Web Services and Applications Deployment Guide

Web Services and Applications 8.5.0

1/6/2022

Table of Contents

Workspace Web Edition & Web Services Deployment Guide

RESTRICTED

Note: This is **restricted release** documentation, and therefore is subject to change and is not complete. Some features that are described in this documentation might not be fully implemented in the application.

This guide provides information about deploying and configuring Workspace Web Edition & Web Services.

Other Resources

- Workspace Web Edition & Web Services 8.5.x Landing Page
- Workspace Web Edition & Web Services 8.5.x Release Note
- Workspace Web Edition Help
- Web Services API Reference

For information about configuring the Workspace Web Edition (agent desktop) interface, refer to Workspace Web Edition Configuration Guide.

Installing and Deploying Workspace Web Edition & Web Services

Premise Implementation

Deploying the Web Application

Start of Procedure

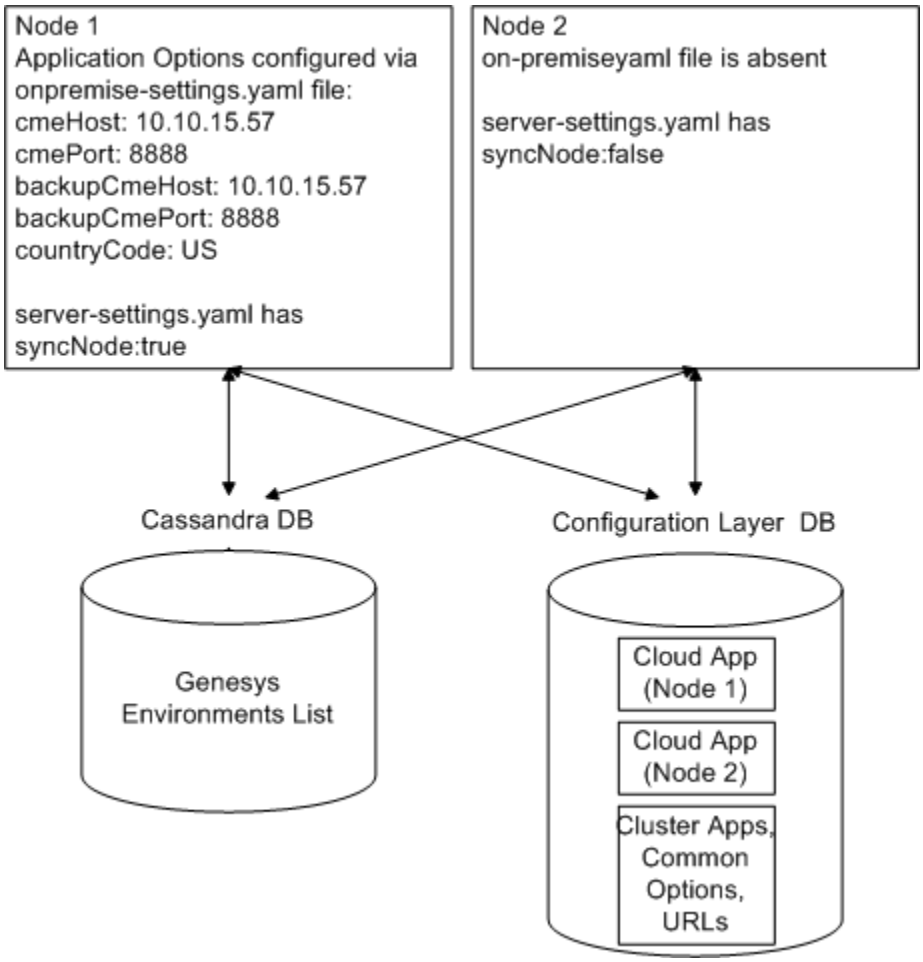
1. Review the Prerequisites, including installation of Cassandra. |See the page.
2. Install Workspace Web Edition & Web Services.
3. Configure Workspace Web Edition & Web Services.
4. Start Web Services and test the configuration.
5. (Optional) Change the default Contact Center feature configuration.
6. Test the User Interface with Workspace Web Edition & Web Services.

End of Procedure

Architecture

On-Premise Architecture

This configuration assumes that you have one contact center for each Workspace Web Edition & Web Services cluster. The following example depicts a two-node configuration, one node for the Synchronization Node (SyncNode) and one Client node. You can have any number of Client nodes. Each node must have a connection to the Cassandra database and the Configuration Layer database.



