knowlagent®

Configuring ACD Services and Connections

This chapter covers the steps to configure .NET services and sub-services that support the Knowlagent Genesys ACD integration. .

K nowlagent's **Services.config.xml** file is the .NET configuration file used to configure the Knowlagent .NET services and sub-services that support the ACDRuleEngine(s) and establish the connection to the Genesys ACD that the Knowlagent server uses to receive ACD statistics.

In this chapter, we will explain:

- The structure and use of the **Services.config.xml** configuration file

- Requirements and steps to configure the sub-services to support the Genesys ACD integration for the Knowlagent Enterprise Server.

Services.Config.xml Configuration File

The Knowlagent **Services.config.xml** .NET configuration file is used to configure the sub-services for the Knowlagent Enterprise Server service (**KAServicesX.exe**). Knowlagent installs this file in the following location:

[Installation Drive\Directory]\Knowlagent\APS\Config\Services.config.xml

The sub-services in the **Services.config.xml** configuration file include the various ACD integration sub-services (Genesys, Avaya, Aspect, and so on) and several other sub-services used by the Knowlagent Enterprise Server such as the **EventService**, **MessagingServer**, **SessionSchedulerService**.

Services.Config.xml Sub-Service Configuration

Sub-services in the **Services.config.xml** configuration file are configured within **<profile> </profile>** XML tag pairs. Each "profile" configures a single instance of a sub-service. Except for the profiles related to ACD integrations, which can have multiple instances configured, the other sub-services will only have one instance (and thus only one profile) configured.

Genesys ACD Default Profile Instance

The following figure shows the default, unedited **Genesys** profile instance.

```
<profile>
    <instance>Genesys</instance>
    <description>Implements Genesys near Real Time statistical report.</description>
    <assembly>Knowlagent.EnterpriseServer.GenesysSTA</assembly>
    <type>Knowlagent.EnterpriseServer.GenesysSTA.GenesysGIS</type>
    <context>0</context>
    <parameters>
        <acddefinitionname>UNKNOWN</acddefinitionname>
        <gishostname>UNKNOWN</gishostname>
        <gisport>9090</gisport>
        <statservername>StatServer</statservername>
        <username>default</username>
        <password>password</password>
        <pollingdelay>15</pollingdelay>
        <objecttype>GroupQueues</objecttype>
        <genesysswitchname>UNKNOWN</genesysswitchname>
        <queues>$KA_QUEUES$</queues>
        <metrics>
            <metric>ServiceFactor,1</metric>
            <metric>TotalNumberCallsAband,2</metric>
            <metric>AbandCallsPercentage,3</metric>
            <metric>AverAbandCallTime,4</metric>
            <metric>EstimTimeToDistribCall,5</metric>
            <metric>CurrNumberWaitingCalls,6</metric>
            <metric>CurrMaxCallWaitingTime,7</metric>
            <metric>AverDistribCallTime,8</metric>
            <metric>TotalNumberCallsDistrib,9</metric>
            <metric>DistribCallsPercentage,10</metric>
        </metrics>
    </parameters>
</profile>
```

Figure 18: Services.config.xml Profile Instance - Genesys ACD default instance

knowlagent®

Genesys ACD .NET sub-services configuration

In this section, we'll explain how to configure the .NET sub-services that allow Knowlagent to retrieve the Genesys ACD data from the GIS. The ACD sub-service profiles of the Services.Config.xml file you need to configure include :

- ACDStateManager instance

- ACDRulesEngine instance

- Genesys instance(s)

> **IMPORTANT NOTE:** Before you can do the actions described in the following section, the ACDs must have been configured in the Knowlagent Manager Console as described in Chapter 2.

ACD Configuration Process

The process of configuring the ACD sub-services in the **Services.config.xml** file consists of editing the value defined by a tag, such as the changing the default port number defined within the **<gisport> </gisport>** tag pair in the Genesys ACD profile:

```
<parameters>
    <acddefinitionname>UNKNOWN</acddefinitionname>
    <gishostname>UNKNOWN</gishostname>
    <gisport>9090</gisport>
    <statservername>StatServer</statservername>
```

One **ACDStateManager** and one **ACDRuleEngine** sub-service support all ACDs defined in the Knowlagent **Configure** tool interface. The ACDStateManager sub-service is configured by default to take the inputs from the ACDRuleEngine sub-service. Configuring a Genesys ACD integration consists of configuring the ACDRuleEngine to take the inputs from one or more Genesys ACD sub-service(s).

> **CAUTION! You should not modify** the non-ACD profiles in the Services.config.xml configuration file, or the **<assembly>** and **<type>** tags within the ACD (and all other) profiles. The **<assembly>** and **<type>** tags define the software component called by each profile, so changing these tags would cause the Knowlagent software to malfunction.

