are used in the master document as a merge field variable to personalize the e-mail message. From a survey perspective, the only merge field variables required are respondent e-mail address and survey URL. The minimum set of merge fields that are required in the Oracle Marketing list may be larger, and require certain fields, based on the method of populating the list with records and the specified list type.

The merged HTML documents are then sent to the electronic mail server designated by Oracle One-to-One Fulfillment. When the message is sent out by the mail server, all list members with valid e-mail addresses receive the message as an invitation (or reminder) to participate in a Web-based survey. Clicking on a hyperlink or otherwise accessing the unique survey URL contained in the invitation results in a Scripting Engine interaction in the Web interface as the respondent participates in the survey.

This model is only possible with one or more valid lists, which must be created and defined using Oracle Marketing functionality.

This section includes the following topics:

- Section 9.3.1, "List Creation Overview"

- Section 9.3.2, "Creating a List"

- Section 9.3.3, "Identifying Record Sources for a List"

- Section 9.3.4, "Adding Records to a List"

- Section 9.3.5, "Generating a List"

- Section 9.3.6, "Campaigns as an Oracle Applications Dependency"

See Also

- Section 9.1, "Administering Campaigns in Oracle Marketing"

9.3.1 List Creation Overview

There are various uses for lists, and correspondingly various ways to identify records to load into an Oracle Marketing list. The list creation itself is a simple process resulting in a list object, at first an empty container awaiting records. Thereafter, you must identify a record source, add the records to the list, and generate the list.

This document assumes the Marketing Administration Console is used for these steps. However, the list management features of Oracle Marketing are also accessible from the Survey Administration Console under the Audience tab, which corresponds to the Audience tab in the Marketing Administration Console. Each appropriate subtab or side panel link required for Oracle Marketing administration can be accessed from either administrative console. If administering lists from the Survey Administration Console, the survey administrator must also be assigned the appropriate marketing responsibility.

See Also

- Section 9.3.2, "Creating a List"

- Section 9.3.3, "Identifying Record Sources for a List"

- Section 9.3.4, "Adding Records to a List"

- Section 9.3.5, "Generating a List"

- Section 9.3.6, "Campaigns as an Oracle Applications Dependency"

9.3.2 Creating a List

Use this procedure to define a list in Oracle Marketing. Until record sources are subsequently defined for the list, it will remain in Draft status.

Prerequisites

You must know the list type and source type before creating a list. For more information, see Oracle Marketing product documentation.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.

2. Click the List subtab.

   The Lists page appears.

3. Click **Create**.

   The Create List page appears.

4. In the Name field, type a name for your list.

5. From the list in the Type field, select **Standard** or **Manual**.

6. From the list in the Source Type field, select the source type you want to use with your list.

   Two typical source types for a list are **Persons** and **Organization Contacts**. These map to fields in the TCA architecture. Other source types can be added in Oracle Marketing if necessary (for example, from HR, the Employees source type). For more information, see Guidelines below.

7. If the appropriate person is not populated in the Owner field, do the following:

   a. Enter your search criteria in the Owner field and click the search icon.

      You can use the % wildcard search character if it is the second or subsequent character in your search string. For exact title matches, enter search criteria using the syntax <Lastname>, <Title> <Firstname>. For example, to obtain the record Mr. George Washington, type **Washington, Mr. George** and click the search icon.

   b. In the Resource Selector page, refine your search criteria if necessary.

   c. From the resulting list on the Resource Selector page, click the name of the appropriate owner.

      The Resource Selector page closes. The Create List page refreshes, containing the selected value.

8. Optionally, in the Description field, enter a description or purpose for this list.

9. Click **Create**.

   The Create List page refreshes, becoming the List - Main page.

If successful, at the bottom of the page, confirmation messages appear, indicating that the record has been created and updated.

Guidelines

- List types are not enforced by all applications. Manual lists can have each list entry manually designated from existing records. For all other lists, including lists that will be populated by Oracle Discoverer workbook, use Standard. Suppression lists contain list entries that other lists can use to filter out or suppress, effectively creating a stop-list mechanism that can be used for various other lists. Suppression lists are generally not used as lists of contacts to execute a campaign.

- A list source type is a concept used by Oracle Marketing to define how fields from the database map to the columns of a list entry, and which fields will be populated. A list entry, stored in the table AMS_LIST_ENTRIES, has many data fields. Two typical source types are Persons and Organization Contacts. These list source types define whether a set of list entries are contacts in an organization (in which case certain fields related to an organization are required, such as Organization, Organization Contact, and Country) or are persons (no organization information is necessary, but FIRSTNAME and LASTNAME are required). For the Organization Contacts source type, the PARTY_ID field contains the name of the organization, and the P_CONTACT_PARTY_ID field contains the name of the individual. However, for the Persons source type, the PARTY_ID field contains the name of the individual, and the P_CONTACT_PARTY_ID field is null.

See Also

- Section 9.3.1, "List Creation Overview"

- Section 9.3.3, "Identifying Record Sources for a List"

- Section 9.3.4, "Adding Records to a List"

- Section 9.3.5, "Generating a List"

- Section 9.3.6, "Campaigns as an Oracle Applications Dependency"

9.3.3 Identifying Record Sources for a List

You must identify record sources to populate lists with list entries. Oracle Marketing lists can be populated in one of three ways:

- Importing a list from a flat file of comma-separated values.

- Manually building a list from existing records in other Oracle Marketing lists or in the TCA architecture.

- Using Oracle Discoverer Workbook from Oracle Marketing to identify list details and conditions to serve as a query to generate a list from the TCA architecture.

Tasks

You can perform the following tasks:

- Section 9.3.3.1, "Importing Record Sources"

- Section 9.3.3.2, "Manually Selecting List Entries as Record Sources"

- Section 9.3.3.3, "Using Oracle Discoverer Workbook to Define Record Sources"

9.3.3.1 Importing Record Sources

Use this procedure to import record sources into an Oracle Marketing list. Data types that can be imported include:

1. Event registrations

2. Leads

3. Organizations, Contacts, Addresses

4. Persons, Addresses

Prerequisites

- An Oracle Marketing list must already exist.

- You must have a flat file available on your computer's local file system or network, or available using the file transfer protocol.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.

2. Click the Import subtab.

The Imports page appears.

3. Click **Create**.

   The Import: Introduction page appears.

4. From the Import Wizard list, select the type of record source you want to import, and click **Next**.

   See Guidelines below for information on record source types.

   The Import Step 1a: Definition page of the import wizard appears.

5. In the **Name** field of the import wizard, type a name for the record source you want to import.

6. Optionally, in the description field, type a description of this record source.

7. In the Source File area, designate the location of the record source you want to import, and provide details on how the record source is organized.

   a. To locate a file on your computer's file system, click **Client**. To locate a file elsewhere on the network, click **Server**. To use the file transfer protocol to obtain the list, click **FTP**.

   b. Click **Go**.

      The Import List selector window appears.

   c. From the Import List selector window, click **Browse...**.

      Files to import must be in one of the three formats specified in the help text on the page.

   d. Navigate to the appropriate comma-separated or ZIP file, select it, and click **Open**.

   e. From the Import List selector window, click **Submit**.

      The Import List selector window closes. The Definition page refreshes, with your selection populating the appropriate text field.

   f. Optionally, from the Character Set list, select a character set if different from the default selection.

   g. From the Column Delimiter list, ensure the appropriate delimiter type is selected.

      For example, if using comma-separated values, ensure **Comma** is selected.

    h. From the Field Enclosed By list, ensure the appropriate field symbol used in the import file is selected.

    For example, if using comma-separated values, ensure *a value other than Comma* is selected, and that this value matches the symbol used to enclose each field within each record in the source list (such as double quotes).

    i. If the list record source contains file headers such as column definitions, ensure the File Header Exists option is selected. Otherwise, click to clear this option.

    If you select this option, the first row of the import file will display as field names in the Source Fields area when mapping fields. Otherwise, the fields in the Source Fields area appear as Field1, Field2, Field3, etc.

    j. Optionally, if merging lists or otherwise removing duplication, perform the following:

      – In the Remove Duplications Using Rule field, enter search criteria for rules to remove duplicate records in the list, and click the search icon.

      The DeDuplication Rule LOV window appears.

      – In the DeDuplication Rule LOV window, refine your search criteria if necessary.

      – From the resulting list, click the name of the appropriate rule.

      The Definition page refreshes, containing the selected value.

    k. Optionally, select the **Use Word Standardization** option.

8. Click **Next**.

For record sources of the lead or event registration import types, the Import Mapping page of the import wizard appears. Proceed to step 11.

For record sources of the Organizations, Contacts, Addresses import type or the Persons, Addresses import type, the Import Step 1b: Additional Details page of the import wizard appears.

9. Optionally, in the provided fields, enter additional information about the record source you want to import.

If the description import type is a rented list, you must supply either an expiration date, a number of uses, or both.

10. Click **Next**.

The Import Mapping page of the import wizard appears. From here you can import existing mappings, designate new mappings, and view records in the record source you want to import in the Preview field.

11. To map fields using an existing mapping, in the Load Existing Mapping field, search for and select the appropriate existing mapping. Verify that the required fields are included in the mapping, and make changes if required.

Optionally, you can save or rename a mapping.

12. To create a new mapping, establish a match between the source fields from your record source and the target fields derived from the Oracle TCA architecture. Match each set of fields in the order used in your record source. This order (as well as the first few records in your record source) are displayed in the preview field below.

For example, if the first entry in the Preview Count window is **George**, click Field1 or PERSON_FIRST_NAME in the Source Fields list and Person First Name Target Fields list.

13. Click the right arrow to confirm the first field mapping.

14. Repeat steps 12 and 13 above for each source field until complete.

- Each field designated in the final target mapping must have a corresponding record in the Target Fields list.

- The converse is not true. You may exclude from the final mapping some fields in the source fields list, as long as they are not required fields as designated by an asterisk in the Target Fields list. Data from excluded fields will not become part of the list. If you exclude some source fields in your final target mapping, your list may not pass validation used by applications such as Oracle Advanced Outbound.

15. Click **Next**.

If successful, a confirmation appears on the Import Step 3: Review page of the import wizard. If required fields are missing, click **Cancel** and add them to the flat file for subsequent importation. If fields are mismapped, click **Restore** to clear the mapping page and remap until successful.

16. For two import record source types (both Organizations, Contacts, Addresses and Persons, Addresses), you can automatically generate the list during the import process. To do this, click the **Generate a List upon Import** box.

> **Note:** If you select this option, you do not need to perform the task described in Section 9.3.5, "Generating a List".

17. Click **Import**.

    The Confirmation page appears, displaying a confirmation note stating that the list has been submitted for processing.

18. If you want to import another record source, click **New Import**. Otherwise, click **Finish**.

Guidelines

- When you select an import type, you determine which fields, at minimum, are required to be included in the list. Note that fields in addition to those marked by an asterisk may be required, due to varying business rules related to the purpose for the list.

    Based on import type, the following fields are required:

Import Type	Intended Use	Required Fields
Event Registrations	Event Registrations	■ Event Source Code
Leads	Leads	■ Address1 ■ Country ■ Customer Name ■ First Name ■ Last Name ■ Promotion Code ■ Source System
Organizations, Contacts, Addresses	Organizations, Contacts, Addresses	■ Organization Name ■ Party ID
Persons, Addresses	Lists of persons only	■ Person First Name ■ Person Last Name
Persons, Addresses	Lists of persons and their addresses	■ Person First Name ■ Person Last Name ■ Country

Import Type	Intended Use	Required Fields
Any import type to be used with the survey component of Oracle Scripting	Targeted survey campaigns	■ Survey URL

- If creating lists for use with the survey component of Oracle Scripting, you must include the survey URL as a field.

- When specifying additional details, if the description import type is a rented list, you must supply either an expiration date, a number of uses, or both.

- If the source set in the Additional Details page is Purchased List, the records from your import file will be added to the Customer Master list. This may also be true for the Others source type, based on your configuration

9.3.3.2 Manually Selecting List Entries as Record Sources

Using the Add List Entries - Contact Search page, you can search for existing records in the TCA architecture or from an existing Oracle Marketing list, and add them as a list entry on a specified list. You can search by person information, address, organization information, or you can find records for a specified organization based on the individual's relationship to the organization.

Use this procedure to manually select list entries from existing records in the TCA architecture or from an existing Oracle Marketing list.

Prerequisites

- An Oracle Marketing list must already exist.

- The list entries you want to create must already exist.

- You must have information from which to query records to add list entries to the Oracle Marketing list.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.

2. Click the List subtab.

   The Lists page appears.

3. From the Name column of the lists list, click the appropriate hyperlink to select the existing Oracle Marketing list to which you want to add entries.

   For example, click **TestList.**

   The Lists - Main page appears.

4. From the side panel, click **Entries**.

   The List Entries page appears.

5. Click **Add to List**.

   The Add List Entries - Contact Search page appears.

6. Enter your search criteria and click **Search**.

   The Add List Entries - Search Results page appears, with a list of all records that match your search criteria.

7. Refine your search criteria if necessary.

8. In the Select column of the Add List Entries - Search Results page, mark each appropriate record that you want to include as a list entry in the selected list.

   > **Note:** All list entries must have valid e-mail addresses in order to receive invitations or reminders to participate in a survey.

   > **Note:** For testing purposes, it is recommended to add yourself as a list entry. Ensure a valid e-mail address is associated with your list entry record.

9. Click **Add to List**.

   The Confirmation List Entries page appears. The designated contacts are added to the specified list.

10. Make any changes if required, and click **Update.**

Once the list is generated, it can be used in a survey campaign deployment.

11. Click **Return to List Entries** to verify that the selected entries were added to the appropriate list.

12. Make any changes if required, and click **Update.**

For example, add an e-mail address to any missing list entries, and click **Update**.

9.3.3.3 Using Oracle Discoverer Workbook to Define Record Sources

Oracle Discoverer is interface software available from the Marketing Administration Console that allows users to view and select records from complex database structures without requiring knowledge of those structures or knowledge of Structured Query Language (SQL). Discoverer generates workbooks, which are lists of records similar in design to a spreadsheet. Workbooks may then be used to generate Oracle Marketing lists or to provide a SQL statement for a segment.

Use this procedure to create an Oracle Discoverer workbook from Oracle Marketing. Using the folders provided in the Oracle Discoverer layer of Oracle Marketing to define record sources for a list provides access to many data fields in the TCA. The workbook parameters are executed when the list is generated.

Prerequisites

- Oracle Discoverer must be appropriately implemented and configured for use in Oracle Marketing.

- You must have the appropriate responsibility and requisite group roles and membership to access the Discoverer component. For more information, see Oracle Marketing product documentation.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.

2. Click the Discoverer subtab.

Oracle Discoverer application launches in a separate browser window. The Open Workbook window appears.

3. Click **Create a new workbook**.

The Open Workbook window refreshes with added display information options.

4. Ensure the Table display option is selected.

5. Click **Next**.

The Select Items window appears.

6. From the Available list, select the appropriate list management option.

For example, select **Marketing Online Organization List Management**.

The Select Items window refreshes, showing folders appropriate to the option you selected. You can drill down into each folder by clicking the plus sign.

7. From the folder area of the Available list, select the appropriate folder.

For example, for a list based on organizations, select **Organization Contact List Details**, or for a person or address list, select **Marketing Online Person List Management**.

8. Click the plus sign preceding the folder to display the folder's contents.

The Select Items window refreshes, showing fields in the database associated with your selection.

9. Select **Mandatory Identifier** and click the select arrow between the Available list and the Select list to move that field to the Select list.

10. Add to the select list all other mandatory fields.

Include in the select list all fields preceded with the word "Include." Other fields may be mandatory, based on the purpose for the list.

For example, for an organization list, you must also select the **Party Site Identifier, Email Address, Person First Name, Person Last Name**, and **Person Title** fields.

11. Review your field selections. When satisfied, click **Next**.

12. Click **Next**.

The Table Layout window appears. No data is required for this window.

13. Click **Next**.

The Format window appears. No data is required for this window.

14. Click **Next**.

    The Conditions window appears.

15. Optionally, enter any conditions by doing the following:

    a. From the Conditions window, click **New**.

       The New Conditions window appears.

    b. Optionally, in the name field, type a name for the new condition.

    c. Optionally, in the description field, enter a description for the new condition.

    d. In the Formula area, start defining the condition. From the Item list, select the appropriate item for which you want to create a condition.

       Each field in the Item list is preceded by its parent list management option and a period. For example, the Mandatory Identifier is named Organization Contact Details List.Mandatory Identifier.

    e. In the Formula area, from the Condition list, select the condition operator.

       For example, select the equal sign.

    f. In the Formula area, in the Values list, type a new value, or select **Create Calculation...** to create a calculation, **Select Item...** to select an item, or **New Parameter...** to define a new condition parameter.

       For example, type an organization ID.

    g. If your condition is case-sensitive, ensure the Match case option is selected. If not, click to clear this option.

    h. Check your condition parameters. When satisfied, click **OK**.

       The New Conditions window closes.

       To add additional conditions, repeat a through h of this step.

16. Click **Next**.

    The Sort window appears.

> **Note:** Do not include any sort conditions in your workbook. Generation of a list of list type Standard remains in Draft status if (a) a Sort condition is included in the workbook, or (b) if all the required fields (Party Id, Mandatory Identifier, One of the Include fields, Title, e-mail Address, First name, Last Name) are not included in the workbook.

17. Click **Next**.

    The Calculations window appears.

18. Optionally, add a calculation. When finished, click **Next**.

    The Percentages window appears.

19. Optionally, add a percentage. When finished, click **Next**.

    The Totals window appears.

20. Optionally, add a total. When finished, click **Next**.

    The Parameters window appears.

21. Optionally, add a parameter.

22. Click **Finish**.

    The Parameters window closes. After processing, your new workbook displays.

23. From the File menu, select **Save As...**, type a name for your workbook, and click **Save**.

24. To exit Oracle Discoverer, from the File menu, select **Exit**.

See Also

- Section 9.3.1, "List Creation Overview"
- Section 9.3.2, "Creating a List"
- Section 9.3.4, "Adding Records to a List"
- Section 9.3.5, "Generating a List"
- Section 9.3.6, "Campaigns as an Oracle Applications Dependency"

9.3.4 Adding Records to a List

After creating a list and identifying record sources, records must be added to the list. Use this procedure to add records to a list in Oracle Marketing.

> **Note:** If you selected the **Generate a List upon Import** box when importing records of source type Organizations, Contacts, Addresses or Persons, Addresses, this procedure is not required.

Prerequisites

- An Oracle Marketing list must already exist.
- Record sources must already be identified.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.
2. Click the List subtab.

   The Lists page appears, displaying a summary view table of all existing lists.
3. From the table, click the name for the list into which you want to add records.

   The List – Main page appears.
4. From the side panel, click **Selections**.

   The List Selections page appears.
5. In the first row:
   a. Type an integer designating the order in which you want the first record source to be incorporated into the list. The lowest numbered record source will be used first. For more information, see Guidelines below.
   b. From the Operation list, select an operation type. Operations available are Include, Exclude and Intersect. For more information, see Guidelines below.

c. From the Type list, select a record source type. Valid record source types include Import List, List, SQL, Segment, or WorkBook. For more information, see Guidelines below.

d. In the Name field, search for and select the name of the record source you want to use to provide record entries to this list.

e. If an Include operation, in the Dedupe Rank field, type a deduplication rank for the record source.

A deduplication rank determines which record to retain when a user removes duplicate records after the list has been generated. Deduplication occurs only during an Include operation, in which cases it is a required parameter.

f. Optionally, if an Include operation, in the Percent of Records field, type an integer designating the percentage of records in the record source you want to include in the list.

For example, if the named record source contains ten thousand records and you only wish to import one thousand, type 10 in this field.

6. Repeat step 5 for each additional record source you want to include, exclude, or intersect to compose this list.

7. Review your list selection criteria. When satisfied, click **Update**.

The List – Selection page refreshes, displaying a confirmation message.

Guidelines

- When designating an order for list selection, each record source in the list must contain a unique number. The lowest numbered record source will be used first.

- The first record source must be of type "include" so that records will be included in the list. Designate all record sources to include prior to designating record sources to exclude.

- The Source Types (Organization Contacts or Persons) should be the same for all the items (lists, workbooks, segments) used in the operation (Include, Exclude, or Intersect) and the main list that you've created.

- If one of the list sources is a suppression list or another list of type "exclude," designate this last in the list selection page, to ensure that records to be included will exist in the set from which exclusions are to occur.

- When designating the operation type, use the following criteria:

Use **Include** to add all entries in the record source to the list. Include is a required operation for the first record source to be used.

Use **Exclude** to remove from the final list any entries in the specified record source. Exclude is generally used to remove a set of entries from a larger record source that is included in the list.

Use **Intersect** to designate a second or subsequent primary list criteria. The list sources to be included must be listed prior to the record source designated as an Intersect record source. For example, to intersect a record source of people making over $100,000 with a record source of people living in Montana, the resulting list will contain entries for people living in Montana who make more than $100,000. Thus, incorporation order is crucial, as other criteria must precede the Intersect designation.

- When designating record source types, use the following criteria:

  If adding records from an existing import list, select **Import List**.

  If adding records using an existing SQL statement, select **SQL**.

  If adding records using an existing marketing segment, select **Segment.**

  If adding records from an existing Discoverer Workbook, select **WorkBook**. Select the Type.

See Also

9.3.5 Generating a List

After creating a list, identifying, and adding record sources to the list, the list must be generated to populate the list with all required records. Use this procedure to generate a list prior to executing a survey using the list.

> **Note:** If you selected the **Generate a List upon Import** box when importing records of source type Organizations, Contacts, Addresses or Persons, Addresses, this procedure is not required.

Prerequisites

- An Oracle Marketing list must already exist.

- Record sources must already be identified.

- Record sources must already be added to the list.

- The AMS Marketing Approvals Process concurrent program needs to be running in order to generate Standard Lists (from a workbook, from SQL, or from import lists) and to change their status to Available. Otherwise, the list status will be trapped in Generating status. For Manual Lists, there is no requirement to run this process. For more information, refer to Oracle Marketing product documentation.

Login

Log into Oracle HTML-based applications.

Responsibility

Oracle Marketing Super User

Steps

1. From the Oracle Marketing administration console, click the Audience tab.

2. Click the List subtab.

   The Lists page appears, displaying a summary view table of all existing lists.

3. From the table, click the name for the list which you want to generate.

   The List - Main page appears.

4. From the side panel, click Generation.

   The List – Generation page appears.

5. If you want to generate the list now, click **Generate immediately**.

6. If you want to schedule a time and date to generate the list in the future:

   a. Click **Generate at a future time**.

    b. Using the date control widget, specify the target date on which to generate the list

    c. Using the hour, minute, and AM/PM lists, specify the time to generate the list on the specified date.

    d. From the time zone list, specify the appropriate time zone.

    e. From the Generation Type, confirm that the selected type is appropriate. If not, select the appropriate generation type from the list.

7. Click **Update**.

The List – Generation page refreshes to display a confirmation message.

See Also

- Section 9.3.1, "List Creation Overview"

- Section 9.3.2, "Creating a List"

- Section 9.3.3, "Identifying Record Sources for a List"

- Section 9.3.4, "Adding Records to a List"

- Section 9.3.6, "Campaigns as an Oracle Applications Dependency"

9.3.6 Campaigns as an Oracle Applications Dependency

A campaign can be generally described as a focused effort to achieve a particular goal from a targeted population over a specific period of time for a particular business purpose.

From an Oracle Applications perspective, campaigns are the vehicle for putting information about an organization's products, services, offers, and messages in front of customers and potential customers. Campaigns are created and administered in Oracle Marketing and used by various Oracle business applications. Although sometimes referred to as a marketing campaign, Oracle Marketing campaigns can also be used for sales, telesales, service and teleservice, collections... for any business purpose.

From an Oracle Scripting perspective, there is another type of campaign that meets the general definition: a survey campaign. This is the collection of business requirements required to execute a script as a survey in a Web browser using the Scripting Engine Web interface. Survey campaigns are created using the Survey Administration console. These will always be referred to specifically as survey campaigns.

