



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Web Services API Reference

Recovering Existing State

12/13/2025

# Recovering Existing State

This is part of the [API Basics](#) section of the [Web Services API](#).

## Overview

Often with voice interactions there is an existing state prior to the client application starting. This could be caused by a failover/recovery scenario or because the agent logged in and started using a physical device prior to opening the application.

The Web Services API provides several means for client applications to discover existing state. For voice applications, the key state to be examined are the state of any devices assigned for the current user and any active calls.

These details can either be queried independently or in one request as shown in the examples below. Once any existing device and call state has been recovered, the application is able to update their UI appropriately and move forward with processing of new unsolicited messages and agent initiated actions.

## Examples

### [+] Reading existing device state

#### Reading existing device state

The current state of any devices assigned to the current user can be read by sending a **GET** request to **api/v2/me/devices?fields=\***:

```
GET api/v2/me/devices?fields=*
{
  "statusCode": 0,
  "devices": [
    {
      "id": "9c14cad7-17c4-48d0-8492-7cf0ff92c224",
      "deviceState": "Active",
      "userState": {
        "id": "900D55CC-2BB0-431F-8BF9-D3525B383BE6",
        "displayName": "Not Ready",
        "state": "NotReady"
      },
      "phoneNumber": "5001",
      "e164Number": "5001",
      "telephonyNetwork": "Private",
      "doNotDisturb": "Off",
      "voiceEnvironmentUri": "http://127.0.0.1:8080/api/v2/voice-environments/370ef5e6-9e3c-4d91-9588-7f4dfe67e011",
      "capabilities": [
        "ForwardCallsOn"
      ]
    }
  ]
}
```

## [+] Reading active calls

### Reading active calls

The active calls for the current user can be read by sending a **GET** request to **api/v2/me/calls?fields=\***:

```
{
  "statusCode": 0,
  "calls": [
    {
      "id": "007102385535e008",
      "state": "Established",
      "callUuid": "011DJV5JI898NB2L04000VTAES000008",
      "deviceUri": "http://127.0.0.1:8080/api/v2/devices/9c14cad7-17c4-48d0-8492-7cf0ff92c224",
      "uri": "http://127.0.0.1:8080/api/v2/me/calls/007102385535e008",
      "participants": [
        {
          "e164Number": "",
          "formattedPhoneNumber": "5000",
          "phoneNumber": "5000",
          "digits": "5000"
        }
      ],
      "dnis": "5001",
      "callType": "Internal",
      "capabilities": [
        "AttachUserData",
        "InitiateConference",
        "UpdateUserData",
        "Hold",
        "SingleStepTransfer",
        "DeleteUserData",
        "SingleStepConference",
        "Hangup",
        "DeleteUserDataPair",
        "SendDtmf",
        "InitiateTransfer"
      ],
      "duration": "363",
      "mute": "Off",
      "supervisorListeningIn": false,
      "monitoredUserMuted": false
    }
  ]
}
```

## [+] Reading device state and active calls together

### Reading device state and active calls together

This example shows the results of getting all information for the current user.

---

This includes both device state and active calls in one request, but also includes other details, such as settings and skills.

GET api/v2/me?subresources=\*

```
{
  "statusCode": 0,
  "user": {
    "id": "8eb52b0724344f67a44389db5aa5f192",
    "userName": "jsmith@uppercape.ca",
    "firstName": "John",
    "lastName": "Smith",
    "roles": [
      "ROLE_AGENT"
    ],
    "devices": [
      {
        "id": "9c14cad7-17c4-48d0-8492-7cf0ff92c224",
        "deviceState": "Active",
        "userState": {
          "id": "900D55CC-2BB0-431F-8BF9-D3525B383BE6",
          "displayName": "Not Ready",
          "state": "NotReady"
        },
        "phoneNumber": "5001",
        "e164Number": "5001",
        "telephonyNetwork": "Private",
        "doNotDisturb": "Off",
        "voiceEnvironmentUri": "http://127.0.0.1:8080/api/v2/voice-environments/370ef5e6-9e3c-4d91-9588-7f4dfe67e011",
        "capabilities": [
          "ForwardCallsOn"
        ]
      }
    ],
    "skills": [],
    "settings": {
      "GWS": {
        "roles": "Agent"
      },
      "provisioning_flags": {
        "modified_At": "4c3fccdb-f942-4c99-a823-6cab68852ef7"
      }
    },
    "calls": [
      {
        "id": "007102385535e008",
```

```
"state": "Established",
"callUuid": "011DJV5JI898NB2L04000VTAES000008",
"deviceUri": "http://127.0.0.1:8080/api/v2/devices/9c14cad7-17c4-48d0-8492-7cf0ff92c224",
"uri": "http://127.0.0.1:8080/api/v2/me/calls/007102385535e008",
"participants": [
  {
    "e164Number": "",
    "formattedPhoneNumber": "5000",
    "phoneNumber": "5000",
    "digits": "5000"
  }
],
"dnis": "5001",
"callType": "Internal",
"capabilities": [
  "AttachUserData",
  "InitiateConference",
  "UpdateUserData",
  "Hold",
  "SingleStepTransfer",
  "DeleteUserData",
  "SingleStepConference",
  "Hangup",
  "DeleteUserDataPair",
  "SendDtmf",
  "InitiateTransfer"
],
"duration": "5",
"mute": "Off",
"supervisorListeningIn": false,
"monitoredUserMuted": false
},
"changePasswordOnFirstLogin": false
}
```