



Omni-Channel Developer's Guide

Version 39.0, Spring '17



CONTENTS

Chapter 1: Omni-Channel SOAP API Objects	1
AgentWork	2
PresenceConfigDeclineReason	7
PresenceDeclineReason	8
PresenceUserConfig	9
PresenceUserConfigProfile	12
PresenceUserConfigUser	12
QueueRoutingConfig	13
ServiceChannel	16
ServiceChannelStatus	17
ServicePresenceStatus	18
UserServicePresence	19
Chapter 2: Omni-Channel Objects for the Salesforce Console Integration Toolkit	21
acceptAgentWork	22
closeAgentWork	23
declineAgentWork	24
getAgentWorks	25
getAgentWorkload	27
getServicePresenceStatusChannels	28
getServicePresenceStatusId	29
login	30
logout	31
setServicePresenceStatus	32
Methods for Console Events	34
Index	38

CHAPTER 1 Omni-Channel SOAP API Objects

In this chapter ...

- [AgentWork](#)
- [PresenceConfigDeclineReason](#)
- [PresenceDeclineReason](#)
- [PresenceUserConfig](#)
- [PresenceUserConfigProfile](#)
- [PresenceUserConfigUser](#)
- [QueueRoutingConfig](#)
- [ServiceChannel](#)
- [ServiceChannelStatus](#)
- [ServicePresenceStatus](#)
- [UserServicePresence](#)

Use SOAP API to create, retrieve, update or delete records, such as accounts, leads, and custom objects. The SOAP API includes several objects that let you control and customize your Omni-Channel records, including Omni-Channel users, routing configurations, and statuses.

If you need more information on Salesforce's SOAP API, see the [SOAP API Developer's Guide](#).

AgentWork

Represents a work assignment that's been routed to an agent. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `getDeleted()`, `getUpdated()`, `retrieve()`, `undelete()`

Fields

Field	Details
AcceptDatetime	Type dateTime Properties Filter, Nillable, Sort Description Indicates when the work item was accepted.
ActiveTime	Type int Properties Filter, Group, Nillable, Sort Description The amount of time an agent actively worked on the work item. Tracks when the item is open and in focus in the agent's console.
AgentCapacityWhenDeclined	Type double Properties Filter, Nillable, Sort Description The agent's capacity when declining work, either explicitly or through push timeout.
AssignedDateTime	Type dateTime Properties Filter, Nillable, Sort Description Indicates when the work item was assigned to an agent,
CancelDateTime	Type dateTime

Field	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work item was canceled.</p>
CapacityPercentage	<p>Type percent</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The percentage of an agent's capacity for work items that's consumed by a specific type of work item from this service channel. When combined work items reach 100% for an agent, the agent won't receive new work items until there is enough open capacity for more work. For example, if you give phone calls a capacity percentage of 100, an agent on a call doesn't receive new work items until the call ends.</p>
CapacityWeight	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The amount of an agent's capacity for work items that's consumed by a work item from this service channel. For example, if cases are assigned a capacity weight of 2, an agent with a capacity of 6 can accept up to 3 cases before the agent is at capacity and can't receive new work items.</p>
CloseDateTime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Indicates when the work item was closed.</p>
CreatedById	<p>Type reference</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description ID of the User who created this record.</p>

Field	Details
CreatedDate	<p>Type dateTime</p> <p>Properties Defaulted on createFilter, Sort</p> <p>Description Date and time when this record was created.</p>
DeclineDatetime	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when this record was declined by an agent.</p>
DeclineReason	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The provided reason for why an agent declined the work request.</p>
HandleTime	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The amount of time an agent had the work item open. Calculated by <code>Close Time - Accepted Time</code>.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, idLookup, Sort</p> <p>Description The ID of the AgentWork object.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	Description Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>).
LastModifiedById	Type reference Properties Defaulted on create, Filter, Group, Sort Description The ID of the user who last modified this record.
LastModifiedDate	Type dateTime Properties Defaulted on create, Filter, Sort Description Date and time when a user last modified this record.
Name	Type string Properties Autonumber, Defaulted on create, Filter, idLookup, Sort Description An automatically generated ID number that identifies the record.
OriginalQueueId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the queue that the work assignment was originally routed to.
PushTimeout	Type int Properties Filter, Group, Nillable, Sort Description The number of seconds set for push timeout. <code>0</code> is returned when push timeout isn't enabled. Available in API version 36.0 and later.
PushTimeoutDateTime	Type dateTime

