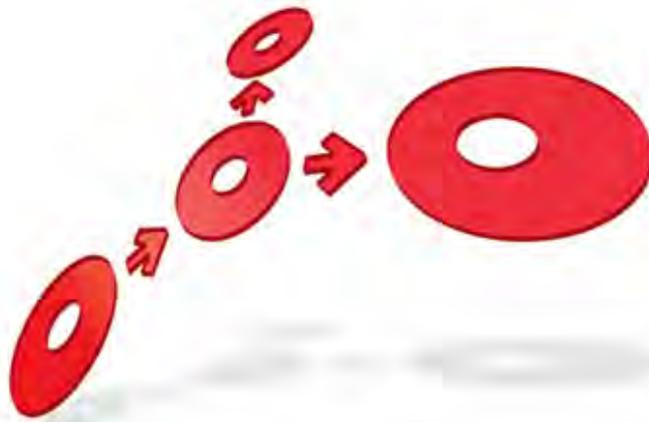


Portrait
Foundation



SQL Server Reporting Services User Guide

Edition 1.2

10 January 2013



 **Pitney Bowes**
Software



Foundation SQL Server Reporting Services User Guide

©2013
Copyright Portrait Software International Limited

All rights reserved. This document may contain confidential and proprietary information belonging to Portrait Software plc and/or its subsidiaries and associated companies.

Portrait Software, the Portrait Software logo, Portrait, Portrait Software's Portrait brand and Million Handshakes are the trademarks of Portrait Software International Limited and may not be used or exploited in any way without the prior express written authorization of Portrait Software International Limited.

Acknowledgement of trademarks

Other product names, company names, marks, logos and symbols referenced herein may be the trademarks or registered trademarks of their registered owners.

About Portrait Software

Portrait Software is now part of [Pitney Bowes Software Inc.](http://www.pitneybowes.com)

Portrait Software enables organizations to engage with each of their customers as individuals, resulting in improved customer profitability, increased retention, reduced risk, and outstanding customer experiences. This is achieved through a suite of innovative, insight-driven applications which empower organizations to create enduring one-to-one relationships with their customers.

Portrait Software was acquired in July 2010 by Pitney Bowes to build on the broad range of capabilities at Pitney Bowes Software for helping organizations acquire, serve and grow their customer relationships more effectively. The Portrait Customer Interaction Suite combines world leading customer analytics, powerful inbound and outbound campaign management, and best-in-class business process integration to deliver real-time customer interactions that communicate precisely the right message through the right channel, at the right time.

Our 300 + customers include industry-leading organizations in customer-intensive sectors. They include 3, AAA, Bank of Tokyo Mitsubishi, Dell, Fiserv Bank Solutions, Lloyds Banking Group, Merrill Lynch, Nationwide Building Society, RACQ, RAC WA, Telenor, Tesco Bank, T-Mobile, Tryg and US Bank.

Pitney Bowes Software Inc. is a division of Pitney Bowes Inc. (NYSE: PBI).

For more information please visit: <http://www.pitneybowes.co.uk/software/>

UK

Portrait Software
The Smith Centre
The Fairmile
Henley-on-Thames
Oxfordshire, RG9 6AB, UK

Email: support@portraitsoftware.com
Tel: +44 (0)1491 416778
Fax: +44 (0)1491 416601

America

Portrait Software
125 Summer Street
16th Floor
Boston, MA 02110
USA

Email: support@portraitsoftware.com
Tel: +1 617 457 5200
Fax: +1 617 457 5299

Norway

Portrait Software
Portrait Million Handshakes AS
Maridalsveien. 87
0461 Oslo
Norway

Email: support@portraitsoftware.com
Tel: +47 22 38 91 00
Fax: +47 23 40 94 99

About this document

Purpose of document

This document shows how it is possible to use SQL Server Reporting Services to build MIS reports on top of real-time Portrait Foundation operational data.

Intended audience

Anyone interested in generating MIS reports on data entities defined in the Configuration Suite and stored in the Portrait Foundation operational database.

Related documents

Some of the reports in this document make use of the Configuration Suite Database views functionality. Details of this can be found in

Database Views User Guide

Software release

Portrait Foundation 4.4 or later.

Contents

1	Introduction	6
2	Installation	7
2.1	SQL Server Reporting Services	7
2.2	Report Deployment	7
2.3	Report Manager	7
3	Agent reports	9
4	Engagement reports	12
5	Workflow reports	16
	Appendix A Workflow Reference	19
A.1	Database API	19

1 Introduction

Provided with this release are some sample reports that are implemented using SQL Server Reporting Services. They build on top of work originally done for the Business Console (Workflow Monitor) and a new feature introduced in Portrait Foundation version 4.2. Making use of the Configuration Suite Database views functionality it is possible to generate MIS reports for data entities defined in the Configuration Suite and stored in the Portrait Foundation operational database.

- FoundationAgentReports – Basic Agent reports covering logon status, roles and teams. Requires the Portrait Platform **“Sample agent report views”** package to be included in your workspace and deployed.
- FoundationEngagementReports – Basic engagement history reports including engagement actions and involved parties. Requires the Portrait Platform **“Sample engagement report views”** package to be included in your workspace and deployed.
- FoundationWorkflowReports – Basic workflow process and task reports that replace the Business Console (Workflow Monitor) functionality. The Portrait Foundation 4.4 release has extended the View Builder functionality to support Workflow Task attributes. See the *Database Views User Guide* for more details.

All these reports were developed using SQL Server 2008 Report Builder 2.0 and then added to a SQL Server Business Intelligence Development Studio Report Server project for deployment.

It is important to ensure that any reports created do not have an impact on the performance of the runtime system. It may therefore be worth considering running your reports against a backed up operational database, that contains these Database Views.

2 Installation

2.1 SQL Server Reporting Services

Install SQL Server Reporting Services on your SQL Server 2008 Database Server.

You should be able to navigate to SQL Server Report Manager:
`http://<sqlserver_machinename>/Reports`

If you are not using the default instance, the URL may have the following format:
`http://<sqlserver_machinename>/Reports_<instance_name>`

The default security role "Content Manager" is set to BUILTIN\Administrators. So as long as you are an Administrator on this machine then you will be able to access this site. In order to deploy reports to this server, you will need to be assigned the Publisher role. Depending of the version of SQL Server you can assign new roles via **Folder Settings** or the **Properties tab** in Report Manager. Please refer to the Microsoft documentation on administering security privileges.

2.2 Report Deployment

There are 3 SQL Server Business Intelligence Development Studio Report Server projects with associated solution files for SQL Server 2008. Navigate to Software\Examples\SQLServerReportingServices on the Release CD, open each solution file and do the following:

- Right click on project and select Properties. Set the Deployment TargetServerURL to point at your Report Server e.g.
 - `http://<sqlserver_machinename>/ReportServer` or
 - `http://<sqlserver_machinename>/ReportServer_<instance_name>`
- Right click on project and select Deploy. Deployment will fail if your domain account is not assigned the Published role.

