



Siebel Analytics Scheduler Guide

Version 7.8.1
April 2005

Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404

Copyright © 2005 Siebel Systems, Inc.

All rights reserved.

Printed in the United States of America

No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photographic, magnetic, or other record, without the prior agreement and written permission of Siebel Systems, Inc.

Siebel, the Siebel logo, UAN, Universal Application Network, Siebel CRM OnDemand, TrickleSync, Universal Agent, and other Siebel names referenced herein are trademarks of Siebel Systems, Inc., and may be registered in certain jurisdictions.

Other product names, designations, logos, and symbols may be trademarks or registered trademarks of their respective owners.

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

U.S. GOVERNMENT RESTRICTED RIGHTS. Programs, Ancillary Programs and Documentation, delivered subject to the Department of Defense Federal Acquisition Regulation Supplement, are "commercial computer software" as set forth in DFARS 227.7202, Commercial Computer Software and Commercial Computer Software Documentation, and as such, any use, duplication and disclosure of the Programs, Ancillary Programs and Documentation shall be subject to the restrictions contained in the applicable Siebel license agreement. All other use, duplication and disclosure of the Programs, Ancillary Programs and Documentation by the U.S. Government shall be subject to the applicable Siebel license agreement and the restrictions contained in subsection (c) of FAR 52.227-19, Commercial Computer Software - Restricted Rights (June 1987), or FAR 52.227-14, Rights in Data—General, including Alternate III (June 1987), as applicable. Contractor/licensor is Siebel Systems, Inc., 2207 Bridgepointe Parkway, San Mateo, CA 94404.

Proprietary Information

Siebel Systems, Inc. considers information included in this documentation and in Siebel Business Applications Online Help to be Confidential Information. Your access to and use of this Confidential Information are subject to the terms and conditions of: (1) the applicable Siebel Systems software license agreement, which has been executed and with which you agree to comply; and (2) the proprietary and restricted rights notices included in this documentation.

Contents

Chapter 1: What's New in This Release

Chapter 2: Using Siebel Analytics Scheduler

| | |
|--|----|
| Concepts and Terms Related to Siebel Analytics Scheduler | 9 |
| Job Scheduling and Scheduler Scripts | 10 |
| iBots Usage | 10 |
| Database Support in Analytics Scheduler | 11 |
| Siebel Analytics Scheduler Schemas | 12 |
| Siebel Analytics Scheduler Log File | 13 |
| Changing Scheduler Table Names | 13 |
| Upgrading Siebel Analytics Scheduler Schemas | 14 |
| Siebel Analytics Scheduler Installation and Configuration Tasks | 15 |
| Configuring Scheduler and Analytics Web Servers Installed on Separate Machines | 16 |

Chapter 3: Configuring Siebel Analytics Scheduler With Job Manager

| | |
|--|----|
| Siebel Analytics Scheduler as a Background Process (UNIX Only) | 18 |
| Opening Analytics Job Manager (Windows Only) | 18 |
| Restoring or Changing Default Job Scheduler Values | 18 |
| Setting Up Analytics Scheduler Jobs | 19 |
| Adding a Scheduler Job in Analytics Job Manager | 19 |
| Modifying a Scheduler Job in Analytics Job Manager | 20 |
| Modifying Analytics Scheduler iBots | 20 |
| Managing Analytics Scheduler Job Instances | 21 |
| Viewing or Refreshing Analytics Scheduler Job Instances | 21 |
| Canceling Analytics Scheduler Job Instances | 22 |
| Purging Analytics Scheduler Job Instances | 22 |
| Analytics Scheduler Job Manager Toolbar | 24 |
| Scheduler Job Manager File Menu | 24 |

| | |
|--|----|
| Scheduler Job Manager Service Management Menu | 25 |
| Scheduler Job Manager Jobs Menu | 25 |
| Scheduler Job Manager Instances Menu | 26 |
| Scheduler Job Manager Instance Properties | 27 |
| Analytics Job Manager Configuration Dialog Box | 28 |
| Analytics Job Manager Scheduler Tab | 28 |
| Analytics Job Manager Mail Tab | 31 |
| Analytics Job Manager iBots Tab | 32 |
| Analytics Job Manager Workflow Tab | 34 |
| General Scheduler Job Properties | 35 |
| Analytics Scheduler Job Actions | 37 |
| Analytics Scheduler Job Triggers | 38 |

Chapter 4: Integrating Siebel Delivers With Siebel Workflow

| | |
|--|----|
| Process of Configuring Siebel Delivers to Launch Siebel Workflow | 44 |
| Configuring the Siebel Enterprise Server for Workflow Integration with Siebel Delivers | 44 |
| Configuring Siebel Analytics Scheduler to Trigger Workflows | 45 |
| Third-Party Triggering of Siebel Analytics Scheduler Jobs | 45 |
| Using Replace Mode | 46 |

Chapter 5: Configuring Siebel Analytics Scheduler Job Scripts

| | |
|--|----|
| Examples of Analytics Scheduler Scripts | 48 |
| Job Manager Script for Cache Clearance | 48 |
| Configuring Custom Script Properties for Analytics Scheduler | 50 |
| Script for iBot Chaining | 50 |
| Configuring Custom Script Properties for Siebel Delivers | 51 |
| Siebel Analytics Scheduler Read-Only Script Object Properties | 53 |
| Siebel Analytics Scheduler Read/Write Script Object Properties | 54 |
| Siebel Analytics Scheduler Script-Defined Constants | 55 |
| DayEnum Constants | 55 |
| DayOfWeekEnum Constants | 55 |
| JobFlagsEnum Constants | 56 |
| MonthEnum Constants | 57 |
| OccurrenceEnum Constants | 57 |

| | |
|--|----|
| Analytics Scheduler Script Object Methods and Events | 58 |
| CreateArray Method | 59 |
| DeregisterCancelCommand Method | 60 |
| GetConfigurationValue Method | 60 |
| GetTempFileName Method | 61 |
| LaunchProcess Method | 61 |
| RegisterCancelCommand Method | 61 |
| ScheduleJobDaily Method | 62 |
| ScheduleJobMonthlyDate Method | 63 |
| ScheduleJobMonthlyDOW Method | 64 |
| ScheduleJobNow Method | 65 |
| ScheduleJobOnce Method | 66 |
| ScheduleJobWeekly Method | 67 |
| OnError Event | 68 |

Index

1

What's New in This Release

What's New in Siebel Analytics Scheduler Guide, Version 7.8.1

Table 1 lists changes described in this revision of the documentation to support Siebel Analytics version 7.8.1.

Table 1. What's New in Siebel Analytics Scheduler Guide, Version 7.8.1

| Topic | Description |
|---|--|
| "Examples of Analytics Scheduler Scripts" on page 48 | These examples of Scheduler scripting have been incorporated into the Scripting chapter. |
| New tabs in Job Manager: "Analytics Job Manager iBots Tab" on page 32 | There are two new fields for the Analytics Job Manager iBots tab: <ul style="list-style-type: none">■ Max Rows Times Columns■ Purge Files Older Than (days) |
| Script constants revised: "JobFlagsEnum Constants" on page 56 | The nqJobIsUserScript constant was removed. |
| Script properties revised: "General Scheduler Job Properties" on page 35 | The Disable NQS Functions property was removed. |

2

Using Siebel Analytics Scheduler

Siebel Analytics Scheduler is an extensible scheduling application for scheduling reports to be delivered to Siebel Analytics users at specified times. Scheduler is the engine behind the Siebel iBots feature of Siebel Delivers and is used by the Job Manager feature of the Siebel Analytics Server Administration Tool.

Under Windows, Scheduler runs as a Windows service. Under Sun Solaris, IBM AIX, and Hewlett-Packard HP-UX UNIX environments, Scheduler runs as a process.

Initial configuration and installation are covered in the *Siebel Analytics Installation and Configuration Guide*, available on Siebel SupportWeb.

TIP: To access *Siebel Analytics Installation and Configuration Guide*, go to SupportWeb and click the link Product Documentation. Under the Product Documentation heading, click the Siebel Bookshelf link. Under the heading Siebel Business Analytics Applications, click the link Siebel Business Analytics 7.7 Bookshelf. On the Analytics Bookshelf page, click the link Individual Guides, then select the book you want.

This chapter contains the following topics:

- “Concepts and Terms Related to Siebel Analytics Scheduler” on page 9
- “Changing Scheduler Table Names” on page 13
- “Upgrading Siebel Analytics Scheduler Schemas” on page 14
- “Siebel Analytics Scheduler Installation and Configuration Tasks” on page 15
- “Configuring Scheduler and Analytics Web Servers Installed on Separate Machines” on page 16

Concepts and Terms Related to Siebel Analytics Scheduler

Siebel Analytics Scheduler is a server in its own right, along with Siebel Analytics Server and Siebel Analytics Web Server, and Scheduler activities are linked with those of the other two Analytics servers. Scheduler is configured through the Analytics Server Administration Tool. Messages to and from Scheduler pass through Siebel Analytics Web Server.

The following terms and concepts are used with Siebel Analytics Scheduler:

Jobs. See “Job Scheduling and Scheduler Scripts” on page 10.

iBots. See “iBots Usage” on page 10.

Scheduler databases. See “Database Support in Analytics Scheduler” on page 11.

Scheduler table schemas. See “Siebel Analytics Scheduler Schemas” on page 12.

Scheduler logs. See “Siebel Analytics Scheduler Log File” on page 13.

Job Scheduling and Scheduler Scripts

The primary purpose of the Siebel Analytics Scheduler is to manage and schedule jobs. In this context, a *job* is a task performed by Siebel Analytics Server. Scheduler supports two types of jobs:

- Scripted jobs that you set up and submit using the Job Manager feature of the Server Administration Tool

NOTE: Scripting for iBots and scripts defined by the Scheduler Job Manager are supported only under Windows platforms and are not supported under UNIX.

For scripted jobs, Siebel Analytics Scheduler supports two scripting languages:

- VBScript
- JScript

An example of a scripted job is to take the Siebel Analytics Server usage statistics that are logged in a file and periodically load them into a back-end database. The script defines which actions are to be performed and when these actions should be executed.

- Unscripted jobs, called *iBots*, that you set up and submit using Siebel Delivers

NOTE: If, while adding or modifying a job in the Job Manager, you enter a script in the Script field, Analytics Scheduler creates a file with an SCS extension in the path defined by this field. However, the Scheduler's back-end database does not actually store the job scripts. In general, do not add or remove scripts from this directory.

Chapter 5, "Configuring Siebel Analytics Scheduler Job Scripts," gives examples of job scripts.

iBots Usage

With unscripted jobs, you define the actions to be performed using Siebel Delivers, to create an iBot. Siebel Analytics Web gathers the necessary information about the priority, delivery devices, user, and other characteristics; packages that information into a job; and tells Siebel Analytics Scheduler when it wants the job to be executed.

NOTE: To create iBots, you must have licensed Siebel Delivers. If you have not licensed Siebel Delivers, you can only create scripted jobs using the Job Manager feature of the Server Administration Tool.

If you are setting up Siebel Analytics Scheduler to use only iBots, you do not have to set up separate jobs using the Job Manager interface. However, because iBots are jobs, you can create iBots using Siebel Delivers and then further configure the iBots using the Job Manager. To do so, you must first configure Siebel Analytics Scheduler. For information about configuration options, see "Analytics Job Manager Configuration Dialog Box" on page 28.

For information about setting up iBots and making them available for subscription, see the online help for Siebel Delivers.

For information about setting up and managing scripted jobs, see the following sections and chapters:

- [“Setting Up Analytics Scheduler Jobs” on page 19](#)
- [“Managing Analytics Scheduler Job Instances” on page 21](#)
- [Chapter 4, “Integrating Siebel Delivers With Siebel Workflow”](#)
- [Chapter 5, “Configuring Siebel Analytics Scheduler Job Scripts”](#)

Database Support in Analytics Scheduler

Siebel Analytics Scheduler uses a single commercial back-end database to store pertinent information about a job, its instances, and its parameters. The Scheduler works with all the supported databases for Siebel Analytics. For the complete list of supported databases, see *System Requirements and Supported Platforms* on Siebel SupportWeb.

TIP: To access *Siebel Analytics Installation and Configuration Guide*, go to SupportWeb and click the link Product Documentation. Under the Product Documentation heading, click the Siebel Bookshelf link. Under the heading Siebel Business Analytics Applications, click the link Siebel Business Analytics 7.7 Bookshelf. On the Analytics Bookshelf page, click the link Individual Guides, then select the book you want.

The Scheduler service starts only if the back-end database satisfies the following conditions:

- The database is configured.