Field	Details
	Properties Filter, Nillable, Sort Description Indicates when the push timeout event occurred. Available in API version 36.0 and later.
RequestDateTime	Type dateTime Properties Filter, Nillable, Sort Description Indicates when the work was requested.
ServiceChannelId	Type reference Properties Create, Filter, Group, Sort Description The ID of the service channel that's associated with the work assignment.
SpeedToAnswer	Type int Properties Filter, Group, Nillable, Sort Description The amount of time between when the work was requested and when an agent accepted it.
Status	Type picklist Properties Filter, Group, Restricted picklist, Sort Description The working status of the work item. Valid values are: <ul style="list-style-type: none"> Assigned – The item is assigned to the agent but hasn't been opened. Opened – The item was opened by the agent. Unavailable – The item was assigned to the agent but the agent became unavailable (went offline or lost connection). Declined – The item was assigned to the agent but the agent explicitly declined it. DeclinedOnPushTimeout – The item was declined because push time-out is enabled and the item request timed out with the agent. Closed – The item is closed.

Field	Details
	<ul style="list-style-type: none"> Canceled – The item no longer needs to be routed. For example: a chat visitor cancels their Omni-Channel routed chat request before it reaches an agent.
SystemModstamp	<p>Type dateTime</p> <p>Properties Defaulted on create, Filter, Sort</p> <p>Description Date and time when a user or automated process (such as a trigger) last modified this record.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user that the work item was assigned to.</p>
WorkItemId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the object that's routed to the agent through Omni-Channel.</p>

Usage

`AgentWork` records can only be deleted if they have the status Closed, Declined, or Unavailable. They can't be deleted if their status is Assigned or Opened because they're active in Omni-Channel.

`AgentWork` records have the status Assigned when they're created. Once created, the record is automatically pushed to the assigned agent.

While the metadata for `AgentWork` indicates support for `upsert()` and `update()`, these calls aren't used with `AgentWork` because none of its fields can be updated.

PresenceConfigDeclineReason

Represents the settings for a decline reason that a presence user provides when declining work. This object is available in API version 37.0 and later.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
PresenceDeclineReasonId	Type reference Properties Create, Filter, Group, Nillable, Sort Description The ID of the PresenceDeclineReason record.
PresenceUserConfigId	Type reference Properties Create, Filter, Group, Sort Description The ID of the PresenceUserConfig record where the decline reasons are added.

PresenceDeclineReason


Represents an Omni-Channel decline reason that agents can select when declining work requests. This object is available in API version 37.0 and later.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
DeveloperName	Type string Properties Create, Filter, Group, Sort, Update Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.

Field	Details
	 Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.
Language	Type picklist Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The language of the PresenceDeclineReason .
MasterLabel	Type string Properties Create, Filter, Group, Sort, Update Description The master label for the PresenceDeclineReason .

PresenceUserConfig


Represents a configuration that determines a presence user's settings. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `describeObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
Capacity	Type int Properties Create, Filter, Group, Sort, Update Description The maximum number of work assignments that can be pushed to an agent at a time.
DeveloperName	Type string

Field	Details
	<p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the presence configuration.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The label of the presence configuration.</p>
OptionsIsAutoAcceptEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether work items that are routed to agents are automatically accepted (<code>true</code>) or not (<code>false</code>). Available only if <code>OptionsIsDeclineEnabled</code> is set to <code>false</code>.</p>
OptionsIsDeclineEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether agents can decline work items that are routed to them (<code>true</code>) or not (<code>false</code>). Available only if <code>OptionsIsAutoAcceptEnabled</code> is set to <code>false</code>.</p>

Field	Details
OptionsIsDeclineReasonEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether agents can select a reason for declining work requests (<code>true</code>) or not (<code>false</code>). This can be selected only if decline reasons are enabled.</p>
OptionsIsDisconnectSoundEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether a sound is played when agents are disconnected from Omni-Channel (<code>true</code>) or not (<code>false</code>).</p>
OptionsIsRequestSoundEnabled	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Indicates whether a sound plays with incoming work requests (<code>true</code>) or not (<code>false</code>). Set to <code>true</code> by default.</p>
PresenceStatusOnDeclineId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the presence status that's automatically assigned to the agent when the agent declines a work item. Available only if <code>OptionsIsDeclineEnabled</code> is set to <code>true</code>.</p>
PresenceStatusOnPushTimeoutId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the presence status that's automatically assigned to the agent when the agent doesn't respond to a work item before push timeout occurs. Available in API version 36.0 and later.</p>

PresenceUserConfigProfile

Represents a configuration that determines the settings that are assigned to presence users who are assigned to a specific profile. User-level configurations override profile-level configurations. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
PresenceUserConfigId	Type reference Properties Create, Filter, Group, Sort Description If an individual user is also assigned a presence configuration through the PresenceUserConfigProfile, this configuration will override that.
ProfileId	Type reference Properties Create, Filter, Group, Sort Description The ID of the profile that's associated with this presence configuration. A profile can be associated with only one presence configuration.