NB: If you are using SQL Server 2008 R2, you may need to make a copy of the solution folders and upgrade the solution files first.

Repeat this process for the other projects. You will see the following warning which can be ignored.

Warning : Cannot deploy data source FoundationDatabase to the server because it already exists and OverwriteDataSources is not specified.

2.3 Report Manager

Follow these steps to view the newly deployed reports:

- Navigate to SQL Server Report Manager (e.g. `http://support-sql08/Reports`).
- Click on the Data Sources folder and then the FoundationDatabase data source.
- Ensure that the "Enable this data source" checkbox is ticked.
- Change the Connection string to point at your Portrait Foundation operational database.

Data Source=**<sql_server_instance>**:Initial
Catalog=**<foundation_operational_database>**

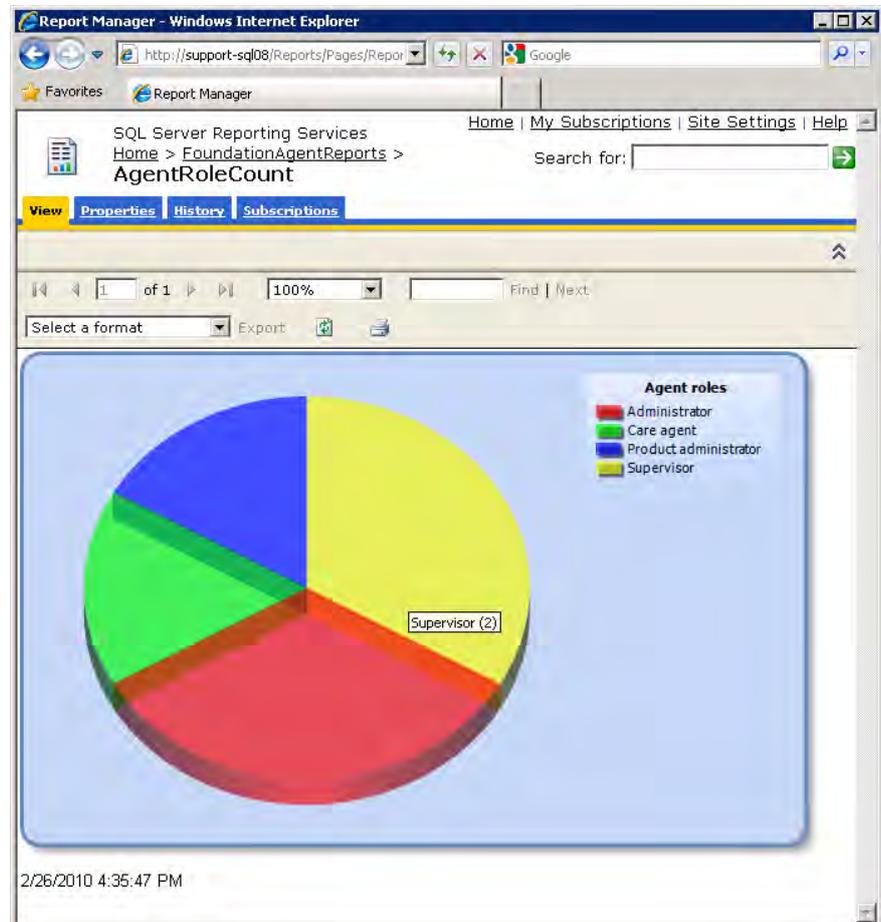
- Scroll to the bottom of the page and click Apply.
- Navigate back to the Report Manager home page, you are now ready to view the reports.

3 Agent reports

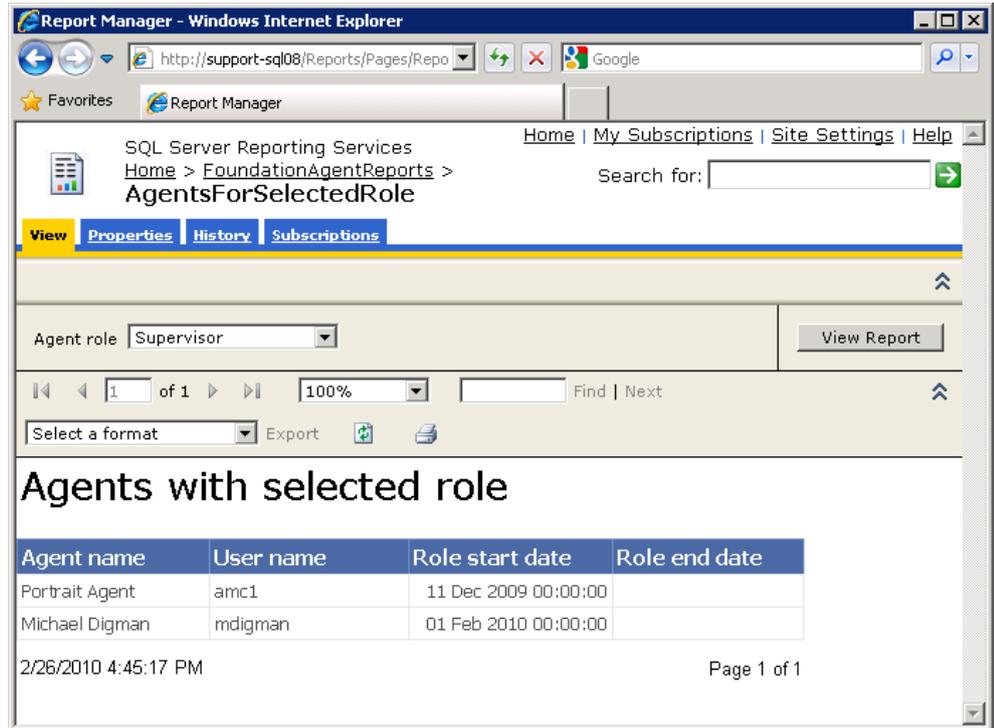
For the following reports to run you must first include the Portrait Platform “**Sample agent report views**” package in your workspace and deploy it to the Portrait Foundation operational database that you configured in [section 2.3](#). For the best results you should connect to a database that already contains Agents.

