Introduction to eServices8: Architecture and Business Process

Marc Sassoon
Master Technical Instructor
September 2012

Arne Stoertenbecker
Solutions Architect

create the new conversation
Class Topics

Introduction to Genesys eServices
eServices Architecture
Introduction to Business Processes and Strategies
Introduction to eServices
What is Genesys eServices?

Genesys eServices is a suite of products that enables automatic and efficient routing for non-voice Genesys eServices products and capabilities include...

- Genesys E-mail, Genesys Chat, Genesys SMS Server, Genesys Web Callback
- Integration with Social Media (Twitter and Facebook)
- Processing of 3rd party interactions (Web Form Email etc)

The Customer Interaction Management (CIM) Platform is a pre-requisite
eServices Benefits

**Improve operational efficiency**

- Prioritize the handling of all interactions for both agents and functional experts across the organization
- Manage all interactions through a single application

**Improve customer satisfaction**

- Apply skills based routing to all interactions
- Leverage customer data to personalize interactions

**Reduce cost**

- Maximize utilization of resources through intelligent routing
eServices supports *Non-voice* Routing

Voice  E-mail  Chat  SMS  Web  Social Media

Customer Interaction Management Platform

Contact Center Resources
The Genesys Media Interfaces (1 of 2)

E-mail Server
• Handles new e-mail interactions from enterprise mail server or a web page
• Sends out replies or other outbound messages

Chat Server
• Handles web-based chat interactions between agents and customers

Web API Server
• Web application
• For all web-based interactions, for example:
  • Support chat sessions
  • Handle callback requests (from a website)
The Genesys Media Interfaces (2 of 2)

SMS Server
• Receives and handles SMS / MMS messages sent from a mobile client

Social Messaging Server
• Monitors a social media site, gathers items that fit a defined profile, and converts them into Genesys interactions so they can be managed
• e.g., Facebook posts and Twitter Tweets
Facebook Interactions

Interactions from Facebook can be based on

• ID of a Facebook object for monitoring
  • Facebook object can be a page, user, event, application, or group.
  • Web developer of the company presence on Facebook would be able to provide the id for each desired object to monitor.

• Any custom valid Facebook query (string)
  • e.g., Genesys
Twitter Interactions

Interactions from Twitter can be based on

• Using Twitter driver's own account
  • Home timeline
  • Mentions - e.g. @Genesys
  • Direct messages

• Any valid Twitter search expressions
  • e.g., from:BusinessCustomer
3rd Party Media

Additional interactions can be retrieved if you use eServices with **SDKs** or **iWD**

**SDKs:** 3rd Party Media SDK and Interaction SDK
  - Programming interfaces

**iWD:** intelligent Workload Distribution
  - Can “capture” interaction from external systems (tasks, workitems)
Media Types

When a new interaction arrives in the system, the Media Server assigns a media type.

Media types are configured as Business Attributes.
How are Media Types used?

In Routing decisions

In Agent Capacity Rules
• Capacity Rules specify the maximum number of interactions of each media type an agent can handle at the same time

In the configuration of the Agent Desktop
• You define what media types the desktop application should handle
eServices Architecture
Genesys eServices includes three major categories of components based on functionality:

- **Media Interface**
- **Workflow Control**
- **Data Storage and Organization**
Media Interface

Interface to an external system
Brings interactions into eServices
Transmits body of interaction to Data Storage
Transmits operational data to Workflow Control
A database stores:

- Interaction content
- Interaction history
- Contact information
Controls where an interaction goes and what happens to it

- Delivered to agent
- Auto-responded
- Stopped

Two groups of components

- Interaction Management
- Routing
Functional Categories – True or False

✔️ Learning Check

Three categories of eServices functionality are Media Interface, Workflow Control, and Agent Desktop.

The Media Interface brings interactions into the system.

The Data Storage device stores interaction content only.
eServices uses Genesys Framework components architecture diagram.