PresenceUserConfigUser

Represents a configuration that determines the settings that are assigned to a presence user. These user-level configurations override profile-level configurations. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
PresenceUserConfigId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the presence configuration.</p>
UserId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the user who's associated with this presence configuration. A user can be associated with only one presence configuration.</p>

QueueRoutingConfig


Represents the settings that determine how work items are routed to agents. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
CapacityPercentage	<p>Type percent</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The percentage of an agent's capacity for work items that's consumed by a specific type of work item from this service channel. For example, you might give phone calls a capacity percentage of <code>100</code>. If an agent receives a phone call, the agent won't receive new work items until the call ends, because at that point the agent's capacity will have reached 100%. This field is available in API version 33.0 and later.</p>

Field	Details
CapacityWeight	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The amount of an agent's capacity for work items that's consumed by a work item from this service channel.</p> <p>For example, if an agent has a capacity of 6, and cases are assigned a capacity weight of 2, an agent can be assigned up to 3 cases before the agent is at capacity and can't receive new work items.</p> <p>This field is available in API version 33.0 and later.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique DeveloperName for each record. If no DeveloperName is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the presence status.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The label of the presence status.</p>

Field	Details
OverflowAssigneeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the user or queue that's set as the Overflow Assignee.</p>
PushTimeout	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The number of seconds set for push timeout. 0 is returned when push timeout isn't enabled. Available in API version 36.0 and later.</p>
RoutingModel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The routing type that determines how work items are routed (pushed) to agents. Possible values are <code>Least Active</code> and <code>Most Available</code>.</p>
RoutingPriority	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The priority in which work items from the service channels that are related to this routing configuration are routed to agents. Work items from routing configurations that have lower priority values (for example, <code>0</code>) are routed to agents first.</p>
ServiceChannelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the service channel that's associated with this configuration.</p>

ServiceChannel


Represents a channel of work items that are received from your organization—for example, cases, chats, or leads. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
CapacityPercentage	<p>Type percent</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The percentage of an agent's capacity for work items that's consumed by a specific type of work item from this service channel.</p> <p>For example, you might give phone calls a capacity percentage of <code>100</code>. If an agent receives a phone call, the agent won't receive new work items until the call ends, because at that point the agent's capacity will have reached 100%.</p> <p>This field is available in API version 32.0 and earlier. For later API versions, you can set the capacity percentage of work items on the QueueRoutingConfig object.</p>
CapacityWeight	<p>Type double</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The amount of an agent's capacity for work items that's consumed by a work item from this service channel.</p> <p>For example, if an agent has a capacity of <code>6</code>, and cases are assigned a capacity weight of <code>2</code>, an agent can be assigned up to 3 cases before the agent is at capacity and can't receive new work items.</p> <p>This field is available in API version 32.0 and earlier. For later API versions, you can set the capacity weight of work items on the QueueRoutingConfig object.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p>

Field	Details
	<p>Description</p> <p>The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>
Language	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description</p> <p>The language of the service channel.</p>
MasterLabel	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Sort, Update</p> <p>Description</p> <p>The label of the service channel.</p>
RelatedEntity	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Create, Filter, Group, Restricted picklist, Sort, Unique, Update</p> <p>Description</p> <p>The type of object that's associated with this service channel.</p>

ServiceChannelStatus

Represents the status that's associated with a specific service channel. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`

Fields

Field	Details
ServiceChannelId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the service channel.</p>
ServicePresenceStatusId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the presence status that's associated with the service channel that's specified by the <code>ServicePresenceChannelId</code>.</p>

ServicePresenceStatus


Represents a presence status that can be assigned to a service channel. This object is available in API version 32.0 and later.

Supported Calls

`create()`, `delete()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>

Field	Details
	 Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.
Language	Type picklist Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update Description The language of the presence status.
MasterLabel	Type string Properties Create, Filter, Group, Sort, Update Description The label of the presence status.