The FoundationAgentReports folder contains 5 reports:

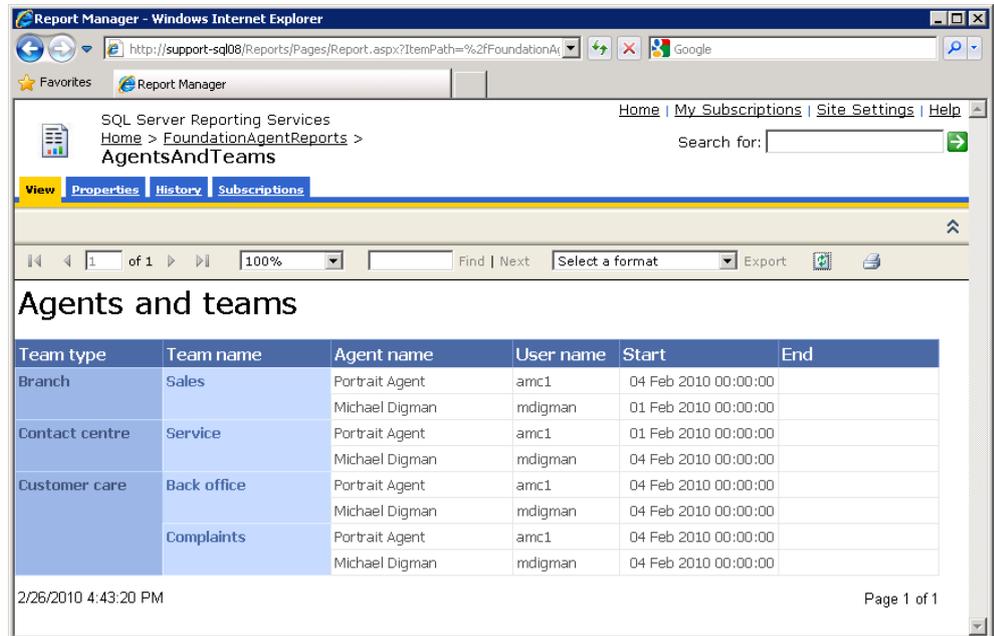
- Agent role count
- Agents for selected role (drill down report)
- Agents and teams
- Logged on agents
- All agents



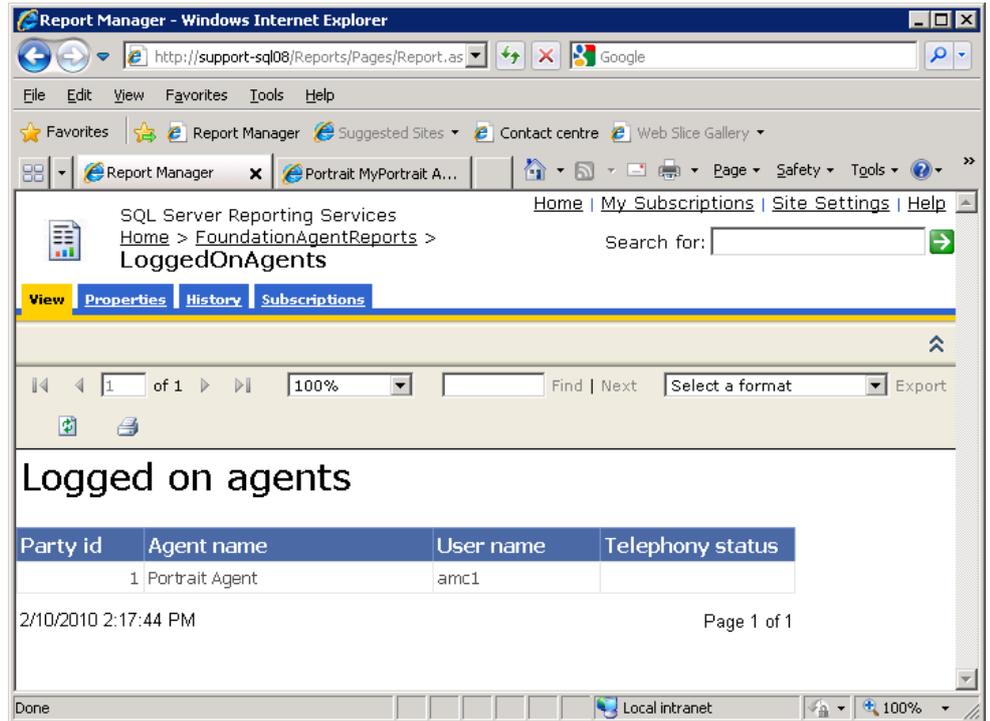
Displays a simple 3D pie chart showing the number of Agent roles. Clicking on one of the segments shows a list of agents that are assigned to this role.



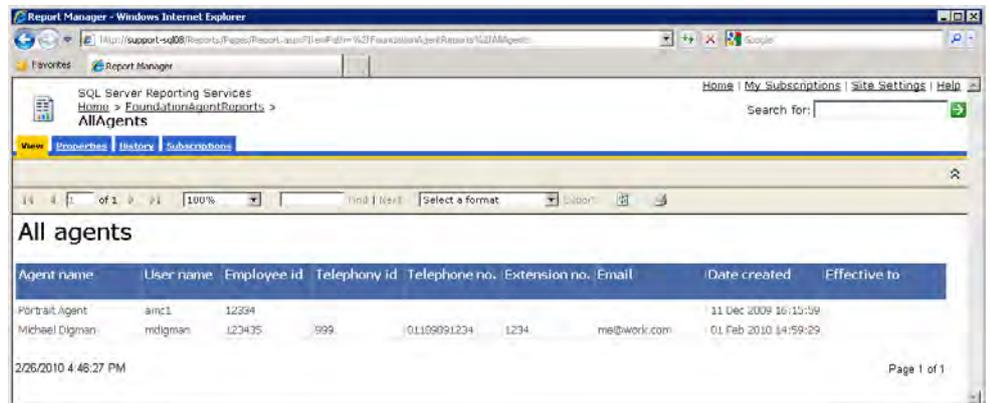
You can select a different Agent role and regenerate the report using the View Report button.



A list of agents grouped by team type and name.



A list of agents currently logged on to the Contact Centre application.



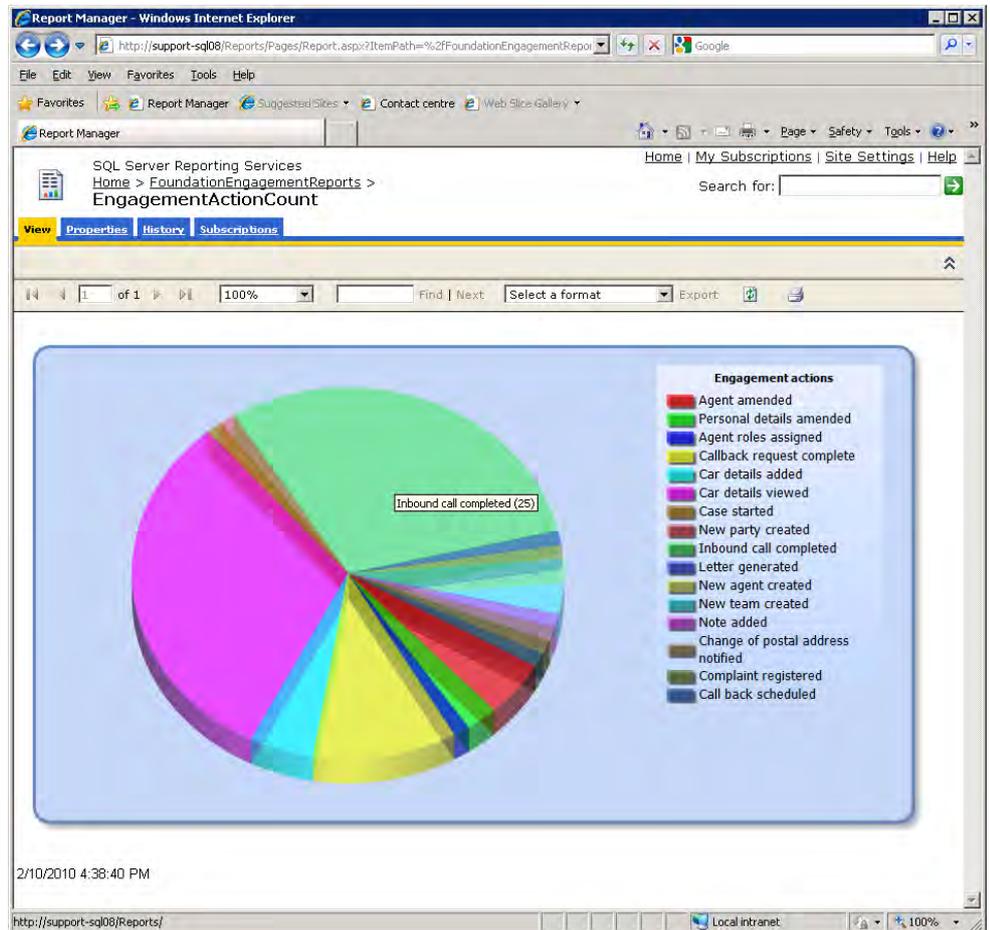
A list of all agents.

