



Context Service Registration API

- [Context Service Registration API, on page 1](#)

Context Service Registration API

Cisco Context Service is a cloud-based omnichannel solution for Cisco Contact Center Enterprise Solutions. It enables you to capture your customer's interaction history by providing flexible storage of customer-interaction data across any channel.

Various components in the CCE Solution provide out of the box integration with Context Service. Context Service also provides an API for integration with your own applications or third-party applications to capture end-to-end customer-interaction data.

For more information about Context Service and to check service availability, see <https://cisco.com/go/contextservice>.

Use the Context Service Registration API to register SocialMiner with Context Service and get the current registration status.

URL

`https://<server>/unifiedconfig/config/contextserviceregistration`

Operations

- **get:** Returns the current Context Service registration state using the URL
`https://<server>/unifiedconfig/config/contextserviceregistration?callbackUrl=(url)`.
- **update:** Updates Context Service connection data in SocialMiner and the CCE database.

Parameters

- **callbackUrl:** This parameter is optional in the Get URL. The browser is redirected to this URL to complete the registration or deregistration process.
- **state:** The Context Service registration state. Values are as follows:
 - notRegistered
 - registered

- **invalid**: The connection data used to register with Context Service is no longer valid. Reregister to recover from this state.
- **managementUrl**: Navigate to this URL to manage the Context Service. This parameter is returned if the state is registered.
- **registrationUrl**: Navigate to this URL to begin Context Service registration. This parameter is returned if you include the `callbackUrl` in the Get URL and the state is either `notRegistered` or `invalid`.
- **deregistrationUrl**: Navigate to this URL to begin Context Service deregistration. This parameter is returned if you include the `callbackUrl` in the Get URL and the state is registered.
- **refURL**: The refURL. See [Shared Parameters](#).
- **connectionData**: This parameter is required and the maximum size is 4 KB. `connectionData` is an opaque string used to connect to the Context Service. Use an empty string to clear the `connectionData` when deregistering.



Note You must deregister before reregistering to Context Service. If the current `connectionData` is not empty, you can set the value either to an empty string or to the current value.

Sample API Workflow

1. The user navigates to a custom Context Service web page.
2. The page calls the Get API to get the current registration status from Context Service. The page displays either a Register or Deregister button.
3. The user clicks Register or Deregister.
4. The page calls the Get API with a callback URL for a custom web page. This page can be the custom Context Service web page from step 1. The API returns either the Context Service registration or deregistration URL.
5. The browser redirects to the Context Service registration or deregistration URL.
6. The user completes the registration or deregistration wizard.
7. The browser redirects back to the callback URL with a `connectionData` query parameter appended to it.
8. The callback page calls the Update API with the connection data from step 7 to send the data to SocialMiner and the CCE database.
9. Go to step 2.

Example Get Response

```
<contextServiceRegistration>
  <state>registered</state>
  <managementUrl>(url)</managementUrl> <!-- Included if state is registered -->
  <registrationUrl>(url)</registrationUrl> <!-- Included if state is notRegistered
    and callbackUrl is provided -->
  <deregistrationUrl>(url)</deregistrationUrl> <!-- Included if state is registered
```

```
    and callbackUrl is provided -->
    <refURL>/unifiedconfig/config/contextservice</refURL>
  </contextServiceRegistration>
```

Example Update Request

```
<contextServiceRegistration>
  <connectionData>(data)</connectionData>
</contextServiceRegistration>
```

Following are the REST responses received during execution of REST API to register the Context Service with CVP or Finesse:

- **Success** - Configuration changes persist in AW DB and synchronized with respective devices.

Code: 200

Response: Successfully saved

- **Partial Success** - Configuration changes persist in AW DB, but failed to synchronize with one or more devices of Finesse and/or CVP.

Code: 201

Response: Configuration update failed for one or more devices. (This occurs when the AW DB is updated but Sync with CVP or Finesse failed.)

Code: 503

Response: The server is currently busy. Please try again later. (This occurs when data synchronization to a device is in progress.)

Example API error:

```
<apiErrors>
  <apiError>
    <errorMessage>Configuration update failed for one or more devices.</errorMessage>
    <errorType>PARTIAL_SUCCESS</errorType>
  </apiError>
</apiErrors>
```

- **Failure**- The configuration updates to AW DB is failed.

