Introduction to Contact Center API’s and Cisco Finesse

BRKCDN-1002
Enabling True Customer Collaboration

Combining Traditional Tools with the Power Tomorrow

- Virtual Contact Center Routing and Reporting
- Social Media Customer Care
- Speech Self-Service
- Collaboration Agent Desktop
- Multi-Media Capture and Storage
- Multichannel
- Video-Enabled Customer Care
- Enterprise Experts

Customer Collaboration

Past 2010
Cisco Finesse
Cisco Finesse Objectives

Collaborative Experience

- Agent tools to reduce handle time and improve caller experience
- Empower agents through the use of a user-center designed product

IT-Friendly

- Seamless integration with Cisco Collaboration portfolio
- Standards Compliant
- Thin Client to reduce TCO

Powerful Ecosystem

- Partner Differentiation through business and vertical applications
- Cisco App Marketplace
Cisco Finesse 8.5(1)
Developer Enablement – December 2010 (Lab Use Only)

Overview
- Web 2.0 SDK for Unified Contact Center Enterprise 8.5
- Gadget-based Agent Desktop UI for CCE 8.5
- For use by partners and customers for lab and development usage only

Features
- Call Control API
- Agent State API
- Real-Time Queue Stat API
- Web Based Admin
- UCS C-Series Support
Open Social Gadgets

“Mini-website” inserted into a larger web-page.
HTML and JavaScript within an XML wrapper.
Creating Gadgets

http://code.google.com/apis/gadgets/docs/gs.html

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<Module>
  <ModulePrefs title="hello world example" />
  <Content type="html">
    <![CDATA[ Hello, world! ]]>
  </Content>
</Module>
```
Cisco Finesse Web Services
Commands sent by URL and responses in name/value pairs

http://.... URL Based Access

