eServices 8.1

Social Media Solution Guide Wiki Redirect

ALERT: This document is available as a PDF only to support searches from the Technical Support Knowledge Base. Click here (eServices Social Media Solution Guide) to be redirected to the content in its original format.

Contents

Articles

References

eServices Social Media Solution Guide

eServices Social Media Solution Guide

Table of Contents

Purpose: Welcome to the eServices 8.x Social Media Solution Guide.

About This Guide

This guide covers areas of information that are required for using the Genesys Social Engagement product. Topics cover the components of Genesys Social Engagement, other associated Genesys products, and some non-Genesys items.

This guide describes the following. Start your reading with the Genesys Social Engagement Overview.

- General information:
  - Genesys Social Engagement Overview
  - Related eServices Documentation Resources
- Deploying Social Messaging Server
- Using and understanding the Sample Business Processes for Social Media
- Deploying the Genesys Desktop Plugin
- Deploying the Interaction Workspace Plugins for Social Media

These pages are valid for all 8.x releases of this product.

Genesys Social Engagement Overview

eServices Social Media Solution Guide

Table of Contents

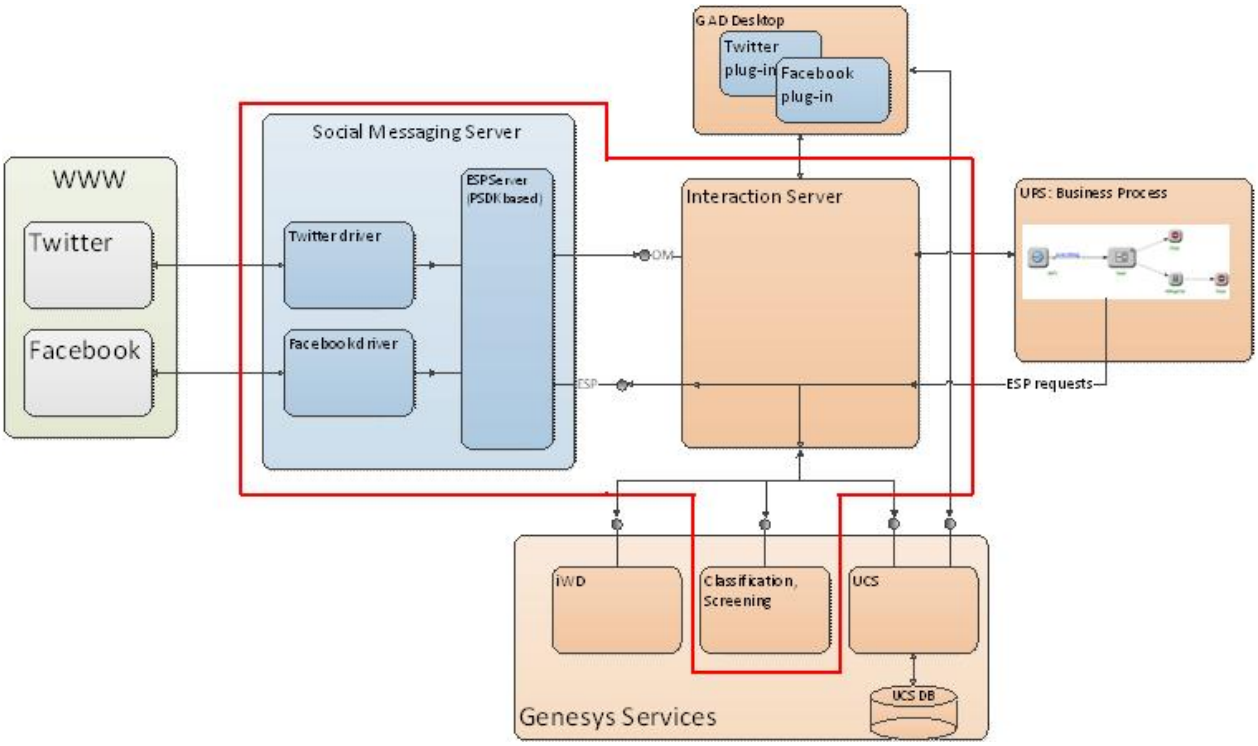
Purpose: To provide a high-level description of what Genesys Social Engagement is and what it does. These pages are valid for all 8.x releases of this product.

What It Is

Genesys eServices takes interactions (which may be defined as attempted acts of communication) and gives them a uniform representation that can be processed in the many ways available in the Genesys suite of products.

Genesys Social Engagement is the latest extension of the range of media that eServices can process. The system monitors a social media site, gathers items that fit a defined profile, and converts them into Genesys interactions. The 8.x releases provide interfaces with Facebook and Twitter.

Here is a simplified architecture diagram:



The red line encloses the components that belong to the eServices line of products.

In place of the Twitter driver and Facebook driver shown in the diagram, you can create a custom media channel driver.

Social Messaging Server

Home > eServices > eServices Social Media Solution Guide

Purpose: Deploying Social Messaging Server

Social Messaging Server is the eServices component that interfaces with social media sites to bring interactions into the Genesys system.

Pages in this chapter describe what you must do to deploy and use Social Messaging Server, including registering applications with the social media sites Facebook and Twitter.

Start your reading with either Creating a Facebook Application or Creating a Twitter Application.

You can also use Social Messaging Server with a Custom Media Channel Driver.

Creating a Facebook Application

eServices Social Media Solution Guide

Table of Contents

Home > eServices > eServices Social Media Solution Guide > Social Messaging Server

Purpose: To create a Facebook application that Social Messaging Server can connect to.

The APIs and other features of social media sites may change with little warning. The information provided on this page was correct at the time of publication (31 January 2012).

Procedure

1. Create a Facebook account if you do not have one. The name of this account will appear as the developer of the application you create.
2. On the web page http://www.facebook.com/developers/#!/developers/createapp.php, enter a name and click Create Application.
3. On the resulting page,
   - Make a note of the Application ID and Application Secret.
   - Fill in other attributes, then click Save Changes.
4. Enter the following URL into a browser, substituting the Application ID for APPLICATION_ID: https://graph.facebook.com/oauth/authorize?client_id=APPLICATION_ID&redirect_uri=http://www.facebook.com/connect/login_success.html&scope=manage_pages,user_status,read_stream,read_insights,publish_stream,offline_access,create_event
5. On the resulting Request for Permission page, click Allow.
6. The resulting page, displaying the single word 'success', has a URL that ends with `code=` followed by a long alphanumeric string. Make a note of that string.
7. Enter the following URL into a browser, substituting the Application ID for APPLICATION_ID, the Application Secret for APPLICATION_SECRET, and the code from the preceding step for CODE: https://graph.facebook.com/oauth/access_token?client_id=APPLICATION_ID&redirect_uri=http://www.facebook.com/connect/login_success.html&client_secret=APPLICATION_SECRET&code=CODE
8. This produces a screen with a single line of text consisting of `access_token=` followed by a long alphanumeric string. Make a note of this string. It will be the value of the Social Messaging Server configuration option `x-facebook-access-token`.

Next Steps

Deploy Social Messaging Server with a Facebook Channel

Creating a Twitter Application

Purpose: To create a Twitter application that Social Messaging Server can connect to.

The APIs and other features of social media sites may change with little warning. The information provided on this page was correct at the time of publication (31 January 2012).

Procedure

1. On the website http://dev.twitter.com/, click `Create an app.`
2. On the resulting page, sign in or create an account.
3. On the Create an Application page, fill in Application Name, Description, and other attributes (you must enter something for Application Website, but it need not be a real URL).
4. Review the resulting Settings page.
5. Copy the values of `Consumer key` and `Consumer secret` and assign them to the `x-consumer-key` and `x-consumer-secret` options of your Social Media Server object, as described in Deploy Social Messaging Server with a Twitter Channel.
6. Click `Create my access token` in the `Your access token` section. `Access Token` and `Access Token Secret` appear in the same section (you may have to refresh the page). Copy their values and assign them to the `x-access-token` and `x-access-token-secret` options of your Social Media Server, as described in Deploy Social Messaging Server with a Twitter Channel.

Next Steps

Deploy Social Messaging Server with a Twitter Channel

Deploy Social Messaging Server

Purpose: Deploy Social Messaging Server

Create a Social Messaging Server application object

1. Create an Application object for Social Messaging Server if it does not already exist.
   1. Import the Social Messaging Server application template from the CD.
   2. Create a new Application object based on the template.
2. Open the `Properties` dialog box of the Application object.
3. On the `Server Info` tab:
   1. In the `Host` box, enter the name of the desired host.
   2. In the `Communication Port` box, enter the port Social Messaging Server will use.
4. On the `Start Info` tab enter some characters in the `Working Directory`, `Command Line`, and `Command Line Arguments` fields. These characters will be over-written with the correct values during the installation, but they cannot be left blank at this point.
5. On the `Connections` tab, add a connection to Interaction Server.
6. If this is for a multi-tenant environment, add the tenant(s) on the `Tenants` tab.

7. Open your Interaction Server Application and add a connection to Social Messaging Server. This connection must be of the type known as `simple`.

Install Social Messaging Server (release 8.1.0 and previous)

Prerequisite: Java Environment and Libraries for eServices and UCS 8.1.0

1. Launch `Setup.exe` (Windows) or `install.sh` (Linux, Solaris, AIX).
2. Enter the login information for your Configuration Server:
    1. Host
    2. Port
    3. User
    4. Password
3. Select the appropriate Social Messaging Server Application object from the list.
4. Click `Install` (Windows) or press `Enter` (Linux, Solaris, AIX).