UserServicePresence

Represents a presence user's real-time presence status. This object is available in API version 32.0 and later.

Supported Calls

`query()`, `getDeleted()`, `getUpdated()`, `retrieve()`

Fields

Field	Details
ConfiguredCapacity	Type int Properties Filter, Group, Nillable, Sort Description The user's total configured capacity.
IsAway	Type boolean

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the user's status is <i>Away</i>.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description An automatically generated ID number that identifies the record.</p>
ServicePresenceStatusId	<p>Type reference</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description The ID of the presence status that's associated with the presence user that's specified by the <code>UserId</code>.</p>
UserId	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The ID of the Omni-Channel user.</p>

CHAPTER 2 Omni-Channel Objects for the Salesforce Console Integration Toolkit

In this chapter ...

- [acceptAgentWork](#)
- [closeAgentWork](#)
- [declineAgentWork](#)
- [getAgentWorks](#)
- [getAgentWorkload](#)
- [getServicePresenceStatusChannels](#)
- [getServicePresenceStatusId](#)
- [login](#)
- [logout](#)
- [setServicePresenceStatus](#)
- [Methods for Console Events](#)

The Salesforce console is designed for users in fast-paced environments who need to find, update, and create records in Salesforce quickly. The Salesforce Console Integration Toolkit provides you with programmatic access to the Salesforce console so that you can extend it to meet your business needs. The Salesforce Console Integration Toolkit includes several objects that let you control how Omni-Channel works within the Salesforce console for your organization.

If you need more information on the Salesforce Console Integration Toolkit, see the [Salesforce Console Integration Toolkit Developer's Guide](#).

acceptAgentWork

Accepts a work item that's assigned to an agent. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.acceptAgentWork(workId:String, (optional) callback:function)
```

Arguments

Name	Type	Description
workId	String	The ID of the work item the agent accepts.
callback	function	JavaScript method to call when an agent accepts the work item associated with the workId.

Sample Code—Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/34.0/integration.js"/>
  <a href="#" onClick="testAcceptWork();return false;">Accept Assigned Work Item</a>

  <script type="text/javascript">
    function testAcceptWork() {
      //First get the ID of the assigned work item to accept it
      sforce.console.presence.getAgentWorks(function(result) {
        if (result.success) {
          var works = JSON.parse(result.works);
          var work = works[0];
          if (!work.isEngaged) {
            //Now that we have the assigned work item ID, we can accept it
            sforce.console.presence.acceptAgentWork(work.workId,
function(result) {
              if (result.success) {
                alert('Accepted work successfully');
              } else {
                alert('Accept work failed');
              }
            });
          } else {
            alert('The work item has already been accepted');
          }
        }
      });
    }
  </script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	true if accepting the work item was successful; false if accepting the work item wasn't successful.

closeAgentWork

Changes the status of a work item to "Closed" and removes it from the list of work items in the Omni-Channel widget. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.closeAgentWork(workId:String, (optional) callback:function)
```

Arguments

Name	Type	Description
workId	String	The ID of the work item that's closed.
callback	function	JavaScript method to call when the work item associated with the workId is closed.

Sample Code–Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/34.0/integration.js"/>
  <a href="#" onClick="testCloseWork();return false;">Close Engaged Work Item</a>
  <script type="text/javascript">
    function testCloseWork() {
      //First get the ID of the engaged work item to close it
      sforce.console.presence.getAgentWorks(function(result) {
        if (result.success) {
          var works = JSON.parse(result.works);
          var work = works[0];
          if (work.isEngaged) {
            //Now that we have the engaged work item ID, we can close it
            sforce.console.presence.closeAgentWork(work.workId,function(result)
          {
            if (result.success) {
              alert('Closed work successfully');
            } else {
```

```
                alert('Close work failed');
            }
        });
    } else {
        alert('The work item should be accepted first');
    }
}
});
}
</script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	true if closing the work item was successful; false if closing the work item wasn't successful.

declineAgentWork

Declines a work item that's assigned to an agent. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.declineAgentWork(workId:String, (optional) declineReason:String,  
(optional) callback:function)
```

Arguments

Name	Type	Description
workId	String	The ID of the work item that the agent declines.
declineReason	String	The provided reason for why the agent declined the work request.
callback	function	JavaScript method to call when an agent declines the work item associated with the workId.

Sample Code—Visualforce