Workspace Web Edition & Web Services On-Premise Architecture

Prerequisites

The following prerequisites must be met before you can install and configure Workspace Web Edition & Web Services:

- The Genesys Platform has been installed, configured, and tested.
- Jetty version 8 has been installed and configured.
- Cassandra has been installed and configured.

Cassandra Installation and Configuration

Cassandra is mandatory for Workspace Web Edition & Web Services and must be installed and configured prior to starting the installation and configuration of Workspace Web Edition & Web Services. The following steps are intended to serve as a quick guide to installing and configuring Cassandra support for Workspace Web Edition & Web Services. For more details, or for advanced scenarios, refer to the Cassandra documentation.

Installing Cassandra

Start of Procedure

1. Download the latest 1.1.x version of Cassandra.
2. Copy the Cassandra archive to the installation directory. For example, `/usr/local`
3. Use a tar utility to extract the files. For example, `tar -zxvf apache-cassandra-1.1.11-bin.tar.gz`
4. Add directories for data, commitlog, saved_caches, and logs.
5. Make Cassandra a service.

End of Procedure

Next Steps

- Configure Cassandra

Configuring Cassandra

 **Purpose:** Modify the default configuration.

Start of Procedure

1. Modify the `cassandra.yaml` file:
    a. Set `cluster_name`. It must be the same name on all nodes.

b. Set `initial_token` according to the place in ring. It must be one of the following:

```
Node #1:                                        0
Node #2:   56713727820156410577229101238628035242
Node #3:  113427455640312821154458202477256070484
```

c. Set seeds to the list of host names of all nodes. For example: `-seeds: “node1, node2, node3”`

d. Set `listen_address` and `rpc_address` to the host name.

e. Set `data_file_directories`, `commitlog_directory`, and `saved_caches_directory` according to the created directories.

f. Change `endpoint_snitch` to `PropertyFileSnitch`.

7. Modify the `log4j-server.properties` file by setting the `log4j.appender.R.File` according to the created folder.
8. Modify the `cassandra-topology.properties` file by updating the cluster topology to match the following:

```
node1=AZ1:RAC1
node2=AZ1:RAC1
node3=AZ1:RAC1
```

End of Procedure

Next Steps

- Verify the installation

Verifying the Cassandra Installation

Start of Procedure

1. Start all Cassandra nodes.
2. Use the nodetool utility to verify that all nodes have connected by entering the following command:

```
[cassandra install dir]/bin/nodetool -h [cassandra host] ring
```

End of Procedure

Client-side Browser Support

Workspace Web Edition agent interface is accessed by a web browser. The following browsers are supported:

- Microsoft Internet Explorer 10
- Google Chrome

Install Workspace Web Edition & Web Services

Perform the following procedures to configure and install the Workspace Web Edition & Web Services application.

Configuring the Application

Use *either* Configuration Manager or Genesys Administrator for this procedure:

Configuring the Application in Configuration Manager

Configuring the Application in Configuration Manager

Workspace Web Edition & Web Services uses two application objects in the Genesys configuration environment. A cluster application of type `Genesys Generic Server`, and a second application of type `Genesys Generic Client`. Together, these two applications provide Workspace Web Edition & Web Services with access to the configuration required for operation.

> **Tip**
> Users, DNs, Places, and Skills are loaded from the Configuration Layer.

The following sections provide details on creating and configuring the Workspace Web Edition & Web Services application pair.

Importing the Application Templates

Start of Procedure

Purpose: Import the templates for the Workspace Web Edition & Web Services Applications. If the Cloud and CloudCluster application templates already exist, skip these steps.

1. To create the Cloud template, right-click in the main content area in Configuration Manager and select `New > Application Template`.
2. Configure the General tab of the template as shown below:

- Name: Cloud
- Type: Genesys Generic Client
- Version: 8.5
- State Enabled: Yes

3. Click OK.
4. To import the Cluster template, right-click in the main content area in Configuration Manager and select `Import > Application Template`.
5. Naviate to the `//templates` folder on the installation CD.
6. Select the `Workspace_Web_Edition_Web_Services_850` template file.
7. Click OK.

End of Procedure
Next Steps

- Create the Cluster Application
- Create the Cloud Application

Creating the Cluster Application

Start of Procedure

1. Navigate to the `Applications` folder in Configuration Manager. Right-click and select `New > Application`.
2. Select the `Workspace_Web_Edition_Web_Services_850` template and click OK.
3. Configure the General tab as shown below:
   - Name: Workspace_Web_Edition_Web_Services_850
   - Template: Genesys Generic Server
   - Component Type: [Unknown]
   - State Enabled: Yes
4. On the Tenants tab do the following action:
   1. Click Add.
   2. Chose the "Environment" tenant.
   3. Click OK.
5. On the Server Info tab, choose the appropriate Host object. This automatically adds a corresponding port entry. The port value is ignored by the server and does not need to be modified.
6. On the Start Info tab, add a “.” to the `Working Directory`, `Command Line`, and `Command Line Arguments` fields. These values are mandatory for all applications and must be entered to save the application object. Workspace Web Edition & Web Services does not use these values, so the “.” is used as a placeholder.