NOTE: There is a one-to-one relationship between the back-end database and Siebel Analytics Scheduler. Do not configure multiple Siebel Analytics Scheduler applications to use a single back-end database.

- The database is operational.

NOTE: For information about the specific back-end databases supported by Siebel Analytics Scheduler, see *System Requirements and Supported Platforms* on Siebel SupportWeb on Siebel SupportWeb.

- For some databases, the database authentication mode must be mixed.

For example, if the database is SQL Server, then the security mode should be set to allow both SQL Server and Windows Authentication for logon.

NOTE: Do not use operating system authentication for the back-end database login. If you do, the Scheduler service may not start in some cases.

Siebel Analytics Scheduler Schemas

The database schemas associated with Scheduler are located in the `$INSTALLDIR\Schema` directory. This directory holds several SQL scripts that can be used by major commercial databases to create the tables in Siebel Analytics Scheduler's schema.

The schemas include four tables:

- S_NQ_JOB
- S_NQ_INSTANCE
- S_NQ_JOB_PARAM
- S_NQ_ERR_MSG

The following are characteristics of Siebel Analytics Scheduler schema tables:

- The Scheduler tables are included in the Siebel operational applications transactional database, version 7.7. If you have version 7.7 of Siebel operational applications installed, you do not need to install Siebel Analytics Scheduler tables. These tables have been included in versions of Siebel operational applications since version 7.5.
- The contents of the schema tables are implementation-specific. For information about creating these tables, see *Siebel Analytics Installation and Configuration Guide*.
- For enterprise applications, you need to use a supported commercial database. For information about setting up a commercial back-end database for use by Siebel Analytics Scheduler, see *Siebel Analytics Installation and Configuration Guide*.

[Table 2 on page 12](#) describes the information stored in each schema table.

Table 2. Analytics Scheduler Schema Tables

| Table | Description |
|----------------|---|
| S_NQ_JOB | Stores information about scheduled jobs. |
| S_NQ_INSTANCE | Stores information about instances. |
| S_NQ_JOB_PARAM | Stores information about job parameters. |
| S_NQ_ERR_MSG | Stores information about job instances that do not complete successfully. |

The data types for each column should remain true to the intent of the schema. For example, if the job ID is defined as an integer type, do not change it to a varchar type. However, increasing the number of characters in a varchar column is an acceptable change.

The schemas also store path and file names to job scripts, providing easy updates to several jobs if they share a script.

Siebel Analytics Scheduler Log File

Siebel Analytics Scheduler logs information about its startup and shutdown activities in the NQScheduler.log file, located in the Log directory in the Siebel Analytics software installation directory.

You can use a text editor to view this file. The entries in the log file are written in UTF-8 format. Set your viewer to UTF-8 to display readable text.

Changing Scheduler Table Names

You can change the names of the tables that Scheduler uses by adding settings to the Windows registry.

NOTE: The same method of changing the registry is used under both Windows and UNIX platforms. For Scheduler installations under UNIX, before beginning the following procedure, follow the steps in “Changing Scheduler Table Names Under UNIX” on page 14.

Changing Scheduler Table Names Under Windows

The following procedure shows how to change Scheduler table names under Windows. For this procedure, a new key, *DB Column Names*, has been created as an example.

To change table names for Analytics Scheduler

- 1 Run regedit and navigate to [HKEY_LOCAL_MACHINE\SOFTWARE\Siebel Systems, Inc.\Siebel Analytics\Scheduler\7.7\].
- 2 Create a new key named *DB Column Names*.
- 3 For each of the entries under the *DB Column Names* key, add the subkey and string values shown in the following table.

| Name | Type | Data |
|-----------------|--------|----------------|
| TABLE_JOBS | REG_SA | S_NQ_JOB |
| TABLE_INSTANCES | REG_SA | S_NQ_INSTANCE |
| TABLE_PARAMS | REG_SA | S_NQ_JOB_PARAM |
| TABLE_ERRMSGs | REG_SA | S_NQ_ERR_MSG |

The values created in the data string become the values used for Scheduler table names.

NOTE: Changing the table names requires a restart of Scheduler.

Changing Scheduler Table Names Under UNIX

For Scheduler installations under UNIX, perform the following procedure before modifying the registry.

To change Scheduler table names under UNIX

- 1 Set up the UNIX environment with the command scripts `sa.sh` or `sa.csh`.
- 2 After the UNIX display is set, continue with the procedure described in [“Changing Scheduler Table Names” on page 13](#).

Upgrading Siebel Analytics Scheduler Schemas

If you are upgrading from Siebel Analytics version 7.5, this section contains notes about upgrading the Siebel Analytics Scheduler database schemas.

NOTE: If you are upgrading from any Siebel Analytics version prior to version 7.5, contact Technical Support for assistance.

The schema upgrade should have been done in the installation procedure outlined in *Siebel Analytics Installation and Configuration Guide*. That installation procedure includes a step to import a Siebel Import File (with the extension `.sif`), which modifies the Siebel Analytics Scheduler database schema.

- In all cases, first install the newer version of Siebel Analytics.
- If you currently use Siebel operational applications version 7 and upgrade to Siebel Analytics version 7.7.1 at the same time, you do not need to upgrade the database schema separately.
- If you install the stand-alone version of Siebel Analytics 7.7.1, then you must upgrade the database schema for Siebel Analytics by running the following file:

```
SAJOBS.7.5To7.7.xxx.sql
```

Where xxx refers to your RDBMS type: DB2, MSSQL, or Oracle.

If you get an error indicating that the TZ_NAME column is not found, shut down Siebel Analytics Scheduler and rerun the `SAJOBS.7.5To7.7.xxx.sql` file.

Siebel Analytics Scheduler Installation and Configuration Tasks

The following high-level steps describe the installation and configuration of Siebel Analytics Scheduler. For further information, see *Siebel Analytics Installation and Configuration Guide* and *Siebel Analytics Web Administration Guide*.

NOTE: If you are migrating a Siebel Analytics environment to a new system (for example, moving from development to testing, or development to production) make sure that you also migrate the Siebel Analytics Server repository file and the Scheduler tables. The Scheduler tables are required for Siebel iBots to function.

- 1 Install Siebel Analytics Scheduler.
- 2 If you are not using a Siebel operational applications transactional database, set up database tables.
- 3 Start the Siebel Analytics Server.
- 4 Populate configuration options as follows:
 - **Windows.** Open the Server Administration Tool, and then open the Job Manager to populate configuration options.
 - **UNIX.** Execute the file `schconfig.exe` on the UNIX machine hosting Siebel Analytics Scheduler. This file is located in the `$INSTALLDIR/Bin` directory.
- 5 Start Siebel Analytics Scheduler.
- 6 Verify that the Siebel Analytics Web Server points to the Siebel Analytics Scheduler.

CAUTION: Do not use operating system authentication for the back-end database login. If you do, the Scheduler service may not start in some cases.

The following registry entry identifies the name of the machine running the Scheduler:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Siebel Systems, Inc.\Siebel Analytics\web\<n.n>
\Alerts
ScheduleServer
```

where `<n.n>` is the version number of Siebel Analytics (for example, 7.7).

The value in the Data column should be the Scheduler machine name. For more information on registry settings in Siebel Analytics, see *Siebel Analytics Web Administration Guide*.

NOTE: If the Scheduler and the Analytics Web Server are not installed on the same machine, you must configure the registry settings as shown in "Configuring Scheduler and Analytics Web Servers Installed on Separate Machines" on page 16.

Configuring Scheduler and Analytics Web Servers Installed on Separate Machines

The registry setting must be specified when the Scheduler and the Analytics Web Server are not installed on the same machine. This registry setting is specified on the Analytics Web Server machine and points to the Scheduler machine address.

Changing the ScheduleServer Registry Setting Under Windows

The following procedure shows how to change the Windows registry setting.

To change the ScheduleServer registry setting under Windows

- 1 Run regedit.
- 2 Navigate to HKEY_LOCAL_MACHINE\SOFTWARE\Siebel Systems, Inc.\Siebel Analytics\Web\<n.n>\Alerts, where <n.n> is the version number of Siebel Analytics.
- 3 For the subkey ScheduleServer, change the value in the Data column to the Scheduler machine name.

Changing the ScheduleServer Registry Setting Under UNIX

The following procedure shows how to change the registry setting.

To change the ScheduleServer registry setting under UNIX

- 1 Open the Analytics/setup/sa.reg file.
- 2 Edit the ScheduleServer entry.

At the following lines:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Siebel Systems, Inc.\Siebel  
Analytics\Web\7.7\Alerts]  
"ScheduleServer"="HOSTNAME"  
"Enabled"="Y"
```

Using the machine IP address, substitute the machine on which the Scheduler process runs as the ScheduleServer parameter for HOSTNAME.

- 3 Save and close the file.
- 4 From the Analytics/setup directory, run the following commands to import the modified sa.reg file:

```
$ sa-cli.sh  
$ ./regedit -c sa.reg
```

You can verify the registration by running regedit from the command line again and seeing the registry entries.

3

Configuring Siebel Analytics Scheduler With Job Manager

To schedule jobs to run at any time, use the Job Manager feature of the Siebel Analytics Server Administration Tool. You can set options for a start time, a start date, an interval between executions, and an optional end time and date. For information about scheduling Siebel iBots, see *Siebel Analytics Web Administration Guide*.

The main topics of this chapter describe how to configure the Scheduler from the Job Manager:

- ["Siebel Analytics Scheduler as a Background Process \(UNIX Only\)" on page 18](#)
- ["Opening Analytics Job Manager \(Windows Only\)" on page 18](#)
- ["Setting Up Analytics Scheduler Jobs" on page 19](#)
- ["Adding a Scheduler Job in Analytics Job Manager" on page 19](#)
- ["Modifying a Scheduler Job in Analytics Job Manager" on page 20](#)
- ["Modifying Analytics Scheduler iBots" on page 20](#)
- ["Managing Analytics Scheduler Job Instances" on page 21](#)
- ["Analytics Scheduler Job Manager Toolbar" on page 24](#)
- ["Analytics Job Manager Configuration Dialog Box" on page 28](#)
- ["General Scheduler Job Properties" on page 35](#)
- ["Analytics Scheduler Job Actions" on page 37](#)
- ["Analytics Scheduler Job Triggers" on page 38](#)

Siebel Analytics Scheduler as a Background Process (UNIX Only)

Starting Siebel Analytics Scheduler as a background process requires that you export the display to an X Window server that is always running. Then, run the initialization script followed by the Siebel Analytics Scheduler executable. The following example is for the bash shell.

```
export DISPLAY="myaccount:0.0"  
sa.sh  
nqscheduler.exe &
```

You may need to modify this example for other shells or to use DISPLAY options.

Opening Analytics Job Manager (Windows Only)

The Job Manager is the interface to Siebel Analytics Scheduler. When the Job Manager is opened, the Siebel administrator can use it to connect to, configure, and start and stop Scheduler, add and manage jobs, and manage job instances.

The following procedure describes how to open the Job Manager Configuration dialog box from the Server Administration Tool.

To open the Job Manager Configuration dialog box in Windows

- 1** Open the Server Administration Tool.
- 2** Choose Manage > Jobs from the toolbar.
The Job Manager window appears.
- 3** Choose File > Configuration Options.
The Machine Name dialog box appears.
- 4** Enter the machine name where Scheduler is located, and then click OK.
The Job Manager Configuration dialog box appears.

Restoring or Changing Default Job Scheduler Values

If you change any of the values in the Job Manager Scheduler tab fields, you can restore the default values as shown in the following procedure.

To restore or change the default values in the Job Scheduler tab fields

- To restore the default values distributed with Siebel Analytics Scheduler, click Defaults.

- To revert to the most recently used settings, click Revert.

Setting Up Analytics Scheduler Jobs

A Scheduler job has properties, one or more actions to perform, and an execution schedule that determines when it runs. A Siebel administrator can add a job through the Job Manager, but jobs are most commonly added through Siebel Delivers.

NOTE: You cannot add iBot jobs using the Job Manager. The Siebel Analytics Server passes the iBots to Siebel Analytics Scheduler. You can, however, modify iBots using the Modify Job dialog box. For more information, see [“Modifying Analytics Scheduler iBots” on page 20](#).

- Use the Add Job and Modify Job dialog boxes in the Job Manager to add and modify jobs. These dialog boxes contain three types of information:
 - General job properties
 - Script area where you can specify job actions
 - Trigger area where you can specify the job trigger
- The tasks to add and modify Scheduler jobs are described in these sections:
 - [“Adding a Scheduler Job in Analytics Job Manager” on page 19](#)
 - [“Modifying a Scheduler Job in Analytics Job Manager” on page 20](#)

Adding a Scheduler Job in Analytics Job Manager

Add Scheduler jobs in the Job Manager using the following procedure.