Campaigns are a fundamental entity for Oracle Marketing, and campaign attributes are prerequisites or dependencies for various Oracle business applications. For example:

- In order to specify a specific script to launch when clicking the View Script button in Oracle TeleSales, a campaign must be defined, campaign schedules associated with the campaign must be defined (including a script associated with the schedule), and agents must be assigned to the schedule or vice versa.

- The same set of campaign administration steps is required:

  - To specify a specific script to launch automatically in Oracle TeleSales upon call delivery to the agent

  - To specify a specific script to launch automatically upon start of interaction in Oracle TeleSales or Oracle Collections

  - To launch a specific script from Oracle Collections by designating a code for promotion, campaign, campaign schedule, or event (assuming a code for the specific object is also defined in Oracle Marketing).

- Campaigns are also required in order to associate Oracle Marketing lists. These lists are sets of customer contact information derived from records in the Oracle Applications Trading Community Architecture (TCA).

- Oracle Advanced Outbound requires use of the campaign schedules created using Oracle Marketing. Each of these campaign schedules can be targeted for different execution channels. For example, a marketing campaign can have a campaign schedule associated with it that is targeted for e-mail execution, and another campaign schedule associated with the campaign that is targeted for telephony execution.

See Also

- Section 9.3.1, "List Creation Overview"

- Section 9.3.2, "Creating a List"

- Section 9.3.3, "Identifying Record Sources for a List"

- Section 9.3.4, "Adding Records to a List"

- Section 9.3.5, "Generating a List"

10

Integrating Oracle Scripting

There are various ways to view integration of Oracle Scripting. Since Oracle Scripting consists of several components, you can consider integration from this perspective. Since Scripting involves technologies using applications built with Oracle Forms Developer (referred to as Oracle forms-based applications) *and* applications using Java Server Pages (JSP) technology built on HTML, you can consider integration by technical stack. Finally, Scripting includes pre-built integration, while also allowing custom integration with Application Program Interfaces (APIs). Integration can also be described from this perspective. Each is described in turn below.

This section includes the following topics:

- Section 10.1, "Integration by Component"

- Section 10.2, "Integration by Technical Stack"

- Section 10.3, "Prebuilt Integration with Oracle Business Applications"

10.1 Integration by Component

Oracle Scripting consists of the Script Author, Scripting Engine, and Survey components. Each has different integration considerations.

This section includes the following topics:

- Section 10.1.1, "Script Author"

- Section 10.1.2, "Scripting Engine"

- Section 10.1.3, "Survey Component"

See Also

- Section 10.2, "Integration by Technical Stack"

- Section 10.3, "Prebuilt Integration with Oracle Business Applications"

10.1.1 Script Author

The Script Author is the component of Oracle Scripting used to create, modify and deploy scripts. As of Oracle Applications release 11.5.8 (CRM Family Pack P), the Script Author is a Java applet that is launched from an existing Oracle Applications session. To launch the Script Author, log into Oracle Applications as a user with the Scripting Administrator responsibility, and click Launch Script Author from the Home tab of the Scripting Administration Console. Oracle JInitiator will launch the Script Author applet on the client. Since the connection is established and associated with a specific database and applications instance, using the Script Author Java applet allows script developers to open, save, and deploy scripts to and from that database instance. Likewise, connection information need not be specified to save commands to the command library or to import library commands into a script.

The Script Author Java applet replaces a stand-alone Script Author Java application available in prior releases which required a Windows operating system on the client. Using the stand-alone application also required the client to be on the same side of the firewall as the applications database, and necessitated apps database user access to deploy scripts or otherwise access the database.

The Script Author can be used on the same computer as any other Oracle Application, but as a Java applet is not in itself integrated with other applications.

See Also

- Section 10.1.2, "Scripting Engine"

- Section 10.1.3, "Survey Component"

10.1.2 Scripting Engine

Agent Interface

The first of the Scripting Engine's two interfaces is the agent interface, which is a Java bean wrapped in an Oracle form. Users of the agent interface are typically interaction center agents. The Scripting Engine agent interface provides enterprises

with a method of scripting interactions with customers or prospects and integrating desktop workflow between various applications.

Integration with the Scripting Engine agent interface occurs in three ways:

- Scripts can get, create, or update records in the applications database or in custom tables.

- Scripts can get or set information from any open Oracle form from any Forms-based application that can be launched from the Navigator.

- Scripts can be launched from three integrated Forms-based business applications (Oracle TeleSales, Oracle Collections, and Oracle TeleService).

The agent interface is intended to be used in combination with integrated Oracle eBusiness applications to take full advantage of the contact handling and sales or service features of those applications. Interaction centers leveraging business applications and the Scripting Engine agent interface therefore have the full benefit of interaction center communications. Agents can also execute scripts in "stand-alone" mode, without accessing other applications. In the stand-alone model, integration is typically limited to reading or writing to the Oracle applications database, and login to an Oracle Applications session is still required.

Web Interface

The Scripting Engine Web interface executes scripts in any Oracle Applications 11*i*-certified Web browser. Scripts executed in the Web interface rely on survey campaign information administered using the Survey component.

The Scripting Engine Web interface is integrated with Oracle iSupport. After using iSupport, customers can be surveyed regarding their self-service Web application experience by clicking on a hypertext link provided in the iSupport interface. This links to a URL for a deployed survey campaign deployment.

See Also

- Section 10.1.1, "Script Author"

- Section 10.1.3, "Survey Component"

10.1.3 Survey Component

The Survey component is a JSP-based HTML administration console. Using the survey component, survey administrators create, administer, and deploy survey campaigns and monitor existing campaigns.

The Survey component integrates directly with Oracle Marketing and Oracle One-to-One Fulfillment from the administration console.

In addition to survey-specific functionality, the survey administration console provides access to an Audience tab. This tab displays the same content listed under the Audience tab in the Marketing Online Administration Console. In order to access this tab from the Survey Administration console, the Oracle Applications user account must have a sales group ID and a sales role. These are administrative setup steps required to be performed by a user with the CRM Resource Manager responsibility. You can also configure the Survey Administrator responsibility to bypass this requirement. For this procedure, see the topic Bypassing the Sales Group Membership Requirement in the Administration section of *Oracle Scripting Implementation Guide*.

The survey administration console also provides an Invitations tab. This tab provides access to the pages you can access using the Template, Query, and Status subtabs in the Fulfillment Administration console. Note that a Fulfillment administrator is still required to set up Fulfillment servers, groups, and to monitor fulfillment requests.

See Also

- Section 10.1.1, "Script Author"
- Section 10.1.2, "Scripting Engine"

10.2 Integration by Technical Stack

This section includes the following topics:

- Section 10.2.1, "HTML-Based Integrated Applications"
- Section 10.2.2, "Forms-Based Integrated Applications"

See Also

- Section 10.1, "Integration by Component"
- Section 10.3, "Prebuilt Integration with Oracle Business Applications"

10.2.1 HTML-Based Integrated Applications

This section includes the following topics:

- Section 10.2.1.1, "Oracle iSupport"

- Section 10.2.1.2, "Oracle One-to-One Fulfillment"

- Section 10.2.1.3, "Oracle Marketing"

See Also

- Section 10.2.2, "Forms-Based Integrated Applications"

10.2.1.1 Oracle iSupport

Using self-service Web-based applications such as Oracle iSupport, customers can launch a script in the Scripting Engine Web interface, using their Web browser.

When Oracle Scripting is integrated with Oracle iSupport, users log into iSupport using an Oracle Applications 11*i*-certified Web browser and access a page with one or more hyperlinks corresponding to active standard (non-list-based) survey campaigns.

From this page, an iSupport user clicks a hyperlink corresponding to an active standard survey campaign deployment. Since an Oracle Applications session is already in progress, the authentication information for the logged-in user is used for the Scripting session. In this model, no guest user is required. Clicking the hyperlink invokes an established hosted survey URL, displaying the script designated for that survey campaign in the same Web browser instance. The page refreshes, displaying the survey questionnaire's first panel as an HTML page. Just as in a list-based survey, the survey resource header appears on each HTML page. In addition, the iSupport navigational menus (tabs) are retained at the top of each page throughout.

The hosted survey URL differs from the survey URL executed in the Web interface from an enterprise's Web site or from an invitation or reminder. The hosted survey URL used the IESSVYMENUBASED.JSP, whereas other surveys use IESSVYMAIN.JSP. All other aspects of the survey URL are the same.

For hosted surveys, the page listing survey links is generally designated as the Final page survey resource. Thus, after the script or survey is completed, the user is returned to the page from which they launched the survey script. Other links on that page can be customized as desired by Oracle iSupport administrators.

This model also relies on a different format for the Error page survey resource. A seeded test resource (IESSVYMENUBASEDTESTERROR.JSP) in this format is shipped with Oracle Applications. Oracle iSupport integration with the survey component can be tested using this JSP page, and this JSP resource can be used by an enterprise as a model for building their own error pages for iSupport integration.

Guidelines

- From the Survey tab within Oracle iSupport, the user clicks a link for an active survey. The survey starts in the current Web browser window, preserving the menu structure.

- The survey is expected to be completed in order to deliver reliable information. Abandoned survey responses are tracked but not displayed in most reports.

See Also

- Section 10.2.1.2, "Oracle One-to-One Fulfillment"

- Section 10.2.1.3, "Oracle Marketing"

10.2.1.2 Oracle One-to-One Fulfillment

Oracle One-to-One Fulfillment provides enterprises with an automated method for sending information to its customers. Interaction center agents use Oracle One-to-One Fulfillment directly or through automated applications such as Oracle Marketing. Agents often receive customer requests for information, literature, and other correspondence. These requests include product and service inquiries, pricing questions, billing inquiries, and general customer care issues. Agents can immediately fulfill these requests using e-mail to automatically deliver predefined fulfillment items to identified customers.

Oracle One-to-One Fulfillment is set up and maintained by a Fulfillment administrator, who must configure and run a Fulfillment server to monitor fulfillment requests, and a Fulfillment e-mail server (a logical channel for the output device that delivers electronic documents to customers).

From an Oracle Scripting perspective, Oracle One-to-One Fulfillment can be used to deliver "invitation" and "reminder" master documents (HTML documents) via electronic mail, inviting identified parties to participate in a survey campaign by visiting a defined URL.

See Also

- Section 10.2.1.1, "Oracle iSupport"

- Section 10.2.1.3, "Oracle Marketing"

10.2.1.3 Oracle Marketing

Oracle Marketing is a tool used to create and manage marketing and sales campaigns, assign scripts to particular campaign schedules, assign agents to specific campaigns, etc. You can create lists of customers in Oracle Marketing, and invite

them via e-mail (using Oracle One-to-One Fulfillment "invitation" and "reminder" master documents) to participate in a survey campaign by visiting a defined URL.

See Also

- Section 10.2.1.1, "Oracle iSupport"

- Section 10.2.1.2, "Oracle One-to-One Fulfillment"

10.2.2 Forms-Based Integrated Applications

Oracle Forms-based eBusiness applications that integrate with Oracle Scripting include Oracle TeleSales, Oracle Collections, and Oracle TeleService. From an interaction in any of these business applications, an agent can launch a script in the Scripting Engine agent interface. Each business application supports at least two models: (1) allowing the agent to select any script from a list of deployed scripts for that instance, and (2) specifying a particular script to launch upon a particular condition or action. No additional configuration is required to enable the first model. To enable launch of a particular script, specific setup steps are required, which vary by application. Each of these is discussed in the section for integrating the particular business application. For more information, see *Oracle Scripting Implementation Guide Release 11i* and the implementation guide for each specific application in question.

To execute scripts in the agent interface, the agent's Oracle Applications account must have the correct responsibility associated with it to launch the business application in question. The table below indicates, by application, responsibilities which may be required, and from which form in the application the script can be launched.

Business Application	Required Responsibility	Form
Oracle TeleSales	TeleSales Agent	eBusiness Center
Oracle Collections	Collections Agent	Collections
Oracle TeleService	Customer Support	Contact Center

Each of these Oracle Forms-based business applications are typically used by interaction center agents for sales, collections, or service, respectively. The functionality of these business applications can be greatly extended and customized by creating and using scripts launched from the appropriate business application.

Guidelines

- After logging into the appropriate business application, no specific Scripting responsibilities are required for an agent to launch a script.

- These scripts are launched in the Scripting Engine agent interface in a separate window.

- If choosing from the full list of deployed scripts, the agent must select the appropriate script name and language from the Script Chooser list.

- The script is expected to be completed prior to returning to the business application.

- If the script commits changes to the customer information in the database, these changes will not be reflected in the business application until a fresh database pull is requested from the business application.

- Individual implementation, setup, or configuration steps may be required in order to access the business applications listed above. For example, membership in a sales or telesales group (for Oracle TeleSales) or a collections group (for Oracle Collections) is required. Other conditions may apply for each application. These steps are outside the scope of this document. For more information, refer to the appropriate documentation for the relevant business application.

Reference

For relevant information regarding Oracle forms-based applications, refer to *Oracle Applications Developer's Guide* Release 11i, Volume 1. Specifically, refer to the section Menus and Function Security.

See Also

- Section 10.2.1, "HTML-Based Integrated Applications"

10.3 Prebuilt Integration with Oracle Business Applications

Interaction centers using the Scripting Engine agent interface can not only interact with customers but can also integrate desktop workflow between various applications. Scripting is intended to be used in combination with integrated Oracle eBusiness applications (Oracle TeleService, Oracle TeleSales, and Oracle Collections) to take full advantage of the contact handling and service, sales, or collections features of those applications, respectively. In terms of prebuilt integration, scripts can be launched from any of these customer-facing, interaction-based business

applications. Additionally, Oracle TeleSales and Oracle Collections automatically send values for a selected customer to the Scripting blackboard for use in a script interaction.

This section includes the following topics:

- Section 10.3.1, "Integrating with Oracle TeleSales"

- Section 10.3.2, "Integrating with Oracle Collections"

- Section 10.3.3, "Integrating with Oracle TeleService"

See Also
- Section 10.1, "Integration by Component"

- Section 10.2, "Integration by Technical Stack"

10.3.1 Integrating with Oracle TeleSales

This section includes the following topics:

- Section 10.3.1.1, "Launching Any Deployed Script from Oracle TeleSales"

- Section 10.3.1.2, "Launching a Specific Script from Oracle TeleSales"

- Section 10.3.1.3, "Parameters Passed to the Scripting Blackboard"

See Also
- Section 10.3.2, "Integrating with Oracle Collections"

- Section 10.3.3, "Integrating with Oracle TeleService"

10.3.1.1 Launching Any Deployed Script from Oracle TeleSales

Using Oracle TeleSales, you can launch a script of your choice in the Scripting Engine agent interface from the full set of scripts deployed to your Oracle Applications instance.

Prerequisites
- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- Oracle TeleSales sends a set of values for the selected customer to the Scripting blackboard when a script is launched from the eBusiness Center. In order to receive those values and use them in a script, the agent must be in an active TeleSales transaction with a customer selected in the eBusiness Center.

Login
Log into Oracle Forms-based applications.

Responsibility
TeleSales Agent

Steps
1. From the Navigate To menu, select **All Scripts**.

   The Oracle Scripting window (commonly referred to as the Script Chooser) appears.

2. From the Script Chooser, select a script name and language.

3. Click **Start Scripting**.

   The script will launch in a separate window.

See Also
- Section 10.3.1.2, "Launching a Specific Script from Oracle TeleSales"
- Section 10.3.1.3, "Parameters Passed to the Scripting Blackboard"

10.3.1.2 Launching a Specific Script from Oracle TeleSales
Using Oracle TeleSales, you can launch a designated script in the Scripting Engine agent interface. There are three models for launching this designated script:

- Automatic launch of a specific script when a call is received.

- Automatic launch of a specific script when an interaction is started in the eBusiness Center

- Launch of a specific script based on explicit agent request (agent clicks **View Script** button)

Oracle Marketing Campaign Administration Requirements
To designate the specific script to launch from Oracle TeleSales using any of these models requires the script to be designated at the campaign schedule level. Thus, as

a prerequisite of launching a designated script, you must administer campaigns in Oracle Marketing. Oracle Marketing requirements include:

- a valid campaign

- a valid campaign schedule associated with the campaign

- an inbound or outbound script associated with the campaign schedule

- agents must be associated to the campaign schedule, or campaign schedules must be assigned to agents

For more information on these steps, see Oracle Marketing product documentation. For your convenience, these steps are also documented in the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing in *Oracle Scripting Implementation Guide*.

Obviously, there is no explicit action to be taken by the agent for automatic launch of a script. Based on the type of automatic launch (on call delivery or on start of interaction), when the event or condition is met, the script launches in a separate window. Thus, the procedure below applies to launching a specified script from the eBusiness Center at a time chosen by the agent.

Specific profile settings are required to accomplish each method of launching a specified script. These are described in the guidelines below.

Prerequisites

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- As a prerequisite of Oracle TeleSales, a campaign must exist in Oracle Marketing, and the agent using the appropriate business application must be associated with the campaign.

- The specific deployed script must be associated to either the campaign or campaign schedule from Oracle Marketing in order to launch without accessing the Script Chooser.

- System profiles must be set according to the guidelines indicated.

- Oracle TeleSales sends a set of values for the selected customer to the Scripting blackboard when a script is launched from the eBusiness Center. In order to receive those values and use them in a script, the agent must be in an active TeleSales transaction with a customer selected in the eBusiness Center.

Login

Log into Oracle Forms-based applications.

Responsibility

TeleSales Agent

Steps

1. Ensure a customer is selected in the eBusiness Center.

2. Ensure a campaign is selected in the Overview tab.

   The selected campaign must have a script associated with it.

3. From the Navigate To menu, select **View Script**, or from the Overview tab, click **View Script**.

   The script will launch in a separate window.

Guidelines

- As indicated above, campaign administration of Oracle Marketing is required.

- For *any* model of launching scripts from Oracle TeleSales or Oracle Collections, set the OTS: Scripting Installation profile to **Yes**.

- For any model of *automatically* launching a *specified* script, also set the OTS: Script Launch on Interaction profile to **Yes**.

- For Oracle TeleSales to launch a script upon delivery of the call to the agent desktop only, also set the OTS: Script Launch on UWQ Delivery profile to **Yes**.

Following is a table describing the effects of the OTS profile settings for Oracle Scripting:

Profile Option	Setting	Result
OTS: Script Launch on UWQ Delivery	Yes	Automatic launch of a specific script in a separate window when a call is received in Oracle TeleSales.
OTS: Script Launch on UWQ Delivery	No or Null	No script launches on call delivery.
OTS: Script Launch on Interaction	Yes	When a TeleSales interaction starts, the specified script automatically launches in a separate window.

Profile Option	Setting	Result
OTS: Script Launch on Interaction	No or Null	No script launches on start of interaction.
OTS: Scripting Installation	Yes	This setting allows scripts to be launched from TeleSales.
OTS: Scripting Installation	No	Regardless of other profile settings, if set to no, scripts will not launch from TeleSales or Collections.

Launching a Specified Script on Call Delivery to Agent Desktop

Profile Option	Setting
OTS: Script Launch on UWQ Delivery	Yes
OTS: Script Launch on Interaction	Yes
OTS: Scripting Installation	Yes

Launching a Specified Script Automatically on Start of Interaction

Profile Option	Setting
OTS: Script Launch on UWQ Delivery	No or null
OTS: Script Launch on Interaction	Yes
OTS: Scripting Installation	Yes

Launching a Specified Script When Agent Clicks View Script

Profile Option	Setting
OTS: Script Launch on UWQ Delivery	No or null
OTS: Script Launch on Interaction	No
OTS: Scripting Installation	Yes

References

■ For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle

Marketing > Associating a Script to a Campaign in *Oracle Scripting Implementation Guide*.

■ For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign Schedule in *Oracle Scripting Implementation Guide*.

See Also

■ Section 10.3.1.1, "Launching Any Deployed Script from Oracle TeleSales"

■ Section 10.3.1.3, "Parameters Passed to the Scripting Blackboard"

10.3.1.3 Parameters Passed to the Scripting Blackboard

Oracle TeleSales and Oracle Collections integrate with Oracle Scripting by passing parameters from a current TeleSales or Collections interaction to the Scripting Blackboard each time a script is launched from the appropriate business application. The parameters passed are listed in the table below.

Parameter
PARTY_ID
CONTACT_ID
PARTY_CONTACT_ID
CUST_ACCOUNT_ID
ADDRESS_ID
CAMPAIGN_ID
CAMPAIGN_SOURCE_CODE
AST_RESOURCE_ID
TM_DNIS
TM_ANI
OPP_ID
LEAD_ID
COLLATERAL_ID
EVENT_ID

Parameter
PARM.PARTY_ID
P_PARTY_ID
PARTY_TYPE
INTERACTION_ID

Guidelines

- Values are passed each time a script is launched from the eBusiness Center or from Oracle Collections.

- Any parameters not available from the appropriate business application will be null in the Scripting Blackboard.

- A customer must be selected in TeleSales or Collections in order to receive some or all values into the script.

- If a script relies on values passed from TeleSales or Collections for critical processing, the script should include a test for a null value before attempting the process. If a null value is found, the script should include panels to collect the required information at runtime, populating them with the appropriate blackboard answer keys to ensure the script can continue its processing.

- If a blackboard value retrieved into a script is subsequently changed in the business application before the script is completed, the updated value will not display in the script. If the script is completed and started again, the updated value will be retrieved into the script.

See Also

- Section 10.3.1.1, "Launching Any Deployed Script from Oracle TeleSales"

- Section 10.3.1.2, "Launching a Specific Script from Oracle TeleSales"

10.3.2 Integrating with Oracle Collections

Oracle Collections shares some interaction center functionality with Oracle TeleSales, and as such relies on many TeleSales application settings and behavior. For example, the blackboard values sent to a script when a script is launched from TeleSales or Collections (see Section 10.3.1.3, "Parameters Passed to the Scripting Blackboard") are identical.

As another example, as indicated in Guidelines below, specific TeleSales profile settings are required in order to launch a designated script from Collections.

Additionally, collections agents must be assigned specific role types and roles, and must have group membership with specific role types, roles, and usages. These are administered in CRM Resource Manager. Oracle Collections requirements include:

■ Collections agent must be imported as a CRM resource. This is administered in Oracle CRM Resource Manager.

> **Note:** To import the agent as a resource, that agent must already exist as an employee in the database. Employees are created in Oracle HRMS Manager (if installed) or in Oracle CRM Resource Manager.

■ The CRM resource (the agent) must have the following attributes:

Role Type	Role
Collections	Collections Agent
Telesales	Telesales Agent

■ The agent must have membership in a resource group. For Oracle Collections, the resource group must have the following attributes:

Role Type	Role	Usage
Collections	Collections Agent	Collections
Telesales	Telesales Agent	Sales and Telesales

> **Note:** In order to use Oracle Collections, *you must ensure the resource and the resource group have attributes set for both Oracle Collections and for Oracle Telesales,* even if the agent is only expected to use Oracle Collections. For more information, see *Oracle Collections Implementation Guide release 11i.*

This section includes the following topics:

See Also

10.3.2.1 Launching Any Deployed Script from Oracle Collections

Using Oracle Collections, you can launch a script of your choice in the Scripting Engine agent interface from the full set of scripts deployed to your Oracle Applications instance.

Prerequisites

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- Oracle TeleSales and Oracle Collections sends a set of values for the selected customer to the Scripting blackboard when a script is launched from the appropriate business application. In order to receive those values and use them in a script, the agent must be in an active TeleSales or Collections transaction with a customer selected in the appropriate business application.

Login

Log into Oracle Forms-based applications.

Responsibility

Collections Agent

Steps

1. From the Collections window, navigate to the Profile tab.

2. From the Profile tab, click **View Script**.

   The Oracle Scripting window (commonly referred to as the Script Chooser) appears.

3. From the Script Chooser, select a script name and language.

4. Click **Start Scripting**.

   The script will launch in a separate window.