7. On the Connections tab, add the following connections:
   - T-Server
   - Stat Server
   - Configuration Server

> **Important**
> Additional connections to Interaction Server and UCS may be configured if you are using eServices in your deployment. However, if you add a connection to Interaction Server or to UCS, you must add a connection to the other. You must connect both or connect neither.

8. Click OK to save the Cluster application.

End of Procedure

Next Steps

- Create the Cloud Application

Creating the Cloud Application

> **Important**
> You must create one Cloud application for EACH node. Each Cloud application must be configured identically.

Start of Procedure

1. Navigate to the `Applications` folder in Configuration Manager. Right-click and select `New > Application`.
2. Select the "Cloud" template and click OK.
3. Configure the General tab as shown below:
   - Name: Cloud
   - Template: Cloud
   - State Enabled: Yes
4. On the Connections tab, add the following connections:

- Cluster application that was configured in the previous procedure.

5. Click OK to save the Cloud application.

Configuring the Application in Genesys Administrator

Configuring the Application in Genesys Administrator

Workspace Web Edition & Web Services uses two application objects in the Genesys configuration environment. A cluster application of type `Genesys Generic Server`, and a second application of type `Genesys Generic Client`. Together, these two applications provide Workspace Web Edition & Web Services with access to the configuration required for operation.

> **Tip**
>
> Users, DNs, Places, and Skills are loaded from the Configuration Layer.

The following sections provide details on creating and configuring the Workspace Web Edition & Web Services application pair.

Creating the Application Templates

Start of Procedure

Purpose: Create the templates for the Workspace Web Edition & Web Services Applications. If the Cloud and CloudCluster application templates already exist, skip these steps.

1. Launch Genesys Administrator (refer to the Genesys Framework 8.1 documentation).
2. In Genesys Administrator, choose the Provisioning tab.
3. To create the Cloud template, select `New > Application Template`.
4. Configure the properties of the template as shown below:
   - Name: Cloud
   - Type: Genesys Generic Client
   - Version: 8.5
   - State: Enabled
5. Click `Save and Close`.
6. Navigate to the `Applications` object.

7. Import the CloudCluster template from the installation CD.
8. Select the `Workspace_Web_Edition_Web_Services_850` template file.
9. Click `OK`.
10. Click `Save and Close`.

End of Procedure
Next Steps

- Create the Cluster Application
- Create the Cloud Application

Creating the Cluster Application

Start of Procedure

1. Launch Genesys Administrator (refer to the Genesys Framework 8.1 documentation)
2. In Genesys Administrator, choose the Provisioning tab.
3. Navigate to the `Applications` object.
4. Select `New > Application`.
5. Select the `Workspace_Web_Edition_Web_Services_850` template and click `OK`.
6. Configure the object properties as shown below:
   - Name: Workspace_Web_Edition_Web_Services_850
   - Template: Genesys Generic Server
   - State: Enabled
7. On the Tenants tab do the following action:
   1. Click `Add`.
   2. Chose the "Environment" tenant.
   3. Click `OK`.
8. On the Server Info tab, choose the appropriate Host object. This automatically adds a corresponding port entry. The port value is ignored by the server and does not need to be modified.
9. On the Start Info tab, add a “.” to the `Working Directory`, `Command Line`, and `Command Line Arguments` fields. These values are mandatory for all applications and must be entered to save the application object. Workspace Web Edition & Web Services does not use these values, so the “.” is used as a placeholder.
10. On the Connections tab, add the following connections:
    - T-Server
    - Stat Server
    - Configuration Server

> **Important**
> Additional connections to Interaction Server and UCS may be configured if you are using eServices in your deployment. However, if you add a connection to Interaction Server or to UCS, you must add a connection to the other. You must connect both or connect neither.

11. Click OK to save the Cluster application.

End of Procedure

Next Steps

- Create the Cloud Application

Creating the Cloud Application

> **Important**
> You must create one Cloud application for EACH node. Each Cloud application must be configured identically.