```
<apex:page >
    <apex:includeScript value="/support/console/34.0/integration.js"/>
    <a href="#" onClick="testDeclineWork();return false;">Decline Assigned Work Item</a>
```

```
<script type="text/javascript">
  function testAcceptWork() {
    //First, get the ID of the assigned work item to accept it
    sforce.console.presence.getAgentWorks(function(result) {
      if (result.success) {
        var works = JSON.parse(result.works);
        var work = works[0];
        sforce.console.presence.declineAgentWork(work.workId, function(result)
{
          if (result.success) {
            alert('Declined work successfully');
          } else {
            alert('Decline work failed');
          }
        });
      }
    });
  }
}
</script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	true if declining the work item was successful; false if declining the work item wasn't successful.

getAgentWorks

Returns a list of work items that are currently assigned to an agent and open in the agent's workspace. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.getAgentWorks(callback: function)
```

Arguments

Name	Type	Description
callback	function	JavaScript method to call when the list of an agent's work items is retrieved.

Sample Code–Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/34.0/integration.js"/>
  <a href="#" onClick="testGetWorks();return false;">Get Agent's Current Work Items</a>

  <script type="text/javascript">
    function testGetWorks() {
      //These values are for example purposes only.
      sforce.console.presence.getAgentWorks(function(result) {
        if (result.success) {
          alert('Get work items successful');
          var works = JSON.parse(result.works);
          alert('First Agent Work ID is: ' + works[0].workId);
          alert('Assigned Entity Id of the first Agent Work is: ' +
works[0].workItemId);
          alert('Is first Agent Work Engaged: ' + works[0].isEngaged);
        } else {
          alert('Get work items failed');
        }
      });
    }
  </script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	<code>true</code> if retrieving the agent's work items was successful; <code>false</code> if retrieving the agent's work items wasn't successful.
works	JSON string of work objects	A JSON string of <code>work</code> objects that represents the work items assigned to the agent that are open in the agent's workspace.

work

The `work` object contains the following properties:

Name	Type	Description
workItemId	String	The ID of the object that's routed through Omni-Channel. This object becomes a work assignment with a <code>workId</code> when it's assigned to an agent.
workId	String	The ID of a work assignment that's routed to an agent.
isEngaged	Boolean	Indicates whether an agent is working on a work item that's been assigned to them (<code>true</code>) or not (<code>false</code>).

getAgentWorkload

In API version 35.0 and later, we can retrieve an agent's currently assigned workload. Use this method for rerouting work to available agents.

Syntax

```
sforce.console.presence.getAgentWorkload(callback: function)
```

Arguments

Name	Type	Description
callback	function	JavaScript method to call when the agent's configured capacity and work retrieved.

Sample Code—Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/35.0/integration.js"/>
  <a href="#" onClick="testGetAgentWorkload();return false;">
    Get Agent's configured capacity and current workload
  </a>

  <script type="text/javascript">
    function testGetAgentWorkload() {
      sforce.console.presence.getAgentWorkload(function(result) {
        if (result.success) {
          alert('Retrieved Agent Configured Capacity and Current Workload
successfully');
          alert('Agent\'s configured capacity is: ' + result.configuredCapacity);

          alert('Agent\'s currently assigned workload is: ' +
result.currentWorkload);
        } else {
          alert('Get Agent Workload failed');
        }
      });
    }
  </script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	<code>true</code> if retrieving the agent's work items was successful; <code>false</code> if retrieving the agent's work items wasn't successful.
configuredCapacity	Number	Indicates the agent's configured capacity (work that's assigned to the current user) through Presence Configuration.
currentWorkload	Number	Indicates the agent's currently assigned workload.

work

The `work` object contains the following properties:

Name	Type	Description
workItemId	String	The ID of the object that's routed through Omni-Channel. This object becomes a work assignment with a <code>workId</code> when it's assigned to an agent.
workId	String	The ID of a work assignment that's routed to an agent.
isEngaged	Boolean	Indicates whether an agent is working on a work item that's been assigned to them (<code>true</code>) or not (<code>false</code>).

getServicePresenceStatusChannels

Retrieves the service channels that are associated with an Omni-Channel user's current presence status. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.getServicePresenceStatusChannels (callback: function)
```

Arguments

Name	Type	Description
callback	function	JavaScript method to call when the channels associated with a presence status are retrieved.