```json
{ "response":"success", 
  "queue": { 
    Response Data
    "numAgentsTalking":15, 
    "numAgentsReady":18, 
    "numAgentsNotReady":9, 
  } } 
```
Modular Architecture enables targeted customization
Out-of-the-box & custom components within the same desktop

Agent Desktop

Unmodified Cisco Gadget
Meets customer need as shipped by Cisco

Customized Cisco Gadget
Minor changes to meet customer need

Customer Gadget
Created and Delivered by customer or partner
High-Level Architecture

Cisco Finesse Server

Web Services  |  Gadget Container

Cisco Finesse Client

Custom Applications

Cisco Unified Contact Center Enterprise PG
“Out of the Box” Deployment
For integrations that include enterprise web applications
Enabled via configuration / no custom development required

http://myserver/acctnum=<callvar1>&name=<callvar2>
Cisco Container Mashup Deployment

For integrations to enterprise server applications

Requires custom gadget development
3rd Party Application Deployment

When the main application is not Cisco’s agent desktop

Fully customized desktop experience
Cisco Finesse Web Services

- Web Service
  - Sets of APIs
    - Agent Sign In/Out
    - Agent States
    - Agent & Queue Statistics
    - Agent, Queue, Supervisor, Team Configurations
    - Call Control
    - Subscriptions to above feeds
  Exposed through HTTP RESTful interface

- Notification Service
  - Real-time event feeds
  - Agent states, statistics, configuration, & call events
  - Uses XMPP protocol as delivery mechanism
Making Requests

- General format (applies to most, but not all)
  http://host:80/webservices/[ServiceType]/[ServiceEntity]/[EntityID]/[ResourceType]/[Action]?param1=<xxx>&param2=<xxx>
  http://host:80/webservices/AgentService/Agent/<agentId>/state/change?newState=3

- Should make HTTP requests to port 80

- HTTP method is GET or POST

- Response will be JSON format
  
  ```json
  {"response": "success", [data]...}
  ```

- Errors
  
  ```json
  {"response": "failure", "errorMessage": "<MESSAGE>", "errorCode": "<CODE>"}
  ```

- Refer to Developer Guide for parameter and payload details
Authentication

- All API request should use Basic Authentication to send user credentials in the header of the HTTP request
  BASE64-encoded string under Authorization header
  For example, johndoe:cisco BASE64 encoded is am9obmRvZTpjaXNjbw==
  Authorization: Basic am9obmRvZTpjaXNjbw==

- Must first Sign In to retrieve token

- The Sign In API request, if successful, will return a JSESSIONID token in the header of response. Clients must also include the JSESSIONID token in the header of all subsequent requests.

- SignIn is the only API that can be accessed without the JSESSIONID.
Request Example

1) Send Agent Sign In request with Basic Auth credentials
   POST http://[host]:80/webservices/ConnectionService/Connection/signIn
   Authorization: Basic am9obmRvZTpjaXNjbw==

2) Response will provide token
   HTTP/1.1 200 OK
   Set-Cookie: JSESSIONID=AC32B27618E3997C16F28F357AB7A852; Path=/webservices

3) Send both Basic Auth and JESSSIONID in Get State request
   GET http://[host]:80/webservices/AgentService/Agent/johndoe/state
   Authorization: Basic am9obmRvZTpjaXNjbw==
   Cookie: JSESSIONID=AC32B27618E3997C16F28F357AB7A852

4) Send request to Change Agent State
   POST http://[host]:80/webservices/AgentService/Agent/johndoe/state/change?newState=3
   Authorization: Basic am9obmRvZTpjaXNjbw==
   Cookie: JSESSIONID=AC32B27618E3997C16F28F357AB7A852

5) Send Agent Sign Out request
   POST http://[host]:80/webservices/ConnectionService/Connection/signOut
   Authorization: Basic am9obmRvZTpjaXNjbw==
   Cookie: JSESSIONID=AC32B27618E3997C16F28F357AB7A852
Agent APIs

- Change State
  POST /AgentService/Agent/<agentId>/state/change*

- Retrieve State