Start of Procedure

1. Navigate to the `Applications` folder in Configuration Manager. Right-click and select `New > Application`.
2. Select the "Cloud" template and click OK.
3. Configure the General tab as shown below:
   - Name: Cloud
   - Template: Cloud
   - State Enabled: Yes
4. On the Connections tab, add the following connections:
   - Cluster application that was configured in the previous procedure.
5. Click OK to save the Cloud application.

Next Steps

- Initialize the Cassandra Schema

Initializing the Cassandra Schema

Start of Procedure

1. Copy the `cf-schema.txt` and `ks-schema-prod_HA.txt` (or `ks-schema-local.txt` for a single node Cassandra configuration) from `[install_dir]/data` to the cassandra node host.
2. Make the following modifications to the file:
   a. Align the datacenter name with that of the Cassandra cluster. You can use nodetool to find the name of the datacenter by examining the output of "nodetool ring" (the tool is located in the `bin` directory of Cassandra) to find the name of the datacenter). The following is sample output from the nodetool:

```
ubuntu@hpe-voicevm-84:/genesys/apache-cassandra-1.1.6/bin$ ./nodetool ring
Note: Ownership information does not include topology, please specify a keyspace.
Address         DC          Rack        Status State   Load
Owns                Token
10.10.15.84     datacenter1 rack1       Up     Normal  14.97 MB
100.00%             0
ubuntu@hpe-voicevm-84:/genesys/apache-cassandra-1.1.6/bin$
```

   b. Edit the contents to adjust the number of datacenter entries.
   c. Edit the contents to adjust the replication factors. (Refer to external documentation on the DataStax website that provide more details for information about how to choose the correct values).
3. Clean the data, commitlog, and saved_caches folders prior to starting the Cassandra cluster.
4. Run the following commands:

```
[cassandra install dir]/bin/cassandra-cli  -h [cassandra host] --file ks-schema-
prod_HA.txt
[cassandra install dir]/bin/cassandra-cli  -h [cassandra host] --file cf-schema.txt
```

End of Procedure
Next Steps

- Deploy the Web Application

Deploying the Web Application

Start of Procedure

1. Stop Jetty.
2. Copy the `jetty.xml` file from `[build_path]/jetty` to `$JETTY_HOME/etc`.
3. Copy the `cloud-web.xml` file from `[build_path]/jetty` to `$JETTY_HOME/contexts`.
4. Copy the `logback.xml` file from `[build_path]/jetty` to `$JETTY_HOME/resources`.
5. Copy the `cloud-web.war` file from `[build_path]/webapp` to `$JETTY_HOME/webapps`.
6. Copy the configuration files from `[build_path]/conf` to `$JETTY_HOME/genconfig` and edit the fields marked ToBeChanged (refer to Premise Configuration).

End of Procedure

Configuring Workspace Web Edition & Web Services

Configuring Web Services Node

Workspace Web Edition & Web Services reads a series of configuration files from the location that is defined by the `config.path` system property. When Workspace Web Edition & Web Services is started, the `config.path` property **must** be defined or the server will not start.

To start each Web Services node, enter the following command line:

```
/usr/bin/java -Xmx4096m -Xms4096m -Djetty.port=80 -Djetty.logs=/var/log/jetty

-Dconfig.path=/opt/jetty/genconfig -Djetty.home=/opt/jetty -Djava.io.tmpdir=/tmp

-Djava.net.preferIPv4Stack=true -jar /opt/jetty/start.jar --pre=etc/jetty-logging.xml --daemon
```

The following parameters are **mandatory** for the web services node:

- The memory allocation: `-Xmx4096m, -Xms4096m`
- The configuration path: `-Dconfig.path=<your path to config>`

Start of Procedure
As described below, modify (or remove) each of the following Workspace Web Edition & Web Services configuration files:

- `server-settings.yaml`
- `onpremise-settings.yaml`
- `cassandra-cluster.yaml`

The following information is specific to deployment of Workspace Web Edition & Web Services in Premise mode. For a more general description of available Workspace Web Edition & Web Services configuration options, please refer to the options reference.

1. In the `server-settings.yaml`, review the options and modify or retain as appropriate for the deployment. This file contains a number of core parameters that are used by the server.