Components will be covered based on functionality:
1. Interaction Server and Routing (Workflow Control)
2. Media Interfaces
3. Data Storage
4. Knowledge Management (Workflow Control)
Core Components – Interaction Server

**Interaction Server (Ixn Server)**

- Receives interactions (operational data) from media interface
- Stores interactions in Interaction Database (Ixn DB)
- Sends routing requests to URS
- Facilitates agent log in
- Provides some statistics to Stat Server
Universal Routing Server (URS)
- Executes routing strategies as requested by Ixn Server
- Communicates with media interfaces through Ixn Server
Core Components – Stat Server

**Stat Server**
- Provides statistical information to routing and reporting clients

**Media Interfaces**

**Data Storage**
Interaction Routing Designer (IRD)

- GUI for defining/managing logic on how to handle interactions
- Define business processes (for Ixn Server) as well as strategies (for URS)
Interaction Server versus URS

Interaction Server

• Accepts new interactions and saves them in a database
• Manages communication with URS
• Delivers interactions to the agent and provides inter-agent operations
• Stops interactions upon request
• Provides reporting information to reporting clients like Stat Server

URS

• Executes strategies for interactions submitted by Interaction Server
• Communicates with other Servers through Interaction Server
  • Communication with media interfaces
  • Communication with external services via External Services Protocol (ESP) to retrieve user data and/or routing data
Interaction Server stores all active interactions in the database. This operational data includes:

- Interaction ID
- Media Type
- Queue information
- Agent ID
- Time stamps

Please note: The actual content of the interaction is NOT stored in the Ixn Database.
E-Mail Server
- Gets e-mails from POP3/IMAP Server
- Stores e-mails in data storage
- Submits operational data to Interaction Server
- Sends e-mails out through SMTP gateway
Web API Server

- Collection of servlets/objects in container such as Tomcat
- Supports a web application submitting interactions to Genesys
- Used for e-mail, chat, web callback, as well as custom media types
Chat Server

- Receives chat request from Web API Server and submits interaction to Ixn Server
- Interfaces with Web API and the agent desktop to support the chat
- Option for web collaboration
Media Interfaces – SMS Server

**SMS Server**
- Receives SMS messages and submits as interactions to Ixn Server
- Sends outbound messages
- Operates in 2 modes:
  - Paging mode
  - Session mode (via Chat Server)
Media Interfaces – Social Messaging Server

Social Messaging Server

- Interfaces with Facebook and/or Twitter to gather desired items and submit as interactions to Ixn Server
- Sends replies back to social media site
Media channel drivers are added to the server as needed

- Twitter, Facebook, ...

Twitter/Facebook search queries are defined in the options of the Social Messaging Server
Examples of Customer Side

- **E-mail**
  - From: customer@training.com (Customer)
  - To: support
  - Cc: 
  - Subject: Account access
  - I need helping accessing my account online.

- **Web form**
  - Media type: webform
  - Workflow queue: webform-inbound
  - Tenant name: Resources
  - Data attached to the interaction:
    - First name: Joe
    - Last name: Customer

- **Twitter**
  - Suzy
  - View my profile page
  - 28 tweets, 1 following, 0 followers
  - I loved the Genesys @EsvTraining class I took last week.
Sample Interactions in Interaction Workspace

- E-mail
- Web form
- Twitter
Agent Response

Possible ways for agent to respond depends on the type of interaction and the choices provided by the agent desktop.