The following Task section describes how to configure the ACD sub-services in the Services.Config.xml configuration file to support Genesys ACDs.

Task - Configuring the ACD sub-services

The following task steps describe the actions required to configure the sub-services in the **Services.config.xml** configuration file to support the Genesys ACD integration.

Step	Action
1.	Obtain the configuration requirements and specifications from the customer.
2.	Find and open the **Services.config.xml** configuration file in a plain-text or XML editor (not MS Word). Knowlagent installs this file in the following location: **[Install Drive\Directory]\Knowlagent\APS\Config\Services.config.xml**
3.	In the default version of the **Services.config.xml** configuration file, the ACD-related sub-services are "commented out" and placed at the end of the file. Copy (do not cut) and paste the ACDRuleEngine and ACDStateManager profiles after the CRS profile instance. The figure below shows an example of this step completed.

```
<profile>
        <instance>CRS</instance>
        <description>Content Repository Service for Content Builder</description>
        <assembly>Knowlagent.APS.Subservices.Core</assembly>
        <type>Knowlagent.APS.Subservices.Core.ContentRepository.ContentRepository</type>
        <parameters>
                <configid>0</configid>
                <virtualaddress></virtualaddress>
        </parameters>
</profile>
<profile>
        <instance>AcdStateManager</instance>
        <description>Manages ACD and Queue states</description>
        <assembly>Knowlagent.APS.Subservices.Core</assembly>
        <type>Knowlagent.APS.Subservices.Core.ACDState.AcdStateManager</type>
        <inputs>
                <input>ACDRuleEngine</input>
        </inputs>
        <parameters>
                <writeruleresults>true</writeruleresults>
        </parameters>
</profile>
<profile>
        <instance>ACDRuleEngine</instance>
        <description>ACD Rule Engine</description>
        <assembly>Knowlagent.APS.Subservices.Core</assembly>
        <type>Knowlagent.APS.Subservices.Core.ACDRuleEngine.AcdRuleEngine</type>
        <inputs>
                <input>[ACD DATA COLLECTOR INSTANCE NAME (ie. AvayaCMS)]</input>
        </inputs>
        <parameters>
                <writestats>false</writestats>
        </parameters>
</profile>
```

Figure 19: ACDStateManagerand ACDRuleEngine profiles after the CRS sub-service

(Steps Continued)

Step	Action

4. In the default version of the **Services.config.xml** configuration file, the Genesys sub-service profile is also commented out and placed at the end of the file. Copy (do **<u>not</u>** cut) and paste this profile before the ACDStateManager profile instance that will process the Genesys ACD statistics. The figure below shows this step completed.

```
<profile>
    <instance>Genesys</instance>
    <description>Implements Genesys near Real Time statistical report.</description>
    <assembly>Knowlagent.EnterpriseServer.GenesysSTA</assembly>
    <type>Knowlagent.EnterpriseServer.GenesysSTA.GenesysGIS</type>
    <context>0</context>
    <parameters>
        <acddefinitionname>UNKNOWN</acddefinitionname>
        <gishostname>UNKNOWN</gishostname>
        <gisport>9090</gisport>
        <statservername>StatServer</statservername>
        <username>default</username>
        <password>password</password>
        <pollingdelay>15</pollingdelay>
        <objecttype>GroupQueues</objecttype>
        <genesysswitchname>UNKNOWN</genesysswitchname>
        <queues>$KA_QUEUES$</queues>
        <metrics>
            <metric>ServiceFactor,1</metric>
            <metric>TotalNumberCallsAband,2</metric>
            <metric>AbandCallsPercentage,3</metric>
            <metric>AverAbandCallTime,4</metric>
            <metric>EstimTimeToDistribCall,5</metric>
            <metric>CurrNumberWaitingCalls,6</metric>
            <metric>CurrMaxCallWaitingTime,7</metric>
            <metric>AverDistribCallTime,8</metric>
            <metric>TotalNumberCallsDistrib,9</metric>
            <metric>DistribCallsPercentage,10</metric>
        </metrics>
    </parameters>
</profile>
<profile>
    <instance>AcdStateManager</instance>
    <description>Manages ACD and Queue states</description>
    <assembly>Knowlagent.APS.Subservices.Core</assembly>
```

