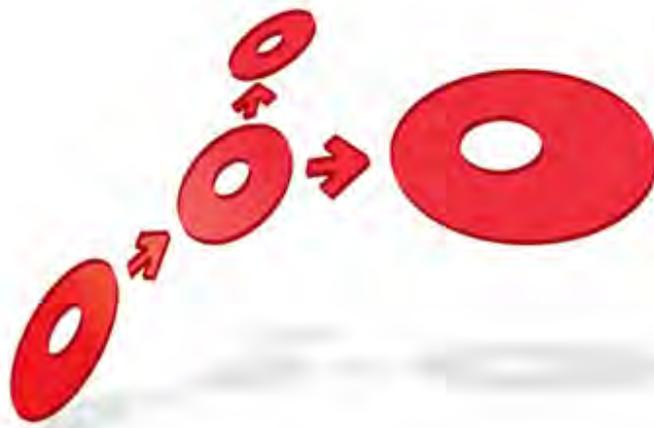


Portrait Foundation



Using the Contact Centre Workspace - Overview

Edition 1.2

11 January 2013





Portrait Foundaion Using the Contact Centre Workspace - Overview

©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.co.uk/software/)

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 is intended to provide information that will allow someone who configures Portrait (a configurer) to implement a solution based on a primitive contact centre workspace.

The workspace provides functionality that will allow a user to log on to a contact centre application and attempt to take a call.

The take call process cannot be completed until the configurer provides a party definition that will represent the parties being served by the contact centre.

An agent management application is also provided that allows agents to be added and linked with teams and roles.

Intended audience

This document is intended to be used by configurers and developers involved in creating a Portrait serviced application e.g. branch, contact centre.

Related documents

Software Development Kit User Guide (*Foundation SDK User Guide*)

Using the Contact Centre Workspace Overview

Software release

Portrait Foundation 4.2 SP1 or later.

Contents

1	Introduction	6
2	Contact centre	7
2.1	Agent desktop	7
2.2	Engagement desktop	9
2.3	Workflow desktop	11
3	Agent manager	12
3.1	Access	12
3.2	Capability	12
3.3	Layout (out of the box)	17
4	Setting application values	19
4.1	Suppressing logon check	20
4.2	Application name	20
4.3	Setting the channel	21
4.4	Setting the Agent engagement type	21
4.5	Setting the language	21
5	Publish and Deploy	23

1 Introduction

The Contact centre workspace provides two applications that are intended to allow agents to service requests from other parties. These are:

- Contact centre
- Agent manager

Both applications are driven by *Application framework models* (AFM). These models manage the lifecycle of each application from Log on to Log off. The AFMs are both defined in packages that can be transferred.

If you wish to modify the flow significantly, you can transfer the packages and edit the AFMs directly (see section 4).

The design and content of the workspace makes no assumptions about the scope, purpose or industry classification of the service requests.

It does however make certain assumptions about the basic structure and flow of the user interface.

This document provides an overview of the standard capabilities of the workspace. Other documents in the *Using the Contact Centre Workspace* series describe how to extend and customise the workspace to meet the needs of your particular implementation.

2 Contact centre



The Contact centre application manages the user's log on/off and navigation between the application's desktops which provide access to functionality:

- for taking/making calls;
- supporting service requests (e.g. loan applications);
- performing workflow tasks.

The Contact centre application is made up of three desktops (also referred to as application contexts). These desktops are described in the following sections.

2.1 Agent desktop

2.1.1 Access

The Agent desktop is presented as a result of a successful log on. The log on process requires the agent to provide a user name and password.



Once the user name and password have been authenticated (see section 4.1), the agent must select a role.

2.1.2 Capability

The desktop provides the launch point for most of the functionality provided by the application. It represents both the point of entry and the point of rest between agent activities. This desktop is also used to present information to agents that relates to agent activities and administration.

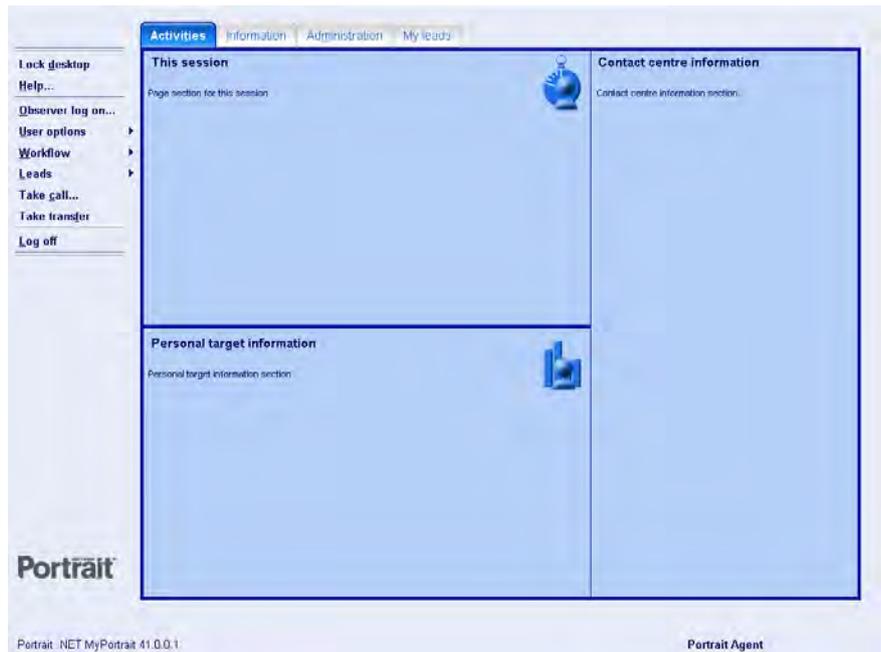
Each operation run from this desktop should register one (or more) engagement actions as part of the agent session engagement. The agent session engagement is started when the agent logs on to the application and is ended when the agent logs off.