- **E-mail**
  - Reply the email
  - Transfer

- **Web form**
  - Transfer

- **Twitter**
  - Reply on Twitter

[Image of agent desktop interfaces for E-mail, Web form, and Twitter interactions]
Sample Twitter Interaction in Interaction Workspace

- Number of user followers
- Number of user tweets
- If the user is a follower
- To follow/unfollow the user
- Actionability
- Sentiment
Sample Facebook Interaction in Interaction Workspace

- Male/Female
- Number of Likes
- Actionability
- Sentiment
- Comment
- Delete (if allowed by author)

Includes original post and associated comments

- Open user profile
- View/Hide to expand or collapse a comment
Demo – Genesys Social Engagement

Twitter
Universal Contact Server (UCS)

- Stores information and history about contacts and interactions
- Provides some statistics to Stat Server
- Supports Knowledge Management components
Data Storage - UCS Manager

**UCS Manager**
- GUI for pruning and archiving data

**Media Interfaces**
- Social Messaging Server
- SMS Server
- Chat Server

**Web Server**
- Twitter/Facebook
- SMS Provider
- IMAP Server

**Data Storage**
- UCS DB
- Ixn DB

**Interaction Server and Routing**
- IRD
- Agent Desktop

**Knowledge Management**

**Web API Server**
- UCS DB

**Data Storage - UCS Manager**

**Agent Desktop**
- UCS Manager
- UCS DB
- DAP
Knowledge Management - Knowledge Manager

- **Knowledge Manager**
  - GUI for managing categories, standard responses, screening rules, and models

**Media Interfaces**
- Twitter/Facebook
- SMS Provider
- WWW
- POP3/IMAP

**Data Storage**
- UCS Manager
- DAP UCS DB

**Interaction Server and Routing**
- IRD

**Classification Server**
- UCS DB

**Agent Desktop**
- WWW

**Web API Server**
- Chat Server

**Social Messaging Server**
- SMS Server

**SMS Provider**
- E-Mail Server

**Media Interfaces**
- Web API Server

**Knowledge Management**
Training Server

- Trains the system to recognize categories
- Produces model for use in classification
Knowledge Management - Classification Server

Classification Server
• Applies models and screening rules

Media Interfaces
- Social Messaging Server
- SMS Server
- Chat Server
- Web API Server
- E-Mail Server

Twitter/Facebook
- SMS Provider
- WWW
- POP3 IMAP

Interaction Server and Routing
- IRD
- Agent Desktop

Classification Server
- Knowledge Manager
- Training Server
- Knowledge Management

Data Storage
- UCS
- DAP UCS DB

Interaction Server
- Ixn DB
- Ixn DB Server
- Ixn Server
- Stat
- URS

Media Interfaces
Create and manage the Standard Response Library

- Organized in categories
- Used as an acknowledgment, auto-response or suggestion to an agent
- Responses can include variables (called Field Codes)
Create and manage Screening Rules

<table>
<thead>
<tr>
<th>Name</th>
<th>Rule Body</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit card number</td>
<td>RegExFind(&quot;\b\d{4}[-]\d{4}[-]\d{4}[-]\d{4}[-]\d{4}\b&quot;, &quot;Card_number&quot;)</td>
</tr>
<tr>
<td>Quality prove</td>
<td>Find(&quot;dumb&quot;,true)</td>
</tr>
<tr>
<td>Tech support</td>
<td>(Find(&quot;problem&quot;,true)</td>
</tr>
<tr>
<td>Unidentified transaction</td>
<td>Find(&quot;transaction&quot;,true) &amp; &amp; (Find(&quot;wrong&quot;,true)</td>
</tr>
<tr>
<td>Wrong transaction amount</td>
<td>Find(&quot;transaction&quot;,true) &amp; &amp; Find(&quot;amount&quot;,true) &amp; &amp; (Find(&quot;wrong&quot;,true)</td>
</tr>
<tr>
<td>Warranty problem</td>
<td>Find(&quot;warranty&quot;,true) &amp; &amp; (Find(&quot;problem&quot;,true)</td>
</tr>
<tr>
<td>Auto Response Available</td>
<td>Find(&quot;customer support&quot;,true) &amp; &amp; (Find(&quot;phone number&quot;,true)</td>
</tr>
<tr>
<td>ScreenForPositiveSentiment</td>
<td>Find(&quot;rox&quot;)</td>
</tr>
<tr>
<td>ScreenForNegativeSentiment</td>
<td>Find(&quot;hate&quot;)</td>
</tr>
<tr>
<td>ScreenForNeutralSentiment</td>
<td>!Find(&quot;hate&quot;) &amp; &amp; !Find(&quot;love&quot;) &amp; &amp; Find(&quot;fail&quot;) &amp; &amp; Find(&quot;dumb&quot;) &amp; &amp; Find(&quot;stupid&quot;) &amp; &amp; !Find(...</td>
</tr>
<tr>
<td>Actionable</td>
<td>Find(&quot;Fail&quot;)</td>
</tr>
</tbody>
</table>
| UnclearIfActionRequired     | !Find("fail") & & !Find("help") & & !Find("how to") & & !Find("how do") & & !Find("assist")...