Install Social Messaging Server (release 8.1.1 and later)

Installation procedures differ slightly depending on your operating system.

Windows

Prerequisite: JDK 1.6 In this release, Social Messaging Server is supported with Java 1.6.0_26 or higher (within the 1.6 family). The 64-bit version is required for 64-bit Windows.

1. Locate the `Setup.exe` for Social Messaging Server on the product CD.
2. Double-click `Setup.exe`.
3. Select Java.
4. Enter the login information for your Configuration Server:
    1. Host
    2. Port
    3. User
    4. Password
5. Select the appropriate Social Messaging Server Application object from the list.
6. Confirm selection of JDK.
7. Click `Install`.

Linux, Solaris, AIX

Prerequisite: JRE 1.6 For Oracle, Java version 1.6.0_26 or higher (within the 1.6 family) is supported. For IBM, Java (OS AIX) version 1.6.0 build pap6460sr9fp2-20110627_03(SR9 FP2) for AIX or higher is supported.

1. Locate the `install.sh` file for Social Messaging Server.
2. Use the command `>install.sh` to start the installation script.
3. On request, enter the absolute path for the directory of the Java executable.
4. Press `Enter` to confirm the host name for the installation.
5. Enter the login information for your Configuration Server:
    1. Host
    2. Port
    3. User
    4. Password

6. From the list of applications, select one and enter its number in the list.
7. Press `Enter` to confirm the suggested destination directory, or choose another one.
8. Answer other questions, if required.

Proxy Server Support

Optionally, add proxy server support.

Next Steps

Continue deployment by adding a Facebook channel, a Twitter channel, or a Custom Media Channel.

Deploy Social Messaging Server with a Facebook Channel

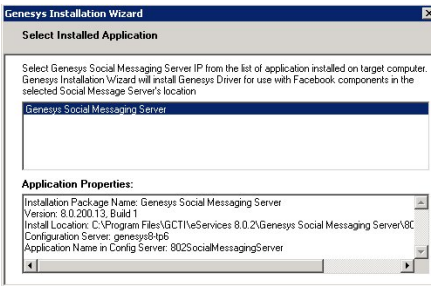
Purpose: To deploy Social Messaging Server to work with Facebook.

For a Facebook channel, you need two installation packages: Social Messaging Server and Genesys Driver for Use with Facebook. The Driver adds Facebook-specific features to Social Messaging Server and does not require its own Application object in the Configuration Server database. You can also create a Custom Media Channel Driver.

Unlike some other eServices components, Social Messaging Server does not require Java Environment and Libraries for eServices and UCS.

Procedure

1. Deploy Social Messaging Server.
2. Run the installation for Genesys Driver for Use with Facebook, selecting the desired Social Messaging Server object:



Select your social Messaging Server Object

3. Locate the `driver-for-facebook-options.cfg` file, which the installation places in the `\<Social Messaging Server application>\media-channel-drivers\channel-facebook` directory.
4. In Configuration Manager, open your Social Messaging Server Application, go to the `Options` tab, and import `driver-for-facebook-options.cfg`.
5. For information on how to configure the options, see the following section.

Configuration Options

Section channel-<any name>

x-thread-pool-size

Type: int

Minimum Value: 25

Maximum Value: 2147483647

Values: Maximum number of threads in application thread pool

Default Value: 50

Takes effect: After restart

Description: Maximum number of threads in application thread pool.

All the monitors and ESP request processors use a thread from the thread pool during processing, and place it back in the queue during idling. A new thread is created for monitor sampling update and individual ESP requests.

x-access-token

Type: String

Minimum Value: N/A

Maximum Value: N/A

Values: String with Facebook access token

Default Value: N/A

Takes effect: After restart

Description: Default Facebook access token that is used by ESP request processors for all communications with Facebook. The page Creating a Facebook Application describes the procedure for obtaining the value for the `x-access-token` option.

x-itx-resubmit-delay

Type: Int

Minimum Value: 0

Maximum Value: Max unsigned int
Values: Number of seconds between each resubmission of interaction
Default Value: 3
Takes effect: After restart
Description: Specifies the period of time that the application pauses between each resubmission of interactions.

x-itx-resubmit-ntimes
Type: Int
Minimum Value: 1
Maximum Value: 2147483647
Values: Number of resubmission attempts
Default Value: 10
Takes effect: After restart
Description: Specifies the number of times that the application tries to resubmit interactions.

x-sampling-time-buffer
Type: Int
Minimum Value: 0
Maximum Value: 172800000
Values: Backward buffer time in seconds for sampling period
Default Value: 180
Takes effect: After restart
Description: Specifies the time that is deducted from the start and end time of each sampling period in the monitor.

x-submit-comments-itx
Type: Boolean
Minimum Value: N/A
Maximum Value: N/A
Values: true or false
Default Value: false
Takes effect: After restart
Description: Ordinarily, for a Facebook post, the system creates a single interaction containing the post plus its associated comments. If this option has a value of `true`, the system also creates a separate interaction (with the Facebook type `comment`) for each comment.

x-first-sampling-period
Type: Int
Minimum Value: 1
Maximum Value: 172800000
Values: Sampling period in seconds
Default Value: 180
Takes effect: After restart
Description: First sampling period that is used when the monitor starts.

x-sampling-period
Type: Int
Minimum Value: 1
Maximum Value: 172800000
Values: Sampling period in seconds; must be greater than or equal to `x-sampling-time-buffer`
Default Value: 180
Takes effect: After restart

Description Sampling period that is used by the monitor.

x-max-posts-per-fql-request

Type Int
Minimum Value 50
Maximum Value 2147483647
Values Maximum number of stream posts that are retrieved per FQL request
Default Value 50
Takes effect After restart
Description Maximum number of stream posts that are retrieved per FQL request.

x-max-comments-per-fql-request

Type Int
Minimum Value 50
Maximum Value 2147483647
Values Maximum number of comments that will be retrieved per FQL request
Default Value 50
Takes effect After restart
Description Maximum number of comments that will be retrieved per FQL request.

x-submit-internal-itx

Type Boolean
Minimum Value N/A
Maximum Value N/A
Values true or false
Default Value false
Takes effect After restart
Description If this option has a value of `false`, Social Messaging Server takes only messages (posts or comments) that have new content created by the customer and submits them as interactions. It does not submit messages that were created prior to monitor read times or messages that were not created by the customer. If it has a value of `true`, it submits all messages regardless of creator or time of creation.

Section channel-<any name>-monitor-<any name>

access-token

Type String
Minimum Value N/A
Maximum Value N/A
Values String with Facebook access token
Default Value x-access-token option value in driver options
Takes effect After restart
Description Default Facebook access token that will be used by ESP requests processors for all communications with Facebook.

itx-resubmit-delay

Type Int
Minimum Value 0
Maximum Value Max unsigned int
Values Number of seconds between each resubmission of an interaction
Default Value x-itx-resubmit-delay
Takes effect After restart

Description Specifies the length of time that the application pauses between each resubmission of an interaction.

itx-resubmit-ntimes

Type Int
Minimum Value 1
Maximum Value 2147483647
Values Number of resubmission attempts
Default Value x-itx-resubmit-ntimes
Takes effect After restart
Description Specifies the number of times that the application tries to resubmit interactions.

sampling-time-buffer

Type Int
Minimum Value 0
Maximum Value 172800000
Values Backward buffer time in seconds for sampling period
Default Value x-sampling-time-buffer
Takes effect After restart
Description Buffer time will be deducted from the start and end time of each sampling period in the monitors.

submit-comments-itx

Type Boolean
Minimum Value N/A
Maximum Value N/A
Values true or false
Default Value x-submit-comments-itx
Takes effect After restart
Description If this option has a value of `true,` a new interaction with the Facebook type `comment` is created for each comment.

first-sampling-period

Type Int
Minimum Value 1
Maximum Value 172800000
Values Sampling period in seconds
Default Value x-first-sampling-period
Takes effect After restart
Description First sampling period that is used when the monitor starts.

sampling-period

Type Int
Minimum Value 1
Maximum Value 172800000
Values Sampling period in seconds; must be greater than or equal to `sampling-time-buffer`
Default Value x-sampling-period
Takes effect After restart
Description Sampling period that is used by the monitor.

max-posts-per-fql-request

Type Int
Minimum Value 50
Maximum Value 2147483647

Values Maximum number of stream posts that will be retrieved per FQL request
Default Value x-max-posts-per-fql-request
Takes effect After restart
Description Maximum number of stream posts that will be retrieved per FQL request.

max-comments-per-fql-request
Type Int
Minimum Value 50
Maximum Value 2147483647
Values Maximum number of comments that will be retrieved per FQL request
Default Value x-max-comments-per-fql-request
Takes effect After restart
Description Maximum number of comments that will be retrieved per FQL request.

submit-internal-itx
Type Boolean
Minimum Value N/A
Maximum Value N/A
Values true or false
Default Value Duplicates the value of `x-submit-internal-itx` in the `channel-<any name>` section
Takes effect After restart
Description If this option has a value of `false`, Social Messaging Server takes only messages (posts or comments) that have new content created by the customer and submits them as interactions. It does not submit messages that were created prior to monitor read times or messages not created by the customer. If it has a value of `true`, it submits all messages regardless of creator or time of creation.

monitor-type
Type String
Minimum Value N/A
Maximum Value N/A
Values generic or search or event
Default Value generic
Takes effect After restart
Description Specifies the monitor type.

id
Type String
Minimum Value N/A
Maximum Value N/A
Values ID of Facebook object for monitoring
Default Value generic
Takes effect After restart
Description The Facebook object can be page, user, application, event, or group.
This option is mandatory when the monitor type is `generic`.

query
Type String
Minimum Value N/A
Maximum Value N/A
Values Facebook search query
Default Value generic

Takes effect After restart
Description This option is mandatory when the monitor type is `search`.