For engagement history purposes, the agent is treated as the *Effecting party*. Whenever you add actions to an engagement, you should provide the ID and Type of the agent party as the values for the *Effecting party ID* and *Type* inputs.

In the case of operations run from the Agent desktop, the agents is also the party *affected*. Consequently you should also provide the agent ID and Party type as the values for the *Affected party ID* and *Type*.

The Contact centre provides functionality to allow a second agent to log on to the same session in order to observe the primary agent. This can be used to support agent training as well as to provide greater access control to certain capabilities. The second agent is treated as the *Observing party*. The presence of the observing party can be used by an implementation as a criterion in managing operation availability.

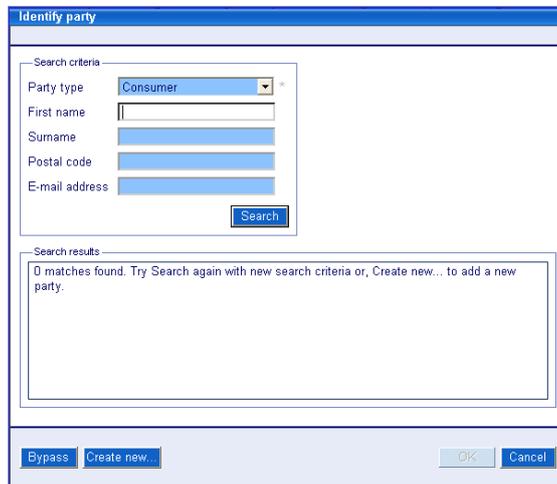
2.1.3 Layout (out of the box)



2.2 Engagement desktop

2.2.1 Access

The *Engagement desktop* is reached as a result of identifying a party in the *Take call* operation.



2.2.2 Capability

The desktop is used to present the functionality that is available during an engagement (interaction with another party). All the activities performed in this desktop are bracketed within a single engagement record that is started as soon as the agent launches the *Take call* operation in the Agent desktop and is ended when the agent selects **Hang up** in the Wrap up dialog and returns to the Agent desktop.

Each operation run from this desktop should register one (or more) engagement actions as part of this engagement.

The application flow (override-able) is for the agent to identify a party. The application then builds tabs relating to that party in the engagement desktop.

The party identified is, for engagement history purposes, treated as the *Requesting party*. You should not implement any operations that change this state for the duration of the engagement.

In many scenarios, the services performed are for the party identified. In this case, the *Requesting party* is also the *Affected party*. However, the Contact centre application provides functionality that allows the identified party to act on **another party's behalf**. In these cases, the *Affected party* is this other party.

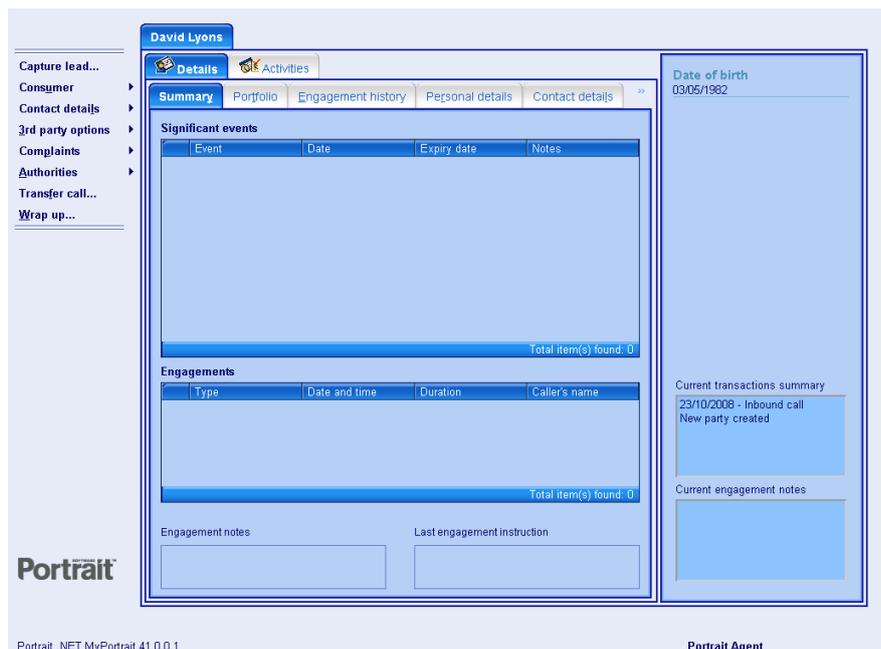
When recording engagement history you should always try to provide the *Affected party* information since this is the key relationship that is used when displaying engagement history in the contact centre.