  GET /AgentService/Agent/<agentId>/state

- State Events
  POST /AgentService/Agent/<agentId>/state/subscribe*
  POST /AgentService/Agent/<agentId>/state/unsubscribe*

- Agent Desk Settings
  GET /AgentService/Agent/<agentId>/settings/desk

* Requires additional parameters (refer to Developer Guide)
Agent & Queue Statistic APIs

### Agent Statistics
- GET /AgentService/Agent/<agentId>/statistics
- POST /AgentService/Agent/<agentId>/statistics/subscribe*
- POST /AgentService/Agent/<agentId>/statistics/unsubscribe*

### Queue Statistics
- GET /QueueService/Queue/<queueId>/statistics
- POST /QueueService/Queue/<queueId>/statistics/subscribe*
- POST /QueueService/Queue/<queueId>/statistics/unsubscribe*

* Requires additional parameters (refer to Developer Guide)
Agent, Queue, Team, Supervisor Config API

- Agent Configuration
  GET|POST
  /AgentService/Agent/<agentId>/config/[subscribe|unsubscribe]*

- Queue Configuration
  GET|POST
  /QueueService/Queue/<queueId>/config/[subscribe|unsubscribe]*

- Team Configuration
  GET|POST
  /TeamService/Team/<teamId>/config/[subscribe|unsubscribe]*

- Supervisor Configuration
  GET|POST
  /SupervisorService/Supervisor/<supervisorId>/config/[subscribe|unsubscribe]*

* Requires additional parameters (refer to Developer Guide)
Call Control APIs

- Call event feed
  POST /AgentService/Agent/<agentId>/call/subscribe
  POST /AgentService/Agent/<agentId>/call/unsubscribe

- Call control actions
  POST /CallService/Call/<callId>/answerCall
  POST /CallService/Call/<callId>/clearCall
  POST /CallService/Call/<callId>/clearConnection*
  POST /CallService/Call/<activeCallId>/conferenceCall*
  POST /CallService/Call/makeCall*
  POST /CallService/Call/<activeCallId>/consultCall*
  POST /CallService/Call/<callId>/holdCall
  POST /CallService/Call/<callId>/retrieveCall
  POST /CallService/Call/<callId>/dtmfDigits*
  POST /CallService/Call/<callId>/callData/set*
  POST /CallService/Call/<activeCallId>/transferCall*

Please note that Finesse Web Services enforce a policy check that only allows call parties to perform call control operations via API’s.

* Requires additional parameters (refer to Developer Guide)
Call Control Events

- BEGIN_CALL
- CALL_CLEARED
- CALL_CONFERENCED
- CALL_CONNECTION_CLEARED
- CALL_DATA_UPDATE
- CALL_DELIVERED
- CALL_DIVERTED
- CALL_ESTABLISHED
- CALL_FAILED
- CALL_HELD
- CALL_ORIGINATED
- CALL_RETRIEVED
- CALL_TRANSMFERRED
- END_CALL
Receiving Events

- Communication must conform to XMPP standard
  http://xmpp.org/

- Establish a connection with Notification Server with proper JID

- Jabber Identifiers (JIDs) uniquely identify individual entities in the Jabber network
  Syntax: agentId@domain/resourceId
  Example: johndoe@cisco.com/home
  resourceId is used to identify a client endpoint
  agentId and password should match configurations on UCCE
  Should use PLAIN authentication only

- Port 5222 used for XMPP
  Used by thick clients who CAN maintain long-lived TCP connection

- Port 7071 for XMPP over BOSH
  Used by thin clients who CANNOT maintain a long-lived TCP connection
  BOSH HTTP URL is http://[host]:7071/http-bind

- Enable/disable events by sending subscribe/unsubscribe REST request with the resourceId matching the one provided in the JID
Receiving Events Example

1) Sign In Agent
   POST http://[host]:80/webservices/ConnectionService/Connection/signIn
   Authorization: Basic am9obmRvZTpjaXNjbw==

2) Establish XMPP connection to Notification Server with JID
   johndoe@cisco.com/resource123

3) Enable events for Calls
   POST http://[host]:80/AgentService/Agent/johndoe/call/subscribe?resource=resource123
   Authorization: Basic am9obmRvZTpjaXNjbw==

4) Trigger call action and see events published
   {"eventType":"Call", [data]...}
Event Payload and Types

- Event payload