Requirement for Posting and Commenting

In order for the Genesys Desktop Plugin to display the `Delete`, `Delete Post`, `Comment`, and `Share` buttons, it checks the `_facebookCanComment` and `_facebookCanDeleteComments` keys in the interaction's attached data.

This means that, if the Facebook account holder did not set permissions to let everybody post/comment on the account holder's wall, the agent must "like" the poster's page or add him as a friend to be able to post on the wall. The same holds for an autoresponse if it is included into the strategy: if we do not "like" the poster, or if he or she did not set permissions to let everybody post/comment on the wall, our post/comment will fail and a `Facebook POST failed` exception will be written in the log.

Deploy Social Messaging Server with a Twitter Channel

Purpose: To deploy Social Messaging Server to work with Twitter.

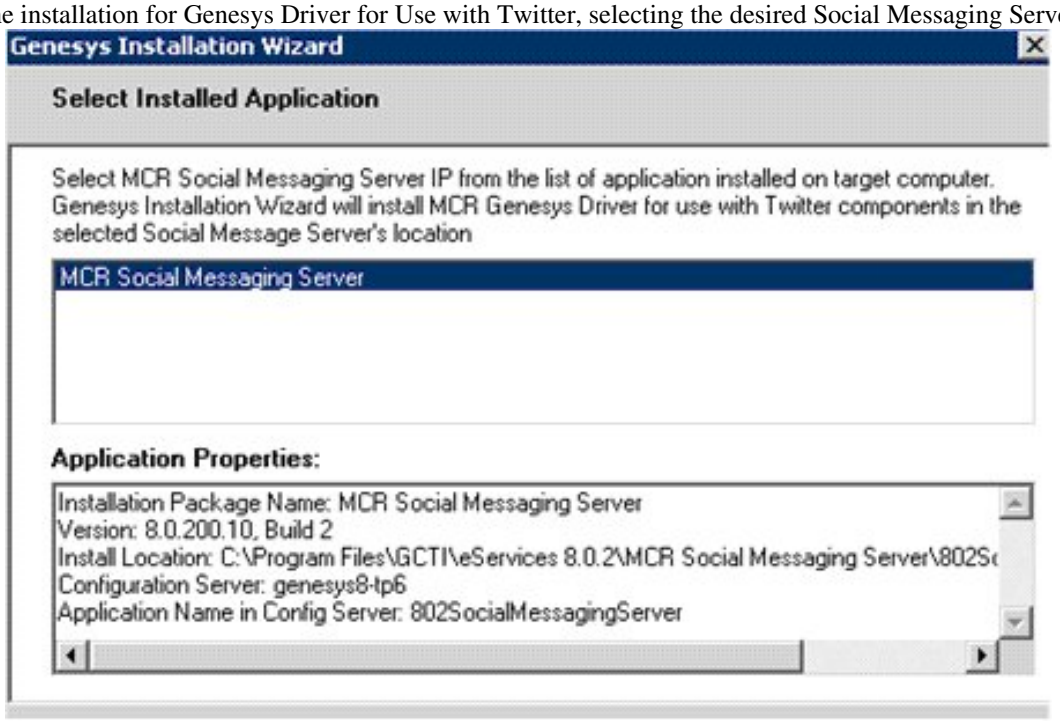
Overview

For a Twitter channel, you need two installation packages: Social Messaging Server and Genesys Driver for Use with Twitter. The Driver adds Twitter-specific features to Social Messaging Server and does not require its own Application object in the Configuration Server database. You can also create a Custom Media Channel Driver.

Unlike some other eServices components, Social Messaging Server does not require Java Environment and Libraries for eServices and UCS.

Procedure

1. Deploy Social Messaging Server.
2. Run the installation for Genesys Driver for Use with Twitter, selecting the desired Social Messaging Server object:



3. Locate the `driver-for-twitter-options.cfg` file, which the installation places in the `\<Social Messaging Server application>\media-channel-drivers\channel-twitter` directory.
4. In Configuration Manager, open your Social Messaging Server Application, go to the `Options` tab, and import `driver-for-twitter-options.cfg`, selecting `No` in response to `Do you want to overwrite the existing data?`
5. Set configuration options as follows:

Configuration Options

Full descriptions of all options can be found in the eServices 8.1 Reference Manual [1].

Name	Description	Example
channel-<any name> section		
Replace <any name> with an informative name, such as `twitter`.		
driver-classname	Value of the first line of `driver-for-twitter-options.cfg`	com.genesyslab.mcr.smserver.channel.twitter.TwitterDriver
inbound-route	<Tenant ID>:<endpoint>, where <endpoint> must match the name of (one of) the option(s) in the `endpoints:*tenant_dbid*` section. Genesys recommends configuring that section before setting this option.	101:twitter_queue
x-access-token	When creating a Twitter application, click My Access Token on the application Settings page (at upper right).	167466696-lksFBUeqs0WmybyhkbpCUWmiE28fEboIjjU91if1
x-access-token-secret	When creating a Twitter application, click My Access Token on the application Settings page.	RGB2LlWD0UkZl7eKvO2fvIsiEQtEX6CAVmlZ7s1w
x-consumer-key	When creating a Twitter application, copy the value of `consumer key` on the application Settings page.	PBevSLoxNVkpBtPGx5NfsA
x-consumer-secret	When creating a Twitter application, copy the value of `consumer secret` on the application Settings page.	8wZKKOSA5dtBqnMj3vcbdsUnpNZCcEwKJEBRyYFpxs
x-inbound-media	The media type for this channel	twitter
x-registered-app-name *Release 8.1.0 and earlier only*	Name of the application you create and register at dev.twitter.com	MyApp
x-submit-own-all *Release 8.1.1 and later*	Inbound messages originating from the media account associated with this channel ("own" messages) are (`true`) or are not (`false`) submitted to Interaction Server.	false
x-user-id *Release 8.1.0 and earlier only*	The user ID of the Twitter account that Social Messaging Server uses to access Twitter resources. To filter out messages originating from this account, specify its Twitter user ID. A Twitter user ID may be obtained at http://api.twitter.com/1/users/show.xml?screen_name=<screen-name> where <screen-name> must be replaced with the appropriate Twitter screen name; for example, http://api.twitter.com/1/users/show.xml?screen_name=bobshow	197203329
endpoints:*tenant_dbid* section		
Specifies interaction queues for inbound messages. Substitute the tenant's database ID (in decimal format) for *tenant_dbid*. Each `endpoints` section can contain multiple options for various queues. In a multiple-tenant environment, you must create a separate `endpoints:*tenant_dbid*` section for each tenant.		
endpoint name for inbound paging	This is a placeholder for the option that represents the queue, in a Business Process, where Social Messaging Server places interactions for processing. The actual option is created as a result of actions in Interaction Routing Designer (IRD), as follows: 1. In IRD, add Social Messaging Server to a Business Process. 2. Add an endpoint to Social Messaging Server. 3. Connect the endpoint to a queue. 4. Save the configuration. Saving the configuration creates an option whose name is the endpoint name and whose value is the queue name. You must use this endpoint name in the value of the `inbound-route-default` option.	Twitter Inbound Queue This is the name of the queue in the sample Twitter Business Process that is supplied with Genesys Social Messaging Management.
settings section		

hide-attached-data	Specifies whether the log file hides (yes, the default) or shows (no) the attached data of interactions submitted to Interaction Server	yes
channel-<any name>-monitor section		
This section defines fetch queries and other parameters for fetching data from a media channel. Replace <any name> with the name of the media channel to which this section will apply. The name must match one of the channels that are defined by the `channel-<any name>` section(s).		
sampling-period	Specifies the periodicity, in seconds, of data fetching from the Twitter data source.	900
sampling-history	Specifies the "historical depth," in seconds, of the first data fetching cycle from the Twitter data source. The first data fetching cycle requests messages with time stamps from (current_time_in_seconds minus sampling-history value) to (current_time_in_seconds). A value of zero (the default) means only new messages are fetched; that is, messages created after the driver's start time.	216000
get-direct-messages	Turns off (false) or on (true, the default) fetching of direct messages from your own account. More information on direct message queries is in Sample Queries for Twitter Channel.	false
itx-submit-timeout *Release 8.1.1 and later*	Time, in seconds, that Social Messaging Server waits for a response after sending a request to Interaction Server. Default is 10.	15
itx-resubmit-attempts *Release 8.1.1 and later*	Number of times that Social Messaging Server resubmits an interaction after the `itx-submit-timeout` timeout expires. Default is 3.	4
itx-resubmit-delay *Release 8.1.1 and later*	Time, in seconds, that Social Messaging Server waits between attempts to resubmit an interaction. Default is 30.	65
get-home-timeline	Turns off (false, the default) or on (true) fetching of messages from your home timeline. More information on timeline queries is in Sample Queries for Twitter Channel.	true
get-mentions	Turns off (false) or on (true, the default) fetching of messages that mention your account. More information on mention queries is in Sample Queries for Twitter Channel.	false
qry-<name>	This is a "class" option, which is instantiated as one or more particular options that specify search queries. Substitute an informative string for <name>; for example, `qry-iPad`, `qry-ALU` and so on. The value must be a valid Twitter search expression. The `driver-for-twitter-options.cfg` file includes four instances of this option, which are described in Sample Queries for Twitter Channel.	Name: qry-from-somebody Value: from:BusinessCustomer Name: qry-ipad Value: iPad :(

There is also a `log` section that holds the usual log options, which are described in the *Framework 8.0 Configuration Options Reference Manual.*

Next Steps

Review the sample queries supplied with the Driver for Use with Twitter.