By default the contact centre application manages the following globals (*ID*, *Type* and *Known as* for each) for you to use as input values for engagement action records:

- **Current party** – This set of globals is controlled by a Party tab. When the tab is clicked the globals are set. Clicking on another party tab will change the values. These globals should be treated as the *Affected party*. The affected party may change several times during the engagement.
- **Effecting party** – This set of globals is managed by the log on and remains unchanged for the entire agent session. These globals should be treated as the *Effecting party*.
- **Requesting party** – This set of globals is managed by the party identification process in the *Take call* operation as well as the *Re-identify* operation. They should remain unchanged for the duration of an engagement (unless re-identification is required). These globals should be treated as the *Requesting party*.
- **Observing party** – This set of globals is managed by the Observer log on/off process. They are set when the observing agent logs on and cleared when that agent logs off. These globals should be treated as the *Observing party*.

Affected party globals are also provided, but these should only be used if it is important to distinguish between the current party and the affected party.

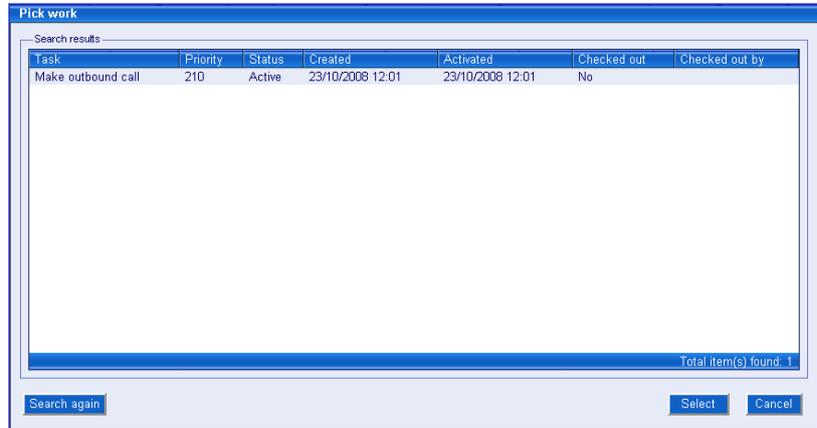
2.2.3 Layout (out of the box)



2.3 Workflow desktop

2.3.1 Access

This desktop is reached as a result of agent selecting *Pick work* or *Get work* from the Agent desktop.



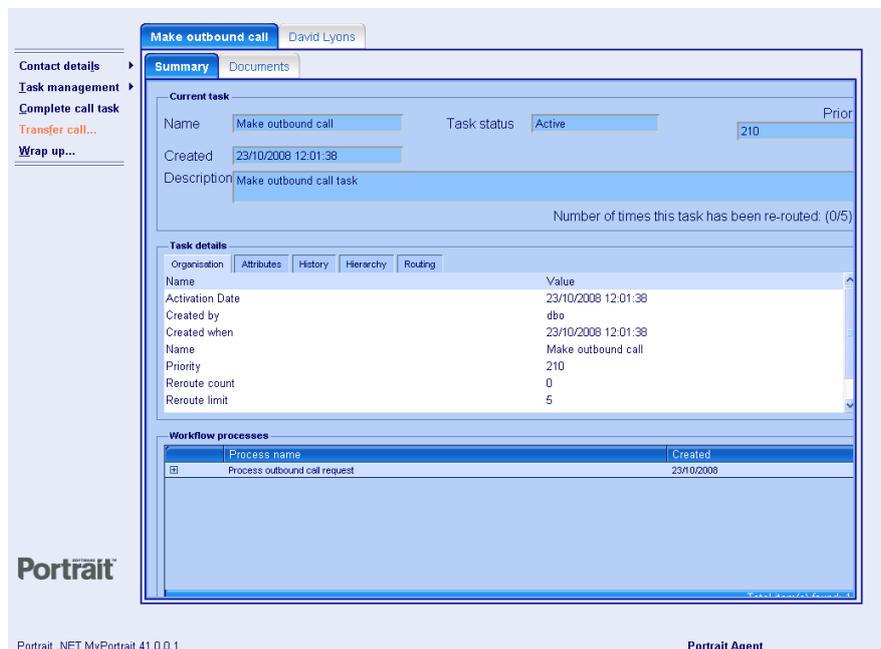
The workflow desktop is intended to present the functionality for performing workflow tasks (some of which may result in an engagement).

2.3.2 Capability

Once a task has been identified, the desktop is built to display information relating to the task. The desktop can also display information about other entities related to the task such as parties, contracts and cases.

In order to allow a workflow related call to be transferred from one agent to another, the application controls a separate set of globals for the *Task agent*. These globals will be the same as the *Effecting party* globals until the task related call is transferred. After this, the *Task agent* globals hold the information about the agent that actually checked out the task. To be certain that your task operations can function correctly after a transfer, you should always use these globals to supply the inputs to your *Get task* and *Complete task* nodes.