   ### The following is an unmodified file:

   ```
   :externalApiUrl: [ToBeChanged: "PUBLIC_SCHEMA_HOST_PORT"]/api/v1
   internalApiUrl: [ToBeChanged: "INTERNAL_SCHEMA_HOST_PORT"]/api/v1
   externalApiUrlV2: [ToBeChanged: "PUBLIC_SCHEMA_HOST_PORT"]/api/v2
   internalApiUrlV2: [ToBeChanged: "INTERNAL_SCHEMA_HOST_PORT"]/api/v2
   reconnectAttempts: [ToBeChanged: "RECONNECT_ATTEMPTS"]
   reconnectTimeout: [ToBeChanged: "RECONNECT_TIMEOUT"]
   activationTimeout: [ToBeChanged: "ACTIVATION_TIMEOUT"]
   ```

```
contactCenterSynchronizationTimeout: 60000
opsUserName: [ToBeChanged: "OPS_USER_NAME"]
opsUserPassword: [ToBeChanged: "OPS_USER_PASSWORD"]
applicationName: [ToBeChanged: "APPLICATION_NAME"]
applicationType: [ToBeChanged: "APPLICATION_TYPE"]
cmeUserName: [ToBeChanged: "CME_USER_NAME"]
cmePassword: [ToBeChanged: "CME_PASSWORD"]
syncNode: [ToBeChanged: true|false]
cmeAuthenticationEnabled: [ToBeChanged: true|false]
nodeId: [ToBeChanged: unique id of the node]
```

The following is a sample of a modified file:

```
externalApiUrl: http://hpe-voicevm-84.genesyslab.com/api/v1
internalApiUrl: http://hpe-voicevm-84.genesyslab.com/api/v1
externalApiUrlV2: http://hpe-voicevm-84.genesyslab.com/api/v2
internalApiUrlV2: http://hpe-voicevm-84.genesyslab.com/api/v2
reconnectAttempts: 1
reconnectTimeout: 10000
activationTimeout: 120000
contactCenterSynchronizationTimeout: 60000
opsUserName: ops
opsUserPassword: ops
applicationName: Cloud
applicationType: CFGGenericClient
cmeUserName: default
cmePassword: password
syncNode: true
nodeId: hpe-voicevm-84
```

2. In the `onpremise-settings.yaml`, review the options and modify or retain as appropriate for the deployment. This file contains a number of core parameters that are used by the server.

```
cmeHost – primary CME host
cmePort – primary CME port
backupCmeHost – backup CME host
backupCmePort – backup CME port
countryCode – Premise contact center's country code
```

`cmeHost` specifies the host or IP of the Configuration Server and `cmePort` specifies the port of Configuration Server.

If there is a backup Configuration Server in the Genesys environment and you want HA support, you can configure two additional parameters in the `onpremise-settings.yaml` file: `backupCmeHost` and `backupCmePort`. These are the host (IP) and port of the backup Configuration Server.

3. Review `cassandra-cluster.yaml`. The options settings are as follows:

```
thrift_port: 9160
jmx_port: 7199
keyspace: sipfs
nodes: hpe-voicevm-84.genesyslab.com
should_sync_schema: TRUE
write_consistency_level: CL_ONE
read_consistency_level: CL_ONE
max_conns_per_host: 16
max_cons:  16
```

```
max_pending_conns_per_host: 80
max_blocked_threads_per_host: 160
retry_sleep_time_ms: 5000
retry_max_attempt: 1
#replication_factor: 1
```

4. For a single-node Cassandra cluster, confirm that `write_consistency_level` and `read_consistency_level` are `CL_ONE`, and `replication_factor` are `1`, as shown above.
5. For a multi-node Cassandra cluster, set `write_consistency_level` and `read_consistency_level` to `CL_QUORUM` and `replication_factor` to `3`.
6. (Optional) Specify `retry_sleep_time_ms` and `retry_max_attempt`, which define the Web Services retry policy as a Cassandra client. If these are not specified, the `retry_max_attempt` is `1` and `retry_sleep_time_ms` is `5000` (which equals 5 seconds).

End of Procedure

Workspace Web Edition & Web Services Sync Node

To facilitate administration by using standard Genesys tools such as Configuration Manager or Genesys Administrator, Workspace Web Edition & Web Services can be configured to read the contents of the Configuration Database, import objects into Workspace Web Edition & Web Services, and subscribe to change notifications from Configuration Server.

In a multi-node Workspace Web Edition & Web Services deployment, only a single node should be configured to be the Synchronization Node. The `syncNode` option in `server-settings.yaml` is used to control this functionality. The Workspace Web Edition & Web Services node that performs synchronization tasks should be configured with the value of syncNode set to `true`, and all the other Workspace Web Edition & Web Services nodes configured with the value of `syncNode` set to `false`.

Reporting Configuration

The following procedure outlines the steps that are required to enable Workspace Web Edition & Web Services to use, store, and expose statistical data. Workspace Web Edition & Web Services uses several statistics internally to support contact availability and also provides a default set of statistics used by the bundled client applications.

The following steps include the configuration of Workspace Web Edition & Web Services node, Statistics Time-to-Live, Stat Server, and Contact Center Stat Server host and port.