To add a Scheduler job in the Job Manager

- 1 In the Job Manager, choose Jobs > Add New Job.
- 2 Enter the appropriate information in the dialog box.
Refer to the following sections for field descriptions:
 - [“General Scheduler Job Properties” on page 35](#)
 - [“Analytics Scheduler Job Actions” on page 37](#)
 - [“Analytics Scheduler Job Triggers” on page 38](#)

Modifying a Scheduler Job in Analytics Job Manager

Modify Scheduler jobs in the Job Manager using the following procedure.

To modify a Scheduler job in the Job Manager

- In the Job Manager, select the job you want to modify, and then select Jobs > Modify Job.

The Modify Job dialog box appears, where you can change job properties.

For field descriptions, see [“General Scheduler Job Properties” on page 35](#).

Modifying Analytics Scheduler iBots

You can modify individual iBots using the Modify Job dialog box in the Job Manager, accessible from the Server Administration Tool in Windows.

To modify an Analytics Scheduler iBot

- In the Job Manager, select the iBot you want to modify, and then choose Jobs > Modify Job.

The Modify Job dialog box displays the iBot you selected. Modify the Analytics Scheduler iBot properties, as described in the following table.

NOTE: A default value in these fields indicates that the value specified in the Siebel iBots tab of the Job Manager Configuration dialog box is active.

| iBot Property | Description |
|---------------|--|
| Web Server | Specifies the Web server this iBot contacts when it runs. Do not change this setting, because the iBot may not exist on a different Web server. This feature was added for debugging purposes only. |
| Debug Log | Determines whether debugging information is written to a log. |

Managing Analytics Scheduler Job Instances

An instance in Siebel Analytics Scheduler is a record that stores information regarding a specific execution of a Scheduler job.

To work with job instances, click the Instances tab in the lower-left corner of the Job Manager window.

- When instances are present, use the tree in the left pane to locate instances and view information about them.
 - The Instances menu is described in [Table 6 on page 26](#).
 - Instances properties are described in [Table 7 on page 27](#).
- You can perform the following tasks using Analytics Scheduler job instances:
 - [“Viewing or Refreshing Analytics Scheduler Job Instances” on page 21](#)
 - [“Canceling Analytics Scheduler Job Instances” on page 22](#)
 - [“Purging Analytics Scheduler Job Instances” on page 22](#)

Viewing or Refreshing Analytics Scheduler Job Instances

View all the Analytics Scheduler job instance information using the following methods.

NOTE: In some environments, if numerous instances have run and instances have not been purged in some time, this process may take a few seconds.

To view Analytics Scheduler job instance information

- 1 Open the Server Administration Tool in Online mode.
- 2 Choose Manage > Jobs from the toolbar.

The Job Manager window appears.
- 3 Navigate to the Instance list.
- 4 Select a particular job instance and choose Instance > View Instance.

This opens the Instance window. A description of the Instance properties is given in [Table 7 on page 27](#).

NOTE: For an iBot, the ExitCode of an instance is set to the number of successful deliveries. The count corresponds to the number of successful deliveries to devices, and there may be more than one device for each recipient of an iBot.

To refresh Analytics Scheduler job instances

- In the Instance List, choose Instances > Refresh Instance List.

Canceling Analytics Scheduler Job Instances

Registered cancel methods are described under the command ["RegisterCancelCommand Method"](#) on page 61.

To cancel an Analytics Scheduler job instance

- Select a particular job instance and choose Instances > Cancel Instance(s).

The cancel event is issued to Siebel Analytics Scheduler and the instance is marked as canceled when its registered cancel methods are called.

Purging Analytics Scheduler Job Instances

Purging a job instance involves removing it from the back-end database. Use one of the following methods (through the Instances list or through the Purge Instances window).

To purge Analytics Scheduler job instances through the Instances List

- 1 Open the Server Administration Tool in Online mode.
- 2 Choose Manage > Jobs from the toolbar.
The Job Manager window appears.
- 3 Navigate to the Instance list.
- 4 Select the instances from the Instance List.
- 5 Press Delete.

To purge Analytics Scheduler job instances through the Purge Instances window:

- 1 Open the Server Administration Tool in Online mode.
- 2 Choose Manage > Jobs from the toolbar.
The Job Manager window appears.
- 3 Navigate to the Instance list.
- 4 Click the Purge Instance(s) icon on the toolbar or choose Instances > Purge Instances to open the Purge Instances window.

You can purge instances by JobID, by UserID, or by End Time.

If you choose the End Time method, all jobs with an End Time less than or equal to the given time are purged.

- 5 Choose the purge method to use.
- 6 Click OK when you are done to return to the Job Manager window.

Analytics Scheduler Job Manager Toolbar

The Job Manager toolbar contains four menus, as described in the following sections:

- “Scheduler Job Manager File Menu” on page 24
- “Scheduler Job Manager Service Management Menu” on page 25
- “Scheduler Job Manager Jobs Menu” on page 25
- “Scheduler Job Manager Instances Menu” on page 26
- “Scheduler Job Manager Instance Properties” on page 27

Scheduler Job Manager File Menu

This topic describes the options available in the various menus on the Job Manager toolbar. [Table 3 on page 24](#) provides a description of the File menu options.

Table 3. Analytics Scheduler Job Manager File Menu Options

| Command | Description |
|----------------------------|---|
| Open Scheduler Connection | Opens the Machine Name dialog box, where you specify the name of the machine running Siebel Analytics Scheduler. If Scheduler resides on the same machine as the Server Administration Tool, you can enter <code>local</code> . |
| Close Scheduler Connection | Closes the Job Manager connection to Scheduler. |
| Configuration Options | Opens the Machine Name dialog box, where you specify the name of the machine running Scheduler. The Job Manager Configuration dialog box then appears, where you can set configuration options. |
| Exit Job Manager | Shuts down the Job Manager and returns you to the Server Administration Tool. If you exit the Job Manager while a connection to Scheduler is still open, the connection closes. |

Scheduler Job Manager Service Management Menu

Table 4 on page 25 provides a description of the Service Management menu options on the Scheduler Job Manager toolbar.

Table 4. Analytics Job Manager Service Management Menu

| Command | Description |
|---------------------|--|
| Pause Scheduling | <p>Stops all jobs from executing until scheduling is continued. Pause Scheduling is sometimes required for maintenance purposes.</p> <p>Sometimes you may need to Pause Scheduling while Siebel Analytics Scheduler is stopped. In this case, scheduling continues when Scheduler is restarted, unless you also set the option Pause When Service Starts in the Scheduler tab of the Job Manager Configuration dialog box.</p> |
| Continue Scheduling | Resumes Scheduler's regular execution. |
| Stop Service | Stops Siebel Analytics Scheduler service. |

Scheduler Job Manager Jobs Menu

Table 5 on page 25 provides a description of the Jobs menu options on the Job Manager toolbar.

Table 5. Analytics Job Manager Jobs Menu

| Command | Description |
|------------------|--|
| Add New Job | Opens the Add Job window, where you specify the properties for a new job. |
| Remove Job(s) | Removes the selected job or jobs from Siebel Analytics Scheduler. When a job is removed, all instances for that job are removed as well. |
| Modify Job | Opens the Modify Job window, where you can modify the properties for an existing job. |
| View Job | Opens the View Job window, where you can view the properties for a job. |
| Refresh Job List | Refreshes the job information displayed in the Job List in the right pane. |

Scheduler Job Manager Instances Menu

A Siebel Analytics Scheduler instance is a record that records information regarding a specific execution of a Scheduler job. [Table 6 on page 26](#) provides a description of the Instances menu options on the Job Manager toolbar. To work with Scheduler job instances, see [“Managing Analytics Scheduler Job Instances” on page 21](#).

Table 6. Analytics Job Manager Instances Menu

| Command | Description |
|--------------------|--|
| Cancel Instance(s) | Cancels the selected running instance. |
| Purge Instance(s) | Opens the Purge Instances dialog box, where you can specify the purge method to use. |
| View Instance | Displays information about the selected instance. |
| Refresh Instance | Refreshes the instance information displayed in the Instance List in the right pane. |

Scheduler Job Manager Instance Properties

Table 7 on page 27 describes the properties of Job Manager instances.

Table 7. Job Manager Instance Properties

| Field | Description |
|--------------|---|
| JobID | ID of the job associated with this instance. |
| InstanceID | Unique ID of this specific instance of the Job. |
| Status | Status of the job instance. It can be one of the following: <ul style="list-style-type: none">■ Running■ Completed■ Cancelled■ Timed Out |
| ExitCode | Exit code of the instance. Depending on the job type, this code can be interpreted differently. <ul style="list-style-type: none">■ For an iBot, the ExitCode of an instance is set to the number of successful deliveries. The count corresponds to the number of successful deliveries to devices, and there may be more than one device for each recipient of an iBot.■ For a script, it is set according to the ExitCode property in the script. The default is 0 (zero). See Table 18 on page 54. |
| ErrorMessage | Text message containing any error information of the instance. |

Analytics Job Manager Configuration Dialog Box

In Windows, you set Siebel Analytics Scheduler configuration options in the Job Manager Configuration dialog box, a feature of the Server Administration Tool. To access the Job Manager Configuration dialog box, see [“Opening Analytics Job Manager \(Windows Only\)” on page 18](#).

In UNIX, you set Siebel Analytics Scheduler configuration options in schconfig.exe, a console-based application. The configuration options in schconfig.exe are identical to those available in the Job Manager Configuration dialog box.

The Job Manager Configuration dialog box has four tabs, as shown in [Table 8 on page 28](#).

Table 8. Job Manager Configuration Dialog Box Tabs

| Tab | See Section in This Book |
|-----------|--|
| Scheduler | “Analytics Job Manager Scheduler Tab” on page 28 |
| Mail | “Analytics Job Manager Mail Tab” on page 31 |
| iBots | “Analytics Job Manager iBots Tab” on page 32 |
| Workflow | “Analytics Job Manager Workflow Tab” on page 34 |

Analytics Job Manager Scheduler Tab

The fields in the Job Manager Scheduler tab describe Siebel Analytics Scheduler access to its back-end database and its general behavior. [Table 9 on page 28](#) provides a description of each field in the Scheduler tab.

Table 9. Analytics Scheduler Job Manager Scheduler Tab Fields

| Field | Description |
|--------------------------------|--|
| Bulk Fetch Buffer Size (bytes) | Used in the database gateways. Specifies the maximum size in bytes of a bulk fetch page for retrieving data from a data source. The default value is 33,792 bytes. |
| Call Interface | The type of call the connection to the database makes. The call you choose from the drop-down list determines the application programmer interface (API) used to access the data source. The types of call interfaces in the list varies depending on the database type selected. |
| Data Source Name | Identifies the data source name (DSN) of the data source to use for the connection. The DSN must contain valid logon information for a data source. If the information is invalid, the database logon fails. For example, if the DSN of the Scheduler back-end database is ORCL8_Scheduler, you enter ORCL8_Scheduler in this field. |

Table 9. Analytics Scheduler Job Manager Scheduler Tab Fields

| Field | Description |
|--|---|
| Database Type | Siebel Analytics Scheduler back-end database type. Choose a type from the drop-down list. |
| Default Script Path | Path where user-created job scripts (not iBots) are stored. If a filename is entered in the Script field when adding or modifying a job, Scheduler examines the contents of this directory for the specified file. However, if a full path is given in the Script field, this directory is not examined. By default, this field is set to <i>\$SiebelAnalytics\Scripts\Common</i> where <i>\$SiebelAnalytics</i> is the location in which the Siebel Analytics software is installed. |
| Maximum Connections | Maximum number of database connections Siebel Analytics Scheduler can open concurrently. Specify a value of 1 or greater. When this limit is reached, the connection request waits until a connection becomes available. The default value is 5. |
| Maximum Execution Threads | Scheduler is multithreading and uses a thread pool. This field specifies the maximum number of threads that are used in Scheduler's thread pool. |
| Minimum Execution Threads | Specifies the minimum number of multiple threads in Scheduler's thread pool. |
| nQS (Siebel) Administrator Name | Specify a user with administrator's privileges in this field to allow Siebel Analytics Scheduler and Siebel Analytics Web to log onto Siebel Analytics Server as other users. Scheduler runs jobs on the Analytics Server and Analytics Web for users without storing the user's password. The user specified must be a Repository Administrator with the ability to log in as other users, and have basic access to the Web catalog (to access the shared folder). |
| nQS (Siebel) Administrator Password/Confirm Password | Password for the Siebel administrator with privileges to log on as other users. The password is encrypted in the registry. You can change it by using the Job Manager (Windows) or schconfig.exe (UNIX). |
| Password/Confirm Password | Password used to log on to the data source. For security, the password is encrypted in the registry. |
| Pause When Service Starts | Specifies that no jobs should execute when Siebel Analytics Scheduler starts. While Siebel Analytics Scheduler pauses, users can add, modify, and remove jobs. However, no jobs execute. Select Service Management > Continue Scheduling to continue with regular execution. |