Sample Code—Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/34.0/integration.js"/>
  <a href="#" onClick="testGetChannels();return false;">
    Get Channels Associated with a Presence Status
```



```
</a>

<script type="text/javascript">
    function testGetChannels() {
        //These values are for example purposes only.
        sforce.console.presence.getServicePresenceStatusChannels(function(result) {
            if (result.success) {
                alert('Retrieved Service Presence Status Channels successfully');
                var channels = JSON.parse(result.channels);
                //For example purposes, just retrieve the first channel
                alert('First channel ID is: ' + channels[0].channelId);
                alert('First channel developer name is: ' + channels[0].developerName);

            } else {
                alert('Get Service Presence Status Channels failed');
            }
        });
    }
</script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	true if retrieving the current presence status channels was successful; false if the retrieving the current presence status channels wasn't successful.
channels	JSON string of channel objects	Returns the IDs and API names of the channels associated with the presence status.

getServicePresenceStatusId

Retrieves an agent's current presence status. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.getServicePresenceStatusId(callback: function)
```

Arguments

Name	Type	Description
callback	function	JavaScript method to call when the agent's presence status is retrieved.

Sample Code–Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/34.0/integration.js"/>
  <a href="#" onClick="testGetStatusId();return false;">Get Omni-Channel Status ID</a>

  <script type="text/javascript">
    function testGetStatusId() {
      sforce.console.presence.getServicePresenceStatusId(function(result) {
        if (result.success) {
          alert('Get Status Id successful');
          alert('Status Id is: ' + result.statusId);
        } else {
          alert('Get Status Id failed');
        }
      });
    }
  </script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	<code>true</code> if retrieving the presence status ID was successful; <code>false</code> if the retrieving the presence status ID wasn't successful.
statusName	String	The name of the agent's current presence status.
statusApiName	String	The API name of the agent's current presence status.
statusId	String	The ID of the agent's current presence status.

login

Logs an agent into Omni-Channel with a specific presence status. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.login(statusId:String, (optional) callback:function)
```

Arguments

Name	Type	Description
statusId	String	The ID of the presence status. Agents must be given access to this presence status through their associated profile or permission set.
callback	function	JavaScript method to call when the agent is logged in with the presence status associated with statusId.

Sample Code—Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/32.0/integration.js"/>
  <a href="#" onClick="testLogin('0N5xx00000000081');return false;">Log In to
Omni-Channel</a>

  <script type="text/javascript">
    function testLogin(statusId) {
      //Gets the Salesforce ID of the presence status entity which the current user
has been assigned through their permission set or profile.
      //These values are for example purposes only.
      sforce.console.presence.login(statusId, function(result) {
        if (result.success) {
          alert('Login successful');
        } else {
          alert('Login failed');
        }
      });
    }
  </script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	true if the login was successful; false if the login wasn't successful.

logout

Logs an agent out of Omni-Channel. Available in API versions 32.0 and later.

Syntax

```
sforce.console.presence.logout((optional) callback: function)
```

Arguments

Name	Type	Description
callback	function	JavaScript method to call when the agent is logged out of Omni-Channel.

Sample Code—Visualforce

```
<apex:page>
  <apex:includeScript value="/support/console/32.0/integration.js"/>
  <a href="#" onClick="testLogout();return false;">Log out of Omni-Channel</a>

  <script type="text/javascript">
    function testLogout() {
      sforce.console.presence.logout(function(result) {
        if (result.success) {
          alert('Logout successfully');
        } else {
          alert('Logout failed');
        }
      });
    }
  </script>
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
success	Boolean	true if the logout was successful; false if the logout wasn't successful.

setServicePresenceStatus

Sets an agent's presence status to a status with a particular ID. In API version 35.0 and later, we log the user into presence if that user is not already logged in. This will remove the need for you to make additional calls.

Syntax

```
sforce.console.presence.setServicePresenceStatus(statusId:String,  
  (optional) callback:function)
```

Arguments

Name	Type	Description
statusId	String	The ID of the presence status you want to set the agent to. Agents must be given access to this presence status through their associated profile or permission set.
callback	function	JavaScript method to call when the agent's status is changed to the presence status associated with statusId.

Sample Code–Visualforce

```
<apex:page>  
  <apex:includeScript value="/support/console/32.0/integration.js"/>  
  <a href="#" onClick="testSetStatus('0N5xx000000000081');return false;">Set Presence  
  Status</a>  
  