References

[1] http://genesyslab.com/support/dl/retrieve/default.asp?item=B9CB405A445E48524C5650A7E01B6A9D&view=item

Custom Media Channel Driver

Purpose: To create a custom media channel driver.

Genesys supplies drivers for media channels that connect Social Messaging Server to Twitter and Facebook. You can also create a custom media channel driver, as described in this API reference [1].

Installation of Social Messaging Server is described on the Deploy Social Messaging Server page.

References

[1] http://en.wikipedia.org/repository/eServices/OpenAPI/index.html

Proxy Server Support for Social Messaging Server

Purpose: To enable Social Messaging Server to work with a proxy server.

Procedure

Edit the appropriate file by adding these Java runtime startup options:

- `-DproxySet=true`
- `-DproxyHost=<host of proxy server>`
- `-DproxyPort=<port of proxy server>`

To enable authentication for the proxy server, add these options as well:

- `-DproxyUser=<username>`
- `-DproxyPassword=<password>`

File to edit:

- Windows: In `JavaServerStarter.ini`, the `[JavaArgs]` section
- Unix and related systems: In `smserver.sh`, the last string, namely `"$GES_HOME_810/jre/bin/java" -classpath $CLASSPATH $JVMPARAMS $MAINCLASS $*`

Sample Queries for Twitter Channel

Purpose: To explain the sample queries supplied with Genesys Driver for Use with Twitter.

Installing the Genesys Driver for Use with Twitter places a file `driver-for-twitter-options.cfg` in the `C:\Program Files\GCTI\eServices 8.1.0\Genesys Social Messaging Server\<SM-Server-Application-name>\media-channel-drivers\channel-twitter` directory.

Importing this file into your Social Messaging Server Application object introduces various configuration options, including four that specify sample queries to Twitter. These options are instances of the "class" option `qry-<name>`, which is described in Deploy Social Messaging Server with a Twitter Channel.

Search Query Options

These options are imported as part of `driver-for-twitter-options.cfg`.

Name	Value	Returns messages that
qry-alu	ALU	Contain the text string *ALU*
qry-from-somebody	from:GenesysCustomer	Have a `from` field equal to *GenesysCustomer*
qry-ipad	iPad :(	Contain the text string *iPad :(*
qry-to-somebody	to:GenesysCustomer	Have a `to` field equal to *GenesysCustomer*

Query Types

Search

Search queries are regular Twitter search expressions, as described at https://dev.twitter.com/docs/using-search [1].

The results are sets of messages with accompanying attributes like addresses, IDs, and so on. Each message is transferred to Social Messaging Server as a separate message and is submitted to Interaction Server as a separate interaction.

Predefined

- *Timeline queries* get statuses from Twitter timelines; more information is at http://dev.twitter.com/doc/get/statuses/home_timeline.The query result is a set of statuses with accompanying data. Each status is transferred to Social Media Server as a separate message and is submitted to Interaction Server as a separate interaction.
- *Mention queries* are a subtype of timeline query. The result is a set of messages that contain references to the user's name, such as @JohnD.
- *Direct message queries* get direct messages sent to an account; more information is at http://apiwiki.twitter.com/Twitter-REST-API-Method%3A-direct_messages.The query result is a set of direct messages with accompanying data about the sender. Each direct message received is transferred to Social Media Server as a separate message and is submitted to Interaction Server as a separate interaction.

References

[1] https://dev.twitter.com/docs/using-search

Sample Business Processes for Social Media

Purpose: Introduce the sample Business Processes and describe their deployment.

The Genesys Social Messaging Management product CD includes two components containing sample Business Processes:

- Business Process for Use with Facebook contains three sample Business Processes.
- Business Process for Use with Twitter contains two sample Business Processes.

Requirements

Interaction Workflows (Business Processes) that handle social media interactions require the following versions of Universal Routing components:

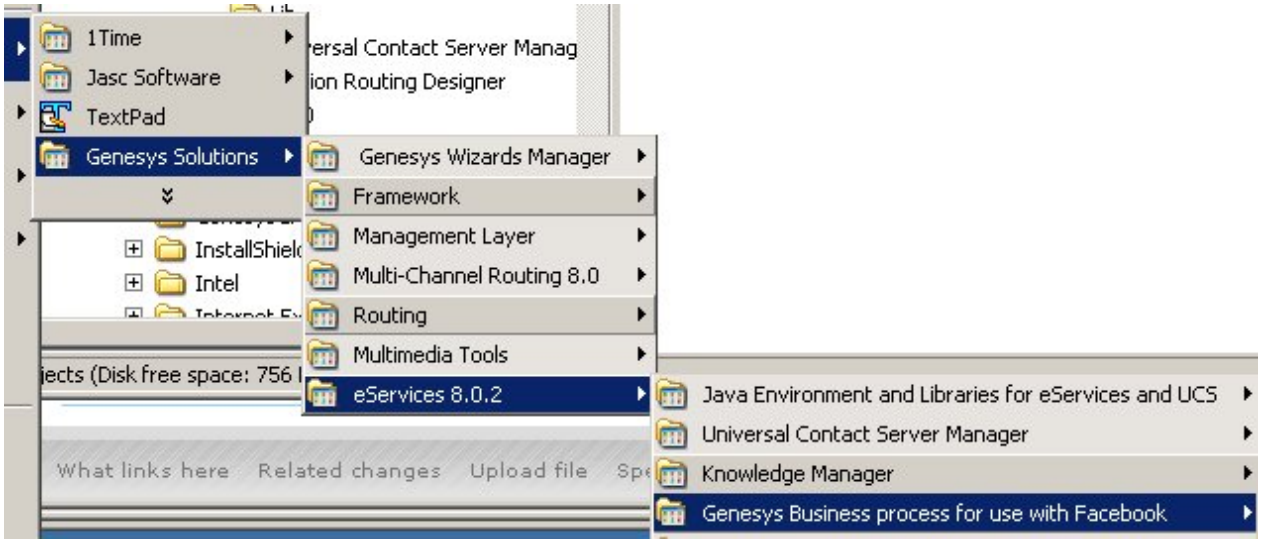
- Interaction Routing Designer (IRD) 8.0.100.12 or later
- Universal Routing Server 8.0.100.17 or later

Procedure: Install the sample Business Processes

Installation is similar for both Business Process installation packages.

1. Double-click `setup.exe`. This installs an application. It also places files in `<eServices_home>\Genesys Business process for use with <SocialMediaName>`
2. Remove any existing versions of these Business Processes in this tenant: In Interaction Routing Designer, deactivate the component strategies, delete the Business Process, and save changes.

3. From the Start menu, launch the application, as shown below with the `Genesys Business process for use with Facebook`.



4. The application asks you to log in to the Configuration Layer in order to launch the wizard for this Business Process.
5. Select a Social Messaging Server and User Account or Access Group.
6. Select a Tenant.
7. Decide whether to add Interaction Custom Properties for iWD integration. If you select this checkbox,
   - The wizard adds `desktop_actionable`, `desktop_influence`, and `desktop_sentiment` to the Interaction Custom Properties under Business Attributes.
   - After completing the wizard, you must run an upgrade script to add the corresponding fields to your Interaction Server database, as the wizard screen reminds you. This script is described below.
8. Select a destination directory for strategy files. This directory will be created (or overwritten if it already exists).
9. Select a resource capacity rule. This concludes the wizard.
10. Use IRD to review the Business Process. There is no need for any import operations.

Configuration Objects Installed

Installing the sample Business Processes creates configuration objects that are listed in Objects Installed with Sample Business Processes.

Database Upgrade Scripts

As part of the installation of the sample Business Processes, the following scripts are installed in `<eServices home>\Genesys Business process for use with <SocialMediaName>\Scripts`:

- Both Facebook and Twitter: `AlterTableInteractionsForIWDIntegration81.sql`. This adds fields for sentiment, actionability, and influence to the database. Identical copies of this script are installed in the Twitter and Facebook directories; you only need to run one of them.
- Facebook only: `AlterTableInteractionsForFacebook8021.sql` (Oracle) or `AlterTableInteractionsForFacebook802.sql`. (MSSQL, DB2). This adds the following fields to the database:
  - `facebookPostId`
  - `facebookInQueueAtTarget`
  - `facebookCommentId`
  - `facebookInBufferBeforeTarget`

Next Steps

Review the sample Business Processes

- Sample Business Process: Facebook BP
- Sample Business Process: Facebook BP - Simplified
- Sample Business Process: Facebook BP-iWD
- Sample Business Process: Twitter BP
- Sample Business Process: Twitter BP with iWD

Sample Business Process: Facebook BP

Purpose: Describe Facebook BP, a sample Business Process.