Table 9. Analytics Scheduler Job Manager Scheduler Tab Fields

| Field | Description |
|----------------------------------|--|
| Purge DB every X minutes | <p>Siebel Analytics Scheduler updates the tables and flags the affected rows as deleted.</p> <p>NOTE: Scheduler does not actually issue SQL DELETE statements when jobs or instances are removed.</p> <p>After every X minutes (where X is defined as the value of this field), the actual SQL DELETE statements are issued.</p> <p>The default value is every 60 minutes.</p> |
| Purge Old instances after X days | <p>Specifies the number of days after which old job instances are deleted from the back-end database automatically. To prevent old job instances from being deleted automatically, set the value to 0 (zero).</p> <p>The default value is to delete after seven days.</p> |
| Scheduler Script Path | <p>Refers to the path where Siebel Analytics Scheduler-created job scripts are stored. In general, you should not add or remove scripts from this directory. By default, this field is set to <code>\$SiebelAnalytics\Scripts\Scheduler</code>, where <code>\$SiebelAnalytics</code> is the location in which the Siebel Analytics software is installed.</p> |
| Temporary File Path | <p>Path that specifies where temporary files are stored during the execution of Siebel Analytics Scheduler.</p> |
| Timeout (Minutes) | <p>Specifies the amount of time in minutes that a connection to the data source remains open after an operation completes.</p> <p>During this time, new operations use this connection rather than open a new one, up to the number specified for Maximum Connections. The time is reset after each completed connection request.</p> <p>Specify a value of 1 or greater. The default value is 60.</p> |
| User name | <p>The user name Siebel Analytics Scheduler uses to log on to the data source.</p> <p>The user name must have read and write permissions to the back-end database and must be the same as the table owner for databases that require fully qualified table names (for example, SIEBEL.S_NQ_JOB).</p> |

Analytics Job Manager Mail Tab

The fields in the Job Manager Mail tab are used to configure the C++ object used by Siebel Delivers to deliver SMTP mail. [Table 10 on page 31](#) provides a description of each field in the Mail tab.

Table 10. Analytics Scheduler Job Manager Mail Tab Fields

| Field | Description |
|----------------------------------|--|
| Authenticate against SMTP Server | Specifies that the SMTP Server requires authentication to send email to an address outside of its domain. When you select this field, fill in the Username and Password fields for an email user on the SMTP server. When this field is not selected, the Username and Password fields are not used. |
| From Display Name | Used in the SMTP From field as a meaningful substitution for the sender's address. The default is Siebel Delivers <iBots@defaultmailserver.com>. |
| Maximum Recipients | Prevents sending to more recipients than your mail server's limit. If the mailing list has more subscribers than the Maximum Recipient value, Siebel Delivers splits the list into smaller lists and sends an email for each list. |
| Number of Retries Upon Failure | If Siebel Delivers fails to deliver an email, it tries to resend it. Number of Retries Upon Failure sets the number of retry attempts. The default is 1. |
| Password/Confirm Password | When you select the option Authenticate against SMTP Server, these fields specify the user's password for the specified user name. |
| Sender Address | Email address on the SMTP Server used as the sender's reply-to address for all mail sent from Siebel Analytics Scheduler. The initial value is defaultuser@defaultmailserver.com, which must be changed to reflect a valid email address. |
| SMTP Port | Port number for SMTP on the given server. Most mail servers use port 25 as the mail port. The default is port 25. |
| SMTP Server | Name of the SMTP server that delivers the mail. The initial value is defaultmailserver.com, which must be changed to a valid server name to deliver mail. |
| Use Bcc Instead of To | Puts the subscribers list on the BCC line instead of the To line. Prevents subscribers from seeing the names of the others on the list and accidentally (or maliciously) replying to the entire list. |
| Username | When you select the option Authenticate against SMTP Server, Username specifies the user name for an email user on the specified SMTP server. If left blank, the email server does not authenticate any email from Siebel Delivers. |

Analytics Job Manager iBots Tab

Siebel iBots are functionally a combination of data stored in Siebel Analytics Web and on Siebel Analytics Scheduler. The fields in the Job Manager iBots tab describe the behavior of all Siebel iBots that run on a specific Siebel Analytics Scheduler. [Table 11 on page 32](#) provides a description of each field in the iBots tab.

If you change any of the values in these fields, you can restore the default values as shown in the following procedure.

Table 11. Analytics Scheduler Job Manager iBots Tab, Configuration Fields

| Field | Description |
|--|--|
| Debug Enabled | Select this to have Siebel Analytics Scheduler generate a log file for each iBot. This log file has useful logging messages when trying to diagnose a problem. |
| iBot Log Directory | <p>Siebel iBots can create log files if exceptional error conditions occur. iBot Log Directory specifies the directory where these files are saved. The directory must be accessible to the Siebel Analytics Scheduler application.</p> <p>In Windows, the default installation runs the service as a system account, which prevents Scheduler from writing to or reading from network directories.</p> <p>If you put script files on network shares, or your scripts access network shares, then Scheduler must be run as a network user.</p> |
| Max Rows Times Columns | When iBots are chained, this value governs the size of filters passed between iBots. When you pass a filter to another iBot in a chain, the Scheduler creates a union of the result sets for the Conditional Report for each personalized recipient. This report can grow very large in some cases (1000 users with 100 unique rows each with ten columns per report = 1,000,000 column values in the filter). The Analytics servers may not be able to handle such a large filter, so this field specifies the maximum number of rows*columns in the filter. The default is 10,000. |
| Minimum/Maximum Delivery Sleep Seconds | The minimum and maximum number of seconds that the iBot randomly sleeps after its connection is refused before it attempts to reconnect to the server to deliver results. |
| Minimum/Maximum Global Sleep Seconds | The minimum and maximum number of seconds that the iBot randomly sleeps after its connection is refused before it attempts to reconnect to the server to get global information about what to deliver and to whom. |
| Minimum/Maximum Request Sleep Seconds | The minimum and maximum number of seconds that the iBot randomly sleeps after its connection is refused before it attempts to reconnect to the server to issue requests. |

Table 11. Analytics Scheduler Job Manager iBots Tab, Configuration Fields

| Field | Description |
|-------------------------------|---|
| Number of Delivery Tries | After a unique request has executed, the iBot tries to deliver the results to specified devices. Specifies the number of times Siebel Analytics Scheduler attempts to connect to the server to deliver the results. |
| Number of Global Tries | A Web or mail server that has too many people logged on may reject new connections, including connections from Siebel Analytics Scheduler. To cope with such overload, an iBot retries the connection. This field sets the maximum number of tries to get global information about what to deliver and to whom before the iBot gives up. |
| Number of Request Tries | After an iBot has received the global information, it issues a series of unique requests to the server for each user. Specifies the number of times Siebel Analytics Scheduler attempts to connect to the server to issue these requests. |
| Purge files older than (days) | Specifies the number of days after which old iBot logs are deleted automatically. To prevent old logs from being deleted automatically, set the value to 0 (zero). The default value is delete after seven days. |
| SAW Machine Name | The machine name and port where the Analytics Web (SAW) server is running. The server name must be specified and optionally can be followed by a colon and port number. If you do not specify a port number, Scheduler uses the default Analytics Web server port number, which is 9710. For example, if the Analytics Web (SAW) server runs on machine1 and port 8900, the value for this field is machine1:8900.] |

Analytics Job Manager Workflow Tab

Siebel Delivers can trigger Siebel operational applications version 7.7 workflows.

NOTE: Before this triggering occurs, you must activate the EAI and Workflow component groups on your Siebel Enterprise Server, following the component activation instructions in *Siebel System Administration Guide*. For more information on setting up operational application workflows, see Chapter 4, “Integrating Siebel Delivers With Siebel Workflow.”

Part of the workflow triggering process requires using the controls on the Job Manager Workflow tab. Table 12 on page 34 provides a description of each field in the Workflow tab.

Table 12. Analytics Scheduler Job Manager Workflow Tab, Configuration Fields

| Field | Description |
|----------|--|
| Server | The name of your Siebel Enterprise Server. Siebel Analytics Scheduler uses the http connector of the server to trigger the workflow. Example: http://localhost |
| System | The name of the Siebel EAI system in your environment. Example: eai_enu |
| Service | The name of the http service as defined in the eai.cfg file. Example: ANALYTICS |
| User | The user name Siebel Analytics Scheduler uses to log on to the Siebel Enterprise Server. |
| Password | Password for the user account. |

General Scheduler Job Properties

Use the fields in the Add Job or Modify Job dialog box to configure or modify the general properties for a job. [Table 13 on page 35](#) provides a description of the general job properties available in the Add Job and Modify Job dialog boxes.

Table 13. General Siebel Analytics Scheduler Job Properties

| Field | Description |
|------------------------|--|
| JobID | Display-only field that becomes visible when choosing Jobs > View Job or Jobs > Modify Job. This field also appears in the Job List display in the right pane of the Job Manager window. After you add a job, this field displays an integer that uniquely identifies the job. The JobID cannot be changed. |
| Name | Short, descriptive name for the job. This field also appears in the Job List display in the right pane of the Job Manager window. |
| Description | Brief description of the job that describes its actions to end users. This field also appears in the Job List display in the right pane of the Job Manager window. |
| UserID | Required for all jobs. For jobs that communicate with Siebel Analytics Server or Siebel Analytics Web, UserID needs to be a valid Siebel Analytics user ID. This field also appears in the Job List display in the right pane of the Job Manager window. When this job runs, Siebel Analytics Scheduler executes it on behalf of the user ID specified in this field. |
| Maximum Run Time MS | Specifies the maximum number of milliseconds this job should run before it is canceled forcibly. If a job exceeds its run time, it fails with a time-out reason code. To prevent the job from timing out, set this field to 0 (zero). |
| Last Run Time | Display-only field that shows the last time this job began execution. This field also appears in the Job List display in the right pane of the Job Manager window. |
| Next Run Time | Display-only field that shows recurrent jobs and the next time this job executes. The trigger is used to determine this value. |
| Running Instance Count | Display-only field that shows the number of currently running instances of this job. |
| Delete Job When Done | When you select this option, Siebel Analytics Scheduler deletes the job after its last scheduled execution as defined by its trigger. When there is no next run time, the job is done. When a job is deleted, all instances are deleted as well. For most jobs, you should not select this option, because you can delete a job manually through the Job Manager. |

Table 13. General Siebel Analytics Scheduler Job Properties

| Field | Description |
|-----------------------------------|--|
| Disabled | When you select this option, the job script does not execute when its trigger expires. However, the next run time is still updated according to the trigger settings. The Disabled field is useful when testing or debugging a new job because a Siebel administrator can quickly disable a job without losing all information. |
| Execute When Missed | If you select this option while the Scheduler is stopped (either all scheduling pauses or the Scheduler application stops), and if the job's next run time was missed, the job runs after Siebel Analytics Scheduler comes back up. If you do not select this option, the job executes at the its next run time, defined by its trigger. |
| Delete Script When Job is Removed | If you select this option, when a job is removed its associated job script is also removed. If many jobs reference the same job script, this option should not be set. |

Analytics Scheduler Job Actions

Use the fields in the Script area of the Add Job or Modify Job dialog box to define the actions a job performs when it executes. [Table 14 on page 37](#) provides a description of the job action properties available in the Add Job and Modify Job dialog boxes.

Table 14. Analytics Scheduler Job Manager Job Action Fields

| Field | Description |
|---------------------------|---|
| Script Type | Currently, Siebel Analytics Scheduler supports two types of scripts—VBScript and JScript. Set this field according to the type of script referenced by the Script field. |
| Script Contains File Name | In the Job Manager, you can enter either a file name or the actual contents of a script in the Script field. If the Script field contains a file name, select this option. |
| Script | This value is either a reference to a job script file or the contents of a job script itself. If it is a reference, enter a file name in this field, such as TestConnect.js. If no path is given, Siebel Analytics Scheduler examines the directory referred to in the Default Script Path configuration value in the Scheduler tab of the Job Manager Configuration dialog. If a path is given, the path must be accessible by the Siebel Analytics Scheduler application. |
| Parameters | Field values are passed to the job script through the Parameters array. Enter one parameter per line. For example: <pre>c:\siebel\data\scheduler cli_snowflake SELECT Lastname FROM Employee</pre> <p>By default, each parameter must be less than 255 characters in length. You can change this restriction by modifying the varchar length in the back-end database. In the NQSJOBPARAMETERS table, the column is JobParameter. For more information about this table, see <i>Siebel Analytics Installation and Configuration Guide</i>.</p> |

Analytics Scheduler Job Triggers

A job trigger determines when and how often a job executes. Use the fields in the Trigger area of the Add Job or Modify Job dialog box to define the actions a job performs when it executes.