Start of Procedure

1. Open the `server-settings.yaml` file.
2. Configure the Workspace Web Edition & Web Services node by setting the `nodeId` property. The value of this option must be a unique identifier for the Workspace Web Edition & Web Services node. Each Workspace Web Edition & Web Services node in a cluster must define a unique `nodeId`. All Workspace Web Edition & Web Services nodes that share the same Cassandra storage will read the contact centers that require statistic and divide the monitoring tasks among the Workspace Web Edition & Web Services nodes.
3. In the same file, configure the Statistics Time-to-Live by doing the following:

1. Define the "time-to-live" for statistic storage using the `statisticsTTL` property. If you do not define this optional parameter, the default value of `86400` (24 hours) is used.
2. Confirm that the `statistics.yaml` file is present in the main Workspace Web Edition & Web Services `config` folder (defined by the `config.path` variable, `/opt/jetty/genconfig`, by default). Modifications to this file are not typically necessary.
   The `statistics.yaml` file defines which statistics, and for what object types, Workspace Web Edition & Web Services will request from Stat Server. A default `statistics.yaml` file is included with the Workspace Web Edition & Web Services distribution and contains all the statistics Workspace Web Edition & Web Services requries internally as well as those required by the bundled UI applications.

4. Configure any Stat Server applications in the Genesys environment(s) to which the Workspace Web Edition & Web Services node/cluster will connect to include a set of statistic definitions that match those included in the `statistics.yaml` file. You must do this for both the primary and backup Stat Servers in each Genesys environment that will be accessed by Workspace Web Edition & Web Services. A sample Stat Server configuration can be found on the installation CD. This file can be copied locally and imported using Configuration Manager or Genesys Administrator for both the primary and backup Stat Servers.

End of Procedure

Starting and Testing Workspace Web Edition & Web Services

Purpose: to test and verify the Workspace Web Edition & Web Services configuration.

Start of Procedure

1. Start Jetty.
2. To verify that Workspace Web Edition & Web Services Server is up and running, type the following URL into a web browser:
   `http://[htcc_host]:[htcc_port]/api/v2/diagnostics/version`
   If the request is successful, the version will be printed as shown below:

```
{
    statusCode: 0
    version: "gws-x.x.xxx.xx"
}
```

End of Procedure

Feature Configuration

Each contact center that is created in Workspace Web Edition & Web Services is assigned a set of features. A feature represents a set of functionality that can be enabled or disabled for each contact center. Workspace Web Edition & Web Services provides APIs that enable clients to read the list of features that are assigned to a contact center and then show or hide UI elements according to what is specified. New features can also be assigned to or removed from contact centers after creation.

Supported Features

The following features are currently supported. If a feature is marked "assigned by default" all contact centers have this feature assigned. **Note:** Default features cannot be unassigned at this time.

Name	Description	Assigned by default
api-provisioning-read	General provisioning read	Y
api-provisioning-write	General provisioning write	Y
api-voice	API for the voice channel	Y
ui-supervisor-provisioning-routing	UI to provision routing	Y
ui-supervisor-provisioning-skill	UI to provision skills	Y
ui-supervisor-provisioning-user	UI to provision users	Y
ui-supervisor-recording	UI to view call recordings	Y
ui-supervisor-reporting	UI to view contact center statistics	Y
api-multimedia	API for non-voice channels	N
api-supervisor-recording	API For Call Recording Supervisor	Y
api-supervisor-monitoring	API For Supervisors to Monitor Agents	N
api-supervisor-agent-control	API For Supervisors to Control Agent States	N

Feature Assignment for New Contact Centers

When a new contact center is created, it is automatically assigned a default set of features. If the Workspace Web Edition & Web Services configuration is unmodified, the contact center features are assigned according to the table in the previous section.

> **Important**
>
> Features that are assigned to contact centers by default **cannot** be removed using the API. To change the default set of features for contact centers, a custom `feature-definitions.json` file must be included as described in the next section.

Changing the Default Feature Assignments

To change the default set of features, you must create a `feature-definitions.json` file and add it to the `config.path` folder. When a custom configuration is included, it changes the default features for all contact centers even after they are created. It is important that all Workspace Web Edition & Web Services nodes in a cluster share the same feature configuration.