  <script type="text/javascript">  
    function testSetStatus(statusId) {  
  
      //Sets the user's presence status to statusID. Assumes that the user was  
      assigned this presence status through Setup.  
      //These values are for example purposes only  
      sforce.console.presence.setServicePresenceStatus(statusId, function(result) {  
  
        if (result.success) {  
          alert('Set status successful');  
          alert('Current statusId is: ' + result.statusId);  
          alert('Channel list attached to this status is: ' + result.channels);  
          //printout in console for lists  
        } else {  
          alert('Set status failed');  
        }  
      });  
    }  
  </script>  
</apex:page>
```

Response

This method is asynchronous so it returns its response in an object in a callback method. The response object contains the following properties:

Name	Type	Description
<code>success</code>	Boolean	<code>true</code> if setting the agent's status was successful; <code>false</code> if setting the agent's status wasn't successful.
<code>statusName</code>	String	The name of the agent's current presence status.
<code>statusApiName</code>	String	The API name of the agent's current presence status.
<code>statusId</code>	String	The ID of the agent's current presence status.
<code>channels</code>	JSON string of <code>channel</code> objects	Returns the IDs and API names of the channels associated with the presence status.

Methods for Console Events

JavaScript can be executed when certain types of events occur in a console, such as when a user closes a tab. In addition to the standard methods for console events, there are a few events that are specific to Omni-Channel.

Standard Console Events

Event	Description	Payload
<code>sforce.console.ConsoleEvent.OPEN_TAB</code>	Fired when a primary tab or subtab is opened. Available in API version 30.0 or later.	<ul style="list-style-type: none"> <code>id</code>—the ID of the opened tab <code>objectId</code>—the object ID of the opened tab, if available
<code>sforce.console.ConsoleEvent.CLOSE_TAB</code>	Fired when a primary tab or subtab with a specified ID in the <code>additionalParams</code> argument is closed. Or, fired when a primary tab or subtab with no specified ID is closed. Available in API version 30.0 or later.	<ul style="list-style-type: none"> <code>id</code>—the ID of the closed tab <code>objectId</code>—the object ID of the closed tab, if available
<code>sforce.console.ConsoleEvent.CONSOLE_LOGOUT</code>	<p>Delays the execution of logging out of a console when a user clicks Logout. When Logout is clicked:</p> <ol style="list-style-type: none"> 1. An overlay appears, which tells a user that logout is in progress. 2. Callbacks are executed that have been registered by using <code>sforce.console.ConsoleEvent.CONSOLE_LOGOUT</code> 3. Console logout logic is executed. <p>If the callback contains synchronous blocking code, the console logout code isn't executed until the blocking code is executed. As a best</p>	None

Event	Description	Payload
	practice, avoid synchronous blocking code or long code execution during logout. Available in API version 31.0 or later.	

Omni-Channel Console Events

Event	Description	Payload
<code>sforce.console.ConsoleEvent.PRESENCE.LOGIN_SUCCESS</code>	Fired when an Omni-Channel user logs into Omni-Channel successfully. Available in API version 32.0 or later.	<ul style="list-style-type: none"> <code>statusId</code>—the ID of the agent's current presence status.
<code>sforce.console.ConsoleEvent.PRESENCE.STATUS_CHANGED</code>	Fired when a user changes his or her presence status. Available in API version 32.0 or later.	<ul style="list-style-type: none"> <code>statusId</code>—the ID of the agent's current presence status. <code>channels</code>—channel JSON string of channel objects. <code>statusName</code>—the name of the agent's current presence status. <code>statusApiName</code>—the API name of the agent's current presence status.
<code>sforce.console.ConsoleEvent.PRESENCE.LOGOUT</code>	Fired when a user logs out of Salesforce. Available in API version 32.0 or later.	None
<code>sforce.console.ConsoleEvent.PRESENCE.WORK_ASSIGNED</code>	Fired when a user is assigned a new work item. Available in API version 32.0 or later.	<ul style="list-style-type: none"> <code>workItemId</code>—the ID of the object that's routed through Omni-Channel. This object becomes a work assignment with a <code>workId</code> when it's assigned to an agent. <code>workId</code>—the ID of a work assignment that's routed to an agent.