2.3.3 Layout (out of the box)

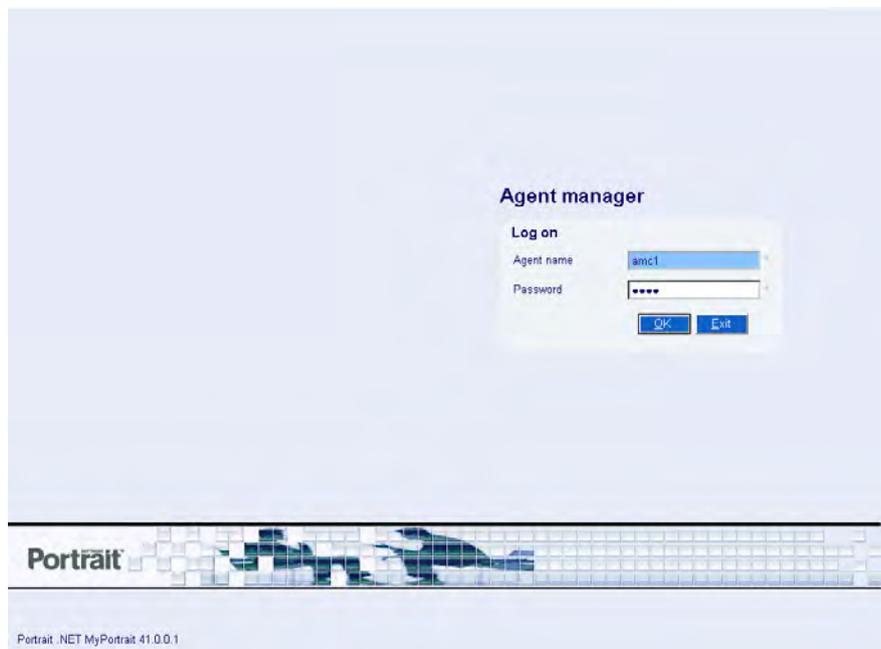


3 Agent manager



3.1 Access

The Agent manager desktop is presented as a result of a successful log on. The log on process requires the agent to provide a user name and password.



The user must have a role of *Supervisor* to log on to this application.

3.2 Capability

The desktop provides the launch point for operations that allow:

- Creation, amendment and deletion of agents;

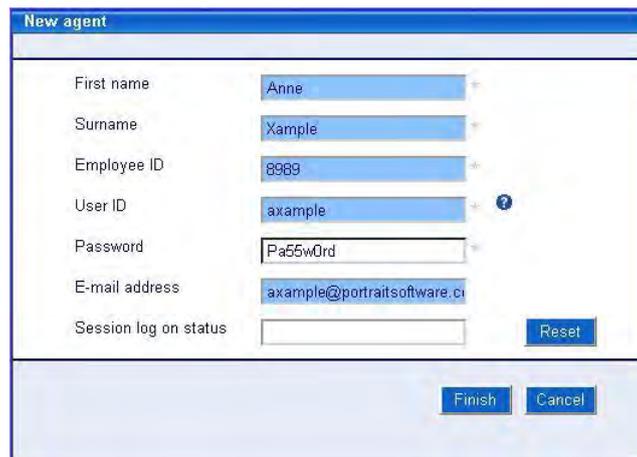
- Creation, amendment and deletion of teams;
- Association of agents to teams and roles.

3.2.1 Agents

Agent manager allows agents to be maintained. The operations provided are:

- Add agent...
- Amend performance alert thresholds...
- Amend agent...
- Assign roles...
- Assign teams...
- Expire agent...
- View agent's history

Once an agent has been created various characteristics can be assigned some of which are optional.

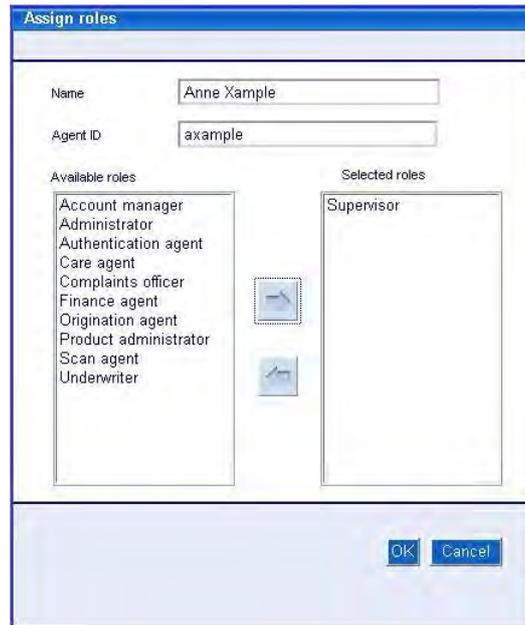


The screenshot shows a 'New agent' form with the following fields and values:

Field	Value
First name	Anne
Surname	Xample
Employee ID	8888
User ID	axample
Password	Pa55w0rd
E-mail address	axample@portraitsoftware.co.uk
Session log on status	

Buttons: Reset, Finish, Cancel

In order to access the out of the box application an agent is expected to be associated to at least one role.