Facebook BP is a sample Business Process that is supplied on the Genesys Social Messaging Management product CD.

Overview

Facebook BP submits Facebook interactions to a configured agent group. It consists of eight strategies, four subroutines, and nine queues.

In Release 8.1.0

In broad terms, processing proceeds as follows:

1. The first step depends on whether you want to create a contact record for each author of a comment to a Facebook post. If you do, you must set the `x-submit-comments-itx` option to `true`, which makes the system create an interaction for each comment (this is in addition to the interaction that it creates containing both the base post and all of the comments on it).
   - If the interaction being processed is one of these comment-only interactions, it is sent to the `Stop Interaction Comments in Parking Queue` strategy, which creates a contact for the author of the comment, then terminates.
   - If the interaction consists of a post plus comments, it continues to the next step.
2. This step determines whether this interaction contains comments on some post that is already in the strategy and so should be merged with it.
   a. A query is sent to Interaction Server to see if there is already an interaction down the flow in the strategy with the same Facebook Post ID.
   b. Then,
      - If such an interaction is found, it is updated with the content of (merged with) the newer interaction, then the newer interaction is terminated.
      - If no such interaction is found in the strategy, then this current interaction proceeds into the buffer stage, which delays the delivery of the interaction to the agent queue.
3. The buffer stage is similar to the preceding, but it determines whether this interaction contains comments on some post that has already been delivered to an agent. If it does, the interaction is held until that agent can accept delivery of it.
   a. The buffer stage periodically checks if there is an interaction with the same Post ID being processed by any agent in the group.
   b. Then,
      - If there is such an interaction, the current interaction is delayed in the buffer stage until it can be delivered to the agent who is processing the earlier related interaction.
      - If there is no such interaction, it proceeds to the classify and screen stage.
4. This stage tries to classify and screen the interactions that will be delivered to the agent group. Note the following:
   - This stage uses:
     - One subroutine that screens for sentiment and actionability.
     - One subroutine that classifies for sentiment.
   - Routing decisions can be made based on the classification/screening results. One way of doing this is presented in this Business Process: All the results are attached to the interaction, and can later be viewed on the agent desktop in the attached data tab.
5. Finally the interaction is created in UCS and is delivered to an agent's virtual queue.

For the `MergeItxData` method in the ESP block of Facebook Inbound Strategy to work properly, the `settings/delay-updates` option in Interaction Server should be set to `false`. With this setting, Interaction Server forces updates of interaction properties in the database each time it processes

`RequestChangeProperties`.

In Release 8.1.1

1. Facebook Inbound Strategy can be described in two main steps.
   a. The first step depends on whether you want to create a contact record for each author of a comment to a Facebook post. If you do, you must set the `x-submit-comments-itx` option to `true`, which makes the system create an interaction for each comment (this is in addition to the interaction that it creates containing both the base post and all of the comments on it).
      - If the interaction being processed is one of these comment-only interactions, it is sent to the Stop Interaction Comments in Parking Queue strategy, which creates a contact for the author of the comment, then terminates.
      - If the interaction consists of a post plus comments, it continues to the next step.

   b. This step determines whether this interaction contains comments on some post that is already in the strategy and so should be merged with it.
      ii. A query is sent to Interaction Server to see if there is already an interaction down the flow in the strategy with the same Facebook Post ID.
      iii. Then,
         - If such an interaction is found, it is updated with the content of (merged with) the newer interaction, then the newer interaction is terminated.
         - If no such interaction is found in the strategy, then this current interaction proceeds into the buffer stage, which delays the delivery of the interaction to the agent queue.
2. Facebook Inbound Buffer Strategy is similar to the preceding, but it determines whether this interaction contains comments on some post that has already been delivered to an agent. If it does, the interaction is held until that agent can accept delivery of it.
   a. The buffer stage periodically checks if there is an interaction with the same Post ID being processed by any agent in the group.
   b. Then,
      - If there is such an interaction, the current interaction is delayed in the buffer stage until it can be delivered to the agent who is processing the earlier related interaction.
      - If there is no such interaction, it proceeds to the classify and screen stage.
3. Classification-Screen Strategy tries to classify and screen the interactions that will be delivered to the agent group. Note the following:
   - The strategy organizes the task as follows:
      - The strategy itself screens and classifies posts for sentiment and actionability.
      - One subroutine screens comments for sentiment and actionability.
      - One subroutine classifies comments for sentiment and actionability.
   - Routing decisions can be made based on the classification/screening results. One way of doing this is presented in this Business Process: All the results are attached to the interaction, and can later be viewed on the agent desktop in the attached data tab.
4. Facebook Calculation Strategy processes all previously-attached classification and screening keys and attaches the keys `desktop_sentiment`, `desktop_actionable`, and `desktop_expand`, which the desktop uses in presenting the interaction in its user interface.
5. Facebook Agent Delivery Strategy delivers the interaction to the agent desktop.
6. Finally,

- If there is a reply, Facebook Outbound Strategy dispatches it to Social Messaging Server, for delivery to Facebook.
- If there is no reply, Facebook Stop Strategy terminates the interaction.

For the `MergeItxData` method in the ESP block of Facebook Inbound Strategy to work properly, the `settings/delay-updates` option in Interaction Server should be set to `false`. With this setting, Interaction Server forces updates of interaction properties in the database each time it processes `RequestChangeProperties`.

Sample Business Process: Facebook BP - Simplified

Home > eServices > eServices Social Media Solution Guide > Sample Business Processes for Social Media

Purpose: Describe Facebook BP - Simplified, a sample Business Process.

Facebook BP - Simplified is a sample Business Process that is supplied on the Genesys Social Messaging Management product CD.

Overview

The simplified Facebook business process is designed to process the submitted Facebook interactions in the most simplistic way.

- If the interaction contains both a post and comments:

1. An interaction is created in the UCS database.
2. The Facebook post ID and comment ID are updated in the UCS database.
3. The interaction is delivered to an agent.

- If the interaction contains only a comment, it is parked in a parking queue.

Sample Business Process: Facebook BP-iWD

Home > eServices > eServices Social Media Solution Guide > Sample Business Processes for Social Media

Purpose: Describe Facebook BP-iWD, a sample Business Process.

Facebook BP-iWD is a sample Business Process that shows how Genesys Social Messaging can work with Genesys Intelligent Workitem Distribution (iWD). Facebook BP-iWD is supplied on the Genesys Social Messaging Management product CD and is used together with two of the Business Processes that are supplied with iWD, as described below.

Requirements

To use Facebook BP-iWD:

- Install Facebook BP. This creates some configuration objects that are required by Facebook BP-iWD.
- Use the iWD Setup Utility to install these Business Processes:
  - Standard Genesys to iWD Adapter
  - IWDBP

For information on how to install and use the iWD Setup Utility, consult the *intelligent Workload Distribution 8.0 Deployment Guide.* [1]

In Release 8.1.0

Facebook BP-iWD and its associated Business Processes do the following things.

1. Facebook BP-iWD proceeds in the following stages:
   1. A query is sent to Interaction Server to see if there is already an interaction down the flow in the strategy with the same Facebook Post ID.
      - If such an interaction is found, it is updated with the content (merged) of the newer interaction, then the newer interaction is terminated.
      - If no such interaction is found in the strategy, then this current interaction proceeds into the Buffer stage, which delays the delivery of the interaction to the agent queue.
   2. The Buffer stage periodically checks if an interaction with this Post ID can be delivered to the agent desktop; in other words, if there is an interaction with the same Post ID being processed by any agent in the group.
      - If there is such an interaction, the current interaction is delayed in the buffer stage.
      - If there is no such interaction, it proceeds to the Classification and Screening stage.
   3. The Classification and Screening stage tries to classify and screen the interactions that will be delivered to the agent group. Note the following:
      - This stage uses:
        - A subroutine that screens for sentiment and actionability.
        - A subroutine that classifies for sentiment.
      - Routing decisions can be made based on the classification/screening results. One way of doing this is presented in the sample BP, which you can modify. All the results are attached to the interaction, and can later be viewed on the agent desktop in the attached data tab.
   4. The last stage passes the interaction to Standard Genesys to iWD Adapter.
2. Standard Genesys to iWD Adapter attaches some required key-value pairs to the interaction, then passes the interaction to IWDBP.
3. IWDBP performs classification, prioritization, distribution, and archiving.

For the `MergeItxData` method in the ESP block of Facebook Inbound Strategy to work properly, the `settings/delay-updates` option in Interaction Server should be set to `false`. With this setting, Interaction Server forces updates of interaction properties in the database each time it processes `RequestChangeProperties`.

For more information about the Standard Genesys to iWD Adapter and IWDBP Business Processes, see the *intelligent Workload Distribution 8.0 Deployment Guide* [1].

In Release 8.1.1 and Later

1. Facebook Inbound Strategy-iWD can be described in two main steps.
   a. The first step depends on whether you want to create a contact record for each author of a comment to a Facebook post. If you do, you must set the `x-submit-comments-itx` option to `true`, which makes the system create an interaction for each comment (this is in addition to the interaction that it creates containing both the base post and all of the comments on it).
      - If the interaction being processed is one of these comment-only interactions, it is sent to the Stop Interaction Comments in Parking Queue strategy, which creates a contact for the author of the comment, then terminates.
      - If the interaction consists of a post plus comments, it continues to the next step.

   b. This step determines whether this interaction contains comments on some post that is already in the strategy and so should be merged with it.

      ii. A query is sent to Interaction Server to see if there is already an interaction down the flow in the strategy with the same Facebook Post ID.

      iii. Then,

      - If such an interaction is found, it is updated with the content of (merged with) the newer interaction, then the newer interaction is terminated.
      - If no such interaction is found in the strategy, then this current interaction proceeds into the buffer stage, which delays the delivery of the interaction to the agent queue.