4 Engagement reports

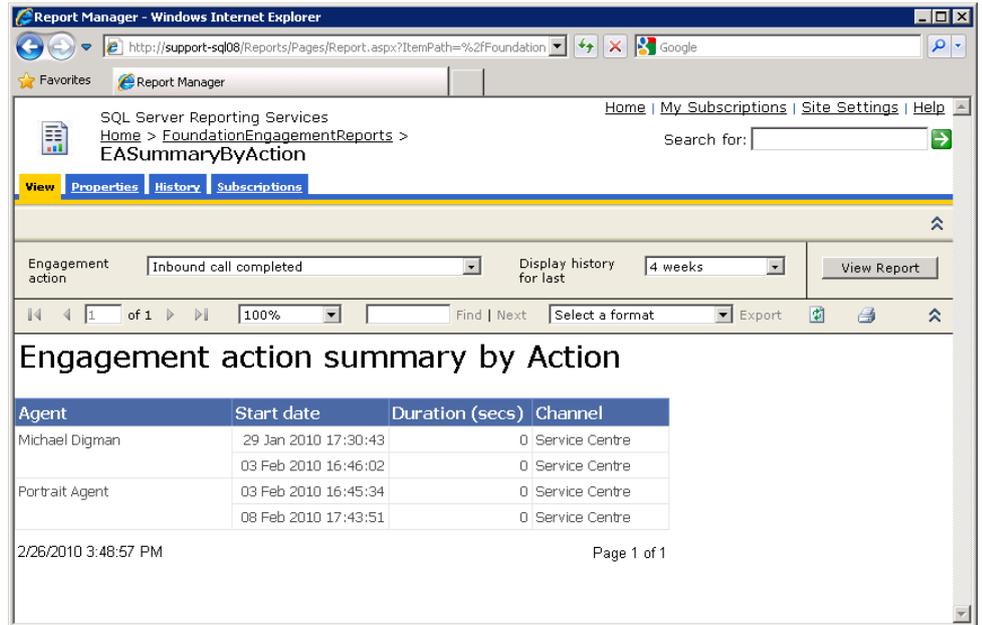
For the following reports to run you must first include the Portrait Platform “Sample engagement report views” package in your workspace and deploy it to the Portrait Foundation operational database that you configured in [section 2.3](#). For the best results you should use the All Applications Workspace.

The FoundationEngagementReports folder contains 6 reports:

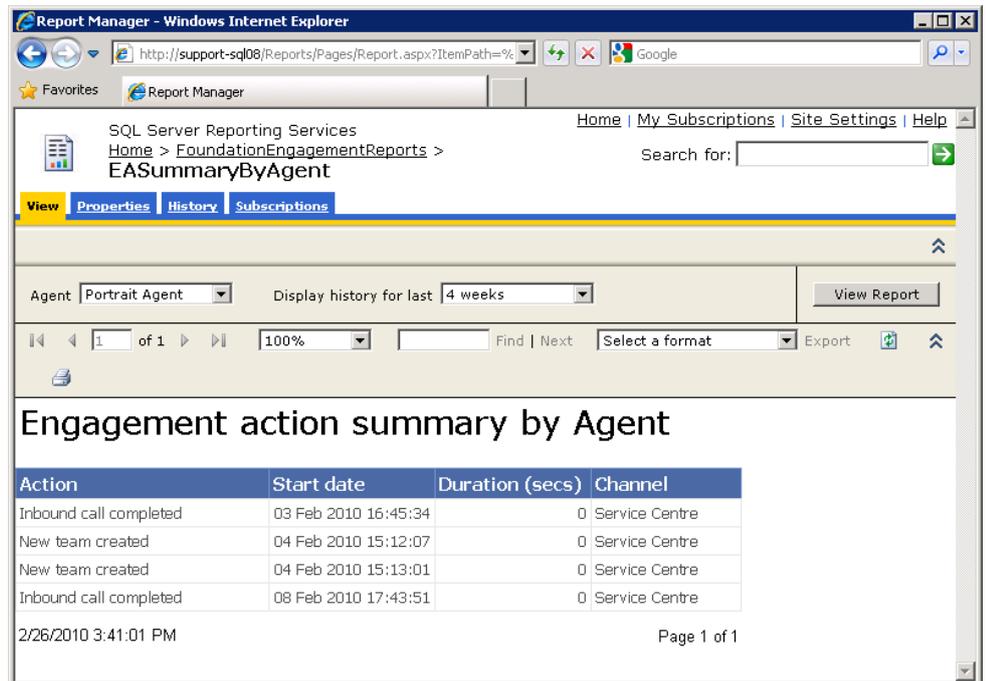
- Engagement action count
- Engagement action summary by Action
- Engagement action summary by Agent
- Engagement action summary
- Engagement history summary
- Personal details update history



Displays a simple 3D pie chart showing the number of Engagement actions.



A list of agent's that have performed the specified action in the selected time period. You can select a different Engagement action and Display history period. Use the View Report button to regenerate the report.



A list of engagement actions that have been performed by the specified agent in the selected time period. You can select a different Engagement action and Display history period. Use the View Report button to regenerate the report. Complaints

Report Manager - Windows Internet Explorer
 http://support-sql08/Reports/Pages/Report.aspx?ItemPath=%2fFoundation
 SQL Server Reporting Services
 Home > FoundationEngagementReports > EngagementActionSummary
 Search for: []

View Properties History Subscriptions

Engagement action summary

Engagement action	Channel	Total	Max duration	Min duration	Avg duration
Agent amended	Service Centre	4	0	0	0
Agent roles assigned	Service Centre	1	0	0	0
Call back scheduled	Service Centre	1	0	0	0
Callback request complete	Service Centre	9	0	0	0
Car details added	Service Centre	4	0	0	0
Car details viewed	Service Centre	26	0	0	0
Case started	Service Centre	1	34	34	34
Change of postal address notified	Service Centre	1	0	0	0
Complaint registered	Service Centre	1	0	0	0
Inbound call completed	Service Centre	25	0	0	0
Letter generated	Service Centre	1	1	1	1
New agent created	Service Centre	1	0	0	0
New party created	Service Centre	1	0	0	0
New team created	Service Centre	4	0	0	0
Note added	Service Centre	1	32	32	32
Personal details amended	Service Centre	2	0	0	0

2/26/2010 4:18:08 PM Page 1 of 1

A summary of the total number of engagement actions with minimum, maximum and average durations.