See Also

- Section 10.3.2.2, "Launching a Specific Script from Oracle Collections"
- Section 10.3.1.3, "Parameters Passed to the Scripting Blackboard"

10.3.2.2 Launching a Specific Script from Oracle Collections

Using Oracle Collections, you can launch a designated script in the Scripting Engine agent interface. There are two models for launching this designated script:

- Automatic launch of a specific script when a Collections interaction is started.
- Launch of a specific script upon agent entry of a specified code and the application's subsequent successful validation of that code.

Oracle Marketing Campaign Administration Requirements

To designate the specific script to launch from Oracle Collections using either of these models requires the script to be designated at the campaign schedule level. Thus, as a prerequisites of launching a designated script, you must administer campaigns in Oracle Marketing. Oracle Marketing requirements include:

- a valid campaign, *including the designation of a campaign source code*
- a valid campaign schedule associated with the campaign
- an inbound or outbound script associated with the campaign schedule
- agents associated to either the campaign or to the campaign schedule

For more information on these steps, see Oracle Marketing product documentation. For your convenience, these steps are also documented in the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing in *Oracle Scripting Implementation Guide*.

Starting a Collections Interaction

There are several ways to start an interaction in Oracle Collections. At any time, an interaction can be started by the agent by clicking the Start Interaction icon in the tool bar. Also, by default, Oracle Collections starts an interaction when a customer is selected in the Collections window. Like most Oracle applications, Collections can

be modified to start an interaction after a customer is selected and any other event on the form is triggered (such as selecting a specific field).

Regardless of how a Collections interaction is started, there is no explicit action to be taken by the agent in order to cause the automatic launch of a script, if Collections is configured to do so upon start of an interaction. Thus, the procedure below applies to an agent causing the launch of a specified script from Oracle Collections by entering a code in the Code field within the Profile tab. The Code field associates with a campaign, campaign schedule, event, or promotion code, as established in Oracle Marketing.

Prerequisites

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- As prerequisites of Oracle TeleSales and Oracle Collections, a campaign must exist in Oracle Marketing, and the agent using the appropriate business application must be associated with the campaign.

- The specific deployed script must already be associated to either the campaign or campaign schedule from Oracle Marketing in order to launch without accessing the Script Chooser.

- System profiles must be set according to the guidelines indicated below.

- Oracle Collections send a set of values for the selected customer to the Scripting blackboard when a script is launched from the Collections window. In order to receive those values and use them in a script, the agent must be in an active Collections transaction with a customer selected.

Login

Log into Oracle Forms-based applications.

Responsibility

Collections Agent

Steps

1. From the Navigator, select Collections.

   The Collections window appears.

2. In the Profile tab, type the appropriate value in the Code field and press the tab key.

   Oracle Collections will automatically validate this code against campaign information established in Oracle Marketing.

3. If the code is valid, the Name and Type fields will populate with information from Oracle Marketing, and the specified script will launch in a separate window.

   If the code is not valid, no script will launch.

   > **Note:** The Name and Type fields cannot be used to launch a script. Only by typing the appropriate code in the Code field can agents launch the specified script automatically. If desired, an agent can elect to click the View Script button and manually select a valid script from the list of available scripts in the Script Chooser.

Guidelines

- For Collections to launch a script upon delivery of the call to the agent desktop, the following Oracle TeleSales profiles must be set:

Profile Option	Setting
OTS: Script Launch on Interaction	Yes
OTS: Scripting Installation	Yes

References

- For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign Schedule in *Oracle Scripting Implementation Guide*.

- For information on associating a script with a campaign schedule, see the topic Administering Oracle Scripting > Administering Campaigns in Oracle Marketing > Associating a Script to a Campaign in *Oracle Scripting Implementation Guide*.

See Also

- Section 10.3.2.1, "Launching Any Deployed Script from Oracle Collections"

- Section 10.3.1.3, "Parameters Passed to the Scripting Blackboard"

10.3.3 Integrating with Oracle TeleService

This section includes the following topics:

- Section 10.3.3.1, "Launching Any Deployed Script from Oracle TeleService"

- Section 10.3.3.2, "Launching a Specific Script from Oracle TeleService"

- Section 10.3.3.3, "Obtaining Values in a Script from Oracle TeleService"

- Section 10.3.3.4, "Creating Service Requests in Oracle TeleService from Oracle Scripting"

See Also

- Section 10.3.1, "Integrating with Oracle TeleSales"

- Section 10.3.2, "Integrating with Oracle Collections"

10.3.3.1 Launching Any Deployed Script from Oracle TeleService

Using Oracle TeleService, you can launch a script of your choice in the Scripting Engine agent interface from the full set of scripts deployed to your Oracle Applications instance.

Prerequisites

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- System profiles must be set according to the guidelines indicated below.

- If the script being called uses any Forms commands from any TeleService form, that form must be open and appropriately populated with information in the Contact Center for those values to be made accessible to the script.

Login

Log into Oracle Forms-based applications.

Responsibility

Customer Support

Steps

1. Ensure a customer is selected in the Contact Center.

2. From the Tools menu, select **Script**, or click on the Script icon in the tool bar.

   The Oracle Scripting window (commonly referred to as the Script Chooser) appears.

3. From the Script Chooser, select a script name and language.

4. Click **Start Scripting**.

   The script will launch in a separate window.

Guidelines

■ For Oracle TeleService to launch a script upon delivery of the call to the agent desktop, the following profile must be set:

Profile Option	Setting
Customer Care: Default Contact Center Script	<global script name>

See Also

■ Section 10.3.3.2, "Launching a Specific Script from Oracle TeleService"

■ Section 10.3.3.3, "Obtaining Values in a Script from Oracle TeleService"

■ Section 10.3.3.4, "Creating Service Requests in Oracle TeleService from Oracle Scripting"

10.3.3.2 Launching a Specific Script from Oracle TeleService

Using Oracle TeleService in a media-enabled environment, you can automatically launch a designated script in the Scripting Engine agent interface. As soon as a call is delivered to the Oracle TeleService Contact Center, if at least one of three parties is identified, the designated script launches in a separate window. In the event that a party is not identified (the TeleService screen pop is blank), the script is not launched. Only one of three possible parties (organization, person, or relationship) is required to launch the script, as identified by the HZ_PARTIES table in the Oracle Applications schema.

There is no explicit action to be taken by the agent for automatic launch of a script other than appropriate setup, as indicated in the guidelines below.

Prerequisites

- Oracle TeleService must be media-enabled.

- In order to launch a designated script automatically, the calling party (rel party) must be identified.

- At least one appropriate script must be deployed from the Script Author to the applications database.

- If custom code is referenced in the script, that code must be available to the Scripting Engine at runtime.

- System profiles must be set according to the guidelines indicated below.

- If the script being called uses any Forms commands from any TeleService form, that form must be open and appropriately populated with information in the Contact Center for those values to be made accessible to the script.

Login

Log into Oracle Forms-based applications.

Responsibility

Customer Support

Steps

1. From the Navigator, select Contact Center and click **Open**.

   The Contact Center window opens.

2. When a call is received, the designated script launches in a separate window.

3. Complete interaction within the script before returning to the Contact Center.

Guidelines

- For Oracle TeleService to launch a script upon delivery of the call to the agent desktop, the following profile must be set:

Profile Option	Setting
Customer Care: Default Contact Center Script	<global script name>

See Also

- Section 10.3.3.1, "Launching Any Deployed Script from Oracle TeleService"

-
-

10.3.3.3 Obtaining Values in a Script from Oracle TeleService

In addition to being able to launch a script from Oracle TeleService, Oracle Scripting can retrieve values from the TeleService Contact Center using Forms commands in the Script Author. Upon execution of the Forms command at script runtime, the Contact Center retrieves the appropriate contact information, and passes this information on to Scripting.

Some typical Forms commands used to pass values from Oracle TeleService to Oracle Scripting are listed in the table below:

Forms Command	Value Passed to Script
GetInteractionId	Interaction ID
GetCustomerId	Account ID/Party ID
GetCustAccountId	Customer Account ID
GetCustomerType	Customer Type
GetContactId	Contact ID
GetContactPointId	Contact Point ID
GetContactPointType	Contact Point Type
GetContactType	Contact Type
GetPrimaryFlag	Primary Flag

References

- For more information on Forms commands, see the Forms Commands section of the *Oracle Scripting Developer's Guide*.
- A script can also be associated with a Relationship Plan. For information on relationship plans, refer to *Oracle Customer Care Implementation Guide release 11i*.

See Also

-
-

10.3.3.4 Creating Service Requests in Oracle TeleService from Oracle Scripting

Oracle Scripting integrates with Oracle Forms-based applications. To create a service request by calling a PL/SQL package from Scripting, the following fields must be populated:

- SR Type
- SR Status
- SR Severity
- Summary
- Group Owner (see guidelines)
- Individual Owner (see guidelines)

Guidelines

- If set to "Yes", the Oracle TeleService profile option, Service: Group Owner Mandatory, requires the user to enter a group owner when creating a service request.
- If set to "Yes", the Oracle TeleService profile option, Service: Default Group Owner for Service Requests, requires either the group owner *or* an individual owner to be entered when creating a service request. This profile takes precedence over the Service: Group Owner Mandatory profile option, if set.
- The usage of a CRM resource must be defined as "Support" for the resource to be available in the Service Request tab in Oracle TeleService. This usage is set when the resource is defined using Oracle CRM Resource Manager.

See Also

- Section 10.3.3.1, "Launching Any Deployed Script from Oracle TeleService"
- Section 10.3.3.2, "Launching a Specific Script from Oracle TeleService"
- Section 10.3.3.3, "Obtaining Values in a Script from Oracle TeleService"

11

Implementation Verification

Implementation verification tasks may differ, based on which components you implement and use at the enterprise.

At minimum, every implementation requires you to create or obtain a functional script and deploy it to your environment using the Script Author. Oracle Corporation recommends creating or obtaining a simple test script without custom Java and a script that includes references to custom Java.

Only scripts to be executed in the Web interface (whether executed as surveys, or as scripts executed in a Web browser from an integrated self-service Web application) require you to perform survey resource and survey campaign administration steps.

All scripts must be executed in the Scripting Engine to verify your implementation. Because there are fewer environmental factors affecting success, Oracle Corporation recommends first verifying test scripts in the Scripting Engine agent interface, even if your implementation only requires you to execute scripts in the Scripting Engine Web interface. If your enterprise requires scripts to be executed in a Web browser, then after testing in the agent interface you should test your scripts in the Web interface.

Other verification tasks specific to your Oracle Applications operating system or environment may apply.

This section includes the following topics:

- Section 11.1, "Script Author Tasks"

- Section 11.2, "Survey Administration Console Tasks"

- Section 11.3, "Scripting Engine Tasks"

- Section 11.4, "Testing Display Server for UNIX Environments"

11.1 Script Author Tasks

All implementations must be verified using one or more scripts. Oracle Corporation recommends first testing your implementation with the Scripting Engine agent interface, using a simple script with no associated custom Java. Step-by-step instructions to create and deploy a Hello World graphical test script using Script Author's standard visual tools is provided in this section for this purpose. If using the Script Wizard feature of the Script Author, follow the instructions provided to create the Hello World wizard test script.

Testing a simple script with no associated custom Java by executing it in the Scripting Engine agent interface verifies that:

- A script can be launched in the designated environment, independent of the technical considerations related to custom Java.

- You eliminate certain environmental variables as the cause of problems if this test succeeds and the subsequent test (launching and executing a script that calls custom Java methods or other custom commands) fails.

- Script Author is compiling and deploying scripts successfully.

- Scripting User or Scripting Agent responsibility functions appropriately.

If your implementation requires custom Java, you must test a script that references custom Java. Oracle Corporation recommends using the regression test script posted on Oracle*MetaLink* and Oracle iSupport for that purpose. If you can successfully execute the regression test script, you quickly verify that:

- A script with custom Java methods associated with it can be launched in the designated environment.

- The Apache Web server can locate and interpret the custom Java classes.

- Your environment is correctly set up to deploy custom Java to the database, map custom Java to scripts, and make the appropriate methods available at runtime.

- Pending the appropriate configuration of environmental factors, a customized script will function properly in your environment.

Tasks

You can perform the following tasks:

- Section 11.1.1, "Creating a Hello World Graphical Test Script"

- Section 11.1.2, "Creating a Hello World Wizard Test Script"

-

See Also

-

-

-

11.1.1 Creating a Hello World Graphical Test Script

Prerequisites

- To create or deploy scripts using the Script Author Java applet, you must use the Scripting Administration console. Thus, you must have access to an Oracle Applications account with the Scripting Administrator responsibility.

- All implementation tasks must be accomplished.

Steps

Follow the steps below.

Step	Perform From	Action	Result
1	Oracle Applications HTML Login	Log into Oracle HTML-based applications using a User ID with the Scripting Administrator responsibility.	The Scripting Administration console appears.
2	Scripting Administration console	From the Home tab, click **Launch Script Author**.	A new browser window opens, Oracle JInitiator launches, and the Script Author Java applet appears in a separate window.
3	File menu *or* Script Author Toolbar	Create a new script using one of the following methods: - From the Script Author **File** menu, select **New.** - From the Script Author toolbar, click the **New Script** icon.	The new script window appears, with options for starting a new graphical script or a new wizard script.

Step	Perform From	Action	Result
4	New Script window	From the New Script window, click Graphical script.	A new, untitled script appears within the Script Author's visual layout region (the "canvas"), including a Start node.
5	Script Author Toolbar	Optionally, adjust the view of the script from the Script Author toolbar. For example: ■ To display a 100% view of the canvas, click the **Zoom to 100%** icon. ■ To zoom in or out as required to display the entire script in the current window, click the **Zoom To Fit Graph In Window** icon. ■ For a closer or further view of the canvas, click the **Zoom In** or **Zoom Out** icon, respectively.	The view of the script on the canvas adjusts per your request.
6	Script Author Toolbar	Click the **Panel Insertion Mode** icon (the blue sphere) from the Toolbar to indicate you wish to insert a new panel.	Script Author Toolbar
7	Script Author canvas	Click *once* in the canvas below the Start node. This action places a panel icon (shown as a blue sphere) into the canvas. **Note:** Until you switch to another Toolbar object or switch to toggle selection mode (performed in the next step), the current Toolbar object remains selected and will continue to place objects (in this case, panels) on the canvas each time the mouse is clicked in the work area. If you prefer to disable this feature, click to clear the "Sticky Mode" option under the **File** menu. The option to disable sticky mode is retained only for the duration of the current Script Author session. If you prefer this option, you must select it each time you launch the Script Author.	Script Author canvas

Step	Perform From	Action	Result
8	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the Toolbar to turn off the panel insertion mode. **Note:** If you clear the "Sticky Mode" option as described in the step above, selecting Toggle Selection Mode will not be required. However, when using the same toolbar tool repeatedly (when dropping multiple objects, or drawing multiple branches of the same type), you will be required to select the desired object from the toolbar each time, which is not required when "Sticky Mode" is enabled.	Script Author Toolbar
9	Script Author canvas	**Right-click** the panel that you just added to the script. The Script Author displays a drop down menu with the following options: Edit, Edit Blob Name, and Edit Blob Properties. **Note:** If the menu displayed shows only two options, Edit and Edit Blob properties, then you have not correctly selected the Panel object. Left-click to close the menu and re-select the Panel object.	Script Author canvas
10	Panel Properties window	Select the **Edit Blob Properties** option from the menu created by right-clicking on the new panel. The Script Author displays the Panel**[n]** Properties window. **Note:** The Script Author automatically assigns numbering to each new panel created (represented by **[n]** above) to enforce unique panel names. When you type a value in the Name field of the panel properties (such as "Hello") and apply the change, the value you entered replaces the **[n]** as the new name of the panel you have just created. Thus, for example, Panel5 Properties becomes PanelHello Properties. For the sake of simplicity, the name of each new panel (the value of **[n]**) will be excluded in these instructions, and the window will be referred to generically as the Panel Properties window.	Panel Properties window

Step	Perform From	Action	Result
11	Panel Properties window	In the **Panel Properties** window, input the following values in the prescribed fields: Name: Overwrite the default "Panel[n]" entry and type **hello** From this point on, only the values you should enter will be provided. In each case, overwrite the default entry. Comments: *Leave blank* Label: **Greeting**	Panel Properties window
12	Panel Properties window	Click **Apply** in the Panel Properties window. **Note:** Clicking the **Apply** button saves the values entered or changed (just as if you had clicked the **Ok** button), but leaves the window open for additional work.	Panel Properties window
13	Panel Properties window	Click the **Answers** property displayed on the left pane of the Panel Properties window. The right pane of the window changes to display a text area which will list any questions - also known as answer definitions - that you define for the script.	Panel Properties window
14	Panel Properties window	Click the **Add** button, which is immediately below the Answers text area. The Answer Entry Dialog window appears.	Panel Properties window
15	Answer Entry Dialog window	Provide the following values for the fields in the Answer Entry Dialog window: Default for Distinct Branching: *checkmark* Selecting this check box is optional. Panels which do not use distinct branching are not required to enable this property. Name: **helloAnswer** UI Type: **Button** Label for reporting: *Leave blank*	Answer Entry Dialog window
16	Panel Properties window	From the bottom of the Answer Entry Dialog window, click the **Edit Data Dictionary** button. The application displays the Edit Data Dictionary dialog.	Panel Properties window
17	Edit Data Dictionary dialog	Select the Lookups tab. **Note:** Lookup values are referred to in the Script Wizard as answer choices.	Edit Data Dictionary dialog

Step	Perform From	Action	Result
18	Edit Data Dictionary dialog	On the Lookups tab, click the **Specify lookups** radio button. The Lookups list and its corresponding Add button enable. This list is currently empty.	Edit Data Dictionary dialog
19	Lookup entry dialog	Click the **Add** button. The application displays the Lookup entry dialog.	Lookup entry dialog
20	Lookup entry dialog Edit Data Dictionary dialog	In the Lookup entry dialog, assign the following values to the fields: Display Value: **Continue** Value: **Continue** Click **Ok** to close the Lookup entry dialog and return to the Edit Data Dictionary dialog. Click **Ok** to close the Edit Data Dictionary dialog and return to the Answer Entry Dialog window.	Lookup entry dialog Edit Data Dictionary dialog
21	Panel Properties window	Click **Ok** to close the Answer Entry Dialog window. The answer that you have defined now appears in the Answers text area in the Panel Properties window. It should appear as an entry with the name **Answer:helloAnswer (default)**.	Panel Properties window
22	Panel Properties window	Click **Ok** to close the Panel Properties window. On the canvas below the Start node, the panel (for which you have just configured name and answer properties) appears.	Panel Properties window
23	Tools menu *or* Script Author canvas *or* keyboard	Click *once* on the panel you have just created. When the markers appear around the panel to indicate that you have selected it, open the panel layout editor. **Note:** The panel layout editor can be opened by selecting **Panel Layout Editor** from the **Tools** menu, by clicking on the **Panel Layout** icon from the Toolbar, or by entering the keyboard combination command Ctrl-Shift-E (Command-Shift-E for the Macintosh OS). It can also be accessed by selecting the **Panel Layout** property displayed on the left pane of the Panel Properties window, and then selecting **Panel Layout Editor** from the Panel Layout details area.	Tools menu *or* Script Author canvas *or* keyboard

Step	Perform From	Action	Result
24	Panel layout editor Toolbar	When the panel layout editor opens in Script Author 11.5.6 or later, representative controls for any answers you have defined appear on the page, with room above for text. Click once in the text entry area. From the toolbar, click the **Set Spoken Text Font** icon. Then enter the following text: **Hello, World!** **Note:** The text is formatted as "Spoken Text," allowing one-click formatting to provide an easy visual clue. For agents, this is typically text that would be read to a customer during an interaction. For scripts run in the Web interface as a survey, this is an easy way to emphasis text or use one of two preexisting formats.	Panel layout editor Toolbar
25	Panel layout editor *or* Unsaved Changes dialog	Save the panel text you have created and exit the panel layout editor. You can do this by selecting **Save** and then **Exit** (both from the **File** menu), or you can simply exit the panel layout editor and click the **Yes** button when the Unsaved Changes dialog asks if you would like to save the changes you have made.	Panel layout editor *or* Unsaved Changes dialog
26	Script Author Toolbar	Select the **Group Insertion Mode** icon from the Toolbar.	Script Author Toolbar
27	Script Author canvas	Click **once** on the canvas, below the **hello** panel. The Script Author will place a Group object into the script at the insertion point.	Script Author canvas
28	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the Toolbar to turn off the Group Insertion mode.	Script Author Toolbar
29	Script Author canvas	**Double-click** the newly-added group. The Group Properties dialog is displayed.	Script Author canvas
30	Group Properties dialog	Fill out the fields in the Group Properties dialog in the following manner: Name: **WrapUpGroup** Comment: **Enables the Disconnect button** **Note:** The comment is not required.	Group Properties dialog
31	Group Properties dialog	Click **Apply** in the Group Properties dialog.	Group Properties dialog

Step	Perform From	Action	Result
32	Group Properties dialog	Click the **Shortcut** property from the items on the left pane of the Group Properties dialog. The right pane is re-written and a new field appears. You must fill it out as follows: Shortcut: **WrapUpShortcut** **Note:** This Shortcut name must appear *exactly as typed here, with upper and lowercase text and no spaces.* For scripts run in the agent application, the Scripting Engine associates the **Disconnect** button with any group containing the Shortcut property of **WrapUpShortcut**.	Group Properties dialog
33	Group Properties dialog	Click **OK** in the Group Properties dialog. The dialog closes and the new group appears on the canvas below the **hello** panel. A new group represents a second graph. The original graph on which the **hello** panel and the **WrapUpGroup** group reside is referred to as the Root Graph.	Group Properties dialog
34	Script Author canvas	From the Script Author canvas, **click once** on the **WrapUpGroup** icon to select it. The selection markers will appear.	Script Author canvas
35	Script Author Toolbar	Click the **Go Down Into Child Graph** icon from the Toolbar. You will now see the child graph of the new group, which contains only a Start node.	Script Author Toolbar
36	Script Author Toolbar	Click the **Panel Insertion Mode** icon from the Toolbar and **click once** in the child graph below the Start node.	Script Author Toolbar
37	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the Toolbar to turn off the Panel Insertion mode.	Script Author Toolbar

Step	Perform From	Action	Result
38	Script Author canvas	**Double-click** the new panel to display the Panel Properties window. Click on the Properties property from the items on the left pane of the Panel Properties window, and enter the following values in the prescribed fields: Name: **goodbye** Comments: *Leave blank* Label: **Last panel** **Note:** The Script Author "remembers" the last property panel or field location displayed in any similar dialog. Thus, you may need to select a different property or field in a given dialog in order to proceed with the steps indicated in these instructions.	Script Author canvas
39	Panel Properties window	Click **Apply** in the Panel Properties window.	Panel Properties window
40	Panel Properties window	Click the **Answers** property displayed on the left pane of the Panel Properties window. The right pane of the dialog changes to display a text area which will list any answers you define for the script.	Panel Properties window
41	Panel Properties window	Click the **Add** button to define your answer in the Answer Entry Dialog window.	Panel Properties window
42	Answer Entry Dialog window	Provide the following values for the fields in the Answer Entry Dialog window: Default for Distinct Branching: *checkmark* Name: **goodbyeAnswer** UI Type: **Button** Label for reporting: *Leave blank*	Answer Entry Dialog window
43	Panel Properties window	From the bottom of the Answer Entry Dialog window, click the **Edit Data Dictionary** button. The application displays the Edit Data Dictionary dialog.	Panel Properties window
44	Edit Data Dictionary dialog	On the Lookups tab, click the **Specify lookups** radio button.	Edit Data Dictionary dialog
45	Lookup entry dialog	Click the **Add** button. The application displays the Lookup entry dialog.	Lookup entry dialog