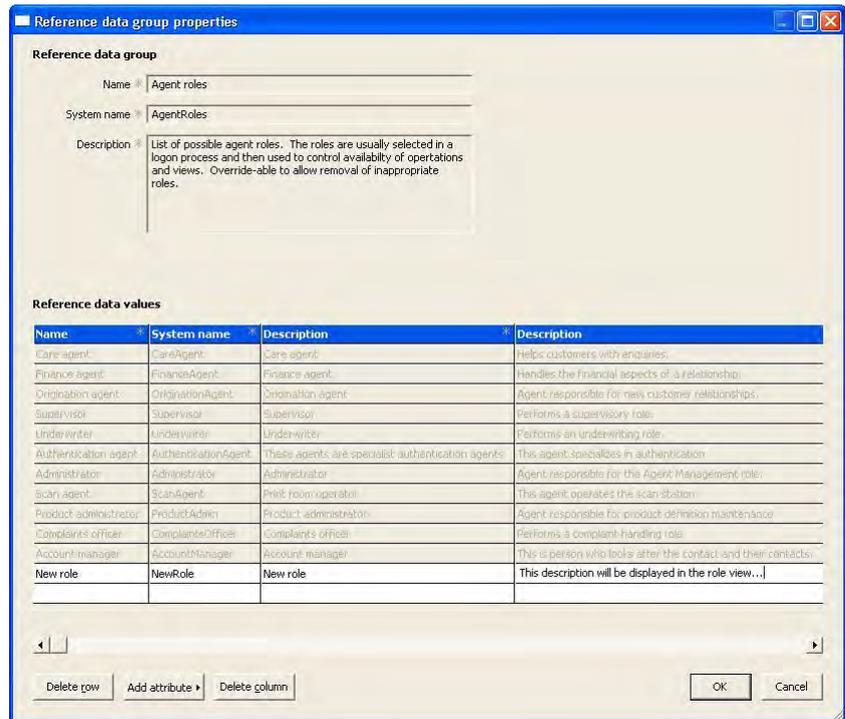
The optional characteristics that can also be assigned to an agent are:

- Performance alert thresholds
- Telephony details
- Teams

3.2.2 Roles

Agent manager allows agents to be assigned to or removed from role(s). The list of agent roles is maintained as reference data that can be extended via an overlay and supplied roles can be overwritten. To extend the roles perform the following steps:

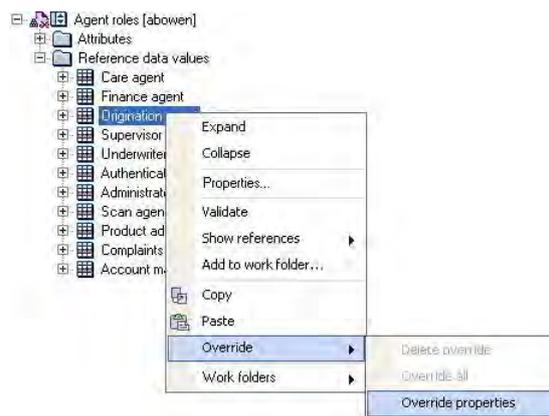
- 1 Select the *Agent roles* RDG in the Explorer view. The RDG can be found in the *Supporting definitions/Reference data* folder.
- 2 From the context menu select **Open item**. This will open the RDG in the object viewer.
- 3 From the context menu select **Overlay→Create overlay**.
- 4 Right click and select **Properties...**. The *Reference data group properties* dialog is displayed.
- 5 Add required reference data items.



- 6 Click on the **OK** button.
- 7 Validate.
- 8 Add to source control and publish.

To amend existing agent roles repeat steps 1 to 3 then undertake the following steps:

- 1 Expand the *Agent roles* RDG and the Reference data values branch in the object viewer. Then select the RDI that you wish to amend.
- 2 From the context menu select **Override**→**Override properties**.



- 3 From the context menu select **Properties**.
- 4 In the *Reference data group properties* dialog select the appropriate row in the grid and change the Name column as required.
- 5 Click on the **OK** button.
- 6 Validate.
- 7 Add to source control and publish.

3.2.3 Teams

Agent manager allows agents to be assigned or removed from team(s). Teams are saved as an instance of a party configurable business entity (CBE) of type Team. The operations provided are:

- Add team...
- Amend team...
- Assign agents to team...
- Expire team...

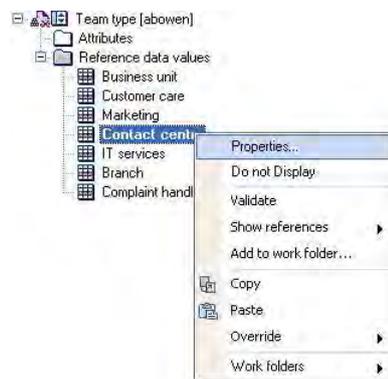
Workflow tasks can be assigned to a team enabling all members to get the task.

When creating a team assigning a Team type is mandatory. The list of team types is maintained as reference data that can be extended via an overlay and supplied types can be overwritten. To extend the team types roles perform the following steps:

- 1 Select the *Team types* RDG in the Explorer view. The RDG can be found in the *Supporting definitions/Reference data* folder.
- 2 From the context menu select **Open item**. This will open the RDG in the object viewer.
- 3 From the context menu select **Overlay→Create overlay**.
- 4 Right click and select **Properties...**. The *Reference data group properties* dialog is displayed.
- 5 Add required reference data items.
- 6 Click on the **OK** button.
- 7 Validate.
- 8 Add to source control and publish.