Report Manager - Windows Internet Explorer
 http://support-sql08/Reports/Pages/Report.aspx?ItemPath=%2fFoundationEngagementReports%2fEngagementHistorySummary
 SQL Server Reporting Services
 Home > FoundationEngagementReports > EngagementHistorySummary
 Search for: []

View Properties History Subscriptions

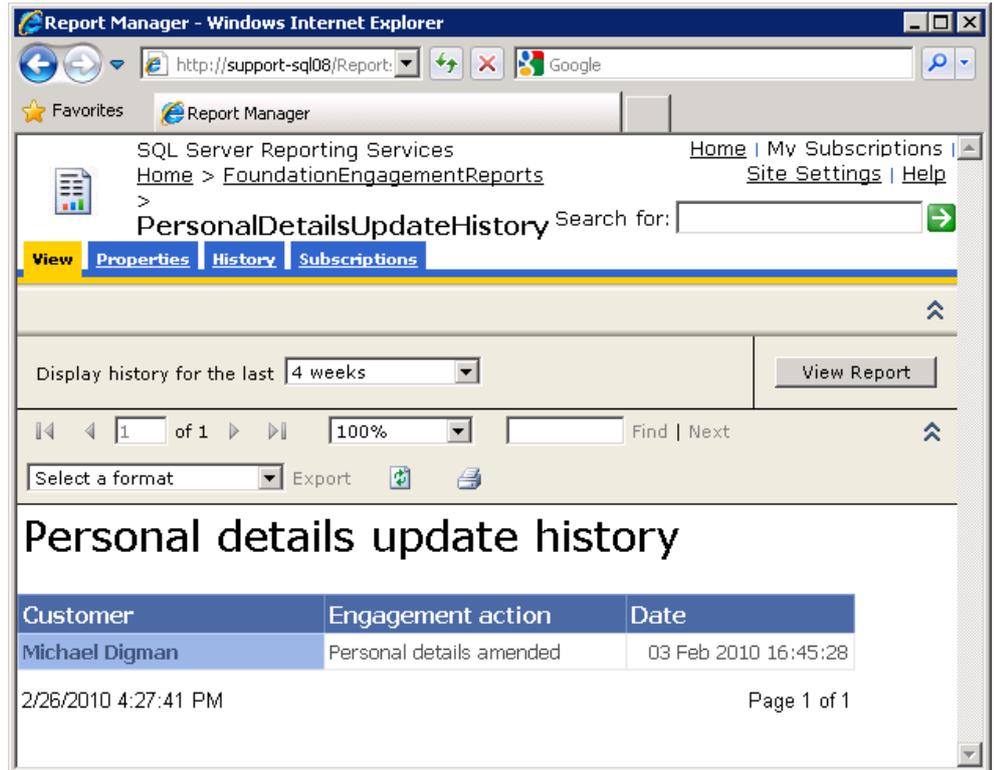
Display history for the last 4 weeks View Report

Engagement history summary

Type	Start date	End date	Duration (mins)	Engagement action	By	For	Action time
Inbound call	29 Jan 2010 16:49:40	29 Jan 2010 17:30:44	41	Inbound call completed	Portrait Agent	Michael Digman	5:30:43 PM
Inbound call	03 Feb 2010 16:44:52	03 Feb 2010 16:45:35	1	Personal details amended	Portrait Agent	Michael Digman	4:45:28 PM
				Inbound call completed	Portrait Agent	Michael Digman	4:45:34 PM
Inbound call	03 Feb 2010 16:45:36	03 Feb 2010 16:46:03	1	Inbound call completed	Portrait Agent	Michael Digman	4:46:02 PM
Inbound call	08 Feb 2010 17:43:24	08 Feb 2010 17:43:52	0	Inbound call completed	Portrait Agent	Michael Digman	5:43:51 PM

2/26/2010 4:26:01 PM Page 1 of 1

A list of engagements with their related actions and involved parties in the selected time period. You can select a different time period and regenerate the report using the View Report button.



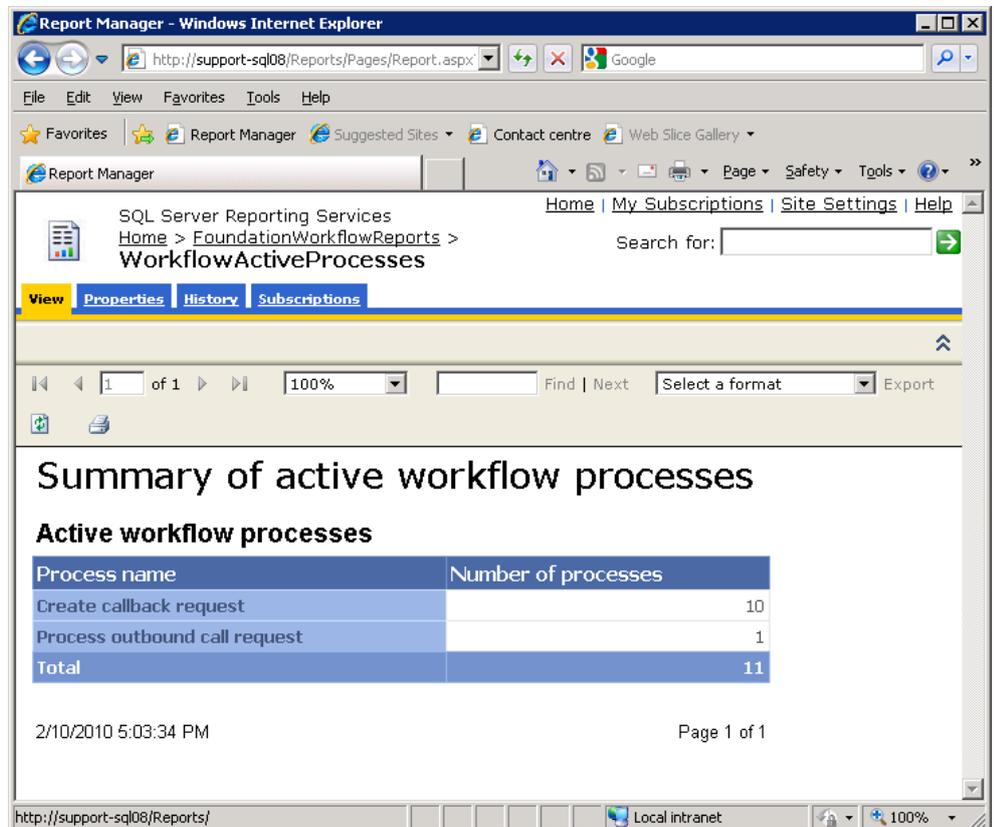
A list of customers who have had their personal data updated, this also includes contact details.