Step	Perform From	Action	Result
46	Lookup entry dialog Edit Data Dictionary dialog	In the Lookup entry dialog, assign the following values to the fields: Display Value: **Finish** Value: **Finish** Click **Ok** to close the Lookup entry dialog and return to the Edit Data Dictionary dialog. Click **Ok** to close the Edit Data Dictionary dialog and return to the Answer Entry Dialog window. Click **Ok** to close the Answer Entry Dialog window.	Lookup entry dialog Edit Data Dictionary dialog
47	Panel Properties window	Click **Apply** in the Panel Properties window.	Panel Properties window
48	Panel Properties window	Click the **Panel Layout** property displayed on the left pane of the Panel Properties window. The right pane of the dialog changes to display the Panel Layout details, including a **Browse** button from which you can import HTML, and a **Panel Layout Editor** button.	Panel Properties window
49	Panel Properties window	Click **Panel Layout Editor.** The panel layout editor for the **goodbye** panel appears.	Panel Properties window
50	Panel layout editor Toolbar	In the panel layout editor dialog, click once in the text entry area. From the toolbar, click the **Set Spoken Text Font** icon. Then enter the following text, followed by: **Goodbye, World!** Press Enter to continue to the next line, and then click the **Set Instruction Font** icon. In the Text entry area, type the following text: **[Click the Finish button.]** **Note:** Instructional text is intended to provide instructions to the end user of the script.	Panel layout editor Toolbar
51	Panel layout editor *or* Unsaved Changes dialog	Save the panel text you have created and exit the panel layout editor. Click **Ok** to save and close the Panel Properties window. The **goodbye** panel is now visible below the Start node of this child graph.	Panel layout editor *or* Unsaved Changes dialog

Step	Perform From	Action	Result
52	Script Author Toolbar	From the Script Author Toolbar, click the **Termination Point Insertion Mode** icon. On the canvas, below the goodbye panel, **click once** to insert a Termination node. **Note:** A Termination node is required for each graph in a script, and exits the graph.	Script Author Toolbar
53	Script Author Toolbar	From the Script Author Toolbar, select the **Default Branch Mode** icon. When over the canvas, the cursor turns into a cross-hairs symbol, indicating that you can begin drawing a line to define flow from one object to another within the canvas.	Script Author Toolbar
54	Script Author canvas	With the cursor displayed as a cross-hairs symbol, click on the **Start** node and drag to **draw an arrow** between the Start node and the **goodbye** panel. Release the mouse. The two objects are connected.	Script Author canvas
55	Script Author canvas Script Author Toolbar	With the cursor displayed as a cross-hairs symbol, again click on the canvas, this time connecting the **goodbye** panel and the Termination node. If you have disabled "Sticky Mode," you must again select the **Default Branch Mode** icon from the toolbar to accomplish this step. You have now completed a syntactically correct graph. Return to the root graph to complete this script.	Script Author canvas Script Author Toolbar
56	Script Author Toolbar	From the Script Author Toolbar, select the **Go up to parent graph** icon. The canvas now displays the root graph.	Script Author Toolbar
57	Script Author Toolbar Script Author canvas	From the Script Author Toolbar, click the **Termination Point Insertion Mode** icon. On the canvas, below the WrapUpGroup object, **click once** to insert a Termination node. **Note:** For the root graph, the Termination node indicates the completion of the script.	Script Author Toolbar Script Author canvas

Step	Perform From	Action	Result
58	Script Author Toolbar	From the Script Author Toolbar, select the **Default Branch Mode** icon.	Script Author Toolbar
		Connect the Start node to the hello panel, the hello panel to the WrapUpGroup object, and the WrapUpGroup object to the Termination node with default branches.	
59	Script Author Toolbar	Click the **Toggle Selection Mode** icon from the Toolbar to turn off the **Default Branch** insertion mode.	Script Author Toolbar
60	Script Author Toolbar	Using click-and-drag techniques, select all the objects on the canvas (Start node, hello panel, Termination node, and Default branches). From the lower Tools menu, select the **Align Selected Objects Vertically** icon.	Script Author Toolbar
		All highlighted items will align vertically, presenting a neat graph on the canvas.	
61	Script Author File Menu *or* Script Author Toolbar *or* keyboard	At the **File** menu, select **Save** to save your work. This command can also be executed by selecting the Save Script (disk) icon in the Toolbar or by entering Ctrl-S from the keyboard (Command-S for the Macintosh OS).	Script Author File Menu *or* Script Author Toolbar *or* keyboard
62	Script Chooser dialog	In the resulting Script Chooser menu, you can save the script to your local file system or to the applications database. Fill in the following:	Script Chooser dialog
		Location: Select the directory in which you wish to save the Script Author file.	
		Note: For location, Oracle Corporation recommends **a child directory of the Script Author directory**, for example (your path may differ):	
		`"[OraHome]\oracle\apps\ies\author\scripts."`	
		File Type: Select the default **Oracle CRM script file (**.SCRIPT, .SCR) option.	
		File name: **helloWorldTestScript.script**	
		Click the **Save** button to save the file in the selected directory.	

Step	Perform From	Action	Result
63	Script Author canvas	Even though you have saved and named your script in your operating system's file system, you still need to assign a name to the script which will be recognized from a database perspective. Right-click on any unpopulated section of the canvas to display two options, Edit and Edit Blob Properties, and select **Edit Blob Properties** to open the [Script Name] Properties dialog. You can also open this dialog by selecting **Script Properties** from the **File** menu. **Note:** If the dialog displayed shows *three* options: **Edit, Edit Blob Name**, and **Edit Blob Properties,** then you have not correctly selected the Script properties dialog, but properties for another object. The global script itself is the Binary Large Object (Blob) for which you want to modify properties. In this case, left-click to close the menu and re-select the **Script Properties** following one of the methods above.	Script Author canvas

Step	Perform From	Action	Result
64	Global [Script Name] Properties dialog	In the resulting global [Script Name] Properties dialog (which should currently be named something like **Untitled1 Properties**), fill in the following: Name: **helloWorldTestScript** Comments: **Graphical hello world test script, built using Script Author visual layout tools.** Script language: As appropriate for your environment.For example, **AMERICAN** for American English. Click the **Ok** button to save the global script properties and return to the canvas. **Note:** For the Name field, fill in a name under which this script will be deployed to the database. We recommend using the same name as that you assigned to the .SCRIPT file in the file system, if practical. Do not assign a file extension to this name. For the Comments field, you may elect to fill this in with information you will find helpful to identify this particular script or script version. This field is for information only and is not processed in any way. For the Script language field, enter the language in which this script is developed, based on the FND_LANGUAGES table of the Oracle 8*i* database. For this example, ensure this field reads **AMERICAN**, for American English.	Global [Script Name] Properties dialog
65	Tools menu or Check Syntax Toolbar icon	Select **Syntax Check** from the **Tools** menu, or click the Check Syntax icon in the Toolbar. Assuming all the syntax in your script is correct, a status line at the bottom of the Author will display the message "Syntax check successful, with no errors." If the syntax check fails, correct any errors and check the syntax of the script again until successful.	Tools menu or Check Syntax Toolbar icon

Step	Perform From	Action	Result
66	File Menu *or* Script Author Toolbar *or* keyboard	At the **File** menu, select **Save** again, to ensure you save the Global Script properties you assigned. **Note:** The method used to save the document is not important. You may elect to click the Save icon in the Script Author Toolbar or enter Ctrl-S from the keyboard (Command-S for the Macintosh OS).	File Menu *or* Script Author Toolbar *or* keyboard
67	Script Author Tools menu *or* Script Author Toolbar	From the Tools menu, select **Deploy Script**, or from the Script Author toolbar, click the **Deploy Script to DataBase** icon. Upon successful deployment, a status message appears on the bottom of the Script Author frame, "Deployment to database successful."	Script deploys, as indicated by status message
68	Forms-based applications	To launch this script in the agent interface, you must perform the following actions: ■ Log into Forms-based Oracle Applications with the appropriate responsibility to launch a script. For example, to launch the script in stand-alone mode, the user must have the Scripting User or Scripting Agent responsibility. ■ From the Navigator, select **Scripting Demo Form** and click **Open**. The Script Chooser window appears. This window is entitled Oracle Scripting. From the Script Chooser, select the appropriate script name and language. For example, select **helloWorldTestScript/AMERICAN**. ■ Launch the script by clicking **Start Scripting**.	Forms-based applications

Step	Perform From	Action	Result

69	Forms-based applications	To launch this script as a survey in a Web browser, you must perform the following actions:
		
		 ■ Log into the Survey Administration console from Oracle HTML-based applications with a user that has the Survey Administrator responsibility
		
		 ■ Define header section, footer section (optional), error page and final page survey resources.
		
		 For example, you can create references to seeded test survey resources in the Survey Resources tab. These are named IESSVYTESTHEADER.JSP, IESSVYTESTERROR.JSP and IESSVYTESTTHANKU.JSP.
		
		 If using the footer section, provide the test header resource name for this field.
		
		 ■ Create a survey campaign, cycle, and standard (non-list-based) deployment. The survey campaign must identify a deployed 11.5.6 or later test script.
		
		 ■ Activate the deployment. When the Survey URL field is populated with a hyperlink, note the deployment ID (dID) by placing the cursor over the hyperlink and viewing the full survey URL in the Web browser status field. The format of the survey URL should be as follows:
		
		 `http://<server>.<domain>:`
		 `<port>/OA_HTML/<hosted or`
		 `stand-alone survey jsp>?`
		 `dID=xx`
		
		 ■ Enter the survey URL in a Web browser to launch the script in the Scripting Engine Web interface.
		
		 ■ Execute the script by participating in the survey to completion.
			Forms-based applications

See Also

11.1.2 Creating a Hello World Wizard Test Script

Step	Perform From	Action	Result
1	Oracle Applications HTML Login	Log into Oracle HTML-based applications using a User ID with the Scripting Administrator responsibility.	The Scripting Administration console appears.
2	Scripting Administration console	From the Home tab, click **Launch Script Author**.	A new browser window opens, Oracle JInitiator launches, and the Script Author Java applet appears in a separate window.
3	Script Author File Menu *or* Script Author Tools Menu *or* keyboard *or* Script Author Toolbar	Start the Script Wizard using one of the following methods: From the File menu: **1** Select **New**. **2** In the resulting New Script window, click **Wizard Script**. From the Tools menu: **1** Select **Wizard**. **2** In the Choose Create or Edit window, select **Create Wizard Script** and click **Next**. From the keyboard: **1** Click **ALT + T** to select the Tools menu. **2** Type **W** to select the Wizard option. **3** Using the **Tab** key to navigate through wizard page options and the **space bar** to select the active option, in the Choose Create or Edit window, select **Create Wizard Script** and then select **Next**. From the Script Author toolbar: **1** Click **Script Wizard**. **2** In the Choose Create or Edit window, select **Create Wizard Script** and click **Next**.	The Script Wizard launches. The Define Script Properties window appears

Step	Perform From	Action	Result
3	Script Wizard - Define Script Properties window	Provide the following global script properties to this wizard script. Global script properties are properties that apply to the entire script, rather than a particular panel. Script Name: **helloWorldWizardScript** Description: **Hello world test script, built using Script Author Script Wizard.** Language: **AMERICAN** When complete, click **Next**.	The wizard script is assigned global script properties.
4	Script Wizard - Define Panel Information for Panel Default Panel 1	Provide panel information for the first panel. Panel Name: **hello** Panel Text: Overwrite the default "Enter text here" entry and type **Hello, World!** From this point on, only panel text values will be provided. In each case, overwrite the default text. Panel Text Style: **Spoken Text** Text Alignment: **Left** Question Alignment: **Left** Exit Panel Sequence: **Go to the next panel in sequence** When complete, click **Next**.	Panel, question, and answer choice properties generally appear at the top of each wizard page to assist you in navigating through the creation or editing process. For this new wizard script, until you click **Next**, none of these properties are defined. A temporary name (Default Panel 1) is automatically supplied by the wizard.

Step	Perform From	Action	Result
5	Script Wizard - Question Manager for Panel hello	After defining panel properties, if relevant, you must define question properties for each question in the panel. This is performed from the Question Manager	Upon selecting Panel Manager, the Panel Manager window appears.
		Since the only function for panel hello is to display panel text and a Continue button, you do not need to define any questions for this panel. Unlike when using Script Author's visual layout tools, the wizard will automatically create the Continue button to route the runtime user to the destination specified using the Exit Panel Sequence property. Based on the panel properties established in the previous step, this script at runtime will display the next panel you define.	
		To save this one-panel script and exit, click **Save**, and proceed to step 17. To continue learning how to create panels, define answer types and questions, and determine panel branching, from the set of navigation buttons to the right of the Question Manager window, select **Panel Manager** and continue this procedure.	
6	Script Wizard - Panel Manager for Script helloWorldWizard Script	Once you have one or more panels defined, from the Panel Manager window you can see a list of existing panels and their status. From this window you can also create new panels, edit the properties of existing panels, copy existing panels, and delete existing panels. If more than one panel is listed here, you can also move panels higher and lower in sequence.	Define Panel Information window appears for new panel.
		From the set of navigation buttons to the right of the Panel Manager window, select **Create**.	
7	Script Wizard - Define Panel Information for Panel Panel2	Provide panel information for the first panel.	The Script Wizard progresses to the Question Manager for panel goodbye.
		Panel Name: **goodbye**	
		Panel Text: **Goodbye, World!**	
		Panel Text Style: **Spoken Text**	
		Text Alignment: **Left**	
		Question Alignment: **Left**	
		Exit Panel Sequence: **End Script**	
		When complete, click **Next**.	

Step	Perform From	Action	Result
8	Script Wizard - Question Manager for Panel goodbye	After defining panel properties, if relevant, you must define question properties for each question in the panel. This is performed from the Question Manager. For this panel you will create a question. From the set of navigation buttons to the right of the Question Manager window, select **Create**.	The Script Wizard progresses to the define question main properties window.
9	Script Wizard - Define Question Main Properties	Questions have three main properties. These are similar to properties in a graphical script created using the Answer Entry dialog. The *question name* identifies this question. This is identical to the name property in a graphical script. The *question label* identifies this script to reporting features accessible through Oracle Discoverer. This is equivalent to the Label for reporting property in a graphical script. The *type* identifies the answer control displayed at runtime. Wizard scripts can contain a text box (text), text area, radio button, drop-down list, checkbox group, multi-select list, or a password field. This is equivalent to the UI type property in a graphical script. Question Name: **measureSuccess** Question Label: **script successful?** Type: **Dropdown** When complete, click **Next**.	The Script Wizard progresses to the answer manager window.
10	Script Wizard - Answer Manager for measureSuccess Question	The answer manager provides wizard script developers the opportunity to provide answer choices to a specific question. (Answer choices are referred to as lookup values when using graphical scripts.) In this case, you will create three answer choices for the measureSuccess question you created. From the set of navigation buttons to the right of the Answer Manager window, select **Create**.	The Script Wizard progresses to the define answer choice window.

Step	Perform From	Action	Result
11	Script Wizard - Define Answer Choice for Untitled	The question types supported by wizard scripts include those questions that can accept only a single answer (text box, text area, password field, drop-down list or button), and those that can accept more than one answer choice (multi-select question types, including checkbox group and multi-select list question types).	The Script Wizard refreshes the answer manager, showing the single answer choice currently defined for this question.
		For text question types (text box, text area or password field answer choices), as well as checkbox group and multi-select list choices, a null answer (no input provided by end user at runtime) is also supported. Radio buttons, drop-down lists, multi-select lists, and checkbox groups must have answer choices provided by the script developer in order to display answer choices at runtime.	
		Answer choices have two properties: the *answer value* (which is passed to the database) and the *answer label* (the answer choice viewed by the script end user). This is equivalent to lookup type value and display value fields in the data dictionary of the graphical Script Author.	
		For this question, which will ask the script end user to determine the level of success for this test script, provide three answer choices.	
		For the first set of answer choices, enter the following:	
		Answer Value: Overwrite the default "Untitled" entry and type **Unsuccessful**. From this point on, only an answer value will be provided. In each case, overwrite the default text.	
		Answer Label: Overwrite the default "Untitled" entry and type **Unsuccessful**. From this point on, only an answer value will be provided. In each case, overwrite the default text.	
		You can also set a default answer choice selection. This will default in the appropriate answer control at runtime, although defaults can be changed at will.	
		For the question "Should this be the default answer for this question," select **No**.	
		This version of the script will use the panel level defaults. Select **Go to next default panel.**	
		When complete, click **Next**.	
12	Script Wizard - Answer Manager for measureSuccess Question	You must now configure another answer choice to display at runtime for this question.	The Script Wizard progresses to the define answer choice window.
		From the set of navigation buttons to the right of the Answer Manager window, select **Create**.	

Step	Perform From	Action	Result
13	Script Wizard - Define Answer Choice for Untitled	For the second answer choice, enter the following: Answer Value: **Moderately successful** Answer Label: **Moderately successful** For the question "Should this be the default answer for this question," select **Yes**. This version of the script will use the panel level defaults. Select **Go to next default panel.** When complete, click **Next**.	The Script Wizard refreshes the answer manager, showing the two answer choices currently defined for this question.
14	Script Wizard - Answer Manager for measureSuccess Question	You must now configure the third and final answer choice to display at runtime for this question. From the set of navigation buttons to the right of the Answer Manager window, select **Create**.	The Script Wizard progresses to the define answer choice window.
15	Script Wizard - Define Answer Choice for Untitled	For the third answer choice, enter the following: Answer Value: **Very successful** Answer Label: **Very successful** For the question "Should this be the default answer for this question," select **No**. If you choose Yes, you will be reminded that this will overwrite the selection of Moderately successful as the default answer choice for this question. This version of the script will use the panel level defaults. Select **Go to next default panel.** When complete, click **Next**.	The Script Wizard refreshes the answer manager, showing all three answer choices defined for this question.
16	Script Wizard - Answer Manager for measureSuccess Question	You have now finished defining two panels, the second containing a single question with three answer choices. At this point, save your work. Click **Save**.	The Script Wizard progresses to the save script window.
17	Script Wizard - Save Script helloWorldWizard Script	From the save script window, for the prompt Do you want to save the script with a different name, select **No**. When complete, click **Next**.	The Script Wizard progresses to the continue with script save window.

Step	Perform From	Action	Result
18	Script Wizard - Continue With Script Save	You can choose one of the following save options: ■ To save the current wizard script and continue editing it, select **Save and continue editing**. ■ To save the wizard script in the database without deploying it, close the wizard tool, and view the Script Author canvas, select **Save and exit**. ■ To save the wizard script, deploy it to the database, close the wizard tool, and view the Script Author canvas, select **Save, deploy and exit**. Select the last option, **Save, deploy and exit**. When complete, click **Next**.	The Script Wizard progresses to the create graphical script copy window.
19	Script Wizard - Create Graphical Script Copy	From this window, by clicking Yes you can view the wizard script you have completed, using Script Author's visual tools. However, the script displayed is a *copy* of the wizard script. Any changes made in the resulting graphical script will *not* be reflected in the original wizard script, which will still be accessible to open, edit, view and deploy from the Script Wizard. Select **Yes** and click **Next**.	The Script Wizard progresses to the script is complete window.
20	Script Wizard - Script is Complete for helloWorldWizard Script	Click **Finish** to exit the wizard and return to the Script Author canvas. The wizard script then deploys to the database per your request, and the flow you created appears on the Script Author canvas. Note that this graphical copy is entitled "Copy of helloWorldWizardScript." At this point, you have the option of saving this graphical copy of the wizard script to the database or your local file system. Since you can always open the wizard script and graph a copy to use with Script Author's graphical tools, select **File > Close** to close the graphical Script Author without saving. In the resulting Select an Option window, click **No** to exit without saving.	Copy of helloWizardScript appears on the Script Author canvas. After closing the script, the graphical Script Author user interface appears, with no script open.

Step	Perform From	Action	Result
21	Script Author File Menu *or* Script Author Tools Menu *or* keyboard *or* Script Author Toolbar	Start the Script Wizard again, using one of the following methods:	The Script Wizard launches.
		From the File menu:	
		▪ Select **Open**.	
		▪ In the resulting Open Script window, click **Wizard script**.	
		From the Tools menu:	
		▪ Select **Wizard**.	
		▪ In the Choose Create or Edit window, select **Edit Wizard Script** and click **Next**.	
		From the keyboard:	
		▪ Click **ALT + T** to select the Tools menu.	
		▪ Type **W** to select the Wizard option.	
		▪ Using the **Tab** key to navigate through wizard page options and the **space bar** to select the active option, select **Edit Wizard Script** and click **Next**.	
		From the Script Author toolbar:	
		▪ Click **Script Wizard**.	
		▪ In the Choose Create or Edit window, select **Edit Wizard Script** and click **Next**.	

Step	Perform From	Action	Result
22	Script Wizard - Script Manager	Scroll down through the list of existing wizard scripts if necessary to locate the appropriate script. Scripts are listed by global script name and include last revised date and script language. For any wizard script in the list, you can perform the following: ■ To modify any panel, question, or answer choice details, select the appropriate script from the list, and from the wizard function buttons to the right of the wizard scripts table, click **Edit**. ■ To retain the original wizard script but to make a copy of the script to modify, select the appropriate script from the list and click **Copy**. ■ To delete the wizard script from the database, select the appropriate script from the list and click **Delete**. *Caution:* This will permanently delete the selected script. Unless you have made a copy of this wizard script, *you will not be able to modify, edit, or execute this script after deletion.* ■ To create a graphical copy of this script for viewing or modification with the Script Author's visual editing tools, select the appropriate script from the list and click **Graph**. This will close the Script Wizard. ■ To deploy a saved wizard script to the database, select the appropriate script from the list and click **Deploy**.	The selected function will execute.

11.1.3 Using the Regression Test Script

Oracle Corporation provides a regression test script for the purpose of testing Oracle Scripting implementations and functionality. This script contains references to custom Java. Custom Java classes, compiled into JAR files, are included in the package and must be managed accordingly.

Prerequisites

■ To create or deploy scripts using the Script Author Java applet, you must use the Scripting Administration console. Thus, you must have access to an Oracle Applications account with the Scripting Administrator responsibility.

■ All implementation tasks must be accomplished.

Steps

To use the regression test script, perform the following:

1. Download and unzip the regression test script files.

   The regression test script is accessible from Oracle*MetaLink* and Oracle iSupport as note 124522.1. To locate, enter the document ID in the Search field. From the support document entitled Oracle Scripting Regression Testing Scripts, download the associated ZIP archive and unzip this package. Included are a Script Author file, a compiled Java class file, and a packaged Java archive.

2. Using the Script Author Java applet, open the Script Author file. Note the global script name (**File > Script Properties**) and change this if necessary to prevent overwriting any existing script with the same name in your environment.

3. Deploy the regression test script to the database from the Script Author Java applet (**Tools** > **Deploy Script**).

   You can view deployed scripts from the Scripting Administration console (Administration tab > Deployed Scripts subtab).

4. Upload the custom Java archive file to the Oracle Applications database from the Scripting Administration console (Administration tab > Jar Listings subtab).

5. Map the script to the custom Java archive from the Scripting Administration console (Administration tab > Jar Mapping subtab).

   The regression test script is now appropriately deployed, and can be tested using any Scripting Engine interface. To test using the Web interface, survey campaign requirements must be established using this script.

See Also

- Section 11.1.1, "Creating a Hello World Graphical Test Script"

11.2 Survey Administration Console Tasks

To verify your implementation for scripts to be executed in the Scripting Engine Web interface, you must establish survey campaign requirements. Steps to perform for implementations that require scripts to be executed in an Oracle Applications 11*i*-compatible Web browser are described below.

Prerequisites

- To use the Survey Administration console, you must have access to an Oracle Applications account with the Survey Administrator responsibility.

- All implementation tasks must be accomplished. For enterprises using Oracle Applications on the UNIX platform, you must have a display X Server set up.

- To access the Invitations tab in the Survey Administration console, you must be a member of a fulfillment group and have the JTF_FM_ADMIN role.

- To establish invitations and reminders for targeted (list-based) survey operations, functioning invitation and reminder master documents and their associated queries must exist for Oracle One-to-One Fulfillment.

- Invitations and master documents must be associated with Oracle One-to-One Fulfillment templates, as appropriate.

- To establish list information in a deployment, you must have a corresponding, accurate list of 150 or fewer records created in Oracle Marketing.