• Run by Classification Server
Content Analyzer allows you to create and use statistical models

The process of creating a Model is called training

• Training Server analyzes messages which are typical for each category and builds a model

Classification Server can apply a Model to a new interaction and “classify” it

• As a result the interaction can be associated with one or more Categories and each is given a probability

• Classification is triggered from a strategy
Sentiment and Actionability

Analyze the sentiment and actionability of interactions that have been brought into the system by Social Messaging Server

• Can be done using Screening Rules or Content Analyzer
• Genesys provides samples for each

Sentiment
• Positive, Negative, Neutral

Actionability
• Actionable, NonActionable
Sample Interactions with Sentiment/Actionability

Samples shown for Interaction Workspace
eServices GUI Components

**IRD**
Create Business Processes and Strategies

**Composer**
Create Interaction Process Diagrams and Workflows

**Knowledge Manager**
Create Standard Responses, Rules, Statistical Models

**UCS Manager**
Archive or prune UCS data
Name that Component

Learning Check

1. GUI for defining business processes and strategies

2. Stores contact information and history, standard responses, screening rules

3. GUI for managing standard responses, screening rules, models
Introduction to Workflows and Strategies
Business Processes and Strategies

Business Process
• An interaction processing scenario that covers the whole interaction life cycle
• Includes Strategies and Workflow Queues
• Executed by Interaction Server

Strategy
• Encapsulates all processing details
• Can update interaction data, apply services, segment, prioritize, target, and so on
• Selects a target
• Executed by Universal Routing Server
Business Processes and Strategies are created in Interaction Routing Designer (IRD)

• Stored in the form of script objects in the Configuration database

IRD is a GUI that has two ‘Design’ windows:

• The Routing Design window, which displays a graphical view of a strategy or subroutine. Using this window, you can build and edit a strategy or subroutine

• The Interaction Design window, which enables you to create Business Processes that route non-voice interactions
IRD: Strategies and Business Processes

Routing Design bar
• Includes Routing Strategies and reusable objects

Interaction Design bar
• Includes the Business Processes
• Can be activated from the IRD options
The Interaction Design window is an IRD interface, in which you create Business Processes.
You can add new components using the...

- Toolbar
- Workflow Viewer
- Object Browser
A place where an interaction can park as long as necessary

- Interaction Queues are defined in IRD as part of a workflow
- Interactions leave a Queue for further processing by a strategy based on default or user-defined conditions
- May be connected to 0 – N strategies
A Business Process can have many Interaction Queues for different purposes

- Interaction pre-processing
- Inbound
- Escalation
- Outbound
- QA Review
Working with Interaction Queues: Use

Interaction Server puts interactions into Queues

• As instructed by a Media Server

• As instructed by a Strategy (URS)
  • Target Object
  • Queue Interaction
Basic Business Process Elements: Strategy

Strategies define processing and routing details

- Strategies are built (or modified) in a separate window from the 'containing' Business Process

- The Business Process displays objects from the strategy, they are called: `strategy-linked-nodes`
Working with Strategies: Important Steps

If you view or edit a strategy:
- It gets deactivated and unloaded from URS.