5 Workflow reports

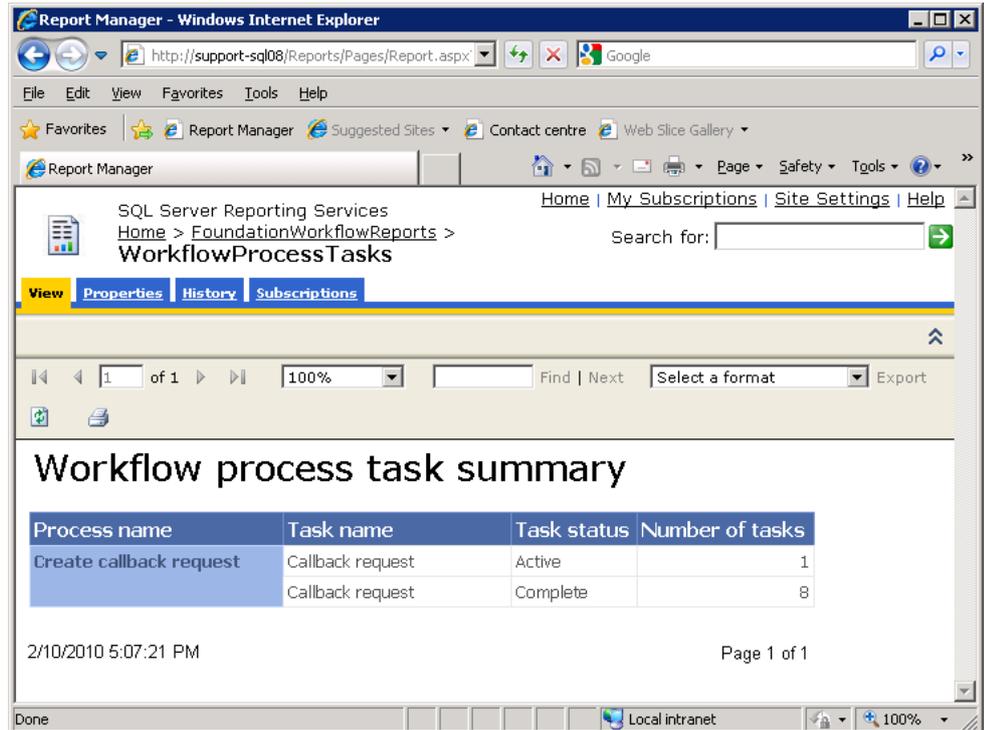
For the best results you should connect to a Portrait Foundation operational database that already contains configured Workflow processes and task instances. If you wish to use a different database follow the steps in [section 2.3](#).

The FoundationWorkflowReports folder contains 5 reports:

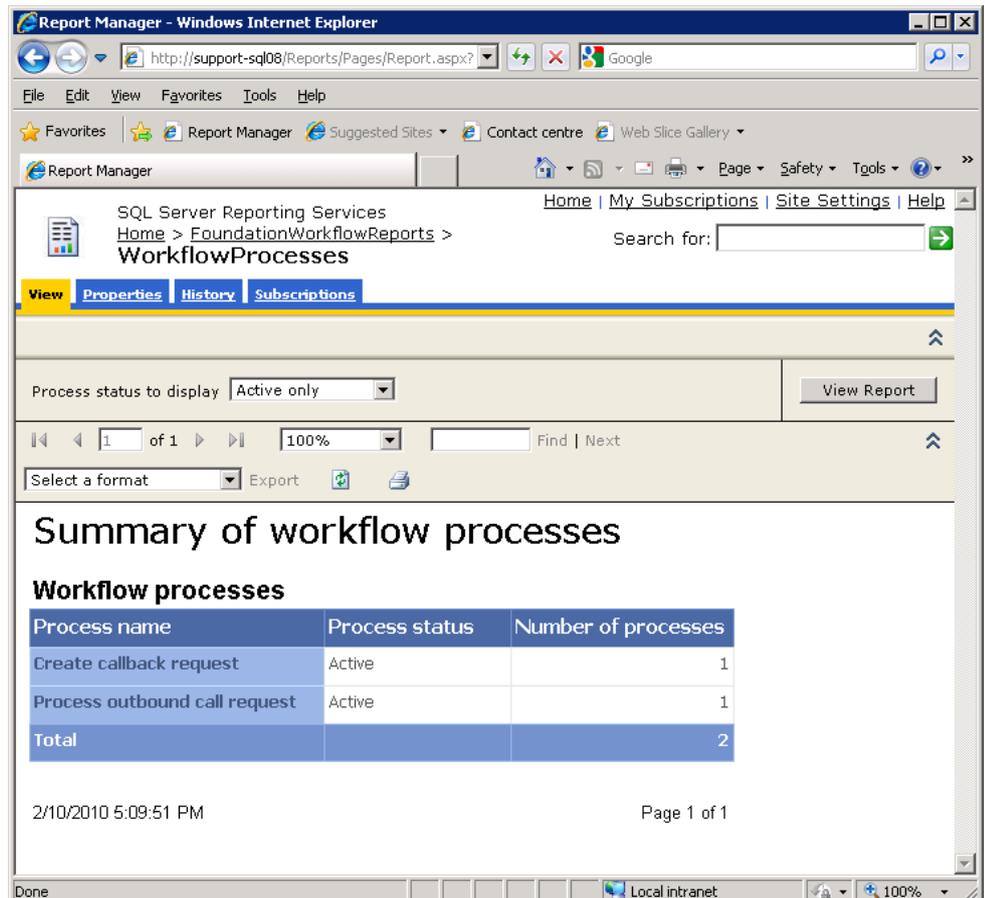
- Active workflow processes
- Workflow process task summary (drill down report)
- Workflow process summary
- Workflow task summary
- Workflow task instances