Steps

1. Identify existing survey resources or create survey resources to use with your script in the Web interface.

   You can use seeded test resources to test your implementation, or you can customize your own resources based on these seeded resources. These JSP files (iessvytestheader.jsp, iessvytestthanku.jsp, iessvytesterror.jsp and iessvymenubasederror.jsp) are located in $OA_HTML in your environment.

2. Using the Survey Administration console, define survey resources (Survey Campaign tab > Survey Resources subtab). At minimum, you must define a header, error, and final page resource.

3. Using the Survey Administration console, create a survey campaign (Survey Campaign tab > Survey Campaign subtab). Identify the script you want to use as your survey questionnaire or Web script (for example, the Hello World or regression test script). For survey resources, reference the resources identified in steps 1 and 2 above.

4. Using the Survey Administration console, create a cycle as a child to the survey campaign (Survey Campaign tab > Cycle subtab).

5. Using the Survey Administration console, create a deployment as a child to the cycle (Survey Campaign tab > Deployment subtab). For deploy date, use a date prior to or equal to SYSDATE.

   Only enter list and reminder information if you have a corresponding, accurate list of 150 or fewer records created in Oracle Marketing and you have functioning invitation and reminder master documents created for Oracle One-to-One Fulfillment.

6. Activate your deployment by clicking **Deploy** (Survey Campaign tab > Deployment subtab > Deployment Detail page). Note the survey URL.

7. Using an Oracle Applications 11*i*-compatible Web browser, access the survey URL, and execute your script in the Scripting Agent Web interface. Test all paths in the script to ensure the implementation is successful.

See Also

- Section 11.1, "Script Author Tasks"

- Section 11.3, "Scripting Engine Tasks"

- Section 11.4, "Testing Display Server for UNIX Environments"

11.3 Scripting Engine Tasks

To verify that the Scripting Engine will execute scripts, you can execute servlet tests designed to ensure scripts will execute in the Apache mid-tier architecture.

Then, test a simple script with no custom Java (such as Hello World), and a script containing custom Java (such as regression test), using the Scripting Engine.

> **Note:** Oracle Corporation recommends that all implementations test first using the agent interface. Implementations using the Web interface must then also perform the appropriate verification steps.

Each Oracle Scripting implementation includes at least one custom script. Oracle Corporation recommends, at minimum, that the first script used at an enterprise is developed by Oracle Corporation consultants or partners. In addition to testing the recommended Hello World and regression test scripts, it is recommended that you completely test your custom script prior to putting that script into production.

If custom Java is required, you must create and test your custom methods, compile and package your Java archive appropriately, and deploy the Java archive to your environment. To test these methods from Oracle Scripting, your script must contain valid references to the code, and must also be deployed to your environment.

This section includes the following topics:

- Section 11.3.1, "Apache Web Server Servlet Verification URLs"

- Section 11.3.2, "Scripting Engine Agent Interface Verification"

- Section 11.3.3, "Scripting Engine Web Interface Verification Tasks"

See Also

-

-

-

11.3.1 Apache Web Server Servlet Verification URLs

Script Servlet Verification

In order to execute Oracle Scripting in the agent application using the Apache Mid-Tier architecture, the Scripting Servlet must be operational.

The following URL can be used to verify that the Scripting Servlet is properly configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_
zone>/oracle.apps.ies.client.application.ScriptingServlet
```

A message of **up and running...** indicates success. This message *must* be obtained to assure that scripts can be executed appropriately.

Apache JServ Servlet Verification

The following URL can be used to verify that the Apache JServ servlet engine is configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>/IsItWorking
```

A message of **Yes, It's working! Congratulations, Apache JServ/<version> is working!** indicates success. Note, however, that this servlet may have been removed by system administration and is not necessarily an indication of whether the Scripting Servlet will function.

AOL Test Servlet Verification

The following URL can be used to verify that the AOL test servlet is configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>/
oracle.apps.fnd.test.HelloWorldServlet
```

A message of **Hello World! From AOL oracle.apps.fnd.test.HelloWorldServlet** indicates success.

Note, however, that this servlet may have been removed by system administration and is not necessarily an indication of whether the Scripting Servlet will function.

See Also

- Section 11.3.2, "Scripting Engine Agent Interface Verification"
- Section 11.3.3, "Scripting Engine Web Interface Verification Tasks"

11.3.2 Scripting Engine Agent Interface Verification

Execute a script without custom Java using the Scripting Engine agent interface. For example, execute the Hello World test script to test your implementation. Then, execute a script that references custom Java using the Scripting Engine agent interface. For example, execute the regression test script to test your implementation.

> **Note:** Since testing for *all* customized scripts is recommended at the lowest HTML interpretation engine available, steps for implementing the Scripting Engine for the agent interface are recommended to be performed for all implementations.

Prerequisites

- To verify scripts using the Scripting Engine agent interface, you must have access to an Oracle Applications account with the Scripting User or Scripting Agent responsibility. This is recommended for all implementations.
- All implementation tasks must be accomplished.

Steps

1. Using the Scripting Engine agent interface, log into Oracle Forms-based applications as a user with the Scripting User or Scripting Agent responsibility.
2. From the Script Chooser, select and execute the regression test script in the Scripting Engine agent interface. Test all paths in the script to ensure the implementation is successful.

See Also

- Section 11.3.1, "Apache Web Server Servlet Verification URLs"
- Section 11.3.3, "Scripting Engine Web Interface Verification Tasks"

11.3.3 Scripting Engine Web Interface Verification Tasks

Once you can execute a script in the Web interface (based on an existing survey campaign deployment), you have verified your implementation. If your attempt to do so fails, you should verify that a guest user has been set up.

Tasks

You can perform the following tasks:

- Section 11.3.3.1, "Executing a Script in the Web Interface"

- Section 11.3.3.2, "Guest User Verification Steps"

See Also

- Section 11.3.1, "Apache Web Server Servlet Verification URLs"

- Section 11.3.2, "Scripting Engine Agent Interface Verification"

11.3.3.1 Executing a Script in the Web Interface

Prerequisites

- To verify scripts in the Scripting Engine Web interface, survey campaign requirements must be established by a survey administrator.

- Survey resources referenced by the appropriate survey campaign must exist in $OA_HTML.

- A guest user account must already be established by an Oracle CRM applications system administrator.

- Advanced JTF properties must be set up by an Oracle CRM applications system administrator.

- You must have access to a survey URL for an active survey campaign deployment.

- For targeted survey deployments, you may require a survey URL with a unique respondent ID.

Steps

1. Using an Oracle Applications 11*i*-compatible Web browser, access the survey URL for the survey deployment you wish to test.

2. Execute your script in the Scripting Agent Web interface. Test all paths in the script to ensure the implementation is successful.

Performing any verification task other than testing the Guest User requires you to also perform all Survey implementation steps. These are in addition to Scripting Engine implementation tasks for all implementations and those required for the Web interface, since testing a script in the Web interface requires a survey campaign to be defined and deployed.

> **Note:** Since testing for *all* customized scripts is recommended at the lowest HTML interpretation engine available, steps for implementing the Scripting Engine for the agent interface are *recommended to be performed for all implementations*.

See Also
- Section 11.3.3.2, "Guest User Verification Steps"

11.3.3.2 Guest User Verification Steps

For Survey operations, ensure you have a guest user defined. If you did not establish the guest user JTF property, it is recommended that you verify the user name and password for the guest user to ensure it is functioning.

Steps

1. Check the guest user name and password JTF properties.

   For information on how to view the guest user settings, see Section 4.5.3, "Administering the Guest User".

2. Log into Oracle HTML-based applications.

3. Use the guest user name and guest password key value pairs.

4. If the following error results in the HTML administration console, the guest user is functioning properly:

   This page is missing or not a valid HTML page.

   This message indicates that there is an Oracle Applications user name that has no responsibilities and no default application associated with it. This is appropriate for the guest user.

See Also

- Section 11.3.3.1, "Executing a Script in the Web Interface"

11.4 Testing Display Server for UNIX Environments

Implementations of Oracle Scripting using UNIX servers require a display server to be set up to process graphics in order to view reports generated under the Reports tab of the Scripting Administration console.

> **Note:** Interaction Center Family Pack P and later implementations no longer require a functional display server for a survey respondent to participate in a survey.

> **Note:** Interaction Center Family Pack Q and later implementations no longer require a functional display survey for the Survey Administration console.

You can verify that the display server is appropriately configured by accessing a seeded JSP test page and viewing the resulting message.

The following URL can be used to verify that the display server is properly configured and operational:

```
http://<servername>.<domain>:<Apache Web server port>/OA_HTML/
iestestdisplayserver.jsp
```

Reference

For more information on the display server, see Section 4.6.5, "Setting Display Server for UNIX Environments".

See Also

- Section 11.1, "Script Author Tasks"
- Section 11.2, "Survey Administration Console Tasks"
- Section 11.3, "Scripting Engine Tasks"

A

Upgrading from Releases Prior to 11.5.6

Oracle Scripting release 11.5.6 was a major functional release including significant improvements and changes. Some of the new features or aspects of Oracle Scripting may impact production environments using previous releases of Oracle Scripting 11*i* and custom scripts. This section is provided to guide you through the changes from previous releases to release 11.5.6 or later with minimum impact.

Upgrading Oracle Scripting as of release 11.5.6 or later can be accomplished using the latest IES rolled-up patch (RUP). Alerts indicating the latest Oracle Scripting RUP are available on Oracle*MetaLink* or Oracle iSupport. For latest release information, alerts, and to obtain patches, always consult Oracle*MetaLink* or Oracle iSupport first. If you continue to have questions or concerns, contact your Oracle Support representative.

The steps required to migrate any Oracle Scripting 11*i* environment from a pre-11.5.6 release to release 11.5.6 include:

Section A.1, "Implementation Upgrade Tasks"

Section A.2, "Installation of Script Author Post-11.5.6 Stand-Alone Application"

Section A.3, "Converting, Re-Deploying, and Testing Scripts"

A.1 Implementation Upgrade Tasks

Implementation upgrade tasks include:

Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"

Section A.1.2, "Modifying System Profile Settings"

Section A.1.3, "Referencing Custom Java"

Section A.1.4, "Modifying Custom Java"

See Also

A.1.1 Upgrading to Oracle Applications 11.5.6 or Later

In order to use Oracle Scripting 11.5.6 or later, you must upgrade Oracle Applications to release 11.5.6 or later. This is accomplished either by executing a Rapid Install for 11.5.6 or later or by upgrading with the appropriate Interaction Center Family Pack for 11.5.6 or later.

> **Note:** Oracle Applications must generally be at the same patch level (for example, release 11.5.6) to function appropriately. Upgrading to release 11.5.6 can be achieved using the appropriate series of patches that meets your specific application requirements, or using the appropriate Oracle Platinum version to support a release 11.5.6 upgrade. Appropriate patching is prerequisite of a successful Oracle Scripting implementation and is not addressed herein. For more information, see Oracle*MetaLink* or Oracle iSupport, or contact your Oracle Support representative.

See Also

A.1.2 Modifying System Profile Settings

To execute Oracle Scripting in the Scripting Engine agent interface in release 11.5.6 or later, you must change the IES : Architecture Type profile upon upgrading to release 11.5.6 or later. The profile options for this system profile setting have changed, as has the recommended architecture.

Guidelines

1. For upgrading to the Apache Mid-Tier architecture, select the value **Apache Mid Tier/Servlet Architecture**.

2. For using Oracle Scripting 11.5.6 or later in the Caching Architecture, select the value **Caching Architecture** (replacing **Two Tier Mode**).

   Note that using Oracle Scripting in the Caching Architecture will *only be supported* for customers previously using this architecture in a production environment, and only for a limited amount of time, as designated by Oracle Support Services.

3. The previous value, **Three Tier Mode**, has been obsoleted. New implementations will not include this lookup value. *Do not use this setting under any circumstances.*

See Also

Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"

Section A.1.3, "Referencing Custom Java"

A.1.3 Referencing Custom Java

There are two types of custom Java used in association with Oracle Scripting:

Custom Commands

Custom Java Beans

See Also

Section A.1.1, "Upgrading to Oracle Applications 11.5.6 or Later"

Section A.1.2, "Modifying System Profile Settings"

A.1.3.1 Custom Commands

The most common type is custom Java methods written using Oracle Scripting APIs to provide additional functionality to a script using a Command (for example, branch actions, embedded values in panel text, statements defining parameters for a conditional or indeterminate branch, and so forth).

These Java commands always run in the Java Virtual Machine (JVM) designated by architecture type. Thus:

- If using the Caching Architecture, the JVM is provided by JInitiator on the client workstation. In this case, your code must be compiled using JDK 1.1.8.

  If using the Caching Architecture, see Guidelines above for caveats.

- If using the Apache Mid-Tier architecture, the JVM is provided by the Apache JServ (which can be JDK 1.1.8, 1.2, or 1.3). The compiling restrictions are based on the JDK used by your version of Apache.

This section includes the following topics:

- Section A.1.3.1.1, "Referencing Custom Commands in the Caching Architecture"
- Section A.1.3.1.2, "Referencing Custom Commands in the Apache Mid-Tier Architecture"

See Also
Section A.1.3.2, "Custom Java Beans"

A.1.3.1.1 Referencing Custom Commands in the Caching Architecture Using the Caching Architecture of Oracle Scripting in releases 11.5.4 and 11.5.5, the APPSWEB.CFG file was required to be modified:

    a. To reference custom Java in support of custom commands,

    b. To obtain JDBC classes, and

    c. To obtain AOL/J classes.

If you are upgrading an Oracle Scripting implementation using the Caching Architecture, you must still modify the APPSWEB.CFG file to reference custom Java code (as well as to download JDBC classes to the client workstation). *Modification of the APPSWEB.CFG file to obtain AOL/J classes is no longer required. For details, see* Appendix D of the *Oracle Scripting Implementation Guide* release 11.5.4 or 11.5.5, as appropriate.

> **Note:** If using the Caching Architecture, see Guidelines in this section for caveats.

See Also
Section A.1.3.1.2, "Referencing Custom Commands in the Apache Mid-Tier Architecture"

A.1.3.1.2 Referencing Custom Commands in the Apache Mid-Tier Architecture Using the new Apache Mid-Tier Architecture, the class path of the JSERV.PROPERTIES file on the Apache Web server must be modified to reference custom Java. The requirement to upload custom Java classes to the applications server has not changed.

See Also

Section A.1.3.1.1, "Referencing Custom Commands in the Caching Architecture"

Section A.1.5, "Adding Custom Java Class Path to JSERV.PROPERTIES File"

A.1.3.2 Custom Java Beans

The second type of custom Java to support Scripting is custom java code compiled in a UI Java Bean that is used to replace an entire panel in the runtime session. Custom Java beans are executed by Oracle JInitiator on the agent client. Thus, this code must be compiled using the version of JDK that is compatible with Oracle JInitiator on the agent client (the same version or lower). This may include JDK 1.3 or 1.1.8. For specific JDK or other software certification information, reference the Certify section of Oracle*MetaLink* or Oracle iSupport.

> **Note:**   While substitution of *an entire panel* with a Java Bean is still supported, substituting an *answer* (or panel *node*) with a Java bean is no longer supported as of release 11.5.6 and later.

See Also

Section A.1.3.1, "Custom Commands"

A.1.4 Modifying Custom Java

Moving from the Caching Architecture to the Apache mid-tier architecture changes the location of error messages. Since Oracle JInitiator is no longer providing the JVM on the client, some messages that were available in the Java console are now written to the JServ error logs.

In any custom Java code where you included print line statements (generally used for debugging or verifying that a certain step in a method was executed), you will need to change your code to continue viewing messages.

Change any instance of "System.out.println" to "System.err.println" to allow you to view the messages in JServ error logs.

Problem:

Prior to upgrading to the Apache mid-tier architecture, certain debug statements or messages appeared in the Java console that are no longer visible.

Cause:

The JVM executing code is located on the Apache web server whereas before the messages were "printed" to the client's Java console.

Solution:

To view debugging messages in the Apache Mid-Tier architecture, change any Java code using "System.out.println" to "System.err.println". This will result in the messages being written to the jserv error logs.

A.1.5 Adding Custom Java Class Path to JSERV.PROPERTIES File

This step is only relevant for users of the Apache Mid-Tier architecture between releases 11.5.6 and 11.5.8 base release (prior to Interaction Center Family Pack P). This step is not required for users upgrading to Oracle Scripting release 11.5.8 patched to Interaction Center Family Pack P or later.

Any time a Script Author script references custom Java *(regardless of whether that script will be executed in the agent interface or in a Web browser as a survey)*, you must have an appropriately configured JAR file available to the class loader, containing compiled Java classes referenced in the script. In the Apache Mid-Tier architecture for Oracle Scripting, prior to Interaction Center Family Pack P, that custom Java needs to be stored on the applications server, and referenced in the Apache JServ classpath in order for the Scripting Engine to be able to call those classes at runtime. Following are steps to add a custom JAR file to the Apache Web server's JServ configuration file, known as the JSERV.PROPERTIES file.

> **Note:** There is more than one way to specify a class path for Apache. *If the method described below is not supported by your enterprise implementation*, consult with your Apache Web server administrator to determine the appropriate method for your environment. This may include identifying the classpath in a control file or another configuration.

Login and Responsibility

To access the JSERV.PROPERTIES file you must have physical or Telnet access to the Web server, and have applmgr or sysadmin privileges. Use the appropriate login for these privileges in your environment.

Prerequisites

- A custom Java method must be written and tested.

- All custom Java methods must be compiled into class files. Oracle Corporation recommends compiling custom Java code for use with Oracle Scripting with JDK 1.1.8. For more information, see Section 3.2.2.4.1, "Compiling Custom Code Using JDK 1.1.8".

- All class files must be packaged into a Java archive format appropriately.

- All JAR files must be uploaded to the applications server in a specified directory. Any directory accessible to the Web server is acceptable. For ease of maintainability, Oracle Corporation recommends creating a directory named ies_custom on the JAVA_TOP (Oracle file system on the applications server) specifically for storing custom JAR files for Oracle Scripting.

Steps

1. Connect using Telnet to the enterprise system.

2. Change to the directory where your Oracle HTTP Server is installed. Refer to the Appendix of your Oracle Applications Installation manual if you are not sure where to find this.

3. Change to the directory in which the JSERV.PROPERTIES file resides.

   > **Note:** The actual name of this file may include an underscore and the Apache JServ port (for example, JSERV.PROPERTIES_9404). For the purposes of this document, the Apache Web server configuration file will be referred to generically as the JSERV.PROPERTIES file. If you have any difficulties locating the file, consult with your Apache Web server administrator.

   The location of this file is environmentally dependent. It *may* be in one of the following paths:

   <ORAHTTP_TOP>/Jserv/etc/jserv.properties

   <ORAHTTP_TOP>/apache39/conf

   The ORAHTTP_TOP variable may not be set for your environment, in which case the path will be fully qualified to identify the entire physical path in the file system. If your configuration is different, consult with your Apache Web administrator to locate this directory.

4. Load the JSERV.PROPERTIES file into a text editor.

5. Look in the JSERV.PROPERTIES file for several lines beginning with **wrapper.classpath=**.

   This is the section in the file that sets the CLASSPATH used by the web server.

6. If this configuration file contains a wrapper.classpath entry, you can modify it to add the classpath of any custom JAR file your script references in this line. For example:

   ```
   wrapper.classpath=/d01/testora/iAS/Apache/xsu/xsu12.jar
   ```

   > **Note:** If there is no listing, consult with your Apache Web server administrator to determine the appropriate location to specify a classpath for your Apache environment in support of Oracle Applications.

7. Save the JSERV.PROPERTIES file, and exit the text editor.

   Your web server is now configured to load custom Java classes for Oracle Scripting.

Guidelines

- Back up your JSERV.PROPERTIES file prior to customization. After configuring and testing, also back up your customized JSERV.PROPERTIES file in a separate location.

  > **Note:** When you apply a patch, any customizations you have made to the JSERV.PROPERTIES file *may be lost*. In this case you will need to restore customizations to this file after patching.

- This process will have to be repeated any time you reference new classes in Oracle Scripting, regardless of whether the script is executed by agents in the Java UI or by survey respondents in a Web browser.

> **Note:** If you write new Java methods supporting scripts already in production, you may wish to combine the compiled code in an existing JAR file (and re-uploading) so you need not update the JSERV.PROPERTIES file. Oracle Corporation recommends backing up custom JAR files prior to changes to any system in production, and storing the backups in a separate location.

A.2 Installation of Script Author Post-11.5.6 Stand-Alone Application

Implementations of Oracle Scripting between release 11.5.6 and baseline release 11.5.8 *prior to Interaction Center Family Pack P* must install the Script Author stand-alone application to create, modify, or deploy scripts to the applications database.

As of release 11.5.8 (Interaction Center Family Pack P and later), the Script Author is a Java applet launched from the Scripting Administration console. The stand-alone application is longer available, and users are encouraged to use the Java applet to develop scripts. Such users will enjoy new capabilities available from the Scripting Administration console, but only for scripts deployed from the applet. For example, custom Java archives in support of scripts can be loaded from the Scripting Administration console and can be mapped from this console to apply to any script or scripts deployed from the console only.

Script Compatibility

Any existing (pre-11.5.6) script *must be converted* to an 11.5.6 (or subsequent) version of the Script Author prior to execution in the Scripting Engine provided by Oracle Applications 11.5.6 or later.

The method for obtaining and implementing the Script Author stand-alone application for Oracle Scripting releases 11.5.6 through baseline 11.5.8 prior to Interaction Center Family Pack P is described below.

This section includes the following topics:

Section A.2.1, "Locating Script Author Installation Package"

Section A.2.2, "Downloading Script Author Installation Package"

Section A.2.3, "Preparing for Script Author Installation"

Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

See Also

A.2.1 Locating Script Author Installation Package

Installation of the Script Author is accomplished using the Oracle Universal Installer. As of release 11.5.6, this is obtained as a ZIP file from the APPL_TOP (Oracle Applications file system), specifically in the product top for Oracle Scripting (product code IES).

This is in contrast to earlier releases of Oracle Applications 11*i*, in which the only way to access the Script Author installation package was on the 11*i* Call Center Applications Install CD, or as a ZIP file downloaded from Oracle*MetaLink*.

See Also

A.2.2 Downloading Script Author Installation Package

Download the iesauthr.zip file from the Oracle Applications 11.5.6 or later APPL_TOP.

Prerequisites

- You must be using an Oracle Applications 11*i*-compliant Web browser.

- You must know the appropriate URL to access your Oracle Applications environment.

- You must have enough space allocated on the target computer to download and extract the Script Author files. The compressed file is approximately 80 megabytes in size. This size may vary on your local machine.

Login

None required

Responsibility

None required

Steps

1. From your Web browser, access the appropriate URL for your environment, using the following syntax, and press Enter.

   ```
   http://<server>.<domain>:<port>/OA_HTML/download/ies/iesauthr.zip.
   ```

   Fill in the server, domain, and port specific to your environment. For example:

   ```
   http://server1.yourcompany.com:7777/OA_HTML/download/ies/iesauthr.zip
   ```

2. Your Web browser will prompt you to perform an action with the specified file. Select the option that allows you to save the file to your local file system.

   a. If you are downloading the files using the target computer, navigate to the desired directory and save the file.

      After installation, you do not need to retain the installation package, since it will be accessible from the same URL at any time. Therefore, you may save the installation package in any location, including your default TEMP directory.

   b. If you are downloading the files for installation on a different computer, save the file to any location, and transfer the ZIP file to the target computer.

See Also

Section A.2.1, "Locating Script Author Installation Package"

Section A.2.3, "Preparing for Script Author Installation"

Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

A.2.3 Preparing for Script Author Installation

This task describes how to prepare the target computer for installation of Script Author 11.5.6 or later.

Login

None required

Responsibility

None required

Prerequisites

- You must have downloaded Script Author 11.5.6 or later from the APPL_TOP of your local environment.

- If required, you must have moved the Script Author installation package to the computer on which the Script Author will be installed. Script Author can be installed on a computer running any Microsoft Windows 95, 98, 2000, or NT operating system.

- You must have enough space allocated on the target computer to accommodate the extracted Script Author files. Once extracted, the Script Author requires approximately 100 megabytes. This size may vary on your local machine.

Steps

1. Locate the Script Author installation package, typically named iesauthr.zip.

2. In the target directory, extract the ZIP file to a directory with the same name.

   This will result in a directory named Iesauthr.

3. Review the readme.txt file located in the extracted path Iesauthr\Oracle Scripting Author Installation\readme.txt prior to installing.

See Also

Section A.2.1, "Locating Script Author Installation Package"

Section A.2.2, "Downloading Script Author Installation Package"

Section A.2.4, "Installing Script Author 11.5.6 Through 11.5.8 Baseline Release"

A.2.4 Installing Script Author 11.5.6 Through 11.5.8 Baseline Release

This task describes how to prepare the target computer for installation of the Script Author stand-alone application for releases 11.5.6 through 11.5.8 MiniPack O. This information is not relevant for releases of Oracle Scripting after release 11.5.8 (Interaction Center Family Pack P and later).

Login

None required

Responsibility

None required

Prerequisites

■   You must have downloaded Script Author 11.5.6 or later from the APPL_TOP of your local environment and extracted the Script Author installation package.

■   You must install the Script Author on a computer with a Microsoft Windows 95, 98, 2000, or NT operating system. Typically, this must be a networked computer that can see the Applications server on the same side of the firewall.

■   You must have read the readme.txt file included in the installation package.

■   You must have enough space allocated on the target computer to accommodate the installed Script Author files. Once extracted, the Script Author requires only approximately 22 megabytes. This size may vary on your local machine.

Steps

1.   Start the Oracle Universal Installer by running setup.exe from path:

```
Iesauthr\Oracle Scripting Author Installation\install\win32\setup.exe
```

Oracle Universal Installer launches and the Welcome screen appears.

2.   Click **Next** to continue.

3.   In the File Locations screen, specify the destination directory for Oracle Scripting Script Author.

   ■   The Source area displays the path to the components available for installation.

   Note: Do not modify the information in the Source area.

   ■   The Destination area displays the destination alias, and the default path to which Script Author will be installed.

> **Note:** The Script Author can be installed in any directory; installation into an Oracle Home directory is *not required*. If the target computer has an Oracle Home directory, this location is recommended and will typically default as the Destination. You may choose another destination if desired.

In the Destination area, type the destination path or click **Browse** to choose a destination path. If the destination path does not have a name, type an alias for the destination path in the Name field. You can provide any alias for the purpose of installing Oracle Scripting Script Author.

Click **Next** to continue.

4. In the Installation Types screen, select an installation type and then click **Next** to continue.

> **Note:** Each installation type installs the same components.

5. In the Summary screen, verify the installation settings.

   - To make changes, click **Previous** to return to the appropriate screen.
   - To being the installation, click **Install**.

   The Install screen appears, and Oracle Scripting Script Author is installed in the oracle\apps\ies\author directory of the destination path. The Oracle Universal Installer also creates an Oracle Scripting folder in the Programs menu.

   When the installation is complete, the End of Installation screen appears.

6. Click **Exit** to close the Oracle Universal Installer.

This completes the installation of Scripting Author. After installing, you may delete the Script Author installation package if desired.

Section A.2.1, "Locating Script Author Installation Package"

Section A.2.2, "Downloading Script Author Installation Package"

Section A.2.3, "Preparing for Script Author Installation"

A.3 Converting, Re-Deploying, and Testing Scripts

When accessing the Script Chooser Forms window to launch a script in the Scripting Engine agent interface, *all scripts deployed to the database* will be listed. Nonetheless, **only scripts that have been recompiled in Script Author 11.5.6 or later will be executable in the 11.5.6 or later Scripting Engine.**

This is true regardless of whether the scripts are launched from an integrated application (Oracle TeleSales, Oracle TeleService, or Oracle Collections), or when launching in stand-alone mode.

Thus, any previously deployed scripts required to be executed in Scripting Engine release 11.5.6 or later must be opened in a matching 11.5.6 or later Script Author, converted (by saving), re-deployed, and tested.

> **Note:** Ensure that all scripts to be converted are still supported. For example, there are two qualifications:
>
> While substitution of *an entire panel* with a Java Bean is still supported, substituting an *answer* (or panel *node*) with a Java bean is no longer supported as of release 11.5.6 and later.
>
> Scripts that contain multiple answers in a panel may not use the "button" UI type. The only supported use of a button UI type is for a panel with a single answer definition.

Steps

The following is a list of the recommended steps for conversion and redeployment:

1. Determine which scripts in the production environment must be converted (i.e., which scripts will be run in the 11.5.6 or later environment).

2. Prepare a test environment that has the 11.5.6 or later release of Scripting (including the 11.5.6 or later version of Script Author) installed.

3. For each script identified in Step 1, locate in your script archive the source .SCRIPT file that was used to deploy the script to the production environment.

> **Note:** If you are unable to locate the exact .SCRIPT file that was used to deploy the script currently in use in the production environment, you can obtain a copy of this script from the production database itself. To do so, from the Script Author, select the Database tab of the **Open Script** dialog to open scripts directly from the referenced database.

4. Make a backup copy of the script.

> **Note:** If you select Save from the Script Author, the pre-11.5.6 script will be saved in the same location, appended with a file extension of .PREVIOUS. Thus, if the previous script were named SAMPLE.SCRIPT, the 11.5.6 or later version of the script will now be named SAMPLE.SCRIPT, and the earlier version will be retained as SAMPLE.SCRIPT.PREVIOUS.
>
> **Warning:** If you use the Save As command, ensure you provide a new name for the converted script, since the <filename>.SCRIPT.PREVIOUS copy will not be created.

5. Launch the release 11.5.6 or later version of the Script Author.

6. Open the script file (identified in Step 3 above) in the Script Author.

7. The Script Author will warn you that continuing will modify the script. Once you do so, you cannot use this script with Script Author releases previous to 11.5.6.

8. Save the converted script.

9. Choose **Tools > Deploy Script** to re-deploy the script to the test environment.

   You are now ready to execute this script in the test environment.

10. Run the script in the Scripting Engine agent interface and test the re-deployed script.

11. After verifying that no new problems have been introduced, follow the same process to re-deploy the script to the production environment.

    If you discover any issues, correct the script, redeploy, and test until successful.

12. Repeat this process for each script identified in Step 1.

Guidelines

To ensure no confusion between source script files, consider creating separate folders or directories on the Script Author workstation, identifying scripts by Script Author version (at minimum, discriminating between pre-11.5.6 scripts and 11.5.6 or later scripts). Move any <filename>.previous copies of updated scripts into the appropriate pre-11.5.6 directory.

See Also

Section A.1, "Implementation Upgrade Tasks"

Section A.2, "Installation of Script Author Post-11.5.6 Stand-Alone Application"

B

Oracle Scripting Profile Options

This chapter describes profile option settings that are required for successful implementation.

This section includes the following topics:

- Setting Profile Options
- Finding Responsibility ID Values
- Oracle CRM Technology Foundation (JTT) Profile Options
- Current IES Profile Options
- Obsolete IES Profile Options
- Additional Required Profile Options

B.1 Setting Profile Options

Use the following procedure to set any profile option.

Prerequisites

Before making Oracle Forms settings, ensure that all Oracle Applications server processes are up and running. In particular, if you stopped concurrent managers before applying Oracle Applications patchsets, restart them now by changing to $COMMON_TOP/admin/scripts, and executing adcmctl.sh <APPS user name/APPS password> start.

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, navigate to **Profile > System**.

    The Find System Profile Values window opens.

2. Check the level(s) at which you want to set the profile option. The available levels are listed below:

    ■   **Site**: If you select this level, the profile will apply to the entire site. Individual profiles with a different value designated at any other level will take precedence.

    ■   **Application**: If you select this level, choose the application from the Application LOV for which you want to set the profile option.

    ■   **Responsibility**: If you select this level, choose the responsibility from the Responsibility LOV for which you want to set the profile option.

    ■   **User**: If you select this level, choose the user from the User LOV for whom you want to set the profile option.

3. In the Profile field, enter the profile name, such as IES: Architecture Type, or a wildcard search criterion such as IES%.

4. Click **Find.**

    The System Profile Values form opens with the results of your search.

5. Verify or set the profile option(s) at the levels that you selected.

B.2 Finding Responsibility ID Values

You must know the APPLICATION_ID value for Oracle Scripting and the RESPONSIBILITY_ID value for the Survey Administrator responsibility before you can set the required Oracle CRM Technology Foundation (JTT) profile options for users with these responsibilities.

> **Note:** The APPLICATION_ID value for the seeded Scripting User (or Scripting Agent) responsibility is 519. The RESPONSIBILITY_ID value for Survey Administrator is 21685. However, in some scenarios, the responsibility identification may differ, based on environmental factors.

Use the following procedure to find the APPLICATION_ID value and RESPONSIBILITY_ID value of a responsibility.

Prerequisites

None

Login

Log into Oracle Forms-based applications.

Responsibility

System Administrator

Steps

1. From the Navigator, navigate to **Security > Responsibility > Define**.

   The Responsibilities form opens.

2. Choose **View > Find.** Search for the responsibility, highlight it, and click **OK** in the search window.

   The Responsibilities form is populated with the record for the responsibility that you chose.

3. With your cursor in any field of the record, choose **Help > Diagnostics > Examine.**

   The Examine Field and Variable Values window opens.

4. In the Examine Field and Variable Values window, choose **APPLICATION_ID** from the Field LOV.

   The Value field in the Examine Field and Variable Values window is populated with the value of APPLICATION_ID.

5. In the Examine Field and Variable Values window, choose **RESPONSIBILITY_ID** from the Field LOV.

The Value field in the Examine Field and Variable Values window is populated with the value of RESPONSIBILITY_ID.

B.3 Oracle CRM Technology Foundation (JTT) Profile Options

This section summarizes the Oracle CRM Technology Foundation (JTT) profile options that you need to set when implementing Oracle Scripting. Set the profile options in this section at the site level.

Table B–1   JTT Profile Options for the Survey Administration Console

Profile Option Name	Description	Value[1]
JTF_PROFILE_DEFAULT_APPLICATION	Default application ID (519=Oracle Scripting)	519
JTF_PROFILE_DEFAULT_BLANK_ROWS	Number of blank rows on HTML forms (can be set to any integer > 0)	3
JTF_PROFILE_DEFAULT_CSS	Default Oracle CRM Technology Foundation Cascading Style Sheet	jtfucss.css
JTF_PROFILE_DEFAULT_CURRENCY	Default currency. Enter the currency code in all uppercase letters.	USD
JTF_PROFILE_DEFAULT_NUM_ROWS	Number of default rows displayed on HTML summary tables or records displayed (can be set to any integer > 0)	10
JTF_PROFILE_DEFAULT_RESPONSIBILITY (application level only)	Default responsibility ID (21685=Survey Administrator)	21685

[1] See Section B.2, "Finding Responsibility ID Values" if you need to find the APPLICATION_ID and RESPONSIBILITY_ID values for a responsibility.

B.4 Current IES Profile Options

This section summarizes the current Oracle Scripting (IES) profile options that you need to set when implementing the Scripting Engine and the Survey components of Oracle Scripting. Set the profile options in this section at the site level.

In addition to the profile name, seeded lookup values, a description of the profile setting's function, and recommended settings are provided.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : Dropdowns Display Default Null Choice in HTML UI	True False	By default, Oracle Scripting displays a text string "- select one - " at runtime for questions with a data type of drop-down list. Although "- select one -" appears in the list, this is a null value. Business rules for a drop-down list require the user to select one value for this question, resulting in a blackboard value other than null for each such question. setting this profile to False suppresses the null value with the "- select one - " one string in the Scripting Engine web interface. For Web browsers only, the first answer choice in the list will appear as the selected value, unless another answer choice has been designated as a default answer.	Set to False to prevent " - select one -" string from displaying for scripts displaying in HTML UI. Leave null or set to True to ensure this string remains in both interfaces.
IES : AUTHOR DEBUG MODE	Debug off Debug on	Controls display of informational and error messages for Script Author debugging. When enabled, provides additional informational and error messages for Script Author debugging, generated in the Apache server logs. Note that you must administer (configure) your Apache Jserv to send console output to the Apache error log.	Generally set to Debug off unless troubleshooting specific issues.
IES : Architecture Type	Apache Mid Tier/Servlet Architecture Three Tier Mode Caching Architecture	Determines mode of Oracle Scripting operations. (1) Use Apache Mid Tier for all new implementations. (2) Never set to Three Tier Mode. (3) Use Caching Architecture only if upgrading existing caching architecture implementations.	Apache Mid Tier/Servlet Architecture
IES : Debug Mode	Debug off Debug on	Controls display at runtime of informational and error messages for Scripting Engine operations. These messages are logged to IEO server logs.	Generally set to Debug off unless troubleshooting specific issues.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : Display Suspend Button on Script Frame	False True	When set to True, the Scripting Engine agent interface displays a Suspend button on the bottom of the script frame. This button is hidden if the profile is set to False (or remains null). When this button is clicked by an agent during script runtime, the current script interaction is suspended. (Requires Suspendable global script property option to be enabled.)	False unless using Oracle Scripting in stand-alone mode
IES : Scripting Panel Display Mode	Display Multiple Panels at a time in Scripting window Display Single Panel at a time in Scripting window	Determines in Scripting Engine Java agent interface whether a single panel is visible upon script execution or whether multiple panels are displayed with active panel having focus.	At the discretion of the implementing enterprise.
IES: Proxy Server Name	None	Name of Proxy server (required if proxy server used for Scripting Engine agent interface only).	Syntax: http://<servername>.<domain>:<Apache Web server port>/<servlet_zone>
IES: Proxy Server Port	None	Port of Proxy server (required if proxy server used for Scripting Engine agent interface only).	N/A

B.5 Obsolete IES Profile Options

This section summarizes obsolete Oracle Scripting (IES) profile options. *These options should not be used for any reason for Oracle Scripting release 11.5.8 or later operations.* Providing settings in some of these options could disable Scripting operations in the enterprise.

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : JNDI name used for Scripting application	None	OBSOLETE	NONE

Profile Option Name	Lookup Values	Description	Recommended Setting
IES : Port for Oracle Scripting application	None	OBSOLETE	NONE
IES : SID of Oracle Scripting Database	None	OBSOLETE	NONE
IES : Scripting Bean Display Mode	Display Scripting bean in separate Frame	OBSOLETE	NONE
	Embed Scripting bean in form		
IES : Server Host Name for Scripting application	None	OBSOLETE	NONE
IES: Agent password	None	OBSOLETE	NONE
IES: Agent user name	None	OBSOLETE	NONE
IES_SVY_OA_TEMP	None	OBSOLETE	NONE
IES : SVY ERROR PAGE	None	OBSOLETE	NONE
IES : SVY HEADER PAGE	None	OBSOLETE	NONE
IES: SVY FINAL PAGE	None	OBSOLETE	NONE
IES: TIME FORMAT	None	OBSOLETE	NONE

B.6 Additional Required Profile Options

This section summarizes additional profile options that you need to set when implementing the Scripting Engine and the Survey components of Oracle Scripting. Set the profile options in this section at the site level.

In addition to the profile name, seeded lookup values, a description of the profile setting's function, and recommended settings are provided.

Profile Option Name	Lookup Values	Level	Description	Recommended Setting
Apps Servlet Agent	None	Site Application	Identifies servlet zone of Apache Web server JServ used for Scripting Engine agent interface only. May be set at site level if one Apache server listener used. If different settings required for multiple applications, set at application level.	Syntax: http://<server name>.<domain>:<Apache Web server port>/<servlet_zone>
ICX: Language	None	Site	Establishes default language for the Internet Cartridge Exchange (ICX) settings for Web-based transactions	As appropriate.

C

Oracle Discoverer Workbooks

Analysis and reporting for data received from survey campaigns has been enhanced. In previous releases of Oracle Scripting, three survey-related reports were accessible from the Survey Administration console. Beginning with Oracle Scripting release 11.5.9 (Interaction Center Family Pack Q or later), Oracle Corporation provides six pre-defined sets of reporting criteria to provide analysis information for the effectiveness of Oracle Scripting scripts executed as survey campaigns. No longer accessible from the Survey Administration console, these expanded survey analysis reports are now available as Oracle Discoverer workbooks, available as part of Oracle Interaction Center Intelligence reporting functionality.

Oracle Discoverer is a tool which enables report builders and analysts to create, modify, and execute ad hoc queries and reports. Casual users can view and navigate through pre-defined reports and graphs. Oracle Discoverer provides a business view (referred to as the end user layer or EUL) to hide the complexity of the underlying data structure. It enables users to focus on solving business problems instead of data access issues.

When scripts are executed as survey campaigns, return information for each deployment is stored in transaction tables in the IES schema of the Oracle Applications database. These transaction tables provide reporting data for three of the six survey analysis workbooks. Executing the Summarize Survey Data Concurrent Program moves relevant information into tables that summarize survey data from individual responses. These summary tables provide reporting data for the remaining three survey analysis workbooks. A matrix containing a list of the workbooks, a description of each, and the tables or views accessed by each workbook is provided below in Appendix C.2, "Survey Workbook Descriptions".

To view survey data using Oracle Discoverer workbooks, Oracle Applications users who are assigned a particular responsibility (based on the implementation) log into Oracle Discoverer User Edition and select the appropriate workbook from the

database. The Oracle Applications user and responsibility must be granted administrative privileges to the end user layer as part of implementation. If you do not have access to the Oracle Scripting survey analysis workbooks, or do not have the appropriate privileges to access the workbooks, consult your Oracle Discoverer administrator, or perform the implementation steps described in this document.

Within the Survey Analysis business area are six pre-defined Oracle Scripting survey summary workbooks. Workbooks are lists of records, similar in design to a spreadsheet, that gather requirements for a Structured Query Language (SQL) database query. The columns in a workbook use ordinary business terms, shielding end users from requiring knowledge of the underlying relational database structure. When the query associated with each workbook is executed, users are prompted for parameters, after which the workbook builds and executes SQL statements which provide the requested information, which can range from simple to complex. The return data generated from the workbooks can be exported to a spreadsheet, allowing managers to create additional ad hoc reports on any data items available within a survey.

Using Oracle Discoverer, you therefore generate survey reports derived from survey data stored in one of two locations: in IES transaction tables within the Oracle Applications database schema, or summarized and compiled for analysis in survey summary tables following execution of the appropriate concurrent program.

Oracle Discoverer administrators can tailor existing workbooks to refine reporting information, or create new workbooks, using the Discoverer Plus or Discoverer Desktop components.

Oracle Discoverer employs an end user layer (EUL). An EUL is a metadata repository that allows users to view and select records from complex database structures without requiring knowledge of those structures or knowledge of Structured Query Language (SQL). The EUL consists of one or more business areas. Each business area contains folders (equivalent to tables and views) and each folder contains items (equivalent to columns in the table or view). Business areas are visible through Oracle Discoverer Administration Edition.

The EUL also contains workbooks. Workbooks are visible through Oracle Discoverer User Edition. Workbooks contain the specific data needed for various analysis tasks. Workbooks belong to business areas. Similar in appearance to a spreadsheet, a workbook references data in tables and views, and may have a query associated with it.

For the purpose of Oracle Scripting survey analysis, six workbooks are provided to generate specific reporting data on various survey-related activities. Each has a

query associated with it which, when appropriately executed, provides an ad-hoc report on the appropriate parameters.

As part of post-installation steps for an Oracle Applications implementation, two EULs are created. Oracle Scripting reports are accessed through the Oracle Business Intelligence System (BIS) EUL.

This implementation guide contains the required steps to implement Oracle Discoverer workbooks to generate survey analysis reports for Oracle Scripting.

Use of Oracle Discoverer workbooks for survey reporting requires upgrade or installation of Oracle Applications 11.5.9 or later. Manual post-installation steps are required to import the survey reports into the EUL. This is accomplished using a shell script or by importing the workbooks into the Oracle Discoverer 4i Admin Edition. If you have not already imported these workbooks as part of the post-installation steps for Oracle Applications 11.5.9, you can do so by following the detailed steps provided in this appendix. To import the workbooks using the shell script or for more information on post-installation steps for Oracle Applications release 11.5.9, refer to Note.227071.1 on Oracle*MetaLink*.

> **Note:** For more information, see Oracle Discoverer product documentation or notes on Oracle*MetaLink*.

This section includes the following topics:

- Appendix C.1, "Implementing Survey Analysis Discoverer Workbooks"
- Appendix C.2, "Survey Workbook Descriptions"

C.1 Implementing Survey Analysis Discoverer Workbooks

To implement Oracle Discoverer workbooks for Oracle Scripting survey reporting, the following tasks are required:

Perform the following to implement Oracle Scripting survey reporting using Oracle Discoverer workbooks:

- Stage loader files (EEX files that provide the Oracle Scripting survey analysis business area, including workbooks) in preparation for import.
- Verify the accessibility of the BIS EUL to your Oracle Discoverer Administration Edition user. If BIS EUL is not accessible to your user, create an EUL.

- Import loader files (EEX files that provide business area and workbooks) into the EUL.

- Assigning User and Responsibility Privileges to the EUL.

EULs are created by Oracle Discoverer administrators. These users have access to Oracle Discoverer Administrative Edition with database administrator (DBA) privileges. These users must also have access to Oracle Discoverer User Edition (client/server version).

The EUL stores metadata for Oracle Discoverer. Oracle Scripting administrators that want to view the Oracle Discoverer reports must access the BIS EUL. If the BIS EUL has not been created, perform the procedure Creating an End User Layer. If this EUL exists and you have access to it from Oracle Discoverer, you may omit this procedure from your implementation steps.

Prerequisites

Prerequisites for implementing Oracle Discoverer reporting for survey analysis include the following:

- Oracle Scripting Release 11.5.9 or later

- Oracle Applications patched to release 11.5.9 or later, equivalent to Interaction Center Family Pack Q or later

- Windows NT server with Oracle Discoverer 4i Administration Edition

  Oracle Discoverer 4i Administration Edition and the corresponding adupdeul utility to import EEX files run only on Windows NT. If using Oracle Discoverer workbooks for reporting on Oracle Scripting survey activities, you must have at least one NT server with Discoverer Admin Edition installed. This may be a separate server, providing that it can communicate with the applications database.

- Appropriate patching of Oracle Discoverer, including Oracle Discoverer Patch 1834171

Tasks

This section includes the following tasks:

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"

- Section C.1.2, "Creating an End User Layer"

- Section C.1.3, "Assigning End User Layer Implementation Privileges"

- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"

- Section C.1.5, "Assigning Responsibility and Business Area Privileges"

- Section C.1.6, "Assigning Workbook Privileges"

- Section C.1.7, "Verifying the End User Layer"

- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

References

- Oracle Discoverer Plus User's Guide Oracle Discoverer 4.1

- Oracle Discoverer 4i Plus User's Guide Oracle Discoverer 4.1

- Oracle Discoverer Administration Edition Administration Guide Oracle Discoverer 4.1

See Also

Appendix C.2, "Survey Workbook Descriptions"

C.1.1 Staging Oracle Discoverer EEX Files for Import

Oracle Discoverer workbooks are created in Discoverer User Edition and can be saved as Discoverer export (EEX) files. The Discoverer files that provide Oracle Scripting survey workbook reports must be imported by an Oracle Discoverer Administrator before they are accessible to Discoverer users.

After installation of Oracle Applications release 11.5.9 or Interaction Center Family Pack Q, these files are installed on the appl_top in the following directory in your Oracle Applications server file structure:

```
$AU_TOP/discover
```

In order for these files to be imported during implementation, you must either map a network drive on the administration client to the appl_top where the EEX files are installed, or you copy the EEX files to the Discoverer administration server.

The appropriate EEX files to import include the following:

- bixiescmpgnrsflo.eex

- bixieslsthdrsflo.eex

- bixiesqstnsflo.eex

- bixiessrvysvflo.eex

- bixsvyqsfrwbo.eex

- bixiescrsstbrwbo.eex

- bixieslstsmmrflo.eex

- bixiesrmndrsvflo.eex

- bixiessvylstnflo.eex

- bixsvyqstxwbo.eex

- bixiescyclsvflo.eex

- bixiesmslstntflo.eex

- bixiesrspntrsflo.eex

- bixsvycmpswbo.eex

- bixiesdplcrssflo.eex

- bixiesnswrsflo.eex

- bixiessmmrystflo.eex

- bixsvydpldwbo.eex

- bixiesdplymntflo.eex

- bixiesqsdtvflo.eex

- bixiessrvynlybao.eex

- bixsvyqsdtwbo.eex

See Also

- Section C.1.2, "Creating an End User Layer"

- Section C.1.3, "Assigning End User Layer Implementation Privileges"

- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"

- Section C.1.5, "Assigning Responsibility and Business Area Privileges"

- Section C.1.6, "Assigning Workbook Privileges"

- Section C.1.7, "Verifying the End User Layer"

- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.2 Creating an End User Layer

Perform the following to create an EUL for Oracle Discoverer. This is not required if you have access to the BIS EUL.

Prerequisites

You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.

Responsibility

Oracle RDBMS DBA privileges

Steps

1. Log into Oracle Discoverer User Edition as an Oracle RDBMS user with database administrator (DBA) privileges.

   For example:

   > **Username:** sysadmin:System Administrator (case-sensitive)

   > **Password:** sysadmin

   > **Connect:** mydatabaseinstance

2. Click **Connect**.

   Upon connection, if no EUL exists accessible to your user account, the following message appears:

   > Oracle Discoverer Administration Edition:

   > You do not have access to any End User Layers (EUL). You must create at least one EUL to which you have access to use Discoverer. Do you want to create an EUL now?

   Click **Yes**.

   The EUL Manager Wizard appears.

3. In the first window of the EUL Manager, click **Create an EUL...**

   The Create EUL Wizard appears.

4. Complete the EUL Wizard options as follows:

   In the Who will own the new EUL area:

- Select the **Select an existing user** option.

  Since you are using the same user, you are not prompted for a password.

- Check **Grant access to PUBLIC**.

- Check **New EUL is for use by Oracle Applications users ONLY**.

  Note: Selection of this last criteria is essential for application security and to ensure supportability.

In the EUL connection information area:

- In the User field, type the name of the user.

- In the Password field, type the user's password.

- In the Confirm Password field, retype the user's password.

Click **Next**.

The Create EUL Wizard Step 2 window appears.

5. Complete the EUL Wizard Step 2 options as follows:

   - From the Schema list, select the APPS schema.

   - In the Password field, type the apps level password.

   - Click **Next**.

     The Create EUL Wizard Step 3 window appears.

6. Complete the EUL Wizard Step 3 options as follows:

   - From the default tablespace list, select an appropriate tablespace into which EUL data will be inserted.

   - From the temporary tablespace list, select an appropriate tablespace into which data is temporarily stored.

     Caution: Choosing the wrong tablespace can greatly impact performance. If you don't know which tablespace is appropriate for the default or temporary tablespace, consult your database administrator.

   - Click **Finish**.

A series of progress indicators appears.

7. An Oracle Discoverer Administration Edition message appears, indicating that the EUL has been created successfully. Click **OK**.

8. Optionally, if you want to install tutorial information, click **Yes**. Otherwise, click **No**.

9. An Oracle Discoverer Administration Edition message appears, asking whether you want to connect as the owner of the new EUL that you have just created.

    Click **No** if you want to exit Oracle Discoverer Administration Edition.

    To proceed with the next procedure:

    ■ Click **Yes.**

       The Load Wizard appears.

    ■ Click **Cancel** to close the load wizard window, and proceed to step 2 of Section C.1.3, "Assigning End User Layer Implementation Privileges".

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"

- Section C.1.3, "Assigning End User Layer Implementation Privileges"

- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"

- Section C.1.5, "Assigning Responsibility and Business Area Privileges"

- Section C.1.6, "Assigning Workbook Privileges"

- Section C.1.7, "Verifying the End User Layer"

- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.3 Assigning End User Layer Implementation Privileges

This procedure is only required if you have created a new EUL. Perform this procedure to assign administrative privileges for this EUL to a specific Oracle Applications user and responsibility (required in order to complete the implementation).

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.

- You must be the owner of the EUL.

- You must have created or imported an Oracle Discoverer EUL.

Responsibility

Oracle RDBMS DBA privileges

Steps

1. Log into Oracle Discoverer Administrator Edition as an Oracle RDBMS user with database administrator (DBA) privileges. This must be the same user that created or imported this EUL.

2. From the Tools menu, select **Privileges**.

   The Privileges window appears with the Privileges tab selected.

3. From the user and responsibility drop-down list, select the appropriate user for whom you want to assign administrative access to this EUL.

   - Ensure the User box is checked.

   - Optionally, you can clear the Responsibility box to reduce the amount of records appearing in this list.

     For implementations with access to the seeded SYSADMIN user, selecting SYSADMIN is appropriate. Otherwise, ensure the user you select has the appropriate responsibility assigned.

4. From the Privileges list, select all options and click **Apply**.

   The selected user is granted administrative access to the EUL.

5. From the user and responsibility drop-down list, select the appropriate responsibility for which you want to assign administrative access to this EUL.

   - Ensure the Responsibility box is checked.

   - Optionally, you can clear the User box to reduce the amount of records appearing in this list.

     For implementations using the seeded SYSADMIN user in step 3 above, select the System Administrator responsibility.

For other implementations, any responsibility is appropriate (including System Administrator), as long as the user selected in step 3 already has this responsibility.

6. From the Privileges list, select all options and click **Apply**.

The selected responsibility is granted administrative access to the EUL.

7. Click **OK** to save your work and exit the Privileges window.

8. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

All remaining implementation or use steps in this document should be performed by the user and responsibility for which administrative access to this EUL is granted.

Guidelines

- For implementations with access to the seeded SYSADMIN user, using this account is appropriate.

- For other implementations, designate a user as appropriate. This user must be assigned administrative privileges to this EUL.

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"

- Section C.1.2, "Creating an End User Layer"

- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"

- Section C.1.5, "Assigning Responsibility and Business Area Privileges"

- Section C.1.6, "Assigning Workbook Privileges"

- Section C.1.7, "Verifying the End User Layer"

- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.4 Importing Oracle Discoverer Objects into an End User Layer

This procedure is required for all enterprises that want to implement Oracle Scripting survey analysis capabilities.

Perform the following to import Oracle Discoverer objects (a business area and workbooks) into an end user layer. Use the BIS EUL if accessible to your user. Otherwise, use the EUL you created.

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.

- This user must have administrative privileges to this EUL. For implementations with access to the BIS EUL and the seeded SYSADMIN user, using this account is appropriate.

- If you have access to the BIS EUL but do not have access to the seeded SYSADMIN user, perform the procedure described in Section C.1.3, "Assigning End User Layer Implementation Privileges" in order to designate an appropriate user and responsibility with administrative access to the appropriate EUL.

Responsibility

Use the responsibility for which EUL administrative privileges are assigned. See prerequisites above for more information.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with administrative privileges to this EUL.

   For example:

   > **Username:** sysadmin:System Administrator (case-sensitive)

   > **Password:** sysadmin

   > **Connect:** mydatabaseinstance

2. Click **Connect**.

   The Load Wizard appears.

3. From the Load Wizard window, select Open an existing business area.

   The Load Wizard: Step 1 window refreshes, listing all business areas accessible to your user account.

4. In the business area selection list, if **Survey Analysis** appears, you do not need to import any EEX files.

   In the business area selection list, if the **Survey Analysis** business area is *not listed,* you must import the Oracle Discoverer export (EEX) files for use with Oracle Scripting survey reporting.

5. From the Load Wizard window, select **Cancel**.

   The Load Wizard window closes.

6. Optionally, close the Administration Tasklist window, which is not required.

7. From the **File** menu, select **Import**.

   The Import Wizard: Step 1 window appears.

8. From the Import Wizard window, click **Add...**

   The Open window appears.

9. In the Open window, navigate to the directory in which the Oracle Scripting EEX files have been staged in preparation for import.

   In this release, there are 22 EEX files required for Oracle Scripting survey reports.

10. Select all required EEX files and click **Open** to add them to the list.

    The Open window closes. The EEX files to import are now listed in the Import Wizard: Step 1 window.

11. Click **Next**.

    The Import Wizard: Step 2 window appears.

12. In the Import Wizard: Step 2 window, set the following options:

    - In the *If two objects match then what action should occur* area, select **Refresh the object** and check **Preserve display related properties**.

    - In the *How should two objects be matched up* area, select **Match objects by identifier**.

    - In the *Should the current user take ownership of imported workbooks* area, select **Always take ownership of imported workbooks**.

13. In the Import Wizard: Step 2 window, click **Next**.

    The Import Wizard: Step 3 page appears.

14. In the Import Wizard: Step 3 page, click **Start** to begin importing the selected EEX files.

15. When the importation completes, click **Finish**.

    The Import Wizard closes.

In the Data tab of the Discoverer Administration Edition work area, the business area you imported appears as an expandable navigation tree.

Each business area contains folders (equivalent to tables and views) and each folder contains items (equivalent to columns in the table or view).

If enabled, the Administration Tasklist also appears.

16. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

See Also
- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.5 Assigning Responsibility and Business Area Privileges

The user and responsibility established in Section C.1.3, "Assigning End User Layer Implementation Privileges" has full administrative access to this EUL.

Perform this procedure to designate a user and responsibility for using Oracle Discoverer User Edition to generate and view workbooks, without administrative access.

For implementations with access to the seeded SYSADMIN user, performing this step allows you to have Oracle Discoverer survey analysis workbook users who do not have unrestricted access to Oracle Applications.

This procedure results in assigning permissions to access the workbooks to a specific responsibility. Following this procedure, assign the designated responsibility to workbook user accounts.

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.

- This user must have administrative privileges to this EUL. For implementations with access to the BIS EUL and the seeded SYSADMIN user, using this account is appropriate.

Responsibility

Use the responsibility for which EUL administrative privileges are assigned.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with administrative privileges to this EUL.

    For example:

      Username: sysadmin:System Administrator (case-sensitive)

      Password: sysadmin

      Connect: mydatabaseinstance

2. Click **Connect**.

    The Load Wizard appears.

3. From the Load Wizard window, select **Cancel**.

    The Load Wizard window closes.

4. Optionally, close the Administration Tasklist window, which is not required.

5. From the Tools menu, select **Privileges**.

    The Privileges window appears with the Privileges tab selected.

6. From the user and responsibility drop-down list, select the appropriate responsibility for which you want to assign user access to the appropriate set of workbooks.

    - Ensure the Responsibility box is checked.

    - Optionally, you can clear the User box to reduce the amount of records appearing in this list.

> For implementations using a responsibility specifically created for this purpose, ensure you select the new responsibility. For example, if you created a responsibility called **Survey Analysis**, select this responsibility.

7. From the Privileges list, select the User Edition box.

   The checkbox options below the User Edition category enable.

8. From the User Edition category, select all options and click **Apply**.

9. Click **OK** to save your work and exit the Privileges window.

10. From the Tools menu, select **Security**.

    The Security window appears with the business area to user tab selected.

11. In the business area to user tab of the Security window, in the Business area list, select Survey Analysis.

    > **Note:** If the Survey Analysis business area is not listed, perform the procedure described in Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer". If you still do not see the Survey Analysis business area, consult your Oracle Discoverer administrator.

12. ensure the appropriate business area is selected.

13. From the Available users/responsibilities list, select the appropriate Oracle Applications responsibility for which you have granted user edition privileges in step 6 above, and click the right arrow.

    The selected responsibility appears in the Selected users/responsibilities list.

14. When satisfied with your selections, click **Apply** and then **OK**.

    The Security window closes.

    You have now granted permission for any user with a specific responsibility to open the Oracle Discoverer User Edition tool and to access the Survey Analysis business area. You must perform the next procedure to grant this responsibility permission to open the appropriate workbooks.

15. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

See Also

-

-

-

-

-

-

-

C.1.6 Assigning Workbook Privileges

Perform this procedure to grant to a specific responsibility the privileges to open the appropriate workbooks for survey analysis.

This step must be performed for all implementations for which the procedure in Section C.1.5, "Assigning Responsibility and Business Area Privileges" was performed.

Prerequisites

- You must have access to Oracle Discoverer Administration Edition on a Windows NT server. This software can be installed on your client or you can access a remote server using software such as Citrix.

- This user must have administrative privileges to this EUL. For implementations with access to the BIS EUL and the seeded SYSADMIN user, using this account is appropriate.

- This user must be the administrative user who imported the workbooks.

Responsibility

Use the responsibility for which EUL administrative privileges are assigned.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with administrative privileges to this EUL who is also the owner of the workbooks.

   For example:

   Username: sysadmin:System Administrator (case-sensitive)

Password: sysadmin

Connect: mydatabaseinstance

2. Click **Connect**.

   The Workbook Wizard appears.

3. From the Workbook Wizard window, select **Cancel**.

   The Workbook Wizard window closes.

4. From the File menu, select **Manage Workbooks > Sharing.**

   The Share Workbooks window appears with the Workbook to user tab selected.

5. From Workbook drop-down list, select the first workbook.

   The list of values in the Workbook list include all six workbooks in the Survey Analysis business area.

6. In the Users area, from the Available list, select the responsibility for which User Edition privileges were granted in the previous procedure and click **Add**.

   The selected responsibility appears in the Shared list in the Users area.

7. Click **OK**.

   The Share Workbooks window closes.

8. Repeat steps 4 through 7 for each workbook in Survey Analysis business area.

9. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

   Implementation of the survey analysis workbooks is now complete. Perform the next procedure to view survey analysis workbooks.

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.7, "Verifying the End User Layer"
- Section C.1.8, "Viewing Oracle Scripting Survey Reports"

C.1.7 Verifying the End User Layer

Once all implementation steps described above have been performed, you can verify an Oracle Discoverer User Edition's access to the Oracle Scripting reports available in the Survey Analysis business area.

Perform this procedure to verify the implementation.

> **Note:** This step is optional, to assist in troubleshooting.

Prerequisites

- Oracle Scripting-specific workbooks must already be imported.

- Your username should have the responsibility for which the User Edition privilege, business area privilege, and privilege to view survey analysis workbooks have been granted.

- You must have access to Oracle Discoverer User Edition (client/server edition).

- You must know the Oracle database username who owns the EUL.

Responsibility

The appropriate responsibility varies, based on implementation choices.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with the appropriate responsibility to view survey analysis workbooks.

   For example:

   > **Username:** anyuser:Survey Analysis (case-sensitive)

   > **Password:** yourpassword

   > **Connect:** mydatabaseinstance

2. Click **Connect**.

   The Workbook Wizard appears.

3. From the Workbook Wizard window, select **Cancel**.

   The Workbook Wizard window closes.

4. From the Tools menu, select **Options...**

The Options window appears.

5. Click the **EUL** tab.

6. From the Select a default EUL list, ensure the Oracle database user that created the EUL appears.

7. Click **OK**.

8. From the File menu, select **Open**.

   The Open Workbook window appears.

9. Click Database and then click **Open**.

   The Open Workbook from Database window appears.

   All workbooks for which this responsibility has permissions appear in the list.

10. Select the appropriate workbook and click **Open**.

11. If the Workbook in Other Database Account window appears, select **Open the workbook in the current database account** and click **OK**.

12. An Oracle Discoverer message will appear asking for confirmation to run the query for the selected workbook.

    You have completed the verification. Since you do not need to execute the query, click **No**.

13. From the **File** menu, select **Exit** to exit Oracle Discoverer Administration Edition.

References

- Summarize Survey Data Concurrent Program
- Appendix C.2, "Survey Workbook Descriptions"

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"

-

C.1.8 Viewing Oracle Scripting Survey Reports

Once implemented, Oracle Discoverer User Edition users with the appropriate responsibility can generate Oracle Scripting survey reports using the workbooks imported into the Survey Analysis business area.

Perform this procedure to view Oracle Discoverer workbooks in the Survey Analysis business area and generate fresh reports.

Prerequisites

- Oracle Scripting-specific workbooks must already be imported.

- Your username should have the responsibility for which the User Edition privilege, business area privilege, and privilege to view survey analysis workbooks have been granted.

- You must have access to Oracle Discoverer User Edition (client/server edition).

- Three workbooks require the Summarize Survey Data concurrent program to be executed prior to generating workbooks, to ensure current data. These include the BIX: Survey Campaign Summary Report, the BIX: Survey Deployment Detail Report, and the BIX: Survey Question Frequency Report.

Responsibility

The appropriate responsibility varies, based on implementation choices.

Steps

1. Log into Oracle Discoverer User Edition as an Oracle Applications user with the appropriate responsibility to view survey analysis workbooks.

   For example:

   Username: anyuser:Survey Analysis (case-sensitive)

   Password: yourpassword

   Connect: mydatabaseinstance

2. Click **Connect**.

   The Workbook Wizard appears.

3. From the Workbook Wizard, select **Open an existing Workbook**.

The Workbook Wizard window refreshes, showing workbook location options.

4. Select **Database**.

   The Workbook Wizard window closes. The Open Workbook from Database window appears.

   All workbooks for which this responsibility has permissions appear in the list.

5. From the Open Workbook from Database window, select the appropriate workbook and click **Open**.

6. If the Workbook in Other Database Account window appears, select **Open the workbook in the current database account** and click **OK**.

7. An Oracle Discoverer message appears asking for confirmation to run the query for the selected workbook. To execute the query, click **Yes**. To view the workbook without data, click **No**.

   > **Note:** To run the query at a later time, select **Sheet > Refresh Sheet**.

   Click **Yes**.

   The Parameter Wizard appears.

   For specific guidance on workbook parameters, refer to the detailed workbook descriptions detailed in Appendix C.2, "Survey Workbook Descriptions".

8. Select the appropriate parameters and click **Finish** to execute the query.

   The query executes, populating the workbook with data.

   > **Note:** Three workbooks (BIX: Survey Campaign Summary Report, BIX: Survey Deployment Detail Report, and BIX: Survey Question Frequency Report) require the Summarize Survey Data concurrent program to be executed prior to generating workbooks, to ensure current data. For any of these workbooks, if a query is executed *prior to the execution of the concurrent program*, and no data is available, the report will appear displaying column headings only. If the concurrent program had been previously executed, but not recently, stale data will appear in the workbook.

9. Optionally, to export the workbook to Microsoft Excel, click the Excel icon in the Discoverer User Edition tool bar.

References

- Summarize Survey Data Concurrent Program
- Appendix C.2, "Survey Workbook Descriptions"

See Also

- Section C.1.1, "Staging Oracle Discoverer EEX Files for Import"
- Section C.1.2, "Creating an End User Layer"
- Section C.1.3, "Assigning End User Layer Implementation Privileges"
- Section C.1.4, "Importing Oracle Discoverer Objects into an End User Layer"
- Section C.1.5, "Assigning Responsibility and Business Area Privileges"
- Section C.1.6, "Assigning Workbook Privileges"
- Section C.1.7, "Verifying the End User Layer"

C.2 Survey Workbook Descriptions

The following table indicates the name of each workbook, its filename, and a description of each workbook.

Workbook Name	File Name	Description	Tables or Views Accessed by Workbook
Survey Question Frequency	bixsvyqsfrwbo.eex	This report provides answer frequency data for each survey deployment, showing only non-text fields. This report can be used to show how people have responded to survey questions at a summary level for each survey deployment. It provides a table of all non-text questions, plus the frequency of responses for all the questions.	■ ies_svy_ques_data_v

Workbook Name	File Name	Description	Tables or Views Accessed by Workbook
Survey Question Detail	bixsvyqsdtwbo.eex	This report provides the detailed response for each survey that has been completed. It compiles all answers that each respondent has provided. It is sorted by response date.	▪ ies_svy_surveys_all ▪ ies_svy_cycles_all ▪ ies_svy_deplyments_all ▪ ies_svy_response_entries ▪ ies_question_data ▪ ies_questions ▪ ies_question_types ▪ ies_panels ▪ ies_answers.ies_svy_list_entries_v ▪ ies_ams_list_entries_v
Survey Question Text	bixsvyqstxwbo.eex	This report provides detail solely on the text data type fields (text field, text area, and password field) collected within a survey. This is useful for providing a collection of comments provided by survey respondents for specific questions.	▪ ies_svy_surveys_all ▪ ies_svy_cycles_all ▪ ies_svy_deplyments_all ▪ ies_svy_response_entries ▪ ies_question_data ▪ ies_questions ▪ ies_question_types ▪ ies_panels ▪ ies_answers.ies_svy_list_entries_v ▪ ies_ams_list_entries_v
Survey Campaign Summary	bixsvycmpswbo.eex	This report provides Summary information on each Survey Campaign, with totals provided at the survey campaign and cycle level. It allows you to get an overview of how many surveys have been sent out, whether there have been errors with the surveys, and the response ratio.	▪ ies_svy_summary_stats_v ▪ ies_svy_reminders_v

Workbook Name	File Name	Description	Tables or Views Accessed by Workbook
Survey Deployment Detail	bixsvydpldwbo.eex	This report provides detail information on each deployment, with totals provided at the cycle and deployment level.	■ ies_svy_summary_stats_v ■ ies_svy_reminders_v
Survey Cross Tabulation	bixiescrsstbrwbo.eex	This report allows the administrator to select a specific survey deployment, and then cross-tabulate the results from two different questions. You can select a question and a target value from one question and use that as the establishing criteria for reporting on a different response. For example, you can select all survey records in which the answer to the Overall Satisfaction question is "3 - somewhat satisfied," and then for that set of records, display the responses for a second question (such as "would you use this service again").	■ ies_svy_surveys_v ■ ies_svy_deplyments_v ■ ies_svy_cycles_v ■ ies_questions ■ ies_answers ■ ies_question_data ■ ies_svy_response_entries

This section includes the following topics:

See Also

C.2.1 Survey Question Frequency Workbook

This report provides answer frequency data for each survey deployment, showing only non-text fields. This report can be used to show how people have responded to survey questions at a summary level for each survey deployment. It provides a table of all non-text questions, plus the frequency of responses for all the questions.

Survey Question Frequency Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Campaign Name	Name of the survey campaign.
Cycle Name	Name of the cycle, in alphabetical order.
Deployment Name	Name of the deployment, in alphabetical order.
Panel Name	Name of the panel. These appear in the order in which the panels displayed in the script.
Question Name	Name of the question. These appear in the order in which the question was asked in the survey questionnaire.
Available Answers	Answers for each question. These appear in the order in which they are listed for the question.
Selected Answers	The answers that have been selected by survey respondents. Excludes abandoned survey responses. In other words, if a respondent did not answer even a single question in a survey questionnaire, then that survey transaction is excluded from these totals. If a respondent answered one or more questions, with null values for any other question, then the responses are included in these totals.
Total Answers per Question	The total number of responses for the question (sum of all the selected answers). Excludes aborted surveys.
Answer Frequency	Percentage of answers for this question. Calculated only for data types of radio buttons, check boxes, and drop-down lists. It is not valid for text fields or text areas.

Selection Criteria

- When executed, this workbook selects a specific survey campaign, and can use wild card characters on the survey campaign's cycles and deployments. For instance, users can see all cycles and all deployments for a particular survey

campaign, or users can select for a specific deployment. If users select multiple cycles and deployments, the workbook sorts by cycle, then deployment, in alphabetical order.

- Excludes aborted survey responses (those with 0 answers). Only for abandoned (1 or more answers) and completed survey responses.

Sort Criteria

Survey campaign

Cycle name/ID

Survey deployment (breaks to a new page for each survey cycle, each deployment)

- Panel name
  - Questions listed in the order in which they appear in the panel

Calculations

- Calculates the total answers for each question for each deployment
- Calculates the answer frequency for each question for each deployment
- For each deployment, provides a sub-total which shows the total number of survey responses that were evaluated in this report

See Also

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.2 Survey Question Detail Workbook

This report provides the detailed response for each survey that has been completed. It compiles all answers that each respondent has provided. It is sorted by response date.

Survey Question Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Submit Date	Date the response was entered - RESPONSE_COLLECTED_DATE
Response ID	Survey Respondent ID
First Name	Contact's first name from the AMS List Entries table (for targeted survey deployments only). Blank if standard (non-list-based) deployment.
Last Name	Contact's first name from the AMS List Entries table (for targeted survey deployments only). Blank if standard (non-list-based) deployment.
Panel Name	Name of the panel. These appear in the order in which the panels displayed in the script.
Question Name	Name of the question. These appear in the order in which the question was asked in the survey questionnaire.
Answer	The survey respondent's selected answer to the question.

Selection Criteria

- Select a specific survey deployment
- Select on a Submit Date Range