There are two types of triggers—*single-run* triggers and *recurrent* triggers.

Scheduler Job Single-Run Triggers

Use the Trigger Type drop-down list to choose the trigger type. Single-run triggers perform the action once. There are two single-run triggers:

- **Run Now.** This trigger specifies that the job runs immediately. It executes only one time.
- **Run Once.** Jobs of this trigger type execute at the date and time specified in the Begin Date and Start Time fields, which become active when you choose Run Once. An error occurs if the given time is in the past. If you select the option Set Start Time To Now, this trigger is equivalent to the Run Now trigger.

Scheduler Job Manager Recurrent Triggers

All recurrent triggers specify that the job execute over a period of time at given intervals. [Table 15 on page 38](#) describes the Job Scheduler fields used by recurrent triggers.

Table 15. Analytics Scheduler Job Manager Recurrent Trigger Fields

| Field | Description |
|------------|--|
| Begin Date | Specifies the date when the first recurrent interval runs. The recurrent interval is defined as the time between Start Time and End Time. This field is hidden if you select the option Set Start Time to Now. |
| End Date | Specifies the date when the last recurrent interval is run. Becomes active when the option Has End Date is set. If no end date is set, the job runs forever. |
| Start Time | Specifies the lower bounds of the recurrent interval. The job's first execution for a day occurs at the time specified in this value. |
| End Time | Specifies the upper bounds of the recurrent interval. The job's last execution for a given day occurs at or before the time specified in this value. If this value is less than the Start Time value, the interval spans midnight of the given day. For example, a trigger with a start time of 11:00 P.M. and an End Time of 2:00 A.M. starts its execution on the date specified in Begin Date at 11:00 P.M. and continues until 2:00 A.M. on the following day. |

Table 15. Analytics Scheduler Job Manager Recurrent Trigger Fields

| Field | Description |
|------------------------------|---|
| Has End Date | <p>If you select this option, specify an End Date. If you do not select this option, the job stays scheduled.</p> <p>NOTE: The schedule is perpetual, the job instance is not. If you restart the Scheduler, the next runtime will be set as dictated by the job schedule. If an instance is running while you shut down the Scheduler, it will be canceled.</p> |
| Set Start Time To Now | If you select this option, the Begin Date and Start Time fields are ignored and their values are populated with Siebel Analytics Scheduler's current date and time. |
| Interval in Minutes | Specifies the number of minutes between subsequent executions of a job during the recurrent interval. A job starts execution promptly at its Start Time and executes again every n minutes, where n is the value of this field. |
| Maximum Concurrent Instances | If a job executes every n minutes (from the Interval in Minutes field), a long-running job may have overlapping executions. Use this field to set the number of concurrent running instances. For an unlimited number of concurrent instances, set this value to zero. |

Scheduler Job Manager Recurrent Trigger Types

The recurrent trigger types available from the Trigger Type drop-down list are described in [Table 16 on page 40](#). The fields described pertain to all recurrent triggers. Depending on the trigger type you choose, additional options become active. The examples in the table illustrate how these additional options can be used.

Table 16. Scheduler Job Manager Recurrent Trigger Types

| Trigger Type | Description and Example |
|--------------|--|
| Daily | <p>Runs a job every day or every few days. The Days Interval field specifies the number of days between each subsequent recurrent interval.</p> <p>For example:</p> <p>To run a job every hour between 8:00 A.M. and 5:00 P.M. starting on January 1, 2005 and ending on January 15, 2005, set the Begin Date to 1/1/05, the Start Time to 8:00 A.M., and the End Time to 5:00 P.M. Set the Has End Date flag, the End Date to 1/15/05, the Interval in Minutes to 60, and the Days Interval to 1.</p> <p>To run a job every five minutes forever, set the Begin Date to the desired date, the Start Time to 12:00 P.M., the end time to 11:59 A.M., the Interval In Minutes to 5, and the Days Interval to 1.</p> |
| Weekly | <p>Runs a job on specified days of the week. The Weeks Interval specifies the number of weeks between each execution. The Days of the Week field specifies on which days the execution occurs.</p> <p>For example:</p> <p>To run a job at noon every other week on Mondays, Wednesdays, and Fridays, set the Begin Date to the desired date, the Start Time and End Time to 12:00 P.M., the Interval in Minutes to 1, the Weeks Interval to 2, and the Days of the Week to Monday, Wednesday, and Friday.</p> |

Table 16. Scheduler Job Manager Recurrent Trigger Types

| Trigger Type | Description and Example |
|-----------------|--|
| Monthly by Date | <p>Runs a job on specific days of the month. The Months field specifies in which months this job executes. The Days field specifies which days of those months. If the given day does not exist for a given month, that day is ignored.</p> <p>For example:</p> <p>To run a job at 5:00 P.M. on the 1st and 15th of January, February, and March, set the Begin Date to January 1, the Start Time and End Time to 5:00 P.M., the Interval in Minutes to 1, the Months to January, February, and March, and the Days to 1 and 15.</p> <p>To run a job at 2:00 A.M. on every leap day (February 29th), set the Begin Date to January 1, the Start Time and End Time to 2:00 A.M., the Interval in Minutes to 1, the Months to February, and the Days to 29.</p> |
| Monthly by DOW | <p>Runs a job on specific occurrences of specified days of the week during given months. The Months field specifies which months this job executes. The Days of the Week field specifies which days of the week the job executes during those months. The Occurrence field specifies which of those days to execute. The occurrence can be any or all of First, Second, Third, Fourth, and Last. The Last value specifies that either the fourth or fifth occurrence of a given day is used, depending on whether there are four or five occurrences during that month.</p> <p>For example:</p> <p>To run a job on the first and third Fridays of December every hour between the hours of 4:00 A.M. and 8:00 P.M., set the Begin Date to the desired date, the Start Time to 4:00 A.M., the End Time to 8:00 P.M., the Interval in Minutes to 60, the Months to December, the Days of the Week to Friday, and the Occurrence to the First and the Third.</p> <p>To run a job at 3:00 A.M. every time that Daylight Saving Time switches over to Standard Time, set the Begin Date to the desired date, the Start Time and End Time to 3:00 A.M., the Months to October, the Days of the Week to Sunday, and the Occurrence to Last.</p> |

4

Integrating Siebel Delivers With Siebel Workflow

Chapter 3, “Configuring Siebel Analytics Scheduler With Job Manager,” describes how to use the Job Manager to schedule jobs. This chapter describes how you can also set up workflows and integrate them with Siebel Delivers, which supports native integration with Siebel Workflow version 7.7 so that iBots can trigger workflow tasks.

NOTE: This functionality requires that the workflow tasks be already set up in the Siebel operational applications. For more information, see *Siebel Business Process Designer Administration Guide*, version 7.7.

When integrating iBots with Siebel Workflow, a workflow is invoked for each row of an Analytics results set. The row and other nondynamic properties are passed as a Siebel Property Set that is particular to each workflow. Using the Advanced tab of Siebel Delivers, you can configure the property set by manually typing in the name for a workflow and matching it with a column.

When doing this configuration, you *must* follow the XML Converter Escape Sequences—for example, if a property name in the workflow is “AAA BBB” type in “AAA_spcBBB.” See *XML Reference: Siebel Enterprise Application Integration Volume V* for details.

Process of Configuring Siebel Delivers to Launch Siebel Workflow

The process of configuring Siebel Delivers to launch workflow applications requires the following tasks:

- [“Configuring the Siebel Enterprise Server for Workflow Integration with Siebel Delivers” on page 44.](#)

This configuration provides Siebel Analytics Scheduler with the location of the Siebel Enterprise Server.

- [“Configuring Siebel Analytics Scheduler to Trigger Workflows” on page 45.](#)

- [“Third-Party Triggering of Siebel Analytics Scheduler Jobs” on page 45.](#)

- Creating workflow triggers in the iBots.

By default, only Siebel Analytics Web administrators have the necessary privileges to set up iBots to trigger workflows. For more information, see *Siebel Analytics Web Online Help*.

NOTE: Before Siebel Delivers can trigger workflows, you must activate the EAI and Workflow component groups on your Siebel Enterprise Server, following the component activation instructions in *Siebel System Administration Guide*.

Configuring the Siebel Enterprise Server for Workflow Integration with Siebel Delivers

This task is a step in the [“Process of Configuring Siebel Delivers to Launch Siebel Workflow” on page 44.](#)

To configure the Siebel Enterprise Server for Workflow integration with Siebel Delivers

- 1 On the Siebel Enterprise Server, create a named subsystem specific to Siebel Analytics using the SrvMgr command line interface with the following command.

```
create named subsystem <subsystem_name> for subsystem
EAITransportDataHandlingSubsys with ConverterService="XML
Converter",DispatchService="workflow Process
Manager",DispatchMethod="RunProcess"
```

Replace <subsystem_name> with a name you create, such as AnalyticsWFDISPATCH.

- 2 Open the eai.cfg configuration file inside the SiebSrvr directory and add the following line under the Http Services section.

```
[Http Services]
SiebelQuery = SiebelQueryDispatch
SiebelUpsert = SiebelUpsertDispatch
SiebelExecute = SiebelExecutedDispatch
ANALYTICS = <subsystem_name>
```

- 3 Restart the Siebel Server.

Configuring Siebel Analytics Scheduler to Trigger Workflows

This task is a step in the [“Process of Configuring Siebel Delivers to Launch Siebel Workflow”](#) on page 44.

To configure Siebel Analytics Scheduler to trigger workflows

- 1 In the Siebel Analytics Server Administration Tool, open the Job Manager.
- 2 Choose File > Configuration Options.
The Job Manager Configuration window opens.
- 3 Click the Workflow tab.
- 4 Fill out the fields using the information in [“Analytics Job Manager Workflow Tab”](#) on page 34.

Third-Party Triggering of Siebel Analytics Scheduler Jobs

Third-party applications and scripts can launch a Siebel Analytics Scheduler job from the command line. They can also change the Job Parameters for a single instance. This change simulates third-party-triggered iBots. The interface for the command line is:

```
saschinvoke.exe -u <Admin Name>/<Admin Password> (-j <job id> |  
-i <iBot path>) [-m <machine name>[:<port>]] ([-r <replace parameter filename>] |  
[-a <append parameter filename>])
```

The required parameters <Admin Name> and <Admin Password> are the same as those configured for the Siebel Analytics Scheduler where you invoke the job. You can invoke the job either by the job ID or by the iBot path. Optionally, you can specify a machine and port for Siebel Analytics Scheduler. If this is omitted, the invoker uses localhost and 9705 respectively.

The invoker also takes an optional job parameter file. Depending on the mode you choose, the parameters configured in Siebel Analytics Scheduler are changed in one of the following ways by the options in the job parameter file:

- Replace the existing parameters, using the following syntax:
[-r <replace parameter filename>]
- Append to the existing parameters, using the following syntax:
[-a <append parameter filename>]

If you use a parameter file, follow these rules:

- Use only one parameter per line.
- Do not ignore white space because it may be custom script-dependent.

Using Replace Mode

In replace mode, the file can specify to leave some parameters as they are in Siebel Analytics Scheduler. To specify this for a specific line, enter `$SCH_DEFAULT$` on the line. This text string acts as a variable and replaces the `$SCH_DEFAULT$` text with the text from the original Siebel Analytics Scheduler parameter.

For example, if the original parameter is `hello`, the line

```
$SCH_DEFAULT$ world, $SCH_DEFAULT$ again
```

is changed to

```
hello world, hello again
```

If you use the `saschinvoke` command, make sure the job parameters are correct. A job may not properly execute if invalid parameters are passed to it.

NOTE: The `saschinvoke` command does not test the parameters for correctness.

5

Configuring Siebel Analytics Scheduler Job Scripts

Siebel Analytics Scheduler includes a Script object that encapsulates a running script. The Script object represents a script and exposes the properties and methods of a script. You can access its methods and properties directly because its name is implied. For example, to access the JobID property, you can specify JobID, not Script.JobID.