  ```json
  {"eventType" : [Type], [data]...}
  ```

  **Example:**

  ```json
  {"eventType" : "AgentState", [data]...}
  ```

- Event Types

  - AgentState
  - AgentConfig
  - SupervisorConfig
  - TeamConfig
  - QueueConfig
  - AgentStatistics
  - QueueStatistics
  - Call
Cisco Finesse Sample

- Web page
  Example HTML and JavaScript
  An example of one way to build an application using Finesse services

- Showcases some API features
  Sign in/out
  Data inquiries (Agent States)
  Subscriptions (Agent States, Call Events)
  Call control (Make, Answer, Hold, Retrieve, Drop)

- Connects to the Notification Service (BOSH) to receive events

- Prints all responses and events to console
Dependencies

- jQuery (simplifies Ajax request and document traversing)
  http://jquery.com

- Cisco Ajax XMPP Library CAXL – JabberWerx (receive events)
  http://developer.cisco.com/web/xmpp/resources

- Proxy (solution to same origin policy)
Browser loads webapp with JavaScript

Same origin policy prevents direct JavaScript request

Proxy is used to redirect JavaScript request

Browser

HTTP Proxy

Web App (Sample)

Web Server

Finesse Server

Notification Server

BOSH

Same Origin Policy
Sample Files

- **index.html**
  - Import JavaScript dependencies.
  - Contains the UI structure with buttons and fields.

- **finesse.js**
  - Class with functions to construct and send Finesse API requests.
  - Uses jQuery ($) to send the Ajax requests.

- **sample.js**
  - Control UI elements.
  - Instantiate Finesse class (finesse.js) and bind functions to buttons.
  - Creates event BOSH connection using CAXL (JabberWerx).
Development Tool Suggestions

- Poster plugin for Firefox
  Sends HTTP requests to backend

- XMPP Console plugin for Pidgin
  Send XMPP & receive XMPP messages
  [http://www.pidgin.im/](http://www.pidgin.im/)

- Firebug plugin for Firefox
  JavaScript debugger
  Analyze HTTP request/response

- Fiddler (HTTP traffic analyzer)
  Analyze HTTP request/response in IE
Developer & Product Resources

Cisco Developer Network

- Developer Q&A
- Developer Guide
- Installation and Administration Guide
- Forums and Coding Tips
- Sample Code

http://developer.cisco.com/web/finesse/home

Cisco Community Central

- Marketing Materials and General Product Questions

https://www.myciscocommunity.com/community/partner/collaboration/contactcenter
Cisco SocialMiner
SocialMiner
Social Media Customer Care

1. Capture
2. Analyze & Prioritize
3. Communication Workflow
4. Assign & Engage

Airline lost my luggage!

Social Media Appliance
Social Media Customer Care Agent

Cisco Unified Contact Center
Social Media Customer Care Agent
SocialMiner
Social Media Customer Care

Overview

- Enable proactive customer service by queuing and assigning customer posts to appropriate staff
- Complement brand monitoring dashboards

Features

- Real-time capture of social media postings
- Social media campaign management
- Route and queue contacts to experts
- Tagging
- Social Screen Pop
- Social media customer care metrics
- Optional Integration with full suite of Cisco Enterprise social software systems
SocialMiner Architecture

References:
• OpenSocial Gadgets
• REST API’s
SocialMiner Campaign Management

- **Features**
  - Configure feeds from various sources of user generated contact web sites
  - Group feeds into “campaigns” to monitor social contacts
  - Trainable filters to sort social contacts

- **Benefits**
  - Consolidated list of prioritized opportunities for engagement across all social media sites
  - Distinct campaigns for different products/purposes
    - Product A, Product B, Product C
    - Customer Service, Sales
    - Customers, Partners, Analysts
Route and Queue Contacts to Experts

**Features**

- Easy-to-use web gadget to reserve and dispatch work
- Pick style work assignment
- State management for social contacts

*Push (ACD-style) assignment with Cisco Unified Contact Center Enterprise universal queue*

**Benefits**

- Scale social media team activities
- Automated distribution of work improves efficiency and effectiveness of social media engagement
- Flexible assignment models and approval queues expands participation

*Future release*
Application Programming Interfaces

- **Features**
  - OpenSocial compliant gadgets
  - REST API’s for nearly all appliance configuration and features

- **Benefits**
  - Flexible user interface options
  - Extensive opportunities for customization
SocialMiner Customer Care Metrics

Feature
- Detailed metrics on social media customer care activities
- Campaign reports
- Team reports

Benefit
- Measure work and results
- Manage to service level goals
- Support brand management
- Optimize staffing
Integration with Full Suite of Cisco Collaboration Tools

- **Features**
  - Integration with Quad, Show and Share, Pulse, and Cisco unified communications solutions
  - Lower cost of ownership with out-of-the-box integration between collaboration technologies
  - Consistent user and administration experience

- **Benefits**
  - Access to efficient and effective internal collaboration tools help agents serve customers better
  - Easy to maintain with existing IT personnel
Cisco MediaSense
What is Cisco’s media capture platform?

Cisco’s media capture platform provides open standards, network-based recording of media, including audio and video, with rich metadata to facilitate use by business and analytics applications.

Cisco’s media capture platform provides an efficient, cost-effective foundation for capturing, preserving, and mining business intelligence from conversations.
Multimedia capture, storage, processing

- **Web 2.0 APIs**
  - Application, User, & Configuration Management
  - Redundant Metadata database
  - Media Management

- **Web 2.0 APIs**
  - Application, User, & Configuration Management
  - Redundant Metadata database
  - Media Management

- **Add media servers for scalability and high-availability**

- **Fibre Channel SAN storage**
  - SAN

- **Services**
  - SIP Call Control
  - Media Capture & Streaming

- **Network**
  - Call Control
  - Gateway, SBC, or Phone