If you activate a strategy:
- It gets loaded on a Virtual Routing Point (created automatically).
Business Process Handling

Interaction Server - distributes interactions from queues to strategies

- Uses a View object in combination with configuration options to determine submission criteria and meet performance objectives.
  For example, to manage throughput to keep URS busy but not overloaded.

URS - executes strategy as soon as it gets an interaction from Interaction Server.
Queue View

A Queue View provides a way to define the criteria for pulling interactions from a queue or a workbin and submitting them to the appropriate routing strategies:

- A Queue can have more than one View
- A View is always connected to a Strategy
A **Condition** on a View specifies criteria that must be met in order for an interaction to be submitted from this View to the connected Strategy

- Until the Condition is met, the interaction stays “parked” in the Queue
- Conditions are translated into SQL run against the interactions table

The **Order** tab specifies a SQL order clause

- Default order (empty tab):
  
  Interactions are submit based on **received_at** (time when Interaction Server received the interaction) and **id** (Interaction ID)
Workbin - A Special Queue

Like a queue, a **workbin** holds (parks) interactions

*However*, a **workbin** is associated with a particular **agent/place/group** and its major function is to hold interactions for **that** agent/place/group

From their desktop application, agents can use workbins to store interactions they plan to work on or complete later

Interactions can also be distributed directly to workbins by routing strategies
Interactions are typically **pushed** to agents

- A strategy defines a target and if an agent is available for the interaction’s media type, it is sent (pushed) to the agent

You can allow agents to **pull** interactions from a workbin

- A strategy puts the interaction into a workbin
- Agents can access the workbin from their desktop application and select (pull) interactions they want to work on
Basic Interaction Management

1. The Media Server sends operational data about the interaction to Interaction Server
2. The interaction is “put” into an Interaction Queue
3. Interaction Server submits the interaction to the strategy associated with the Queue
4. URS executes the strategy and instructs Interaction Server about the next steps:
   - Route interaction to an agent
   - Place interaction into a queue or workbin
   - Stop interaction processing
Interaction Management: Strategy Actions

Universal Routing Server / Strategies

- Strategy 1
- Strategy 2

Interaction Server / Workflow

**Strategy Actions:**
1. Route to Agent
2. Place in Workbin
3. Stop Interaction
4. Place in Queue

Agent Desktop
- PlaceID
- AgentID
Interaction Management: Desktop Actions

Universal Routing Server / Strategies

Strategy 1

Strategy 2

Interaction

Interaction Server / Workflow

Q1

Q2

Qn

Agent Desktop

- PlaceID
- AgentID

Agent Desktop

- PlaceID
- AgentID

Basic Agent Actions:
1. Transfer
2. Place in Workbin
3. Stop Interaction
4. Place in Queue
eServices Reporting and Analytics
Reporting Overview

Solution Reporting:
• CC Pulse+
• CC Analyzer

Analytical Reporting:
• Info Mart
• Interactive Insights
Reporting Products

CCPulse+ and CCAnalyzer

• Bundled with the CIM Platform
• As long as you’ve bought some media channel seats of any type
• Referred to as ‘Solution Reporting’

Info Mart

• Not bundled with the CIM Platform
• Interactive Insights interface for reports

Both provide some templates and reports for media types
Analytical Reporting Overview

Info Mart Components

- Collect data based on reporting events from Interaction Server
- Store data in a database
Solution Reporting Overview

Stat Server

• Calculates data based on reporting events
• Includes Stat Server Java Extensions (SSJE) to handle statistics from Interaction Server and UCS

Diagram:

- Ixn Server
- Stat Server
- UCS
- CCPulse+
- CCAnalyzer Components
Stat Server Java Extensions

Stat Server Java Extensions (SSJE)
• Are a set of Java classes
• Plugins with new statistical types
• When loaded, each SSJE passes its own statistical definitions to Stat Server

Mandatory for some eServices reporting
• Used for interaction queue and system-wide (tenant) statistics
Stat Types (Statistical types)

Stat types are used to define statistics in the Stat Server options.