Siebel Analytics Scheduler supports two types of scripting languages that can be used for job scripts: VBScript and JScript.

NOTE: Scripting (JScript, VBScript) for iBots and scripts defined by the Scheduler Job Manager are supported only under Windows platforms and are not supported under UNIX.

Any VBScript or JScript can be used as a job script. Siebel Analytics Scheduler has language extensions to provide these scripts with additional functionality when defined within the context of Siebel Analytics.

The following topics contain detailed information on working with job scripts:

- [“Examples of Analytics Scheduler Scripts” on page 48](#)
- [“Siebel Analytics Scheduler Read-Only Script Object Properties” on page 53](#)
- [“Siebel Analytics Scheduler Read/Write Script Object Properties” on page 54](#)
- [“Siebel Analytics Scheduler Script-Defined Constants” on page 55](#)
- [“Analytics Scheduler Script Object Methods and Events” on page 58](#)

Examples of Analytics Scheduler Scripts

The following sections provide examples of how to set up scripts for Siebel Analytics Scheduler, including scripts for Siebel iBots:

■ ["Job Manager Script for Cache Clearance" on page 48](#)

This is a generic Scheduler script.

■ ["Script for iBot Chaining" on page 50](#)

This is a custom iBot script.

NOTE: Using Scheduler Job Manager to schedule jobs is described in [Chapter 3, "Configuring Siebel Analytics Scheduler With Job Manager."](#)

Job Manager Script for Cache Clearance

Siebel Analytics Scheduler can be used for general purpose scripts that extend the functionality of Siebel Analytics.

The script *purgeSASCache.js* is used to periodically purge all of the cache from the Siebel Analytics Server:

```
////////////////////////////////////
// purgeSASCache.js
//
// Purges the cache on SAS.
// Parameter(0) - The user name to pass in to NQCMD.
// Parameter(1) - The password for the aforementioned user.
////////////////////////////////////

// The full path to nqcmd.exe
var nqCmd = "D:\\SiebelAnalytics\\Bin\\nqcmd.exe";

// The data source name
var dsn = "Analytics Web";

// The user to execute the queries
var user = Parameter(0);

// The password of the aforementioned user
var pswd = Parameter(1);

// The ODBC procedure call for purging the cache
var sqlStatement = "{call SAPurgeAllCache()}";

////////////////////////////////////
// Returns a string from the file name
////////////////////////////////////

function GetOutput(fso, fileName)
{
    var outStream = fso.OpenTextFile(fileName, 1);
    var output = outStream.ReadAll();
}
```



```

        outStream.Close();
        return output;
    }

    //////////////////////////////////////
    // Get wshShell object and run nqCmd. Capture the output
    // so that we can handle erroneous conditions.

    var wshShell = new ActiveXObject("WScript.Shell");

    // Create a temp file to input the SQL statement.

    var fso = new ActiveXObject("Scripting.FileSystemObject");

    var tempFolder = fso.GetSpecialFolder(2);
    var tempInFileName = fso.GetTempName();
    var tempOutFileName = fso.GetTempName();
    tempInFileName = tempFolder + "\\\" + tempInFileName;
    tempOutFileName = tempFolder + "\\\" + tempOutFileName;
    var tempInFile = fso.CreateTextFile(tempInFileName, true);
    tempInFile.WriteLine(sqlStatement);
    tempInFile.Close();

    try
    {
        // execute
        var dosCmd = nqCmd + " -d \"" + dsn + "\" -u \"" + user
            + "\" -p \"" + pswd + "\" -s \"" + tempInFileName + "\" +
            " -o \"" + tempOutFileName + "\"";

        wshShell.Run(dosCmd, 0, true);

        var output = GetOutput(fso, tempOutFileName);

        // Remove the temp files
        fso.DeleteFile(tempInFileName);
        if (fso.FileExists(tempOutFileName)) {
            fso.DeleteFile(tempOutFileName);
        }

        // Check the output for any errors
        if (output.indexOf("Processed: 1 queries") == -1) {
            ExitCode = -1;
            Message = output;
        }

        else if (output.indexOf("Encountered") != -1) {
            ExitCode = -2;
            Message = output;
        }
        else {
            ExitCode = 0;
        }
    } catch (e) {

```

```

    if (fso.FileExists(tempInFileName)) {
        fso.DeleteFile(tempInFileName);
    }

    if (fso.FileExists(tempOutFileName)) {
        fso.DeleteFile(tempOutFileName);
    }
    throw e;
}

```

Configuring Custom Script Properties for Analytics Scheduler

Use the following procedure to modify Analytics Scheduler script properties.

NOTE: The script has to exist on the Analytics Scheduler machine before you can configure the properties.

To configure custom Analytics Scheduler script properties

- 1 Set the custom properties according to the discussion in ["Analytics Scheduler Job Actions" on page 37](#).

For example, for the script `purgeSASCACHE.js`, use the values shown in the following table:

| Field | Value or Setting |
|---------------------------|-------------------------------|
| Script Type | JScript |
| Script Contains File Name | Checked |
| Script | <code>purgeSASCACHE.js</code> |
| Parameters: | |
| Parameter(0): User | Administrator |
| Parameter(1): Password | SADMIN |

- 2 Click OK.

Script for iBot Chaining

The following script copies the results of an iBot to another directory. It copies the temporary file containing the results of the Conditional Request to the iBot log directory. The JobID, InstanceID, and UserID are used in the file name to guarantee that the result sets will not overwrite each other with each execution of the iBot, for each user, or for other iBots that share this script.

The example script uses the following setup:

- The iBot log directory on the Scheduler machine is \SiebelAnalytics\Log\iBots
- The iBot is run as Administrator
- The Custom Script Properties are set according to the table in [“Configuring Custom Script Properties for Siebel Delivers” on page 51](#)
- The job ID is 101 (assigned by the Scheduler)
- The instance ID is 1208 (assigned by the Scheduler)

The output of this example, after the iBot is run, is a file on the Scheduler machine called D:\SiebelAnalytics\Log\iBots\101-1208-Administrator-iBotScript1.PDF. This file contains the results of the Conditional Request in PDF format.

For all scripts from chained iBots, the full path name to the temporary file is specified in Parameter(0).

```

////////////////////////////////////////
//
// createResultFile.js
//
// Copies the results in the temporary file to a new file name
//
// Parameter(0) = iBot Result File Path
// Parameter(1) = Last Part of Output File Name (no path)
//
////////////////////////////////////////

var FSO = new ActiveXObject("Scripting.FileSystemObject");

var fileName = GetConfigurationValue("Log Dir", "iBots") +
  "\\\" + JobID + \"-\" + InstanceID + \"-\" + UserID + \"-\" +
  Parameter(1);

var fooFile = FSO.CopyFile(Parameter(0), fileName, true);

```

Configuring Custom Script Properties for Siebel Delivers

You set the script properties on the Advanced Tab of an iBot under Siebel Analytics Web. See the *Siebel Analytics Web Administration Guide* for details. See also [“Modifying Analytics Scheduler iBots” on page 20](#).

NOTE: The script has to exist on the Analytics Scheduler machine before you can create the iBot. Create the custom script, and then create the iBot to call the script.

To configure custom Siebel Delivers script properties

- 1 In the Siebel Analytics Web screen, select the iBot, and click the Advanced tab.

- 2** In the Custom Script Properties window, enter the properties.

For example, for the script shown in “Script for iBot Chaining,” you use the values shown in the following table:

| Field | Value or Setting |
|------------------|--|
| Filename | createResultfile.js |
| Type | JScript |
| Results | Pass results of Conditional Request as a script Pass results as PDF |
| Other Parameters | iBotScript1.PDF |

- 3** Click OK.

- 4** Run the iBot.

This script is run after the Conditional Request of the iBot is run.

Siebel Analytics Scheduler Read-Only Script Object Properties

Siebel Analytics Scheduler supports the read-only script object properties shown in [Table 17 on page 53](#).

Table 17. Siebel Analytics Scheduler Read-Only Script Object Properties

| Object Property | Description | Return Value | Syntax |
|-------------------|--|---------------|---|
| JobID | Returns the job identification number associated with this instance. | long | |
| InstanceID | Returns the instance identification number associated with this instance. | | |
| ParameterCount | Returns the number of job parameters associated with the job script. | long | |
| Parameter (index) | Returns a specific parameter associated with the script. Parameter (index) returns an error if the given index is less than zero or greater than ParameterCountminus 1. | string | Parameter(index) Index is the zero-based index of the parameter. |
| Script | Returns the Script object that represents the current script. This object implements the COM IDispatch interface and can be passed as arguments to methods of other objects that exist on the system. Implementing the COM IDispatch is particularly useful when handling cancel events to a running instance. See "RegisterCancelCommand Method" on page 61 . | script object | |
| UserID | Returns the user identification number associated with the instance. | string | |

Siebel Analytics Scheduler Read/Write Script Object Properties

Siebel Analytics Scheduler supports the read/write script object properties shown in [Table 18 on page 54](#).

Table 18. Siebel Analytics Scheduler Read/Write Script Object Properties

| Object Property | Description | Return Value |
|-----------------|--|--------------|
| ErrorMessage | <p>Sets or returns the Message property of the running instance. The ErrorMessage property can be used to convey meaningful error information. Setting this value changes the Error Message field of a Job Instance. without stopping execution of the current Job Script.</p> <p>In Windows environments, if the JScript throw() method is called and this property has been set, the value is appended to the message description in the JScript or VBScript Error object.</p> <p>COM objects that implement the IDispatch interface can be accessed from within Job Scripts. If any method fails and properly provides error information through the SetErrorInfo() method, that information is contained in the Message field of the Job Instance. If the Message property is set before the COM object error is generated, then that string value is appended to the COM object error information.</p> <p>NOTE: If this property is set to a nonempty string value, the job instance has a Failed status and its Message property is set. Resetting this value to the empty string ("") clears the error messages.</p> | string |
| ExitCode | <p>Sets or returns the Exit Code property associated with the instance.</p> <p>The default is 0 (zero).</p> | long |

Siebel Analytics Scheduler Script-Defined Constants

Siebel Analytics Scheduler supports the following script-defined constants. These constants are used by the methods to schedule new jobs.

- [“DayEnum Constants” on page 55](#)
- [“DayOfWeekEnum Constants” on page 55](#)
- [“JobFlagsEnum Constants” on page 56](#)
- [“MonthEnum Constants” on page 57](#)
- [“OccurrenceEnum Constants” on page 57](#)

DayEnum Constants

The DayEnum values are used with the scheduling functions to identify days in a month, from Day 1 to Day 31. [Table 19 on page 55](#) describes DayEnum values.

Table 19. DayEnum Constant Values

| Value | Description |
|---------|-------------|
| nqDay1 | Day 1 |
| nqDay2 | Day 2 |
| nqDay3 | Day 3 |
| ... | ... |
| nqDay31 | Day 31 |

DayOfWeekEnum Constants

The DayOfWeekEnum values are used with the scheduling functions to identify days in a week. [Table 20 on page 55](#) describes DayOfWeekEnum values.

Table 20. DayOfWeekEnum Constant Values

| Value | Description |
|-------------|-------------|
| nqSunday | Sunday |
| nqMonday | Monday |
| nqTuesday | Tuesday |
| nqWednesday | Wednesday |

Table 20. DayOfWeekEnum Constant Values

| Value | Description |
|------------|-------------|
| nqThursday | Thursday |
| nqFriday | Friday |
| nqSaturday | Saturday |

JobFlagsEnum Constants

The JobFlagsEnum values are used with the scheduling methods of the Script object to control how a job behaves. [Table 21 on page 56](#) describes JobFlagsEnum values.

Table 21. JobFlagsEnum Constant Values

| Value | Description |
|---------------------------|---|
| nqJobNoFlags | Job has no special behavior. |
| nqJobDeleteWhenDone | Job is deleted when there are no more scheduled run times. |
| nqJobDisabled | Job is disabled. This is useful for preventing a job from running at the scheduled time or times. |
| nqJobHasEndDate | Job has a valid end date. |
| nqJobExecuteWhenMissed | If for some reason Siebel Analytics Scheduler is down when the job is supposed to start, this flag indicates that the job should run when Siebel Analytics Scheduler starts up again. |
| nqJobDeleteScriptWhenDone | When a job is removed and this flag is set, the script associated with the job is deleted. This is useful only in conjunction with the nqJobScriptContainsPath flag. |
| nqJobScriptContainsPath | This flag indicates that the script associated with the job contains a path to a file containing the actual script code. |
| nqJobStartNow | When this flag is set, the begin date and start time are ignored. Instead, these fields get set to the current time of Siebel Analytics Scheduler. |

MonthEnum Constants

The MonthEnum values are used with the scheduling functions to identify months. [Table 22 on page 57](#) describes MonthEnum values.