The following sample `feature-definitions.json` removes the two features that are related to call recording from the default features.

```
[

{
     "id":"api-provisioning-read",
     "displayName":"API Provisioning Read",
     "description":"General provisioning read",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"api-provisioning-write",
     "displayName":"API Provisioning Write",
     "description":"General provisioning write",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"api-voice",
     "displayName":"Voice API",
     "description":"API for Voice",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"ui-supervisor-provisioning-routing",
     "displayName":"UI Supervisor Provisioning For Routing",
     "description":"Routing Provisioning UI",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"ui-supervisor-provisioning-skill",
     "displayName":"UI Supervisor Skill Provisioning",
     "description":"UI Supervisor Skill Provisioning",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"ui-supervisor-provisioning-user",
```

```
     "displayName":"UI Supervisor Provisioning User",
     "description":"UI Supervisor Provisioning User",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"ui-supervisor-recording",
     "displayName":"UI Supervisor Recording",
     "description":"UI For Call Recording Supervisor",
     "autoAssignOnContactCenterCreate": false
},

{
     "id":"api-supervisor-recording",
     "displayName":"API Supervisor Recording",
     "description":"API For Call Recording Supervisor",
     "autoAssignOnContactCenterCreate": false
},

{
     "id":"ui-supervisor-reporting",
     "displayName":"UI Supervisor Reporting",
     "description":"UI For Reporting Supervisor",
     "autoAssignOnContactCenterCreate": true
},

{
     "id":"api-multimedia",
     "displayName":"Multimedia API",
     "description":"API for Multimedia",
     "autoAssignOnContactCenterCreate": false
}
]
```

After making this change, call recording functionality is unavailable.

API Samples

Listing All Supported Features

Request:

```
GET http://localhost:8080/api/v2/ops/features
```

Response:

```
{ "statusCode" : 0,
  "uris" : [ "http://localhost:8080/api/v2/system/features/api-voice",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-provisioning-user",
      "http://localhost:8080/api/v2/system/features/api-provisioning-write",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-provisioning-routing",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-reporting",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-recording",
      "http://localhost:8080/api/v2/system/features/api-supervisor-recording",
      "http://localhost:8080/api/v2/system/features/api-provisioning-read",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-provisioning-skill",
```

```
      "http://localhost:8080/api/v2/system/features/api-multimedia"
    ]
}
```

Review Features Assigned to a Contact Center

Request:

```
GET http://localhost:8080/api/v2/ops/contact-centers
```

Response:

```
{ "statusCode" : 0,
  "uris" : [ "http://localhost:8080/api/v2/ops/contact-centers/
95c81988-b435-48b2-8046-ff7b7e11e524",
      "http://localhost:8080/api/v2/ops/contact-centers/8fd8e342-3e25-4224-a485-bf4533e0e2bf"
    ]
}
```

Request:

```
GET http://localhost:8080/api/v2/ops/contact-centers/95c81988-b435-48b2-8046-ff7b7e11e524/
features
```

Response:

```
{ "statusCode" : 0,
  "uris" : [ "http://localhost:8080/api/v2/system/features/api-voice",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-provisioning-user",
      "http://localhost:8080/api/v2/system/features/api-provisioning-write",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-provisioning-routing",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-reporting",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-recording",
      "http://localhost:8080/api/v2/system/features/api-supervisor-recording",
      "http://localhost:8080/api/v2/system/features/api-provisioning-read",
      "http://localhost:8080/api/v2/system/features/ui-supervisor-provisioning-skill"
    ]
 }
```

Node Start-Up Procedures

When installation is complete, you must start the nodes in the following order:

1. Start the Synchronization Node (Node1)
   When you first start the node, no contact center exists in Cassandra. This node has to create the new contact center in your database and import all the necessary objects. The following steps are performed by the node during node start-up:
   - Reads the `onpremise-settings.yaml` file.
   - Reads the list of Genesys environments from Cassandra and test that the environment for the specified `cmeHost` and `cmePort` does not already exist. If it does exists, the existing Genesys environment object is loaded.
   - Connects to the Configuration Layer (CME) by using the specified settings and read the Cloud application “Node 1” and the Cluster application.
   - From the `server-settings.yaml` file, “Node 1” determines that this host has to synchronize CME with Cassandra.
   - Attempts to read contact center with predefined name from Cassandra DB:
     - If the contact center does not exist, create a new Contact Center and store it in Cassandra.
     - Read Cluster (server) application connections, then configure and store or update the Genesys Environment record.
     - Import all objects from CME to Cassandra.
     - Subscribe to notifications for CME updates and continuously synchronize updated objects.
2. Subsequent start-up of the Syncronization Node
   The second time that the synchronization node is started the contact center exists. The following steps are performed by the node once the contact center exists:
   - Reads the list of Genesys environments from Cassandra and test that the environment for the specified `cmeHost` and `cmePort` does not already exist. If it does exists, the existing Genesys environment object is loaded.
   - Connects to the Configuration Layer (CME) by using the specified settings and load Cloud application “Node 1” and the Cluster application.
   - From the `server-settings.yaml` file, “Node 1” determines that this host has to synchronize CME with Cassandra.
   - Attempts to read the contact center by using the predefined name from the Cassandra database:
     - Since the contact center already exists in Cassandra, the contact center initial synchronization and creation step is skipped and the existing contact center and the existing Genesys Environment are used.
     - Retrieves the update history log from CME and synchronizes with the updates that occurred since the node was shutdown.
     - Subscribes to notifications for CME updates and continuously synchronizes with updated objects.
3. Start the non-synchronization node(s) (Node2)