2. Facebook Inbound Buffer Strategy-iWD is similar to the preceding, but it determines whether this interaction contains comments on some post that has already been delivered to an agent. If it does, the interaction is held until that agent can accept delivery of it.
   a. The buffer stage periodically checks if there is an interaction with the same Post ID being processed by any agent in the group.
   b. Then,
      - If there is such an interaction, the current interaction is delayed in the buffer stage until it can be delivered to the agent who is processing the earlier related interaction.
      - If there is no such interaction, it proceeds to the classify and screen stage.
3. Facebook Classification-Screen Strategy-iWD tries to classify and screen the interactions that will be delivered to the agent group. Note the following:
   - The strategy organizes the task as follows:
     - The strategy itself screens and classifies posts for sentiment and actionability.
     - One subroutine screens comments for sentiment and actionability.
     - One subroutine classifies comments for sentiment and actionability.
   - Routing decisions can be made based on the classification/screening results. One way of doing this is presented in this Business Process: All the results are attached to the interaction, and can later be viewed on the agent desktop in the attached data tab.
4. Facebook Calculation Strategy-iWD processes all previously-attached classification and screening keys and attaches the keys `desktop_sentiment`, `desktop_actionable`, and `desktop_expand`, which the desktop uses in presenting the interaction in its user interface.
5. Facebook iWD Delivery Strategy delivers the interaction to iWD.

For the `MergeItxData` method in the ESP block of Facebook Inbound Strategy-iWD to work properly, the `settings/delay-updates` option in Interaction Server should be set to `false`. With this setting, Interaction Server forces updates of interaction properties in the database each time it processes `RequestChangeProperties`.

References

[1] http://genesyslab.com/support/dl/retrieve/default.asp?item=B9CB405A445E48524C5650A7D2276193&view=item

Sample Business Process: Twitter BP

eServices Social Media Solution Guide

Table of Contents

Purpose: Describe Twitter BP, a sample Business Process.

Twitter BP is a sample Business Process that is supplied on the Genesys Social Messaging Management product CD.

In Release 8.1.0

Twitter BP performs the following simple processing, using two strategies:

1. Twitter Inbound Strategy
    1. Filters out tweets that already exist as interactions in the Universal Contact Server (UCS) database
    2. Creates an interaction in the UCS database
    3. Sends a request to Classification Server to screen and classify the interaction
    4. Delivers the interaction to an agent, along with the results of the screening and classification
2. Twitter Outbound Strategy dispatches a reply from the agent to Social Messaging Server, for delivery to Twitter.

In Release 8.1.1 and Later

Twitter BP uses the following strategies:

1. Twitter Inbound Strategy
    1. Initializes the required variables.
    2. Ensures that the interactions will be delivered to agents in the order they were created.
    3. Checks whether the interaction already exists in the UCS database and, if not, creates it.
    4. Associates the interaction with the correct thread.
    5. Creates contacts in UCS.
2. Twitter Classify-Screen Strategy performs classification and screening, and attaches the resulting Actionability and Sentiment attributes to the interaction.
3. Twitter Influence Calculation Strategy calls up the Klout [1] service and attaches the appropriate keys showing the influence values.
4. Twitter Calculation Strategy processes all previously-attached classification and screening keys and attaches the keys `desktop_sentiment`, `desktop_actionable`, and `desktop_expand`, which the desktop uses in presenting the interaction in its user interface.
5. Twitter Agent Delivery Strategy delivers the interaction to the agent desktop.
6. Twitter Outbound Strategy dispatches a reply from the agent to Social Messaging Server, for delivery to Twitter.

Related Topics

There is also a sample Twitter BP with iWD.

References

[1] http://klout.com/home

Sample Business Process: Twitter BP with iWD

Purpose: Describe Twitter BP with iWD, a sample Business Process.

Twitter BP with iWD is a sample Business Process that shows how Genesys Social Messaging can work with Genesys Intelligent Workitem Distribution (iWD). Twitter BP with iWD is supplied on the Genesys Social Messaging Management product CD and is used together with two of the Business Processes that are supplied with iWD, as described below.

Requirements

To use Twitter BP with iWD:

- Install Twitter BP. This creates some configuration objects that are required by Twitter BP with iWD.
- Use the iWD Setup Utility to install these Business Processes:
  - Standard Genesys to iWD Adapter
  - IWDBP

For information on how to install and use the iWD Setup Utility, consult the *intelligent Workload Distribution 8.0 Deployment Guide* [1].

In Release 8.1.0

Twitter BP with iWD and its associated Business Processes do the following things:

1. Twitter BP with iWD performs initial processing, using the Twitter Inbound Strategy-iWD strategy to:
    1. Filter out tweets that already exist as interactions in the Universal Contact Server (UCS) database.
    2. Create an interaction in the UCS database.
    3. Pass the interaction to Standard Genesys to iWD Adapter.
2. Standard Genesys to iWD Adapter attaches some required key-value pairs to the interaction, then passes the interaction to IWDBP.
3. IWDBP performs classification, prioritization, distribution, and archiving.

For more information about the Standard Genesys to iWD Adapter and IWDBP Business Processes, see the *intelligent Workload Distribution 8.0 Deployment Guide* [1].

In Release 8.1.1 and Later

1. Twitter Inbound Strategy-iWD
    1. Initializes the required variables.
    2. Ensures that the interactions will be delivered to agents in the order they were created.
    3. Checks whether the interaction already exists in the UCS database and, if not, creates it.
    4. Associates the interaction with the correct thread.
    5. Creates contacts in UCS.
2. Twitter Classify-Screen Strategy-iWD performs classification and screening, and attaches the resulting Actionability and Sentiment attributes to the interaction.
3. Twitter Influence Calculation Strategy-iWD calls up the Klout [1] service and attaches the appropriate keys showing the influence values.
4. Twitter Calculation Strategy-iWD processes all previously-attached classification and screening keys and attaches the keys `desktop_sentiment`, `desktop_actionable`, and `desktop_expand`, which the desktop uses in presenting the interaction in its user interface.
5. Standard Genesys to iWD Adapter attaches some required key-value pairs to the interaction, then passes the interaction to IWDBP.
6. IWDBP performs classification, prioritization, distribution, and archiving.

For more information about the Standard Genesys to iWD Adapter and IWDBP Business Processes, see the *intelligent Workload Distribution 8.0 Deployment Guide* [1].

Objects Installed with Sample Business Processes

eServices Social Media Solution Guide

Table of Contents

Purpose: To list the objects that are installed with Business Process for Use with Facebook and Business Process for Use with Twitter.

Overview

Installing the two sample Business Processes creates various objects in your environment, most of them being objects in the Configuration Layer.

This page lists configuration objects by their Display Name. Where the Name differs from the Display Name, it is given in parentheses.

Common

These objects are used by both Business Processes and are installed with both.

- Agent Group
  - Social Media Agent Group
- Business Attributes: Contact Attribute
  - _umsMediaAccount
- Capacity Rule
  - Default_Social_Media_Capacity_rule
- Language
  - English for Sentiment Analysis (English_Sentiment)

Facebook

Business Process for Use with Facebook also installs the following objects:

- Business Attributes: Contact Attributes
  - Facebook Actor ID (_facebookActorId)
  - Facebook Actor Name (_facebookActorName)
- Business Attributes: Interaction Attributes
  - FacebookCommentId
  - FacebookOutboundCommentId
  - FacebookPostId
- Business Attributes: Interaction Custom Properties
  - _facebookCommentId
  - _facebookInBufferBeforeTarget
  - _facebookInQueueAtTarget
  - _facebookPostId
- Business Attributes: Media Type
  - facebook

Facebook with iWD Integration

- Business Attributes: Interaction Custom Properties
  - desktop_actionable
  - desktop_influence
  - desktop_sentiment

Twitter

Business Process for Use with Twitter also installs the following objects:

- Business Attributes: Contact Attribute
  - Twitter User ID (_twitterFromUserId)
  - Twitter From Address (_twitterFrom Addr)
- Business Attributes: Interaction Attribute
  - TwitterMsgId
- Business Attributes: Media Type
  - twitter

Knowledge Manager Export Files

Installing either sample Business Process also places the following files in `\eServices 8.1.1\Genesys Business process for use with <socialmedia>\KnowledgeManagerExport`:

- SentimentAndActionabilityScreeningRules.kme
- EnglishSentiment.kme
- Actionability.kme

Deployment and use of these files is described in the eServices 8.1 User's Guide [1].

- For SentimentAndActionabilityScreeningRules.kme, see "Screening for Sentiment and Actionability" in the "Genesys Knowledge Management: Basics" chapter.
- For EnglishSentiment.kme and Actionability.kme,see "Analyzing Sentiment and Actionability with Content Analyzer" in the "Genesys Knowledge Management: Content Analyzer" chapter.

References

[1] http://genesyslab.com/support/dl/retrieve/default.asp?item=AFD71A7A5A3A9AFADABA1A648CF2FA07&view=item

Genesys Desktop Plugin

Purpose: Describe deployment and use of the Social Media plugin for Genesys Agent Desktop.