Table 22. MonthEnum Constant Values

| Value | Description |
|-------------|-------------|
| nqJanuary | January |
| nqFebruary | February |
| nqMarch | March |
| nqApril | April |
| nqMay | May |
| nqJune | June |
| nqJuly | July |
| nqAugust | August |
| nqSeptember | September |
| nqOctober | October |
| nqNovember | November |
| nqDecember | December |

OccurrenceEnum Constants

The OccurrenceEnum values are used with the scheduling functions to identify the occurrence of a given day. [Table 23 on page 57](#) describes OccurrenceEnum values.

Table 23. OccurrenceEnum Constant Values

| Value | Description |
|----------|-------------------|
| nqFirst | First occurrence |
| nqSecond | Second occurrence |
| nqThird | Third occurrence |
| nqFourth | Fourth occurrence |
| nqLast | Last occurrence |

Analytics Scheduler Script Object Methods and Events

The Siebel Analytics Scheduler script object methods and events, with their usage, syntax, and arguments, are the following:

- ["CreateArray Method" on page 59](#)
- ["DeregisterCancelCommand Method" on page 60](#)
- ["GetConfigurationValue Method" on page 60](#)
- ["GetTempFileName Method" on page 61](#)
- ["LaunchProcess Method" on page 61](#)
- ["RegisterCancelCommand Method" on page 61](#)
- ["ScheduleJobDaily Method" on page 62](#)
- ["ScheduleJobMonthlyDate Method" on page 63](#)
- ["ScheduleJobMonthlyDOW Method" on page 64](#)
- ["ScheduleJobNow Method" on page 65](#)
- ["ScheduleJobOnce Method" on page 66](#)
- ["ScheduleJobWeekly Method" on page 67](#)
- ["OnError Event" on page 68](#)

CreateArray Method

Creates an Array object.

Usage: This method is provided only for JScript because local JScript Array objects cannot be passed directly to the Script methods. This method is called to create an array object and pass the array object to Script methods that accept an array as arguments.

Syntax 1: Set array = CreateArray ()

Syntax 2: Set array = CreateArray (size)

Syntax 3: Set array = CreateArray (element 0, element 1, ..., element *n*)

The different syntax versions create arrays as follows:

- Syntax 1 creates an array of size 0 (zero).
- Syntax 2 creates an array with the specified size.
- Syntax 3 creates an array filled with the specified elements.

Example

```
var i;  
var array1= CreateArray(2);  
for (i = 0; i < array1.Size; i++)  
{  
    array1(i) = i;  
}  
  
    array1.Resize(4);  
for (i = 2; i < array1.Size; i++)  
{  
    array1(i) = i;  
}  
  
var array2 = CreateArray(0, 1, 2,3);  
for (i = 0; i < array2.Size; i++)  
{  
    if (array1(i) != array2(i))  
        break;  
}
```

Arguments: See [Table 24 on page 60](#) for CreateArray method arguments.

Return Value: Returns an Array object.

Table 24. CreateArray Method Arguments

| Argument | Description |
|-----------------------|--|
| size | A long value that specifies the initial size of the array. |
| element0 ... elementn | The values to place in the array. This creates an array with the lower and upper bounds of 0 (zero) and n, respectively. |

DeregisterCancelCommand Method

Deregisters a previously registered cancel method.

Usage: Call this method to deregister the most recently registered cancel method after a long operation has completed successfully. You do not need to call this method if the script was canceled.

Syntax: DeregisterCancelCommand

GetConfigurationValue Method

Returns the value in Siebel Analytics Scheduler configuration relative to Siebel Analytics Scheduler's root registry entry.

Usage: GetConfigurationValue() returns the string value for a registry setting relative to Siebel Analytics Scheduler. The configKey and subkeyPath strings must be identical to those in the registry.

Syntax: value = GetConfigurationValue(*configKey* [, *subkeyPath*])

Arguments: See [Table 25 on page 60](#) for GetConfigurationValue method arguments.

Return Value: Returns a string value.

Table 25. GetConfigurationValue Method Arguments

| Argument | Description |
|------------|--|
| configKey | A string that specifies the registry key name to return. |
| subkeyPath | (Optional) A string value that specifies the registry path below Siebel Analytics Scheduler's root path. |

GetTempFileName Method

Returns a temporary file name.

Usage: GetTempFileName() does not create a file. It only provides a temporary file name that can be used to create a file. Files created in job scripts are not deleted automatically when the script terminates.

Syntax: tfname = GetTempFileName()

Return Value: Returns a string value.

LaunchProcess Method

Executes a command line in a new process.

Usage: Call this method to execute a command line in a new process. If wait is set to True, this method returns the exit code returned by the process.

Syntax: exitcode = LaunchProcess (commandLine [, wait, terminateOnCancel])

Arguments: See [Table 26 on page 61](#) for LaunchProcess method arguments.

Return Value: Returns a long value.

Table 26. LaunchProcess Method Arguments

| Argument | Description |
|-------------------|---|
| commandLine | A string that specifies the command line to execute. |
| wait | (Optional) A Boolean value that specifies whether the method should wait for the process to terminate. The default is True. |
| terminateOnCancel | (Optional) A Boolean value that specifies whether the method should terminate the process when the script is canceled. The default is True. |

RegisterCancelCommand Method

Registers a method to be called when the script is canceled.

Usage: Occasionally, an object's method takes a long time to complete. If the job is canceled before the call returns, the script engine still must wait until the call returns. This could potentially take hours and limit resources. This method solves the problem by registering a method that is asynchronously called by the script engine if the script gets canceled.

Cancel methods should be registered before calling the method that executes a long operation. When the method returns, the cancel method should be deregistered by calling DeregisterCancelCommand().

A good practice is to hide implementation details of a COM object from the caller, having the COM object itself handle all registration and deregistration of cancel commands. Pass an instance of the Script object to the COM object, then call the RegisterCancelCommand() and DeregisterCancelCommand() methods because the Script object implements the IDispatch interface.

Syntax: RegisterCancelCommand source, methodName [, arguments]...

Arguments: See [Table 27 on page 62](#) for RegisterCancelCommand method arguments.

Table 27. RegisterCancelCommand Method Arguments

| Argument | Description |
|------------|--|
| source | An object whose method is being registered. |
| methodName | A string that specifies the method name. |
| arguments | Optional arguments to be passed into the method. |

ScheduleJobDaily Method

Schedules a new job with a Daily trigger.

Syntax: ScheduleJobDaily name, description, scriptType, script, startDate, startTime, endTime, minutesInterval, daysInterval [, parameters, flags, maxRunTimeMS, maxConcurrentInstances, endDate]

Arguments: See [Table 28 on page 62](#) for ScheduleJobDaily method arguments.

Table 28. ScheduleJobDaily Method Arguments

| Argument | Description |
|-----------------|---|
| name | A string that specifies the name of the job. |
| description | A string that specifies the description of the job. |
| scriptType | A string that specifies the script type associated with the job (either VBScript or JScript). |
| script | A String that specifies the script code or path (if the nqJobScriptContainsPath flag is set) associated with the job. |
| startDate | A date value that specifies the date the job is activated. |
| startTime | A date value that specifies the time the job is activated. |
| endTime | A date value that specifies the time the job is deactivated. |
| minutesInterval | A long value that specifies the number of minutes between consecutive job executions. |
| daysInterval | An integer value that specifies the number of days between job invocations. |

Table 28. ScheduleJobDaily Method Arguments

| Argument | Description |
|------------------------|--|
| parameters | (Optional) A string array of parameter values passed to the script. The default is an empty array. |
| flags | (Optional) A long value that specifies the flags associated with the job. For valid settings, see "JobFlagsEnum Constants" on page 56 . The default is nqJobNoFlags. |
| maxRunTimeMS | (Optional) A long value that specifies the maximum time in milliseconds that a job runs before it is terminated. The default is 0 (zero), which means the job can run indefinitely. |
| maxConcurrentInstances | (Optional) A long value that specifies the maximum number of concurrent running instances of this job. The default is 0 (zero), which means no limit. |
| endDate | (Optional) A date value that specifies the time the job is deactivated. |

ScheduleJobMonthlyDate Method

Schedules a new job with a Monthly by Date trigger.

Syntax: ScheduleJobMonthlyDate name, description, scriptType, script, startDate, startTime, endTime, minutesInterval, whichDays, whichMonths [, *parameters*, *flags*, maxRunTimeMS, maxConcurrentInstances, endDate]

Arguments: See [Table 29 on page 63](#) for ScheduleJobMonthlyDate method arguments.

Table 29. ScheduleJobMonthlyDate Method Arguments

| Argument | Description |
|-----------------|---|
| name | A string that specifies the name of the job. |
| description | A string that specifies the description of the job. |
| scriptType | A string that specifies the script type associated with the job (either VBScript or JScript). |
| script | A string that specifies the script code or path (if the nqJobScriptContainsPath flag is set) associated with the job. |
| startDate | A date value that specifies the date the job is activated. |
| startTime | A date value that specifies the time the job is activated. |
| endTime | A date value that specifies the time the job is deactivated. |
| minutesInterval | A long value that specifies the number of minutes between consecutive job executions. |

Table 29. ScheduleJobMonthlyDate Method Arguments

| Argument | Description |
|------------------------|--|
| whichDays | An long value that specifies the days of the month the job runs. For valid settings, see "DayEnum Constants" on page 55 . |
| whichMonths | An integer value that specifies the months the job runs. For valid settings, see "MonthEnum Constants" on page 57 . |
| parameters | (Optional) A string array of parameter values passed to the script. The default is an empty array. |
| flags | (Optional) A long value that specifies the flags associated with the job. For valid settings, see "JobFlagsEnum Constants" on page 56 . The default is nqJobNoFlags. |
| maxRunTimeMS | (Optional) A long value that specifies the maximum time in milliseconds that a job runs before it is terminated. The default is 0 (zero), which means the job can run indefinitely. |
| maxConcurrentInstances | (Optional) A long value that specifies the maximum number of concurrent running instances of this job. The default is 0 (zero), which means no limit. |
| endDate | (Optional) A date value that specifies the time the job is deactivated. |

ScheduleJobMonthlyDOW Method

Schedules a new job with a monthly by day of the week (DOW) trigger.

Syntax: ScheduleJobMonthlyDOW name, description, scriptType, script, startDate, startTime, endTime, minutesInterval, whichOccurrences, whichDays, whichMonths [, parameters, flags, maxRunTimeMS, maxConcurrentInstances, endDate]

Arguments: See [Table 30 on page 64](#) for ScheduleJobMonthlyDOW method arguments.

Table 30. ScheduleJobMonthlyDOW Method Arguments

| Argument | Description |
|-------------|---|
| name | A string that specifies the name of the job. |
| description | A string that specifies the description of the job. |
| scriptType | A string that specifies the script type associated with the job (either VBScript or JScript). |
| script | A string that specifies the script code or path (if the nqJobScriptContainsPath flag is set) associated with the job. |
| startDate | A date value that specifies the date the job is activated. |
| startTime | A date value that specifies the time the job is activated. |

Table 30. ScheduleJobMonthlyDOW Method Arguments

| Argument | Description |
|------------------------|--|
| endTime | A date value that specifies the time the job is deactivated. |
| minutesInterval | A long value that specifies the number of minutes between consecutive job executions. |
| whichOccurrences | An integer value that specifies the occurrences of days of the week the job runs. For valid settings, see "DayEnum Constants" on page 55 . |
| whichDays | An integer value that specifies the days of the week the job runs. For valid settings, see "DayOfWeekEnum Constants" on page 55 . |
| whichMonths | An integer value that specifies the months the job runs. For valid settings, see "MonthEnum Constants" on page 57 . |
| parameters | (Optional) A string array of parameter values passed to the script. The default is an empty array. |
| flags | (Optional) A long value that specifies the flags associated with the job. For valid settings, see "JobFlagsEnum Constants" on page 56 . The default is nqJobNoFlags. |
| maxRunTimeMS | (Optional) A long value that specifies the maximum time in milliseconds that a job runs before it is terminated. The default is 0 (zero), which means the job can run indefinitely. |
| maxConcurrentInstances | (Optional) A long value that specifies the maximum number of concurrent running instances of this job. The default is 0 (zero), which means no limit. |
| endDate | (Optional) A date value that specifies the time the job is deactivated. |

ScheduleJobNow Method

Schedules a new job with a Run Now trigger.