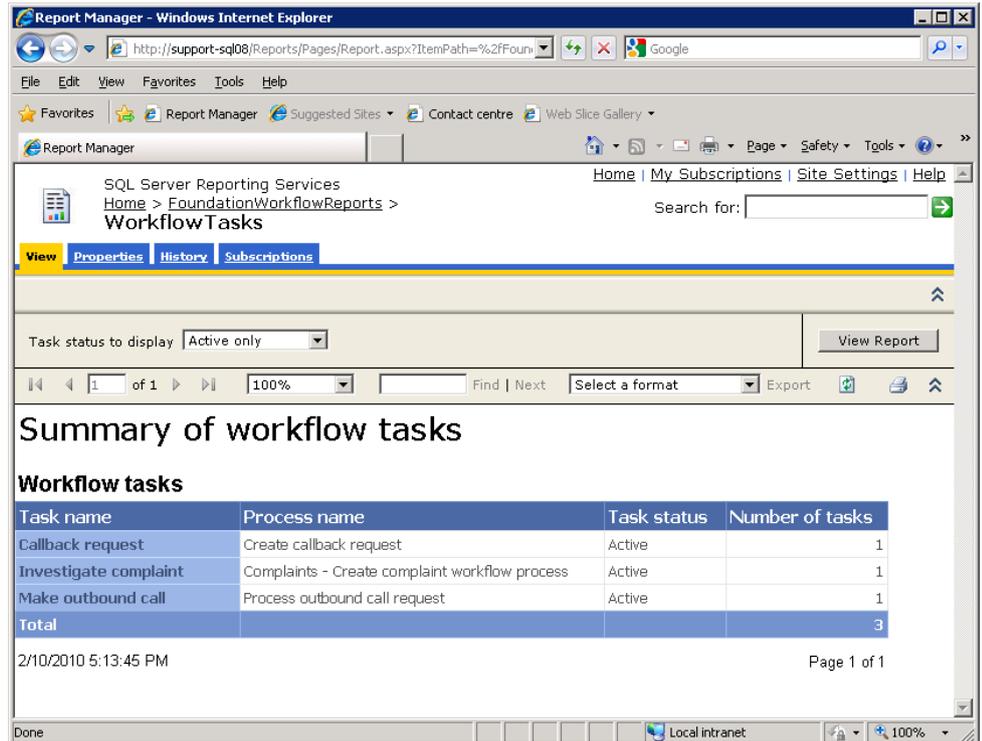
A list of all current active workflow processes. Selecting one of these processes displays the following drill down report.



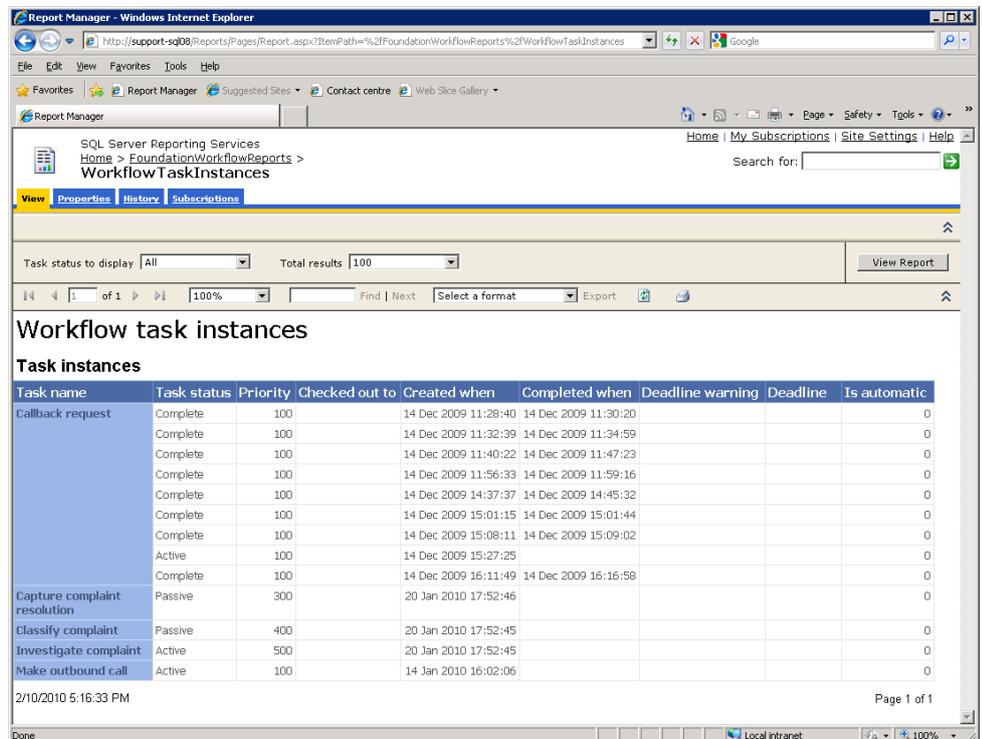
A list of all tasks and their status within the previously selected process.



A summary of all active workflow processes. You can select to display all statuses and regenerate the report using the View Report button.



A summary of all active workflow tasks. You can select to display all statuses and regenerate the report using the View Report button. You can also access the drill down report by clicking on the process name.



A list of all task instances, grouped by task name. You can select to only display active tasks or change the number of results returned. Use the View Report button to regenerate the report.

Appendix A Workflow Reference

This section describes the part of the Portrait operational database that provides the data used by the sample Workflow reports.

This information might be useful to people who needs to customize the Workflow reports in a way that requires changes to the database access. For example, you might need to pass date range parameters to some of the stored procedures in a customized Workflow report, whereas the standard does not use these parameters.

A.1 Database API

The following stored procedures and views enable the retrieval of workflow data from the Operational Database:

- p_amc_te_rep_process_type
- p_amc_te_rep_task_type
- p_amc_te_rep_task_instance
- v_amc_te_rep_task_instance

The stored procedures simplify access to the workflow process and task information. All of the procedures take parameters that filter the results (for example, by process type or by the time period in which an entity was created). However, the procedures are all designed to return information about all active workflow processes or tasks when no parameters are provided.

A.1.1 p_amc_te_rep_process_type

This stored procedure returns a record for each *type* of workflow process. A workflow process type is represented by a Portrait model of type Workflow Process Model. The procedure takes the following parameters, all of which are optional.

Parameter name (and type)	Default	Description
@p_active_only (bit)	1 (true)	If set to 1, the procedure only considers processes that currently have a status of <i>Active</i> . If set to 0, all processes are considered. In general, the default (1) should be used unless one or more started/completed dates are specified to filter the results.
@p_process_system_name (varchar)	null	If a process name is specified, only a single record is returned—the record representing processes of that type. (Note that if the @p_active_only flag is set to false, multiple records may be returned, one for each status.)
@p_started_before (datetime)	null	If a date and time are specified, the procedure only considers processes that were started (created) before that time.
@p_started_after (datetime)	null	If a date and time are specified, the procedure only considers processes that were started (created) after that time. Can be used in conjunction with @p_started_before to specify a range.
@p_completed_before (datetime)	null	If a date and time are specified, the procedure only considers processes that were completed before that time.

@p_completed_after (datetime)	null	If a date and time are specified, the procedure only considers processes that were completed after that time. Can be used in conjunction with @p_completed_before to specify a range.
----------------------------------	------	---

The following columns are returned.

Column name (and type)	Description
Process_Status (varchar)	The status of the Workflow Process Model instances. Possible values are <i>Active</i> and <i>Complete</i> .
Process_Display_Name (varchar)	The display name of the Workflow Process Model.
Process_System_Name (varchar)	The system name of the Workflow Process Model.
Number_Of_Processes (int)	The number of Workflow Process Models matching the supplied parameters.