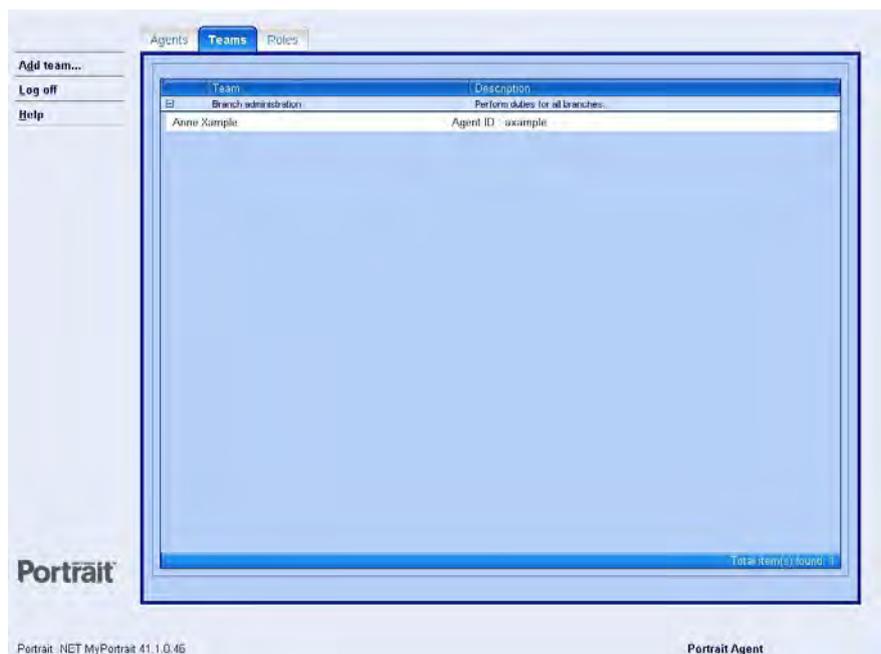
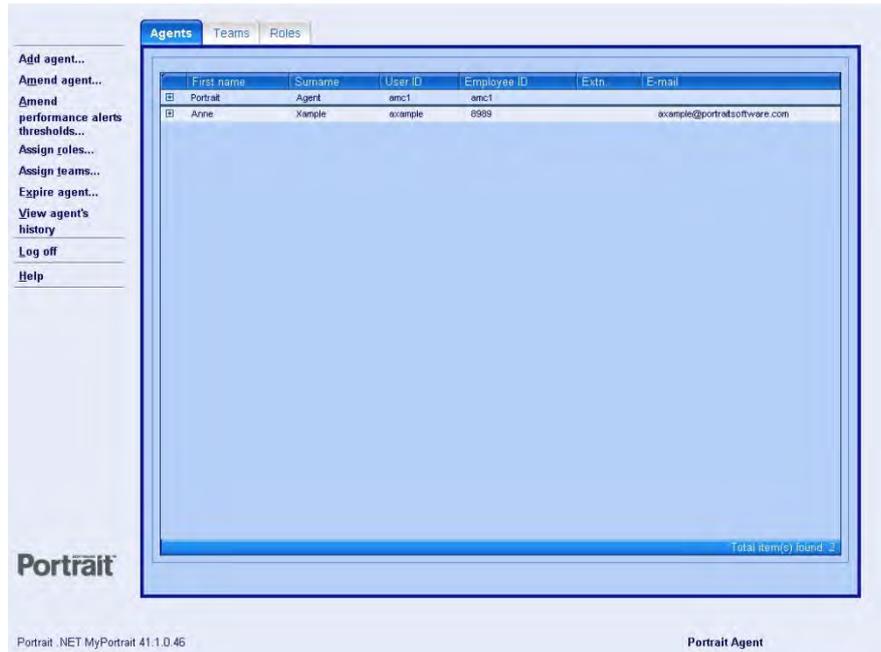
To amend existing team types roles repeat steps 1 to 3 then undertake the following steps:

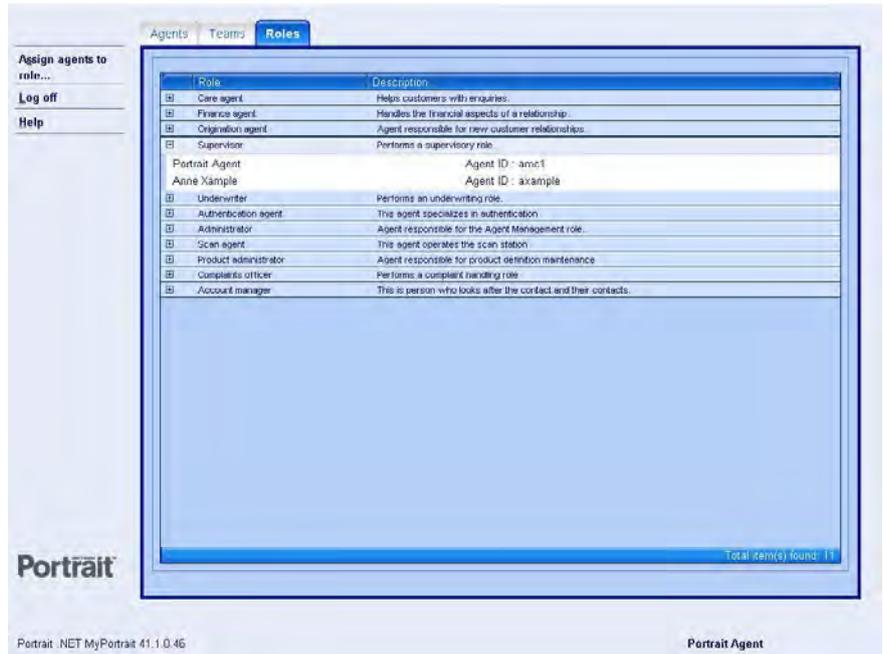
- 1 Expand the *Team types* RDG and the Reference data values branch in the object viewer. Then select the RDI that you wish to amend.
- 2 From the context menu select **Override→Override properties**.
- 3 From the context menu select **Properties**.