The plugin for Genesys Desktop is a set of extensions for Genesys Agent Desktop (GAD) that enable the Desktop to work with social media such as Twitter and Facebook messages.

Requirements

Genesys Agent Desktop, version 7.6.302.13 or later.

Deployment

The procedure for deployment differs according to which application container you are using:

- Deploying Genesys Desktop Plugin with Tomcat
- Deploying Genesys Desktop Plugin with WebSphere and WebLogic

Configuration Options

Configuration options for the plugin are described in the "eServices Social Messaging Plugin for Genesys Agent Desktop Options" section of the "Configuration Options" chapter of the eServices 8.1 Reference Manual. [1]

Adding a Social Media Site After Deployment

The installation IP asks you to select Facebook, Twitter, or both. If you select Twitter only, you can still add Facebook later, as follows:

1. In the `multimedia` section of your Social Messaging Server Application object, locate the `facebook-server-app-name` option.
2. Set the option value to the name of your Facebook application. For example `facebook-server-app-name = SM_Server_Facebook`.

General Information for Facebook

- Generally an agent must be logged in to Facebook to see a user's wall page. If the agent is not logged in, the Facebook screen pop will still open, with a suggestion to register or login to see the information.
- You can create just one Facebook account or multiple agent accounts according to your needs. What will display when an agent clicks the link depends upon permissions. Usually page/application walls are seen, while user walls are not. To view/post something from Facebook, the agent must "Like" it. If the Facebook account is shared among agents, then all agents using this account will then be able to see the personal wall.
- To put it another way, an agent can open a link in a separate browser window to the Facebook object (page, user, event, and so on); it is up to the customer to decide what to do with it. If the data is public, the agent sees the corresponding page. If not, he or she must log in to Facebook with some account that has permission to see that data.
- In order to post a Facebook message as an administrator, you must use the specific access token for the page or application. To do so,

1. First obtain an initial access token, as described in the procedure presented in Creating a Facebook Application.
2. Substitute this access token for `<your_access_token>` in this URL: https://graph.facebook.com/me/accounts&access_token=<your_access_token>
3. The result is a page listing the various objects that you have created and registered, with name, category, ID, and access token for each. Here is a fictional example:

```
{
   "data": [
      {
         "name": "MrZowie",
         "category": "Local_business",
```

```
         "id": "108926930336652",
         "access_token": "108926930336652|e8da8a88e83de6635dac62ac-100000952952880|108977979136652|blBz1BlBc72cu72OupXv1wmzL24"
      },
      {
         "name": "SuperApp",
         "category": "Application",
         "id": "110694465630007",
         "access_token": "110694465630007|e8da8a88e83de6635dac62ac-100000952952880|110694465630461|9ABCcE9qw_1X4B0L-qDYe14QZkM"
      }
   ]
}
```

Uninstalling

After uninstalling the plugin, you must remove the inserted <init-param> tag from the `web.xml` file.

You must also perform the following adjustments:

Section name	Option name	Uninstallation actions
multimedia	facebook-comment-limit	Delete option.
multimedia	facebook-post-limit	Delete option.
multimedia	twitter-reply-limit	Delete option.
multimedia	facebook-outbound-queue	Delete option.
multimedia	facebook-server-app-name	Delete option.
multimedia	media	Remove the previously added `facebook,twitter` string. For example, if the previous value was `email,chat,webcallback,facebook,twitter,sms,mms,` the new value should be `email,chat,webcallback,sms,mms.`
multimedia	open-media-saved-list	Remove the previously added `facebook,twitter` string. For example, if the previous value was `sms,facebook,twitter,mms,` the new value should be `sms,mms.`
multimedia	twitter-outbound-queue	Delete option.
contact	directory-displayed-columns	Remove the previously added `_twitterFromAddr,_twitterFromUserId,_facebookActorId, _facebookActorName.`
contact	displayed-attributes	Remove the previously added `_twitterFromAddr,_twitterFromUserId,_facebookActorId, _facebookActorName.`
contact	directory-search-attributes	Remove the previously added `_twitterFromAddr,_twitterFromUserId,_facebookActorId, _facebookActorName.`
contact	multiple-values-attributes	Remove the previously added `_facebookActorName`

Deploying Genesys Desktop Plugin with Tomcat

Purpose: Describe deployment of the Social Media plugin for Genesys Agent Desktop with Tomcat.

Procedure

1. Run the Installation Package.
   The installation places the plugin files in the following GAD application directories:

```
\< Genesys Application Desktop Root folder>          (GAD root directory)

   \<webapps>                                      (Installed applications root)

         \<gdesktop>                                (GAD application directory)
```

   After completing the installation, make the following adjustments to the `web.xml` file:
2. Locate the `web.xml` file, normally in the `<GAD_root\webapps\gdesktop\WEB-INF>` folder. Analyze the file and find the description of the `<web-app><servlet><servlet-name>initUAD</servlet-name>` servlet.
3. Add the following set of XML tags to this file:

```
<init-param>
  <param-name>customFile</param-name>
  <param-value>/custom/custom.xml</param-value>
</init-param>
```

For example:

```
<web-app>
 ..... some other tags ….
    <servlet>
        <servlet-name>initUAD</servlet-name>
        <servlet-class>com.genesyslab.uadthin.InitUAD</servlet-class>
..... some other tags ….
      <init-param>
            <param-name>customFile</param-name>
            <param-value>/custom/custom.xml</param-value>
        </init-param>
..... some other tags ….
    <servlet>
```

More Information

See the Genesys Desktop Plugin page for information on configuration options, adding a social media site after deployment, and uninstalling.

Deploying Genesys Desktop Plugin with WebSphere and WebLogic

Purpose: Describe deployment of the Social Media plugin for Genesys Agent Desktop (GAD) with WebSphere and WebLogic.

Procedure

1. Install GAD. For information on how to do this, see the Genesys Desktop 7.6 Deployment Guide [1]. *Installation* of GAD is normally followed by *deployment* of GAD. If you have already deployed GAD, you must undeploy it before proceeding.
2. Locate the WAR file created by the GAD installation and extract its contents to some location.
3. Run the Plugin Installation Package.
   When prompted, select or create a destination folder different from the folder that you extracted the GAD WAR file to. The Plugin installation creates a subfolder called `SocialMediaPlugin` in this destination and places files in it.
4. In `SocialMediaPlugin`,
   - Locate the folders `custom` and `extension` and copy them to the GAD root folder.
   - Locate the folder `WEB-INF/lib.` Copy the three libraries that it contains to GAD's `WEB-INF/lib` subfolder.
5. In GAD's `WEB-INF\classes` folder, create a `custom.properties` file, containing the string `contactExt=Contact Ext.`
6. Locate the `web.xml` file, which normally resides in the `<GAD_root>\webapps\gdesktop\WEB-INF` folder, and make the following adjustments to it:
   1. Analyze the file and find the description of the `<web-app><servlet><servlet-name>initUAD</servlet-name>` servlet.
   2. Add the following set of XML tags to this file:

```
<init-param>
  <param-name>customFile</param-name>
  <param-value>/custom/custom.xml</param-value>
</init-param>
```

   For example:

```
<web-app>
 ..... some other tags ....
    <servlet>
        <servlet-name>initUAD</servlet-name>
        <servlet-class>com.genesyslab.uadthin.InitUAD</servlet-class>
..... some other tags ....
      <init-param>
            <param-name>customFile</param-name>
            <param-value>/custom/custom.xml</param-value>
        </init-param>
..... some other tags ....
    <servlet>
```

7. Locate the instance of Java that is used by WebSphere or WebLogic. Usually it is in the `<WebSphereRoot>\AppServer\java\bin` folder.
8. Change the default folder to the folder where all the GAD files and plug-in are located.
9. Launch `<WebSphereRoot>\AppServer\java\bin\jar.exe` with the following parameter:

```
<WebSphereRoot>\AppServer\java\bin\jar cvf gdesktop.war ./
```

10. Deploy the resulting `gdesktop.war` file using your WebSphere or WebLogic administrative console.

For More Information

See the Genesys Desktop Plugin page for information on configuration options, adding a social media site after deployment, and uninstalling.

References

[1] http://genesyslab.com/support/dl/retrieve/default.asp?item=B2BDC3D6C9CCCFD2C5D8BBCA3846ABD6&view=item

Related eServices Documentation Resources

eServices Social Media Solution Guide

Table of Contents

Purpose: To present documentation resources related to the Social Media Solution.

The following resources provide additional information that is relevant to this software. Consult these additional resources as necessary.

eServices

- eServices 8.1 Deployment Guide [1], which describes deployment procedures for all eServices components.
- eServices 8.0 User's Guide [2], which provides overall information and recommendations on the use and operation of eServices.
- eServices 8.1 Reference Manual [1], which provides a reference listing of all configuration options and of field codes used in standard responses.
- eServices 8.1 Universal Contact Server Manager Help [3], which is a guide to the Universal Contact Server Manager user interface.
- eServices 8.1 Knowledge Manager Help [4], which is a guide to the Knowledge Manager user interface.
- For a listing of classes, methods, fields, and constants of the Web API portion of the Web API Server component, see:
  - eServices 8.0 .NET Web API Reference for the .NET Web API
  - The API References of the Platform SDK for the Java-based Web API
- eServices 8.1 Web API Client Developer's Guide [5], which describes the structure of the Web API, explains the Simple Samples, and describes procedures for customizing them.
- "eServices Log Events" in Framework 8.0 Combined Log Events Help [6], which is a comprehensive list and description of all events that may be recorded in logs.
- Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at http://genesyslab.com/support.
- Documentation on the other three members of the Genesys Customer Interaction Platform: Universal Routing, Reporting, and Management Framework. Some of this is listed in the following sections.