- Reads the list of Genesys Environments from Cassandra.
- Opens connections to CME, to the Genesys servers, and to the T-Servers on demand, according to Genesys Environment configuration.

Testing the User Interfaces with Workspace Web Edition

Workspace Web Edition is the agent interface of Workspace Web Edition & Web Services. It is deployed as part of the Workspace Web Edition & Web Services deployment. You configure it by using the Workspace Web Edition & Web Services Agent Interface Configuration Guide.

Workspace Web Edition Agent Desktop

1. Once the configuration is complete, launch Workspace Web Edition by navigating to `/ui/ad/v1/index.html`.
2. Enter the credentials for any of the previously created agents. Note that the user must be of the agent type.
3. After clicking Log In, the Workspace Web Edition agent desktop view is displayed and the agent is ready to work.

Web Services Configuration Options Reference

server-settings.yaml

Option	Description	Mandatory	Default
externalApiUrl		Yes	
internalApiUrl		Yes	
connectionTimeout		No	120000
externalApiUrlV2		Yes	
reconnectAttempts		Yes	
reconnectTimeout		Yes	
activationTimeout		Yes	
contactCenterSynchronizationTimeout		Yes	
opsUserName		Yes	
opsUserPassword		Yes	
applicationName		Yes	
applicationType		Yes	
cmeAuthenticationEnabled		No	True
cmeUserName		Yes	
cmePassword		Yes	
syncNode		No	
logoutAgentWhenNoActiveCometSessionTimeout		No	
NodeId		No	
statisticsTTL		No	86400
statConnectionTimeout		No	5000
statReconnectAttempts		No	1
statReconnectTimeout		No	10000
createCallRecordingCF		No	False
crClusterName		No	
crRegion		No	
awsS3AccessKey		No	
awsS3SecretKey		No	
awsS3BucketName		No	
awsS3SocketTimeout	should >0	No	50000

Option	Description	Mandatory	Default
awsS3MaxErrorRetry	should >=0	No	3
awsS3MaxConnection	should >0	No	50
awsS3ConnectionTimeout	should >0	No	50000

Troubleshooting Workspace Web Edition & Web Services

This section describes logs, configurations, and so on that you should check when you are troubleshooting issues with Workspace Web Edition & Web Services.

If you open an Issue Request for Workspace Web Edition & Web Services, you must provide **ALL** of the following information to assist with troubleshooting and investigating defects:

General Info

- A detailed description of steps that you took to reproduce the issue and the results.
- Screenshots that demonstrate the problem or concern (marked up to indicate the location in the screenshot where the issue can be seen)

Workspace Web Edition & Web Services

Workspace Web Edition & Web Services Version

- To get this information, type the following URL into a web browser

```
http://[htcc_host]:[htcc_port]/api/v2/diagnostics/version
```

- The version will be printed on the browser as shown below:

```
{
    statusCode: 0
    version: "gws-x.x.xxx.xx"
}
```

Workspace Web Edition & Web Services Logs:

- To enable DEBUG mode in Workspace Web Edition & Web Services log: within the Workspace Web Edition & Web Services VM go to **/opt/jetty/resources/** and change the log level in **logback.xml** to DEBUG or TRACE (instead of INFO/WARN).

```
<logger name="com.genesyslab" level="DEBUG" />
```

- Attach the Workspace Web Edition & Web Services log file: Found under **/var/log/jetty/cloud.log**.

- (There may also be parts of log for a particular date/time already zipped, attach those as well).
  - **Note:** Change log back to INFO as soon as possible, otherwise the VM's disk space will be filled, which can cause various random errors. Use the `df` command to check disk space.

Workspace Web Edition & Web Services Configuration Snapshot:

- Workspace Web Edition & Web Services configuration files can be found on VM in: **/opt/jetty/genconfig/**
- Zip up all those files and attach them to the issue report.