- 4 In the *Reference data group properties* dialog select the appropriate row in the grid and change the Name column as required.
- 5 Click on the **OK** button.
- 6 Validate.
- 7 Add to source control and publish.

3.3 Layout (out of the box)





4 Setting application values

The application framework model (AFM) used by the *Contact centre workspace* uses a data manipulation node to set some basic values that are used throughout the application.

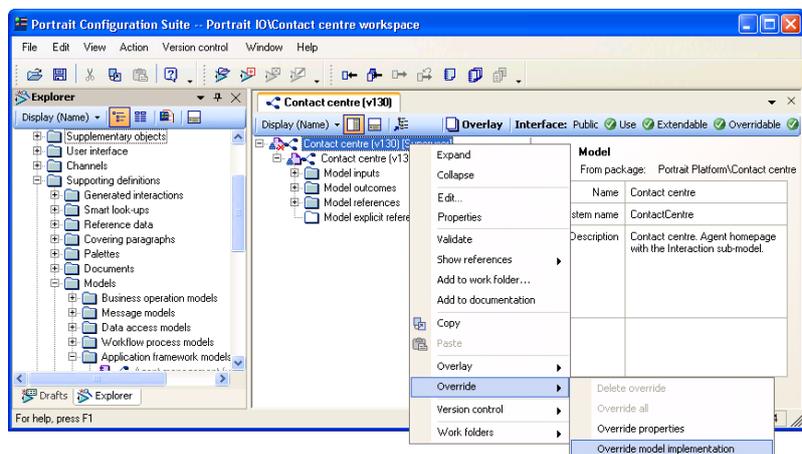
The *Contact centre* AFM is provided by the Contact Centre package. As such you cannot edit it unless you take one of the following actions:

- Copy the AFM and adjust the AIT_HRZ_ContactCentre.hta
- Override the standard AFM
- Transfer the Contact Centre package so that you can edit its content.

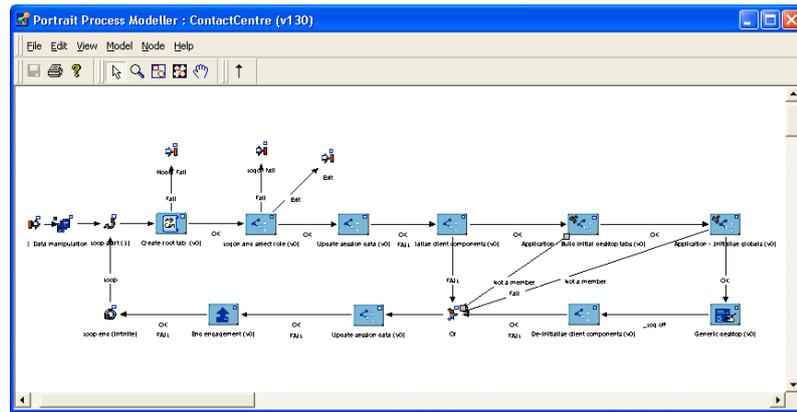
If you transfer the package, you will lose the opportunity to take updates in the future and you should not take this step unless you plan to modify its contents significantly

This document assumes that you will override the AFM by performing the following steps:

- 1 Select the Contact centre AFM in the Explorer view. The AFM can be found in the *Supporting definitions/Models/Application framework models* folder.
- 2 From the context menu select **Open item**. This will open the model in the object viewer.
- 3 From the context menu select **Overlay**→**Create overlay**.
- 4 From the context menu select **Override**→**Override model implementation**.



- 5 From the context menu select **Edit...**. This will launch the process modeller.



- 6 Select the first data manipulation node.
- 7 From the context menu select **Map data inputs...** You are now ready to edit the settings described in the following section.

4.1 Suppressing logon check

The log on process has been specifically designed to prevent agents from logging on more than once. You can suppress this behaviour by modifying the *Ignore session status* input value.

Node Inputs	Type	Context
Telephony general ("Telephony", "G...")	DataObject	
Ignore session status	String	"True"
Setup ("Application", "ContextSetup")	DataObject	
Application ("PortraitApplications")	Reference	"ContactCentre"
Automatic operation name	String	""
Channel ("Channels")	Reference	"ServiceCentre"
Default role ("AgentRoles")	Reference	
Engagement type ("EngagementTy...")	Reference	"AgentSession"
Language ("ImplementationLangua...")	Reference	"Default"

If you set the input to "True" then agents will be allowed to log on to the application many times without constraint.

4.2 Application name

The *Application* input is used to control that name that is displayed on the splash screen as well as being used as a criterion value in many smart lookups (especially operation availability).