The sample BDC metadata file uses this stored procedure in its *Finder* method to retrieve a list of all workflow processes that have active instances, and in its *Specific Finder* method to retrieve details about one particular type of workflow process. (The Specific Finder method passes a process name into the @p_process_system_name parameter.)

A.1.2 p_amc_te_rep_task_type

This stored procedure returns a record for each type of workflow task. By default it returns information about only the active tasks, but it can also provide information for tasks with other statuses.

Parameter name (and type)	Default	Description
@p_active_only (bit)	1 (true)	If set to 1, the procedure only considers tasks that currently have a status of <i>Active</i> . If set to 0, all tasks are considered. In general, the default (1) should be used unless one or more started/completed dates are specified to filter the results.
@p_process_system_name (varchar)	Null	If a process name is specified, records are returned only for tasks that belong to the specified process type.
@p_task_system_name (varchar)	Null	If a task name is specified, only a single record is returned - the record representing tasks of that type. (Note that if the @p_active_only flag is set to false, multiple records may be returned, one for each status.)
@p_started_before (datetime)	null	If a date and time are specified, the procedure only considers tasks that were started (created) before that time.
@p_started_after (datetime)	null	If a date and time are specified, the procedure only considers tasks that were started (created) after that time. Can be used in conjunction with @p_started_before to specify a range.
@p_completed_before (datetime)	null	If a date and time are specified, the procedure only considers tasks that were completed before that time.

@p_completed_after (datetime)	null	If a date and time are specified, the procedure only considers tasks that were completed after that time. Can be used in conjunction with @p_completed_before to specify a range.
-------------------------------	------	---

The following columns are returned.

Column name (and type)	Description
Task_Status (varchar)	The status of the Task instances. Possible values are <i>Active</i> , <i>Passive</i> , <i>Suspended</i> , <i>Superseded</i> and <i>Complete</i> .
Process_Display_Name (varchar)	The display name of the Workflow Process Model that contains this task type.
Process_System_Name (varchar)	The system name of the Workflow Process Model that contains this task type.
Task_Display_Name (varchar)	The display name of the Task type.
Task_System_Name (varchar)	The system name of the Task type.
Number_Of_Processes (int)	The number of Task instances matching the supplied parameters.

The sample BDC metadata file uses this stored procedure in its *Finder* method to retrieve a list of all task types that have active instances, and in its *Specific Finder* method to retrieve details about one particular type of task. (The Specific Finder method passes a task name into the @p_task_system_name parameter.) The *GetTasksForProcess* method defined in the BDC metadata passes a process type name into this method to retrieve a list of all task types that belong to a specific type of process.

A.1.3 p_amc_te_rep_task_instance

This stored procedure returns a record for each workflow task instance. By default it only returns information about the active tasks, but it can also provide information for tasks with other statuses. It can also restrict the results to only a certain task type if a task system name is supplied or to a specific task instance if an instance ID is supplied.

Parameter name (and type)	Default	Description
@p_active_only (bit)	1 (true)	If set to 1, the procedure only considers tasks that currently have a status of <i>Active</i> . If set to 0, all tasks are considered.
@p_task_system_name (varchar)	null	If a task name is specified, only tasks of the specified type are returned. If no name is provided, tasks of all types are returned.
@p_task_instance_id (bigint)	null	If a task instance id is specified, the results contains only the record for that instance (if found).
@p_max_results (int)	null	Restricts the number of records that the procedure returns. If no value is specified, all relevant tasks are returned. This could potentially include tens of thousands of records and would be likely to cause performance problems and overwhelm the user with more data than they can understand. It is therefore recommended that a moderate max results value (around 100) is always used so that the results can be restricted.

The following columns are returned.

Column name (and type)	Description
Task_Instance_Id (bigint)	The unique id of the task instance.
Task_Display_Name (varchar)	The display name of the task type.
Task_Status (varchar)	The status of the task instance. Possible values are: <i>Complete, Active, Passive, Suspended, Superseded</i> .
Priority (int)	The priority of the task instance.
Checked_Out (numeric)	If set to 1 the task is currently checked out.
Checkout_Time (datetime)	Timestamp of when the task was checked out (or null if the task is not checked out).
Checked_Out_To (varchar)	Name of the user who the task is checked out to (or null if the task is not checked out). The user name is taken from the Agent's <i>known_as</i> field.
Reroute_Count (numeric)	The number of times the task has been rerouted.
Routed_To (varchar)	Name of the user to whom the task is routed.
Reroute_Limit (numeric)	Maximum number of times that the task can be rerouted.
Activated (datetime)	When the task last entered an activation window.
Created_Datetime (datetime)	When the task was created.
Completed_Datetime (datetime)	When the task was completed, or null if the task has not been completed.
Deadline_Warning_Datetime (datetime)	When a deadline warning is due, or null if no deadline warning has been specified.
Deadline_Datetime (datetime)	When a deadline is due, or null if no deadline warning has been specified.
Is_Automatic (numeric)	If set to 1 this task can be processed automatically.
Process_Display_Name (varchar)	Name of the process model that created this task.

The Task Instance *Finder* method uses this procedure to return a set of tasks, and the *SpecificFinder* method uses it to return a specific task instance. The *Finder* method supplies parameters for *active_only*, *max_results* and *task_system_name*. The *SpecificFinder* passes the *task_instance_id*.

A.1.4 v_amc_te_rep_task_instance

This view joins a number of different workflow tables to provide comprehensive information about workflow tasks. The *p_amc_te_rep_task_instance* stored procedure uses this view and it is recommended that any additional task-related lookups are also performed against this view.

The following columns are returned.

Column name (and type)	Description
Task_Instance_Id (bigint)	The unique id of the task instance.
Task_System_Name (varchar)	System name of the task type.
Task_Display_Name (varchar)	Display name of the task type.
Task_Status (varchar)	The status of the task instance. Possible values are: <i>Complete, Active, Passive, Suspended, Superseded.</i>
Priority (int)	The priority of the task instance.
Checked_Out (numeric)	If set to 1 the task is currently checked out.
Checkout_Time (datetime)	Timestamp of when the task was checked out (or null if the task is not checked out).
Checked_Out_To (varchar)	Name of the user who the task is checked out to (or null if the task is not checked out). The user name is taken from the Agent's <code>known_as</code> field.
Reroute_Count (numeric)	The number of times the task has been rerouted.
Routed_To (varchar)	Name of the user to whom the task is routed.
Reroute_Limit (numeric)	Maximum number of times that the task can be rerouted.
Activated (datetime)	When the task last entered an activation window.
Created_Datetime (datetime)	When the task was created.
Completed_Datetime (datetime)	When the task was completed, or null if the task has not been completed.
Deadline_Warning_Datetime (datetime)	When a deadline warning is due, or null if no deadline warning has been specified.
Deadline_Datetime (datetime)	When a deadline is due, or null if no deadline has been specified.
Is_Automatic (numeric)	If set to 1 this task can be processed automatically.
Process_Display_Name (varchar)	Display name of the process model that created this task.
Process_System_Name (varchar)	System name of the process model that created this task.