<code>sforce.console.ConsoleEvent.PRESENCE.WORK_ACCEPTED</code>	Fired when a user accepts a work assignment, or when a work assignment is automatically accepted. Available in API version 32.0 or later.	<ul style="list-style-type: none"> <code>workItemId</code>—the ID of the object that's routed through Omni-Channel. This object becomes a work assignment with a <code>workId</code> when it's assigned to an agent. <code>workId</code>—the ID of a work assignment that's routed to an agent.

Event	Description	Payload
<code>sforce.console.ConsoleEvent.PRESENCE.WORK_DECLINED</code>	Fired when a user declines a work assignment. Available in API version 32.0 or later.	<ul style="list-style-type: none"> <code>workItemId</code>—the ID of the object that's routed through Omni-Channel. This object becomes a work assignment with a <code>workId</code> when it's assigned to an agent. <code>workId</code>—the ID of a work assignment that's routed to an agent.
<code>sforce.console.ConsoleEvent.PRESENCE.WORK_CLOSED</code>	Fired when a user closes a tab in the console that's associated with a work item. When the tab for that work item is closed, the status of the AgentWork object associated with it automatically changes to "Closed." Available in API version 32.0 or later.	<ul style="list-style-type: none"> <code>workItemId</code>—the ID of the object that's routed through Omni-Channel. This object becomes a work assignment with a <code>workId</code> when it's assigned to an agent. <code>workId</code>—the ID of a work assignment that's routed to an agent.
<code>sforce.console.ConsoleEvent.PRESENCE.WORKLOAD_CHANGED</code>	Fired when an agent's workload changes. This includes receiving new work items, declining work items, and closing items in the console. It's also fired when there's a change to an agent's capacity or Presence Configuration or when the agent goes offline in the Omni-Channel widget.	<ul style="list-style-type: none"> <code>ConfiguredCapacity</code>—the configured capacity for the agent. <code>PreviousWorkload</code>—the agent's workload before the change. <code>NewWorkload</code>—the agent's new workload after the change.

channel

The `channel` object contains the following properties:

Name	Type	Description
<code>channelId</code>	String	Retrieves the ID of a service channel that's associated with a presence status.
<code>developerName</code>	String	Retrieves the developer name of the the service channel that's associated with the <code>channelId</code> .

Methods for Console Events

Method	Description
addEventListener()	Adds a listener for a custom event type or a standard event type when the event is fired. This method adds a listener for custom event types in API version 25.0 or later; it adds a listener for standard event types in API version 30.0 or later.
fireEvent()	Fires a custom event. This method is only available in API version 25.0 or later.
removeEventListener()	Removes a listener for a custom event type or a standard event type. This method removes a listener for custom event types in API version 25.0 or later; it removes a listener for standard event types in API version 30.0 or later.

INDEX

A

[acceptAgentWork\(\)](#) 22
[AgentWork](#) object 2

C

[closeAgentWork\(\)](#) 23

D

[declineAgentWork\(\)](#) 24

E

[events](#) 34

G

[getAgentWorks\(\)](#) 25, 27
[getServicePresenceStatusChannels\(\)](#) 28
[getServicePresenceStatusId\(\)](#) 29

L

[login\(\)](#) 30
[logout\(\)](#) 31

M

Methods

- [acceptAgentWork\(\)](#) 22
- [closeAgentWork\(\)](#) 23
- [console events](#) 34
- [declineAgentWork\(\)](#) 24
- [events](#) 34
- [getAgentWorks\(\)](#) 25, 27
- [getServicePresenceStatusChannels\(\)](#) 28
- [getServicePresenceStatusId\(\)](#) 29
- [login\(\)](#) 30
- [logout\(\)](#) 31
- [setServicePresenceStatus\(\)](#) 32

O

Objects

- [AgentWork](#) 2
- [PresenceConfigDeclineReason](#) 7
- [PresenceDeclineReason](#) 8
- [PresenceUserConfig](#) 9
- [PresenceUserConfigProfile](#) 12
- [PresenceUserConfigUser](#) 12
- [QueueRoutingConfig](#) 13
- [ServiceChannel](#) 16
- [ServiceChannelStatus](#) 17
- [ServicePresenceStatus](#) 18
- [UserServicePresence](#) 19

Omni-Channel

- [Salesforce Console Integration](#) 21
- [SOAP API objects](#) 1

P

- [PresenceConfigDeclineReason](#) object 7
- [PresenceDeclineReason](#) object 8
- [PresenceUserConfig](#) object 9
- [PresenceUserConfigProfile](#) object 12

Q

[QueueRoutingConfig](#) object 13

S

- [ServiceChannel](#) object 16
- [ServiceChannelStatus](#) object 17
- [ServicePresenceStatus](#) object 18
- [setServicePresenceStatus\(\)](#) 32

U

[UserServicePresence](#) object 19