- **Applications**
  - Cisco & 3rd Party Apps
  - Apps
Product Features: Recording

- Requires UCM 8.5 - SIP trunk enhancements provide additional recording session metadata; TAPI/JTAPI not required.
- Requires endpoint supporting Built-in Bridge (“BIB”)
- Codecs: G.711 and G.729
dual media streams recorded
- Phones are enabled for recording by configuring them with an appropriate Recording Profile in Unified Communications Manager.
- Recordings can also be initiated by directly dialing a number which is associated with the media capture platform. This does not require a phone with a B-I-B
Product Features: Monitoring and Playback

- While a recording is in progress, the platform allows it to be *monitored* through a streaming media player.
- Each track is accessed via its own URI; real time mixing is NOT currently supported.
- If live monitoring of both tracks is required, UCM's Silent Monitoring capability should be used.
- Playback is also on a per-stream basis unless the streams are mixed by conversion to mp4.
Product Features: Metadata

- The media capture platform maintains a database containing a great deal of information about recorded sessions. The database is stored redundantly on the Primary and Secondary servers. The data includes:
  - Various track, participant, call and session identifiers
  - Time stamps and durations
  - Real time session state
  - URIs for streaming and downloading in various formats
  - Server address where recorded files are stored

- The media capture platform allows database users to attach tags to recordings
Product Features: Events

- Event notifications
  - when recordings start and stop
  - when disk space usage exceeds thresholds.

Clients may use these events to keep track of system activities and to trigger their own actions.
Product Features: APIs

- The media capture platform API offers a number of methods to search and retrieve information in the metadata database.
  - **Authentication**
  - **Control Recordings**
    - start, stop, pause, resume etc
  - **Manage Recordings**
    - add a tag, convert, delete, save, automatic pruning etc
  - **Search for Recordings**
    - search for recordings in a flexible way by combining various criterion.
  - **Subscribe for Recording related events**
    - get notified asynchronously for events such as recording started, stopped etc.
    - events are for “server-based clients”
    - Symmetric Web Services (SWS) approach is used for communicating the events.
Benefits:
- Re-use existing network elements
- Record anywhere, under central control
Key Feature: Phone-based Audio Recording

- Dual-stream recording of audio conversations
- Requires UCM 8.5 and IP Phones with Built-in-Bridge (BiB)
- Can support live/silent monitoring
- No additional charge for media forking support in UCM or phones
- Additional applications are required to:
  - invoke/control recording
  - search/playback recordings
  - invoke/control live monitoring
  - provide analytics
Phone-based Media Forking

Support for Media Forking (media packet replication) in specific IP endpoints
UCM SIP recording trunk enhanced with additional call metadata
Recorder captures dual media streams (each side of conversation separately)

Supported Phones

<table>
<thead>
<tr>
<th>Phone Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>7906</td>
</tr>
<tr>
<td>7911</td>
</tr>
<tr>
<td>7921</td>
</tr>
<tr>
<td>7925</td>
</tr>
<tr>
<td>7941</td>
</tr>
<tr>
<td>7942</td>
</tr>
<tr>
<td>7945</td>
</tr>
<tr>
<td>7951</td>
</tr>
<tr>
<td>7961</td>
</tr>
<tr>
<td>7962</td>
</tr>
<tr>
<td>7965</td>
</tr>
<tr>
<td>7970</td>
</tr>
<tr>
<td>7971</td>
</tr>
<tr>
<td>7975</td>
</tr>
</tbody>
</table>

IP Communicator

First available in UCM 6.0
Updated in UC 8.5
FCS: Nov 2010
(SIP trunk metadata enhancements)
Phone-based Media Forking

First available in UCM 6.0
Updated in UC 8.5
FCS: Nov 2010
(SIP trunk metadata enhancements)

Only one IP endpoint required to support media forking
Sizing

- 150 concurrent sessions (bi-directional streams) per server
- Storage: Using G.711 codecs, requirement is 60MB per hour of recording per stream
Clustered Deployment

- 1 to 3 nodes (initial release)
- 1 or 2 servers (replicated) support database & APIs
- All nodes can capture, stream & manage media

Three types of servers
- Primary: Supports all database operations as well as media operations.
- Secondary: Provides high-availability for the database. Also supports all database operations as well as media operations.
- Expansion: Provides additional capacity for media operations, but not for database operations.
Open Platform drives Application Marketplace

Partner Applications

- Compliance
- Quality Monitoring
- Workforce Management
- Speech/Behavioral Analytics

Open Web 2.0 Application APIs

Multimedia Services
- Audio Capture
- Video Capture
- Streaming
- Metadata / database
- Media Storage / archiving
- Media export / transcoding
- Content management

Network infrastructure
- Network-based Media Replication
- Dynamic Resource Control, Discovery and Access
- Seamless media control for complex scenarios and network topologies
- Metadata transmission
- Medianet integration

SAN

DMS

Show and Share

Session Management

Compatibility

Open Web 2.0 APIs drive Application Innovation and Global Partner Ecosystem
Closing
BRKCDN-1002

Recommended Reading

Please browse on-site Cisco Store for suitable reading.
Please complete your Session Survey

- We value your feedback - don't forget to complete your online session evaluations after each session. Complete 4 session evaluations & the Overall Conference Evaluation (available from Thursday) to receive your Cisco Networkers 20th Anniversary t-shirt.

- All surveys can be found on our onsite portal and mobile website: www.ciscovliveeurope.com/connect/mobi/login.ww

- You can also access our mobile site and complete your evaluation from your mobile phone:
  1. Scan the Access Code
     (See http://tinyurl.com/qrmelist for software, alternatively type in the access URL)
  2. Login
  3. Complete and Submit the evaluation