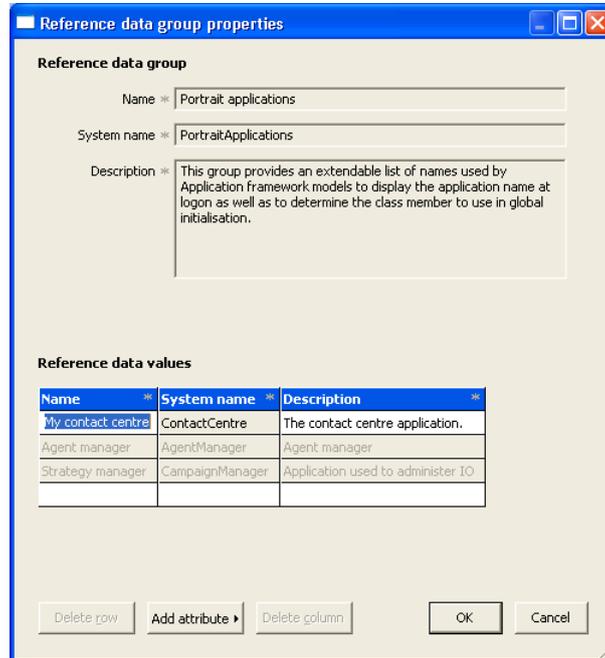
The input is a reference value. Consequently you can only change it to another value in the *Applications* reference data group (RDG).

Because of the impact on smart lookups, you should not be tempted to change this value for cosmetic purposes only.

If all you need to do is change the display name of the application, use the following steps:

- 1 Select the *Portrait applications* RDG in the Explorer view. The RDG can be found in the *Supporting definitions/Reference data* folder.
- 2 From the context menu select **Open item**. This will open the RDG in the object viewer.
- 3 From the context menu select **Overlay→Create overlay**.
- 4 Select the *Contact centre* reference data item (RDI) in the object viewer. The RDI can be found in the *Reference data items* folder.

- 5 From the context menu select **Override**→**Override properties**.
- 6 From the context menu select **Properties**.
- 7 In the *Reference data group properties* dialog select the Contact centre row in the grid and change the Name column as required.



- 8 Click on the **OK** button.
- 9 Validate.
- 10 Add to source control and publish.

4.3 Setting the channel

The *Channel* input is used to set the *Channel* global. This global has significant impact on the availability of operations, the runtime selection of user interface components as well as the paths that some models take.

You should NOT alter this value without an excellent (even exceptional) reason.

4.4 Setting the Agent engagement type

The standard contact centre provides operations that are executed for the Agent (as opposed to the Consumer). For example, *Change role*.

Each of the operations adds an engagement action to record the activity. This engagement action is added to an engagement that starts when the agent logs on and ends when the agent logs off. This engagement represents the agent's session and can be used to track the agent's activities during the session.

This input determines the type of this engagement. It must be the system name of one of the engagement types identified in the *Engagement types* reference data group (RDG). By default the value is set to *AgentSession*.

4.5 Setting the language

The Portrait Configuration Suite provides various components that use language to determine run time capability. In particular, Generated Interactions (GIs),

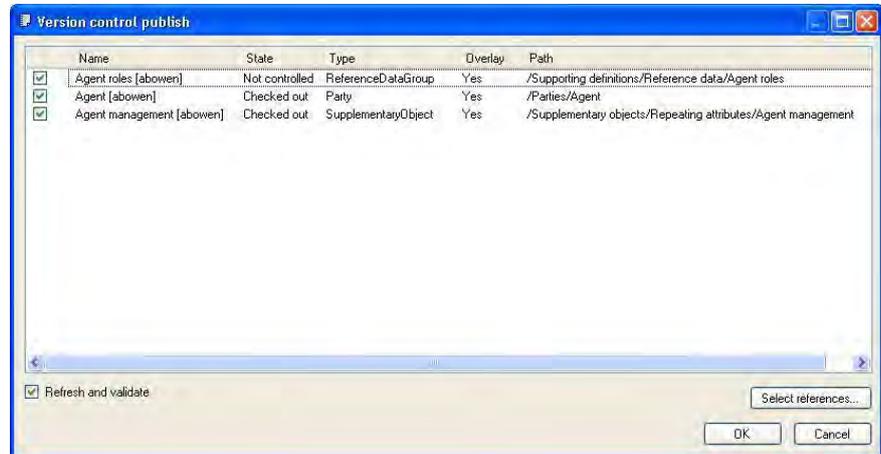
Custom Interactions (CIs) and Custom Controls (CCs) use language as a criterion to determine which page to display and which transformation to run.

This input is used to set the *Language* global which, in turn, is used to make these selections. If you are running a single language implementation, you should set this value to *Default*. For multi-language implementations you can also use *Default* or you may choose to add processing that will take the language **from the Agent's profile or some other suitable source**.

The value must be the system name of one of the languages identified in the *Implementation languages* RDG.

5 Publish and Deploy

You should publish all the configuration changes that have resulted from the steps described in this document and then deploy the changes into a suitable database.



Before you can run the integrated functionality it will be necessary to create a suitable Portrait installation using the Portrait Build Environment (see *Foundation_SDK_User_Guide.pdf*).