There are two main types:

• **Core stat types**: Values are calculated directly within Stat Server.
• **Java stat types**: Values are provided to Stat Server by another Genesys server through the Java Extensions.
Objects for Reporting

Objects used in Java categories:

- StagingArea (Interaction Queue)
- Tenant
- RoutingStrategy
Reporting Templates CD

Sample statistic definitions and reporting templates

Provided on Reporting Templates Software CD
• E-mail Reporting Templates (erms)
• Chat Reporting Templates (liveweb)
• Samples for other media types (om-sample)
CCPulse Template Samples for eServices

E-mail Templates
• E-mail Queue
• General E-mail Handling (for Tenant)
• Resource E-mail Handling

SMS Template
• Agent Performance SMS

Chat Templates
• General Chat Handling (for Tenant)
• Resource Chat Handling

Model Templates
Pre-defined Stat Types for Email and Chat

Stat Types for email and chat are ready to use, for example:

<table>
<thead>
<tr>
<th>Stat Type</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>General_Email_Entered</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Forwarded</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_In_Processing</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Internal</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Maximum</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Minimum</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Not_Submitted</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Oldest_Age</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Outbound</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Redirected</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Responded</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_ShortResponse_Time</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Response_Time</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>General_Email_Waiting_Processing</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stat Type</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chat_Current_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Current_Waiting</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Abandoned</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Answered</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Abandoned</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Answered</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Handle_Time</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Handled</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Inbound_Terminated</td>
<td>5</td>
</tr>
<tr>
<td>Chat_Total_Transfers</td>
<td>5</td>
</tr>
</tbody>
</table>
Stat Types for Other Media Types

Stat types for other media types
• Based on MediaType = 'x'
• Use as reference to create custom stat types for specific media types

```
MediaX_Current_In_Processing_In_Queue (6 Items)
MediaX_Current_In_Queue (6 Items)
MediaX_Current_Waiting_Processing_In_Queue (6 Items)
MediaX_Maximum_Interactions_In_Queue (6 Items)
MediaX_Minimum_Interactions_In_Queue (6 Items)
MediaX_Stopped_Processing_In_Queue (6 Items)
MediaX_Total_Entered_Queue (6 Items)
MediaX_Total_Moved_From_Queue (6 Items)
```
Monitoring the Capacity in CCPulse+

You can configure CCPulse to display agent and agent group capacity

• In the Object Tree

![Image of Object Tree](image)

(capacity available/not available)

• In a Report View

<table>
<thead>
<tr>
<th>IA - Ian.A</th>
<th>Statistic</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>CurrentState</td>
<td>Capacity SMS</td>
<td>sms(0, 1, 0)</td>
</tr>
<tr>
<td>Capacity Chat</td>
<td>chat(0, 1, 1)</td>
<td></td>
</tr>
<tr>
<td>Capacity email</td>
<td>email(1, 1, 0)</td>
<td></td>
</tr>
</tbody>
</table>

Values: media (current, maximum, routable interactions)
To view the capacity details:

1. Right-click an agent or agent group
2. Select Monitor Extended Status
3. Open the Capacity Pane
Genesys University eServices 8 Classes

**eServices 8.1 Foundations (ESV8.1-FND)**
2 days

**eServices 8.1 Development (ESV8.1-DEV)**
3 days

**eServices 8.1 Deployment (ESV8.1-DPL)**
3 days
eServices Course Paths 1/2

Consultants

Pre-Requisites

- Framework 8 Foundations
- Framework 8 Deployment
- Inbound Routing and Reporting 8 Deployment

- eServices 8.1 Foundations
- eServices 8.1 Deployment

© 2012, Genesys Telecommunications Laboratories, Inc. All rights reserved.
Developers

Pre-Requisites

- Framework 8 Foundations
- Building Basic Routing Strategies 7

- eServices 8.1 Foundations
- eServices 8.1 Development
Thank You