Syntax: ScheduleJobNow name, description, scriptType, script [, parameters, flags, maxRunTimeMS]

Arguments: See [Table 31 on page 65](#) for ScheduleJobNow method arguments.

Table 31. ScheduleJobNow Method Arguments

| Argument | Description |
|-------------|---|
| name | A string that specifies the name of the job. |
| description | A string that specifies the description of the job. |
| scriptType | A string that specifies the script type associated with the job (either VBScript or JScript). |

Table 31. SchedulJobNow Method Arguments

| Argument | Description |
|--------------|--|
| script | A string that specifies the script code or path (if the nqJobScriptContainsPath flag is set) associated with the job. |
| parameters | (Optional) A string array of parameter values passed to the script. The default is an empty array. |
| flags | (Optional) A long value that specifies the flags associated with the job. For valid settings, see "JobFlagsEnum Constants" on page 56 . The default is nqJobNoFlags. |
| maxRunTimeMS | (Optional) A long value that specifies the maximum time in milliseconds that a job runs before it is terminated. The default is 0 (zero), which means the job can run indefinitely. |

ScheduleJobOnce Method

Schedules a new job with a Run Once trigger.

Syntax: ScheduleJobOnce name, description, scriptType, script, startDate, startTime [, parameters, flags, maxRunTimeMS]

Arguments: See [Table 32 on page 66](#) for ScheduleJobOnce method arguments.

Table 32. ScheduleJobOnce Method Arguments

| Argument | Description |
|--------------|---|
| name | A string that specifies the name of the job. |
| description | A string that specifies the description of the job. |
| scriptType | A string that specifies the script type associated with the job (either VBScript or JScript). |
| script | A string that specifies the script code or path (if the nqJobScriptContainsPath flag is set) associated with the job. |
| startDate | A date value that specifies the date the job is activated. |
| startTime | A date value that specifies the time the job is activated. |
| parameters | (Optional) A string array of parameter values passed to the script. The default is an empty array. |
| flags | (Optional) A long value that specifies the flags associated with the job. For valid settings, see "JobFlagsEnum Constants" on page 56 . The default is nqJobNoFlag. |
| maxRunTimeMS | (Optional) A long value that specifies the maximum time in milliseconds that a job runs before it is terminated. The default is 0 (zero), which means the job can run indefinitely. |

ScheduleJobWeekly Method

Schedules a new job with a Weekly trigger.

Syntax: ScheduleJobWeekly name, description, scriptType, script, startDate, startTime, endTime, minutesInterval, weeksInterval, whichDays [, parameters, flags, maxRunTimeMS, maxConcurrentInstances, endDate]

Argument: See [Table 33 on page 67](#) for ScheduleJobWeekly method arguments.

Table 33. ScheduleJobWeekly Method Arguments

| Argument | Description |
|------------------------|--|
| name | A string that specifies the name of the job. |
| description | A string that specifies the description of the job. |
| scriptType | A string that specifies the script type associated with the job (either VBScript or JScript). |
| script | A string that specifies the script code or path (if the nqJobScriptContainsPath flag is set) associated with the job. |
| startDate | A date value that specifies the date the job is activated. |
| startTime | A date value that specifies the time the job is activated. |
| endTime | A date value that specifies the time the job is deactivated. |
| minutesInterval | A long value that specifies the number of minutes between consecutive job executions. |
| weeksInterval | An integer value that specifies the number of weeks between job invocations. |
| whichDays | An integer value that specifies the days of the week the job runs. See "DayOfWeekEnum Constants" on page 55 for valid settings. |
| parameters | (Optional) A string array of parameter values passed to the script. The default is an empty array. |
| flags | (Optional) A long value that specifies the flags associated with the job. For valid settings, see "JobFlagsEnum Constants" on page 56 . The default is nqJobNoFlags. |
| maxRunTimeMS | (Optional) A long value that specifies the maximum time in milliseconds that a job runs before it is terminated. The default is 0 (zero), which means the job can run indefinitely. |
| maxConcurrentInstances | (Optional) A long value that specifies the maximum number of concurrent running instances of this job. The default is 0 (zero), which means no limit. |
| endDate | (Optional) A date value that specifies the time the job is deactivated. |

OnError Event

Occurs when the script engine encounters a run-time error while executing the script. This is intended for cleanup purposes, but the creative use of try/catch blocks in JScript and proper Error Handling in VBScript are often superior alternatives to using this event.

Usage: The script engine calls this procedure when it encounters a run-time error while executing the script. Define this procedure in your script if you want to perform some cleanup activities before the script terminates, such as deleting temporary files and releasing resources.

Syntax: OnError

Examples

- Using VBScript:

```
Public Sub OnError()  
    LogFile.WriteLine "Encountered a runtime error in the script."  
    LogFile.Close  
End Sub
```

- Using JScript:

```
function OnError()  
{  
    LogFile.WriteLine("Encountered a runtime error in the  
    script.");  
    LogFile.Close();  
}
```

Index

B

Bulk Fetch Buffer Size (bytes) configuration field 28

C

Call Interface configuration field 28

canceling instances 26

concurrent connections, maximum number 29

CreateArray object method 59

D

daily trigger type 40

Data Source Name configuration field 28

Database Type configuration field 29

database, back-end

- bulk fetch buffer size 28
- call interface type 28
- configuration options 28
- data source logon name 28
- database type 29
- default configuration options, restoring 28
- job script path 30
- logon username 30, 34
- maximum concurrent connections field 29
- password field 29
- purging 30
- Siebel Analytics Scheduler requirement 11
- timeout specification 30

databases supported 11

day-of-month flags 55

day-of-week constants 55

DayOfWeek Enum values 55

Default Script Path configuration field 29

DeregisterCancelCommand object method 60

E

ErrorMessage instance property 27

ErrorMessage script object property 54

ExitCode instance property 27

ExitCode script object property 54

G

GetTempFileName object method 61

I

InstanceID instance property 27

InstanceID script object property instances 53

- canceling 26
- canceling, purging, or viewing 26
- definition of 9
- ErrorMessage property 27
- ExitCode property 27
- InstanceID property 27
- job instances, managing 21
- JobID property 27
- purging 26
- Status property 27
- viewing 26

J

job action

- Parameters (script) field 37
- Script Contains File Name field 37
- Script field 37
- Script Type field 37

job actions, defining 37

job instances, managing 21

Job Manager

- File menu options, table of 24
- Instances menu options, table of 26
- job instances, managing 21
- Jobs menu options, table of 25
- overview 18
- Service Management menu options, table of 25
- shutting down 24
- Siebel iBot jobs, adding and modifying 19

Job Manager Configuration dialog box

- accessing 28
- opening 24
- role of 28
- Scheduler Tab fields 28
- Siebel iBots tab fields 32
- Siebel Workflow tab fields 34

job parameters, definition of 9

job script path 30

job scripts

- day-of-month flags 55
- day-of-week flags 55
- definition of 9

- Delete Script When Job is Removed 36
 - deleting at end of job 36
 - Disabled field 36
 - disabling 36
 - locating 12
 - month-of-year constants 57
 - occurrence constants 57
 - Script object 47
 - Script object read/write properties 54
 - Script object read-only properties 53
 - Script-defined constants 55
 - job triggers**
 - recurrent trigger fields 38
 - recurrent trigger types 40
 - setting 38
 - single-run triggers 38
 - JobFlagsEnum flag values** 56
 - JobID instance property** 27
 - JobID script object property** 53
 - jobs**
 - about setting up 19
 - adding or modifying jobs 19
 - automatic deletion after execution 35
 - definition of 9
 - Delete Job When Done field 35
 - Execute When Missed field 36
 - job property configuration fields 35
 - JobID field 35
 - last run time and next run time fields 35
 - Last Run Time field 35
 - Maximum Run Time MS field 35
 - number of instances running 35
 - Running Instance Count field 35
 - scheduling and canceling 25
 - time out setting 35
- L**
- LaunchProcess object method** 61
 - login username, back-end database** 30
- M**
- Mail tab**
 - Authenticate against SMTP Server configuration field 31
 - From Display Name configuration field 31
 - Maximum Recipients field 31
 - Number of Retries Upon Failure configuration field 31
 - Password/Confirm Password configuration field 31
 - Sender Address configuration field 31
 - SMTP Port configuration field 31
 - SMTP Server configuration field 31
- Use Bcc Instead of To field 31
 - Username configuration field 31
 - managing job instances** 21
 - Maximum Connections configuration field** 29
 - Maximum Execution Threads configuration field** 29
 - Minimum Execution Threads configuration field** 29
 - Monthly by Date trigger type** 41
 - Monthly by DOW trigger type** 41
 - month-of-year constants** 57
- N**
- nqJobDeleteScriptWhenDone script-defined constant** 56
 - nqJobDeleteWhenDone script-defined constant** 56
 - nqJobDisabled script-defined constant** 56
 - nqJobExecuteWhenMissed script-defined constant** 56
 - nqJobHasEndDate script-defined constant** 56
 - nqJobNoFlags** 56
 - nqJobScriptContainsPath script-defined constant** 56
 - nqJobStartNow script-defined constant** 56
 - nQS (Siebel) Administrator Name configuration field** 29
 - nQS (Siebel) Administrator Password/Confirm Password configuration field** 29
 - NQScheduler.log file** 13
- O**
- occurrence constants, table of values** 57
 - OnError object event** 68
- P**
- ParameterCount script object property** 53
 - Password/Confirm Password configuration field** 29
 - Pause When Service Starts configuration field** 29
 - pausing and continuing Siebel Analytics Scheduler** 25
 - Purge DB every X minutes configuration field** 30
 - Purge Old instances after X days configuration field** 30
 - purging instances** 26

R**recurrent trigger**

- Begin Date 38
- End Date 38
- End Time 38
- Has End Date 39
- Interval in Minutes 39
- Maximum Concurrent Instances 39
- Set Start Time To Now 39
- Start Time 38

recurrent triggers

- fields related to 38
- trigger types 40

RegisterCancelCommand object method**running job, definition of** 9**S**

- S_NQ_ERR_MSG** 12
- S_NQ_INSTANCE** 12
- S_NQ_JOB** 12
- S_NQ_JOB_PARAM** 12
- ScheduleJobDaily object method** 62
- ScheduleJobMonthlyDate object method** 63
- ScheduleJobMonthlyDOW object method** 64
- ScheduleJobNow object method** 65
- ScheduleJobOnce object method** 66
- ScheduleJobWeekly object method** 67

Scheduler

- Scheduler tables, about database support 11

Scheduler Log File 13**Scheduler Script Path configuration field** 30**Scheduler tables**

- about 12

schema tables 12**Script object**

- overview 47
- read/write properties 54
- read-only properties 53

script-defined constants

- about 55
- day-of-month flags 55
- day-of-week constants 55
- job status flags 56
- month-of-year flags 57
- occurrence flags 57

scripts

- object methods 58

Siebel Analytics Scheduler

- connection, opening and closing 24
- databases supported 11
- execution threads 29
- Job Manager, role of 18
- overview 9
- pausing and continuing 25
- pausing execution 29
- role of 9
- scheduling and removing jobs 25
- schema tables 12
- stopping 25

Siebel Analytics Scheduler configuration

- configuration options 28
- Job Manager Configuration dialog box, opening 24
- registry settings, configuring 16
- registry settings, confirming 15
- Scheduler Tab options 28

Siebel iBots

- default values, restoring 32
- iBot Log Directory configuration field 32
- Job Manager Configuration fields 32
- jobs, adding and modifying 19
- log file location 32
- Max Rows Times Columns configuration field 32
- Minimum/Maximum Delivery Sleep Seconds configuration field 32
- Minimum/Maximum Global Sleep Seconds configuration field 32
- Minimum/Maximum Request Sleep Seconds configuration field 32
- modifying 20
- number connection attempts 33
- number delivery attempts 33
- Number of Delivery Tries configuration field 33
- Number of Global Tries configuration field 33
- Number of Request Tries configuration field 33
- property fields 20
- Purge files older than (days) configuration field 33
- SAW Machine Name configuration field 33
- Siebel Analytics Scheduler, role of 9
- sleep time 32

Single-Run Triggers 38**Status instance property** 27**stopping Siebel Analytics Scheduler** 25**T****Temporary File Path configuration field** 30

temporary files, location of 30
Timeout (Minutes) configuration field 30
trigger, definition of 9

U

User name configuration field 30
UserID script object property 53
username 30

V

viewing instances 26

W

weekly trigger type 40

Windows

Scheduler tables, about 12
Scheduler, about database support 11

Workflow tab

logon username, back-end database 34
username 34

X

XML Converter Escape Sequences 43