Figure 20: Pasting Genesys profile before the ACDStateManager sub-service

5. Re-name the default Genesys value in the **<instance>** tag.

> **IMPORTANT NOTE:** Knowlagent recommends that the ACD <instance> tag name be correlated to the ACD name defined in the Knowlagent UI at the following location:
>
> **RIGHTTIME DELIVERY > ACD DEFINITION**
>
> For example, if the ACD name in the Knowlagent UI is defined as "**TechSupport_01**," the recommended instance name configured for the **<instance>** tag could be similar to the following:
>
> **GenesysACDForTechSupport_01**

(Steps Continued)

Step	Action
6.	If a second Genesys instance is required, copy and paste another Genesys instance profile after the first. Repeat for as many Genesys ACDs as are required.
7.	Re-name any additional Genesys instances so that they correlate to the appropriate ACD names defined in the Knowlagent UI **RIGHTTIME DELIVERY > ACD DEFINITION**.
8.	For each Genesys ACD <instance>, configure the **<parameters>** settings as specified in the following table according to requirements of the GIS.

Parameter	Entry Specification	Default Entry	Data Type
acddefinitionname	ACD name that exactly matches a corresponding ACD name defined in the **RIGHTTIME DELIVERY > ACD DEFINITION** page of Knowlagent UI.	UNKNOWN	String
gishostname	Hostname or TCP/IP address of the machine that contains the GIS Web service.	UNKNOWN	String
gisport	The TCP/IP port value on which the GIS Web service listens for requests. This entry must specified in the GIS Properties Server Info tab in the Genesys Configuration Manager.	9090	Int
statservername	Genesys Statistical Server name from which this integration will obtain values.	StatServer	String
username	Logon identifier for the GIS system that has rights to obtain statistical and configuration data.	default	String
password	The password that accompanies the username to obtain access to the GIS.	password	String
pollingdelay	This value controls how often (in seconds) the Knowlagent integration will request current statistical values from the Genesys system. This value cannot be any lower than what is defined for the StatServer listed under the GIS in the Genesys Configuration Manager. See the ACD Polling Delay section in **Chapter 2**.	15	Int
objecttype	Queue type and addressing scheme used for String requesting queue information. Queues can be **Queue** or **GroupQueues** object types.	queue	String
genesysswitchname	Name of the internal Genesys switch where the queue is defined. This value is only applicable when the object type is set to **Queue**. Switch names are case sensitive and must exactly match the switch name defined in the **\Configuration\Resources\Switches** folder of the Genesys Configuration Manager.	UNKNOWN	String
queues	Queues that will be monitored by this integration. The default **KA_QUEUES** value causes the integration to monitor all queues associated with the **acddefinitionname**. Alternatively, a comma-separated list of specific queue numbers can be entered such as: **<queues>8000,8001,9000<\queues>**	KA_QUEUES	String
metrics	Configured list of Calculated Parameters that the integration will monitor. See ACD Calculated Parameters section in **Chapter 2**.	See Chapter 2	XML

Table 13: Genesys instance parameter specifications in the Services.Config file.

(Steps Continued)

![knowlagent® logo]

Step	Action

9. In the ACDRuleEngine profile instance, replace the default input text (highlighted below) with the name you configured for the Genesys ACD <instance> tag in the preceding steps.

```
<profile>
    <instance>ACDRuleEngine</instance>
    <description>ACD Rule Engine</description>
    <assembly>Knowlagent.APS.Subservices.Core</assembly>
    <type>Knowlagent.APS.Subservices.Core.ACDRuleEngine.AcdRuleEngine</type>
    <inputs>
        <input>[ACD DATA COLLECTOR INSTANCE NAME (ie. AvayaCMS)]</input>
    </inputs>
    <parameters>
        <writestats>false</writestats>
    </parameters>
</profile>
```

Figure 21: ACDRuleEngine profile instance - Default <input> text

10. If you are configuring more than one ACD, you must add an additional **<input> <\input>** tag pair in the **ACDRuleEngine** profile and configure the appropriate **<instance>** name for each ACD. The following figure shows this step with the "**GenesysACDForTechSupport_01**" and "**GenesysACDForCustServ_02**" inputs configured.

```
<profile>
    <instance>ACDRuleEngine</instance>
    <description>ACD Rule Engine</description>
    <assembly>Knowlagent.APS.Subservices.Core</assembly>
    <type>Knowlagent.APS.Subservices.Core.ACDRuleEngine.AcdRuleEngine</type>
    <inputs>
        <input>GenesysACDForTechSupport_01</input>
        <input>GenesysACDForCustServ_02</input>
    </inputs>
    <parameters>
        <writestats>false</writestats>
    </parameters>
```

Figure 22: ACDRuleEngine profile instance -Two <input> settings specified

11. Save the **Services.config.xml** configuration file when done.

You have completed this task. Go on to Chapter 4 – Completing the integration deployment.