Genesys Desktop

- Genesys Desktop 7.6 Deployment Guide [7], which describes deployment procedures for the Genesys Desktop.
- Genesys Desktop 7.6 Developer's Guide [8], which describes customizing the Genesys Desktop.
- Genesys Desktop 7.6 Agent Help [9], which is a guide to the Genesys Agent Desktop.
- Genesys Desktop 7.6 Supervisor's Help [9], which is a guide to the Genesys Supervisor Desktop.

Universal Routing

- Universal Routing 8.1 Reference Manual [10], which contains descriptions of all routing strategy objects, including those that are specific to eServices.
- Universal Routing 8.1 Strategy Samples [11], which describes the sample strategies supplied with Universal Routing.
- Universal Routing 8.1 Business Process User's Guide [12], which contains step-by-step instructions for using Interaction Routing Designer to design interaction workflows. It also describes the sample business processes supplied with eServices.
- Universal Routing 8.1 Interaction Routing Designer Help [13], which is a guide to Interaction Routing Designer, including the portion of it that designs interaction workflows and business processes for eServices.

Genesys

- Genesys Models Reference Manual [14], which includes a set of basic voice and interaction models, showing the components involved and the relevant event messages sent among them. For authoritative description of the event messages, see the next item.
- The API References of the Platform SDK, which provide the authoritative information on methods and functions for each SDK, including requests and events. The class Message includes all event and request messages.
- Genesys Technical Publications Glossary [15], which ships on the Genesys Documentation Library DVD and which provides a comprehensive list of the Genesys and computer-telephony integration (CTI) terminology and acronyms used in this document.
- Genesys Migration Guide [16], which ships on the Genesys Documentation Library DVD, and which provides documented migration strategies for Genesys product releases. Contact Genesys Technical Support for more information.
- Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at http://genesyslab.com/support.

Information about supported hardware and third-party software is available on the Genesys Technical Support website in the following documents:

- Genesys Supported Operating Environment Reference Manual [17]
- Genesys Supported Media Interfaces Reference Manual [18]

Consult these additional resources as necessary:

- Genesys Hardware Sizing Guide [19], which provides information about Genesys hardware sizing guidelines for the Genesys 8.x releases.
- Genesys Interoperability Guide [20], which provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and Gplus Adapters Interoperability.
- Genesys Licensing Guide [21], which introduces you to the concepts, terminology, and procedures relevant to the Genesys licensing system.
- Genesys Database Sizing Estimator 7.6 Worksheets [22], which provides a range of expected database sizes for various Genesys products.

For additional system-wide planning tools and information, see the release-specific listings of System Level Documents on the Genesys Technical Support website, accessible from the system level documents by release tab in the Knowledge Base Browse Documents Section.

Genesys product documentation is available on the:

- Genesys Technical Support website at http://genesyslab.com/support.
- Genesys Documentation Library DVD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com [23].

References

[1] http://genesyslab.com/support/dl/retrieve/default.asp?item=BCD129B633B02DBA27C4E11965A1D728&view=item
[2] http://genesyslab.com/support/dl/retrieve/default.asp?item=B0B9C5D6C7D8B91A7B5C3D41AED51D64&view=item
[3] http://genesyslab.com/support/dl/retrieve/default.asp?item=BCD129B633B02DBA27C4E1196595E718&view=item
[4] http://genesyslab.com/support/dl/retrieve/default.asp?item=BACD3BA6415C47627D687389F30D81F6&view=item
[5] http://genesyslab.com/support/dl/retrieve/default.asp?item=BCD129B633B02DBA27C4E11965A1D832&view=item
[6] http://genesyslab.com/support/dl/retrieve/default.asp?item=BACD3BA6415C47627D68738AF505738A&view=item
[7] http://genesyslab.com/support/dl/retrieve/default.asp?item=B2BDC3D6C9CCCFD2C5D8BBCA3846ABD6&view=item&fzip=True
[8] http://genesyslab.com/support/dl/retrieve/default.asp?item=B8C93DA63FA831AA33AC3542B5C73148&view=item&fzip=True
[9] http://genesyslab.com/support/dl/retrieve/default.asp?item=B6C52FB62DB42BB229B02752A3CF3742&view=item
[10] http://genesyslab.com/support/dl/retrieve/default.asp?item=B8C93DA63FA831AA33AC3541B9CE3F48&view=item
[11] http://genesyslab.com/support/dl/retrieve/default.asp?item=BBCF3EAA36B22EBA26C2DE2F5AA6DA31&view=item
[12] http://genesyslab.com/support/dl/retrieve/default.asp?item=B5C334AA30A63CA2389E046F94EA1967&view=item
[13] http://genesyslab.com/support/dl/retrieve/default.asp?item=BCD129B633B02DBA27C4E1196596EA18&view=item
[14] http://genesyslab.com/support/dl/retrieve/default.asp?item=B0B9C5D6C7D8B91A7B5C3D42ABD51B64&view=item
[15] http://genesyslab.com/support/dl/retrieve/default.asp?item=AFD71A7A5A3A9AFADABA1A6393DB244D&view=item
[16] http://genesyslab.com/support/dl/retrieve/default.asp?item=AFD71A7A5A3A9AFADABA1A638CF1F803&view=item
[17] http://genesyslab.com/support/dl/retrieve/default.asp?item=B6C52FB62DB42BB229B02755A3D92054&view=item
[18] http://genesyslab.com/support/dl/retrieve/default.asp?item=A9CB309AF4DEB8127C5640A3C32445A7&view=item
[19] http://genesyslab.com/support/dl/retrieve/default.asp?item=B3BFC6DABE22B62AAE32A6D32753A0D1&view=item
[20] http://genesyslab.com/support/dl/retrieve/default.asp?item=B9CB405A445E48524C5650A6E01767A1&view=item
[21] http://genesyslab.com/support/dl/retrieve/default.asp?item=B6C52FB62DB42BB229B02755A1D12650&view=item
[22] http://genesyslab.com/support/dl/retrieve/default.asp?item=BCD129B633B02DBA27C4E11A718D007A&view=item&fzip=True
[23] mailto:orderman@genesyslab.com

Article Sources and Contributors

eServices Social Media Solution Guide *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5431 *Contributors*: Dsolnit, Edjamer, Rpfeiffer, WikiSysop

Genesys Social Engagement Overview *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5369 *Contributors*: Edjamer, Rpfeiffer, WikiSysop

Social Messaging Server *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5465 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Creating a Facebook Application *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3152 *Contributors*: Rpfeiffer, WikiSysop

Creating a Twitter Application *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3156 *Contributors*: Rpfeiffer, WikiSysop

Deploy Social Messaging Server *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5455 *Contributors*: Dsolnit, WikiSysop

Deploy Social Messaging Server with a Facebook Channel *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5457 *Contributors*: Dsolnit, WikiSysop

Deploy Social Messaging Server with a Twitter Channel *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5459 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Custom Media Channel Driver *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5453 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Proxy Server Support for Social Messaging Server *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3164 *Contributors*: Rpfeiffer, WikiSysop

Sample Queries for Twitter Channel *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3166 *Contributors*: Rpfeiffer, WikiSysop

Sample Business Processes for Social Media *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5451 *Contributors*: Dsolnit, WikiSysop

Sample Business Process: Facebook BP *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5467 *Contributors*: Dsolnit, WikiSysop

Sample Business Process: Facebook BP - Simplified *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5447 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Sample Business Process: Facebook BP-iWD *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5461 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Sample Business Process: Twitter BP *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5469 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Sample Business Process: Twitter BP with iWD *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5463 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Objects Installed with Sample Business Processes *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=5471 *Contributors*: Dsolnit, Rpfeiffer, WikiSysop

Genesys Desktop Plugin *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3201 *Contributors*: Rpfeiffer, WikiSysop

Deploying Genesys Desktop Plugin with Tomcat *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3186 *Contributors*: Rpfeiffer, WikiSysop

Deploying Genesys Desktop Plugin with WebSphere and WebLogic *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3190 *Contributors*: Rpfeiffer, WikiSysop

Related eServices Documentation Resources *Source*: http://docs.genesyslab.com/wiki/index.php?oldid=3192 *Contributors*: Rpfeiffer, WikiSysop

Image Sources, Licenses and Contributors

File:welcome.png *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:Welcome.png *License*: unknown *Contributors*: WikiSysop

Image:SolutionGuide.png *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:SolutionGuide.png *License*: unknown *Contributors*: WikiSysop

File:SocialMediaArchitecture.jpg *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:SocialMediaArchitecture.jpg *License*: unknown *Contributors*: WikiSysop

Image:UsersGuide.png *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:UsersGuide.png *License*: unknown *Contributors*: WikiSysop

file:selectSMServerFB.jpg *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:SelectSMServerFB.jpg *License*: unknown *Contributors*: WikiSysop

file:selectSMServer.jpg *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:SelectSMServer.jpg *License*: unknown *Contributors*: WikiSysop

file:launchBPapplication.jpg *Source*: http://docs.genesyslab.com/wiki/index.php?title=File:LaunchBPapplication.jpg *License*: unknown *Contributors*: WikiSysop