- Excludes aborted surveys (those with 0 answers)

Sort Criteria

Survey campaign

Cycle name/ID

Survey deployment

- Panel name
    - Questions listed in the order in which they appear in the panel.

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

C.2.3 Survey Question Text Workbook

This report provides detail solely on the text data type fields (text field, text area, and password field) collected within a survey. This is useful for providing a collection of comments provided by survey respondents for specific questions.

Survey Question Text Responses Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Submit Date	Date the response was entered - RESPONSE_COLLECTED_DATE
Survey respondent ID	RESPONSE_ID
Panel Name	Name of the panel. These appear in the order in which the panels displayed in the script.
Question Name	Name of the question. These appear in the order in which the question was asked in the survey questionnaire.
Answer	The survey respondent's selected answer to the question.

Selection Criteria

- Must select a specific survey deployment.

- Can select a specific Response Collected Date range

- Excludes abandoned surveys (those with 0 answers)

Sort Criteria

Submit Date

- Panel name

    - Questions listed in the order in which they appear in the panel.

Calculations

Total responses evaluated.

See Also

C.2.4 Survey Campaign Summary Workbook

This report provides Summary information on each Survey Campaign, with totals provided at the survey campaign and cycle level. It allows you to get an overview of how many surveys have been sent out, whether there have been errors with the surveys, and the response ratio.

Survey Question Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Survey Campaign Name	Survey Campaign Name
Cycle Name	Name of Survey Cycle
Deployment Name	Name of Survey deployment
Deployment Status	Status of the deployment. Status types include Open, Pending, Error, Active, Cancelled or Incomplete. For a description of each status, see Deployment Status.
Number of Targeted Respondents	For targeted deployments, the total number of survey targets in a list. For standard deployments, the sum of abandoned and completed surveys (represents the number of opportunities to get survey feedback).
Number of Survey Errors	Number of surveys that could not be delivered to list member because of e-mail address or server errors.
Number Surveys Aborted	Number of surveys that have been opened, but no answers have been recorded.

Column Name	Description
Number Abandoned	Number of surveys that have been opened, but for which no answers were provided.
Number Surveys Completed	Number of surveys that have one or more answers.
Response Ratio	Completed responses divided by total number of invitations

Selection Criteria

- Can select survey campaign name to get a report on a specific survey campaign.

- Can also selectively report on survey campaigns by using wildcard search characters.

Sort Criteria

Survey campaign

- Cycle

  - Deployment

Calculations

- Number of survey errors

- Number of opportunities

- Number of abandoned surveys

- Number of completed surveys

- Response ratio

- Totals for Opportunities, Errors, Abandons, Completed (rolled up at cycle level)

- Totals for Opportunities, Errors, Abandons, Completed (rolled up at survey campaign level)

See Also

C.2.5 Survey Deployment Detail Workbook

This report provides detail information on each deployment, with totals provided at the cycle and deployment level.

Survey Deployment Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Campaign Name	Name of the survey campaign.
Cycle Name	Name of the cycle, in alphabetical order.
Deployment Name	Name of the deployment, in alphabetical order.
Deployment Status	Status of the deployment. Status types include Open, Pending, Error, Active, Cancelled or Incomplete. For a description of each status, see Deployment Status.
Start Date	The start date for a deployment. This will be blank at the survey campaign and cycle levels, which do not have start dates.
End Date	The deployment end date for a deployment. This will be blank at the survey campaign and cycle levels, which do not have end dates.
List Name	Name of list used to send invitation or reminder master documents.
Number of Invitations Sent	The number of survey invitations sent. This value will be blank for standard (non-list-based) survey deployments.
Number of Reminders Sent	The number of survey reminders sent. This value will be blank for standard (non-list-based) survey deployments.
Number Surveys Completed	Number of surveys that have one or more answers.
Number of Abandoned Surveys	Number of surveys that have been opened, but for which no answers were provided.
Number of Errors	Number of surveys that could not be delivered to list member because of e-mail address or server errors.
Response Ratio	Completed responses divided by total number of invitations

Selection Criteria

- Can select from survey campaign name, cycle name, and deployment name.

- Can use % wildcard search character.

- Can select by deployment status (Active, Pending, Open, Closed).

Sort Criteria

Survey campaign

- Cycle

    - Deployment

        - Deployment Status

Calculations

- Number of Invitations Sent

- Number of Reminders Sent

- Number of survey errors

- Number of opportunities

- Number abandoned surveys

- Number completed surveys

- Response ratio

- Totals for Opportunities, Errors, Abandons, Completed - At Deployment level

- Totals for Opportunities, Errors, Abandons, Completed - Rolled up at Cycle level

- Totals for Opportunities, Errors, Abandons, Completed - Rolled up at Survey Campaign level

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.6, "Survey Cross Tabulation Workbook"

C.2.6 Survey Cross Tabulation Workbook

This report allows the administrator to select a specific survey deployment, and then cross-tabulate the results from two different questions. You can select a question and a target value from one question and use that as the establishing criteria for reporting on a different response. For example, you can select all survey records in which the answer to the Overall Satisfaction question is "3 - somewhat satisfied," and then for that set of records, display the responses for a second question (such as "would you use this service again").

Survey Deployment Detail Workbook Columns

The following table lists column headings for this workbook and a description of each column.

Column Name	Description
Selection Question	The question that is to be used as the primary selection
Cross Tab Question	The question that is cross tabulated against the selection question
Selection Question Choices	The available answers under the selection question
Cross Tab Answers	The available answers under the cross tab question
Cross Tab Answer Frequency	The frequency at which each answer was chosen for this particular selection question and cross tab question
Cross Tab Answer Ratio	The percentage of the individuals who had responded with a particular value under the selection question
Cross Tab Answer Percentage	The percentage of individuals who had this particular value for their selection question, out of all people surveyed.

Selection Criteria

■　Must select a specific Survey Deployment

■　Select a specific question and answer pair as the primary selection criteria

■　Select the cross-tabulation question and answer definitions

■　Excludes aborted surveys (those with 0 answers)

Sort Criteria

Sort by the answers in the Selections Question Choices column, then by the Cross Tab Answers column

Calculations

- Percentage of respondents who had each specific response for the cross-tabulation question.

- Total ratio of respondents out of the entire deployment who had the specific question/answer pair. For instance, if selecting for an age range of between 40-50, this would show the percentage of total respondents for the entire survey who were in this age range.

See Also

Appendix C.2.1, "Survey Question Frequency Workbook"

Appendix C.2.2, "Survey Question Detail Workbook"

Appendix C.2.3, "Survey Question Text Workbook"

Appendix C.2.4, "Survey Campaign Summary Workbook"

Appendix C.2.5, "Survey Deployment Detail Workbook"

Abbreviations and Acronyms

Abbreviations and Acronyms

Acronym	Definition
ANI	Automatic Number Identification
AOM	Application Object Model
API	Application Programming Interface
CRM	Customer Relationship Management
CTI	Computer-Telephony Integration
DBA	Database Administrator
DNIS	Dialed Number Identification Service
ERP	Enterprise Relationship Planning
EUL	End User Layer
GIF	Graphics Interchange Format
GUI	Graphic User Interface
HR	Human Resources
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol – Secure
IG	Implementation Guide

Acronym	Definition
JAR	Java Archive
JDBC	Java Database Connectivity
JDK	Java Development Kit
JRAD	Java Rapid Application Development
JSP	Java Server Pages
JTF	Java Technology Foundation
JVM	Java Virtual Machine
PL/SQL	Programming Language/Structured Query Language
SID	System Identifier
SQL	Structured Query Language
sysadmin	Systems Administrator
UI	User Interface
URL	Uniform Resource Locator

Product-Related Acronyms

Acronym	Definition
HRMS	Oracle Human Resources Management System
ICX	Internet Cartridge Exchange
JTA	Oracle Common Application Components
JTF	Oracle CRM Data Model
JTT	Oracle CRM Technology Foundation

Oracle Product Designations

Acronym	Definition
AMS	Oracle Marketing
AST	Oracle TeleSales
BIS	Oracle Business Intelligence System
BIX	Oracle Interaction Center Intelligence
IEO	Oracle Interaction Center Technology
IES	Oracle Scripting
IEX	Oracle Collections
ICX	Oracle Self-Service Web Applications
JTA	Oracle Common Application Components
JTF	Oracle CRM Data Model
JTT	Oracle CRM Technology Foundation
JTO	Oracle One-to-One Fulfillment

Glossary

action

A Script Author command that executes at runtime upon evaluation of the object containing the action. Actions can be associated with branches, or with blocks if designated as an API block. See also pre-action and post-action.

answer collection

The recording of end user responses ("answers") to all answer controls ("questions") during a script transaction. Answer collection is performed by the Scripting Engine for a script executed in either the agent interface or the Web interface, when the Answer Collection option is selected as a global script property. Answers are collected in the IES_QUESTION_DATA table in the Oracle Applications schema. Script end user responses are collected for each "question" or answer definition designated as collectable.

answer definition

See question.

answer UI type

See answer user interface type.

answer user interface type

The type of control that appears for a panel question at runtime and which accepts a response (an answer) from the script end user. Each question or answer definition has a single answer user interface (UI) type. Once selected in the Script Author, this feature of a question cannot be changed. Some answer UI types accept alphanumeric entry (text fields, text areas and password fields). Others accept keyboard entries or mouse clicks (radio buttons, check boxes and checkbox groups,

buttons, drop-down lists and multi-select list boxes). Some support more than one response per question (checkbox groups and multi-select list boxes). All answer UI types support null values except buttons and radio buttons.

argument

Data provided to a function or method for use in its calculations and processing. Arguments are passed to the function or method by listing them in parentheses in the function or method name. In a Java method, these are also called parameters.

best practice Java method

Oracle Applications includes Java methods written specifically for Oracle Scripting to quickly enable frequently requested script runtime functionality. Class files containing these "best practice" Java methods are included in APPS.ZIP, each in the package oracle.apps.ies.bestpractice.<Class Name>.

best practice survey

Oracle Corporation provides some pre-built survey questionnaire scripts flows to serve as examples of how to customize scripts specifically intended for use as survey questionnaires. Aspects of these scripts can also be used in scripts to be executed in the agent interface. Best practice survey scripts are placed on the 11i APPL_TOP in the directory $IES_TOP/scripts during installation or patching of Oracle Scripting. These are provided on an as-is basis. Use of best practice surveys requires customization and is not supported.

blackboard

Virtual memory space in Oracle Scripting where information is stored in key/value pairs. This information is only retained for the duration of a single Scripting interaction.

building block

A portion of a Script Author script that contains functionality integrating with other Oracle Applications. Building blocks typically contain one or more API and typically perform a task-based operation such as creating a service request, registering for an event, and so forth. Building blocks are not intended to be used as stand-alone scripts, but are expected to be imported into a script and modified as appropriate. These are provided on an as-is basis. Use of building blocks requires customization and is not supported.

campaign

A focused effort to achieve a particular goal from a targeted population over a specific period of time for a particular business purpose.

class

A set of related objects that share common characteristics. In Java, a class contains a set of methods.

collectable

A boolean property of a Script Author question or answer definition within a panel. This option is selected by default in the general tab of the data dictionary of each specific question defined. Individual answers can be designated as uncollectable by clearing the collectable option. When answer collection is enabled at the global script level, all script end user responses provided in a script session will be saved for questions designated as collectable. Also see answer collection.

CRM Resource Manager

Oracle CRM Resource Manager is a responsibility included with a Rapid Install of Oracle CRM Applications so that HRMS person types (most often, employees) required by Oracle products can be created. It does not have the full functionality of HRMS. Note that if HRMS is installed, HRMS employees must be created using that tool and cannot be created with CRM Resource Manager.

cycle

A cycle is a repeatable survey event, utilizing the same script. Multiple cycles will also allow the same survey to be given through different media types, when supported. Multiple cycles are typically used to measure the same characteristics at a different moment in time.

deployment

A deployment is the lowest and most granular set of requirements for a survey campaign. A deployment is associated with a specific survey and cycle, and identifies specific response start and end dates. List-based deployments identify Oracle Marketing lists and Oracle One-to-One Fulfillment templates. Deployments must be set to active status (they must be "deployed") before a survey campaign commences.

deployment ID

An identification number for each deployment in a particular instance. This is created upon setting a deployment to Active status. All respondents in a particular deployment will have the same deployment ID.

dID

See deployment ID.

error Page

This survey resource is a JSP file (or static HTML with a JSP file extension) that displays as a full page any time an error results during survey participation (when a respondent is taking a survey). For example, an error might occur if a resource is unavailable, the script designated by the survey campaign information is not loaded, or other unexpected conditions are encountered. This page may contain graphics in GIF or JPEG format.

final Page

This survey resource is a JSP file (or static HTML with a JSP file extension) that displays after the last panel in the survey questionnaire has been viewed by the survey respondent, It typically includes a thank-you message to the survey respondent, and indicates to the survey respondent that the survey has ended successfully. This page may contain graphics in GIF or JPEG format, and optionally may contain a hypertext link, for example to the enterprise's home page.

footprinting

Footprinting is the recording in the Oracle Applications database of the names of each panel in a script transaction that is visited during a script transaction, and the duration of time (in milliseconds) prior to the activation of the next panel. This feature can provide useful script tuning data. The default for this property is True. When enabled (or when the Answer Collection property is True), this data is saved in the IES_PANEL_DATA table in the Oracle Applications schema.

graphical script

A script contains the business rules, text and graphics to progress the script end user from panel to panel at runtime. Runtime scripts are executed using any interface of the Scripting Engine component of Oracle Scripting. A graphical script is created using the graphical tools of the Script Author, the authoring and development component of Oracle Scripting. Graphical scripts can also be created by graphing a wizard script. At runtime, scripts display panels containing

questions, answers, and graphics. Each panel includes a Continue button. The graphical script end user progresses through each panel. The flow of the script is controlled by business rules built using the standard Script Author graphical tools, including associated custom code, and may be a rigid flow or may change dynamically, based on the end user's responses. Graphical scripts are created, edited and deployed using the Script Author Java applet, and can be listed, deleted, or associated with custom Java from the Scripting Administration console.

header Page

This survey resource is a JSP file (or static HTML with a JSP file extension) that displays at the top of each page throughout the survey session, except the Error and Final pages). Just below the Header, the content placed in the panel layout editor for that panel (if any) will display, followed by all answer definitions in the panel (each in its own horizontal row). The Header Page often contains corporate logos or other standardized graphics in GIF or JPEG format.

invitation

An e-mail message master document, inviting potential survey respondents (identified in an Oracle Marketing list) to participate in a survey. Each e-mail message contains a customized URL, from which the invited list member can respond to the specified survey.

JRAD

Java Rapid Application Development, an Oracle proprietary technology stack. The Oracle JRAD technology stack is the technology upon which the updated Survey Administration console is built.

JTA

Java Technology Applications, an Oracle proprietary technology stack referring to CRM Common Application Components.

JTF

Java Technology Framework, an Oracle proprietary technology stack referring to CRM Foundation products. Oracle applications using underlying CRM Foundation technology are further divided into the CRM Applications Foundation technology stack (known as JTA) and CRM Technology Foundation (JTT) technology stack.

JTT

Java Technology Tools, an Oracle proprietary technology stack referring to CRM Common Application Components.

list

The list of potential survey respondents who will receive an invitation to participate in the questionnaire. List members may also receive reminders to participate in a survey, which includes the survey end date. Lists are built using Oracle Marketing's list management feature. Use of list-based surveys requires Oracle Marketing.

list member

Any individual included in a list built using Oracle Marketing's list management feature. All list members are potential survey respondents.

master document

A master document is a message template used in Oracle One-to-One Fulfillment. Master documents typically include merge fields populated by an associated SQL query. Upon execution of the query, a copy of the master document is created for each list member, containing the data returned by the query. This creates a personalized message. From a survey perspective, master documents are invitation or reminder messages in HTML format that are sent by e-mail from the fulfillment engine to members of a defined Oracle Marketing list. Master documents are required for targeted survey deployment operations. Master documents and their associated queries are associated with specific Oracle One-to-One Fulfillment templates.

merge field

Merge fields are data elements in an Oracle One-to-One Fulfillment master document surrounded by open and close guillemet (for example, «merge_field»). When a fulfillment request is executed, specified table columns or views are queried. For each list member record, the master document is replicated, with the actual data returned from the query embedded or merged in the master document replicate in place of the merge field. List-based surveys require at minimum one merge field (the survey URL that the recipient must access to participate in the survey). Merge fields must correspond to data returned from a query.

method

A task that can be performed with specific data items or properties. A Java method can be a function (resulting in a return value) or a procedure.

Oracle HRMS

Oracle HRMS is a full-featured enterprise Human Resources Management System. It is an ERP application using Oracle Forms technology to manage human resources used by a variety of ERP and CRM applications.

PHP

Personal home page. Accessed when logging into Oracle Self Service applications, the personal home page lists the responsibilities available to the logged in Oracle Applications account in two categories: Self Service, to access self-service applications, and Applications, for access to ERP applications and Forms-based CRM applications.

post-action

A Script Author command that executes immediately following the evaluation of the object with which the post-action is associated, and prior to evaluation of the next object in the flow of the script. Post-actions can be associated with panels, blocks, groups, or the global script. See also action and pre-action.

pre-action

A Script Author command that executes prior to the evaluation of the object with which the pre-action is associated. Pre-actions can be associated with panels, blocks, groups, or the global script. See also action and post-action.

prototype

Prototype is a boolean characteristic of survey campaigns. Prototype survey campaigns are identical to standard survey campaigns, except that the script used as the survey campaign questionnaire is not locked. The purpose is to allow survey campaign administrators more freedom to refine requirements for survey campaigns, including modification to the script for the designated survey campaign. You can exclude prototype survey campaigns from survey campaign lists by selecting the Exclude prototypes option. This characteristic can only be selected or cleared while a survey campaign status is Open.

query

A specific statement of request for information to be returned in the form of records from the database. Queries are constructed in structured query language (SQL). For targeted survey deployments, each Oracle One-to-One Fulfillment master document must have an associated query which obtains information populated into

the resulting document in placeholders called merge fields. A query must be valid to return any of the requested information.

question

The answer definition or data properties of a panel in the Script Author that will result in the appearance of an answer user interface (UI) control at runtime. All panels must contain at least one question and may contain any number. Questions may have commands associated with them in the Script Author to validate responses at runtime, or to provide default user responses. Questions can be as simple as a continue button, or as complex as a database lookup providing dynamic lookup values (answer choices) for the answer control at runtime. Questions support various answer UI types. Question properties are accessed in the Script Author by selecting the **Answer** option from the Panel Properties tree.

reminder

An e-mail message master document, reminding potential survey respondents (already invited to take a survey) of the appropriate URL to access to respond to the survey. Reminders often include information such as the last day the recipient may participate in the survey. Using the same master document, reminders can be sent in a batch process on a predefined schedule, or can be sent manually for any individual list member.

respondent

An individual responding to a request for information (in the form of participating in a questionnaire). For targeted survey deployments, respondents are list members who respond to an invitation sent via e-mail to participate in the survey.

respondent ID

A unique identification number for each list-based respondent in a particular instance. The URL for a list member includes as parameters both the deployment ID common to all participants in a given deployment and each list member's unique respondent ID.

rID

See respondent ID.

responsibility

A responsibility is a level of authority that allows a user to access specific functionality and data in Oracle Applications. Responsibilities are assigned by a user with the System Administrator responsibility.

sample

The population from which information is solicited by survey.

script

A script contains the business rules, text and graphics to progress the script end user from panel to panel. Scripts are created using the Script Author component of Oracle Scripting. At runtime, scripts are executed using any interface of the Scripting Engine component of Oracle Scripting. The same script can be executed in the agent interface, or (after creating and deploying a survey campaign) executed in a Web browser as a survey questionnaire or web script accessed from a self-service Web application. Scripts are further qualified by type. See graphical script and wizard script.

script information area

The script information area is a programmable feature of the Script Author in which script developers can place information that appears as a header in any script executed in the Scripting Engine agent interface at runtime. The script information area is a global script property and appears in every session of a script executed in the agent interface when this information is defined. It is referred to in the Script Author user interface as the Static Panel. You can access this area by selecting **File > Script Properties > Static Panel** or by right-clicking on an empty area of the canvas and selecting **Edit Blob Properties > Static Panel**.

static panel

See script information area.

survey campaign

A survey campaign is the collection of requirements needed to execute a script in a Web browser using the Scripting Engine Web interface. This is the super class that references the global script. A survey campaign must have information for at least one cycle and at least one deployment. Survey campaigns are created using the Survey Administration console. As prerequisites to creating a survey campaign, you must create and deploy from the Script Author the script containing questions and answer choices, and you must define survey resources.

survey questionnaire

Set of questions built using the Script Author component of Oracle Scripting for execution in the Scripting Engine as a survey questionnaire, typically to gather feedback, obtain market data, tabulate opinion, measure satisfaction, or otherwise collect data directly from specific (customers, employees, prospects) or non-specific target populations. A survey questionnaire script is physically identical to a script executed in the agent interface, although it may be customized to take advantage of sophisticated HTML interpretation available to Web browsers.

survey resource

Survey resources are JSP-format files (or static HTML files saved with a JSP file extension) that provide functionality to a script executing in the Scripting Engine Web interface. The header survey resource displays as the top portion of each HTML page. The footer survey resource displays as the bottom portion of each HTML page. The final page resource is displayed after a survey questionnaire or script running in a Web browser has been completed. The error page resource is the page that appears when an error condition occurs during script execution in a Web browser. Self-service Web applications use different error page resources than survey questionnaires. Survey resources are uploaded into the $OA_HTML directory on the applications server and are served by the Web server as appropriate.

survey respondent

See respondent.

SYSDATE

This refers to the current Systems Date of the database. When creating records in a database the SYSDATE is typically time-stamped as creation date, and SYSDATE is also used as a default setting for creating responsibilities from an Oracle Applications perspective.

template

Oracle One-to-One Fulfillment's term for a group of master documents and any collateral. Templates may also include campaign specifications. Templates are identified with a survey campaign at the deployment level to identify invitation and reminder master documents and their associated queries. Only required for targeted (list-based) survey deployments. No collateral is typically associated with a survey campaign.

trading community architecture (TCA)

The trading community architecture is a customer model designed to support complex trading communities. The entire Oracle E-Business Suite uses the TCA customer model, providing a common model for customer data in all Oracle business applications. TCA strives to model *all* relationships within a trading community, supporting business-to-business (B2B) *and* business-to-consumer (B2C) models equally.

Uniform Resource Locator (URL)

A specific string of characters, including protocol identifier (such as HTTP or FTP), that together forms the address to view or retrieve resources deployed on the Internet.

wizard script

A wizard script is a script created using the Script Wizard feature of the Script Author, the authoring and development component of Oracle Scripting. Like graphical scripts, wizard scripts are executed in any Scripting Engine interface. At runtime, they display panels containing questions, answers, and graphics. Each panel includes a Continue button required to be clicked by the script end user at runtime to progress through each panel. The flow of the script is controlled by business rules built using the Script Wizard. Based on the choices of the script developer, the end user's flow through the script may be rigid (the same panels are seen regardless of how the user answers) or may change dynamically (demonstrating different flows based on the end user's responses). Wizard scripts can be listed, created, copied, edited, deployed, or converted to a graphical script only from the Script Wizard feature of the Script Author.