



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.

Genesys Mobile Services API Reference

`request-inbound-poll`

request-inbound-poll

```
<scxml version="1.0" xmlns="http://www.w3.org/2005/07/scxml"
  xmlns:ixn="http://www.genesyslab.com/modules/interaction"
  xmlns:service="http://www.genesyslab.com/modules/dfm/gsgBasedServices/v1"
  xmlns:ws="http://www.genesyslab.com/modules/ws"
  xmlns:session="http://www.genesyslab.com/modules/session"
  xmlns:yahoo_placefinder="http://www.genesyslab.com/modules/dfm/yahoo_placefinder/v1"
  initial="outside">
  <!--
  This is an advanced service which helps an application/end user contact the contact
center.
  It has the following characteristics:
  It supports customer initiated voice contacts.
  It supports waiting for the appropriate available agent before providing the
necessary access information.
  It stores and maintains application data with the service.
  Access number allocation/reservation will be done by the application using the
reserve request.
  This request will process the appropriate allocation algorithm in conjunction with
the GMS based A2C-Reserve-AccessInfo service.
  It only supports polling for the availability of an agent.
  -->
  <script>
    var statusId = null;
    var reserveId = null;
    var provideCode = false;
    var attemptCounter = 0;
    var callerCity = null;
    var callerState = null;
  </script>
  <datamodel>
    <data id="reqid"/>
  </datamodel>
  <state id="outside" initial="initial">
    <state id="initial">
      <onentry>
        <!-- Create an overall timer to ensure that the session does
not go on forever -->
        <send target="_internal" event="timer" delay="_data._ttl
+ 's' />
      </onentry>
      <transition event="gms.start" target="run">
        <script>
          <!-- Save the session id in the result object as it
must be returned to the client in the response to gms.start -->
          var result = new Object();
          result._id = _sessionid;
        </script>
        <!-- Do the appropriate logic to wait to get an agent; ie. do
the <queue:submit> or wait for a period of time, etc.) -->
        <queue:submit route="true" timeout="300">
          <queue:targets>
            <queue:target name="'Billing'"
type="agentgroup" statserver="'Stat_Server' />
          </queue:targets>
        </queue:submit>
        <!-- Send a response to the client indicating the result -->
        <ws:response requestid="_event.sendid"
```

```

resultcode="JSON.stringify( result )" />
    </transition>
  </state>
  <state id="run">
    <!-- Session specific request to get the status -->
    <transition event="status" target="status" >
      <script>
        statusId = _event.sendid;
      </script>
    </transition>
    <!-- Session specific request to reserve an access number using
location information -->
    <transition event="reserve" cond="_event.data.param.hasOwnProperty(
'_latitude' ) && _event.data.param.hasOwnProperty( '_longitude' )" target="getLocation" >
      <script>
        reserveId = _event.sendid;
        if ( _event.data.param.hasOwnProperty(
'_provide_code' ) ) provideCode = _event.data.param._provide_code;
        if ( _data.hasOwnProperty( '_provide_code' ) )
provideCode = _data._provide_code;
      </script>
      <!-- Request the location using yahoo placefinder given the
latitude and longitude -->
      <yahoo_placefinder:query_location appid="'yourappid'"
lat="_event.data.param._latitude" long="_event.data.param._longitude" />
    </transition>
    <!-- Session specific request to reserve an access number -->
    <transition event="reserve" target="default_reserve" >
      <script>
        reserveId = _event.sendid;
        if ( _event.data.param.hasOwnProperty(
'_provide_code' ) ) provideCode = _event.data.param._provide_code;
        if ( _data.hasOwnProperty( '_provide_code' ) )
provideCode = _data._provide_code;
      </script>
    </transition>
    <!--
      Handler for Router's response indication an agent is
reserved. Practically speaking, for this scenario there is
      nothing we can do as most likely the agent reservation will
be lost before the next status request.
    -->
    <transition event="queue.submit.done">
      <!-- Change the state of the service based on finding an
agent. -->
    </transition>
  </state>
  <!-- This request provides the application with the current status of the
service; no request specific parameters -->
  <state id="status">
    <transition target="run" >
      <!--
        These are specific response parameters from the status
request; this sample doesn't actually track a status for the
        agent (it just keeps a counter of the number of requests).
It should really give some indication of the current
        queue time. Perhaps if the remaining queue time is low,
let the user get into the queue.
      -->
      <script>
        var result = new Object();
        result._status = "" + ++attemptCounter;
        result._dialog = "Whatever text we want";

```

```

                                </script>
                                <!-- Send a response to the client indicating the result -->
                                <ws:response requestid="statusId" resultcode="JSON.stringify(
result )" />
                                </transition>
                                </state>
                                <state id="getLocation">
                                    <!--
state of the caller
                                Build a response to the yahoo placefinder request; includes city and
-->
                                <transition event="yahoo_placefinder.query_location.done"
target="reserve_by_location" >
                                    <script>
                                        var myResultSetObject = eval('(' +
_event.data.content + ')');
                                        callerCity =
myResultSetObject.ResultSet.Results[0].city;
                                        callerState =
myResultSetObject.ResultSet.Results[0].state;
                                    </script>
                                </transition>
                                <transition event="error.yahoo_placefinder.query_location"
target="error"/>
                                </state>
                                <!--
                                This request provides the application with access information that is to be
used with this service; request specific parameters are provide code, longitude, and/or
latitude.
                                This service will use the A2C-Reserve-AccessInfo service to reserve the
access information for this service using the service specific data (resource_group, etc.).
-->
                                <state id="reserve_by_location">
                                    <onentry>
                                        <script>
                                            var resource_group = ( callerCity + ":" + callerState
).toLowerCase();resource_group;
                                        </script>
                                        <!-- Request a resource for this user; that is a phone number
and perhaps an access code -->
                                        <service:reserve requestid="_data.requestId" _id="_sessionId"
_resource_group="resource_group" _provide_code="provideCode"
_phone_number="_data.phone_number" />
                                    </onentry>
                                    <transition event="error.gsgBasedServices.reserve"
target="default_reserve" />
                                    <!--
                                Build a response to the request; it may include
access_number, access_code and/or expiration time
-->
                                    <transition event="gsgBasedServices.reserve.done"
target="waitForCall" >
                                        <script>
                                            var myResultSetObject = eval('(' +
_event.data.content + ')');
                                            var result = new Object();
                                            result._id = _sessionId;
                                            if ( myResultSetObject.hasOwnProperty(
'_access_number' ) )
                                                result._access_number =
myResultSetObject._access_number;
                                            if ( myResultSetObject.hasOwnProperty( '_access_code'
) )

```

```

                                result._access_code = myResultSetObject._access_code;
                                if ( myResultSetObject.hasOwnProperty(
'_expiration_time' ) )
                                result._expiration_time =
myResultSetObject._expiration_time;
                                </script>
                                <!-- Send a response to the client indicating the result -->
                                <ws:response requestid="reserveId"
resultcode="JSON.stringify( result )" />
                                </transition>
                                </state>
                                <state id="default_reserve">
                                    <onentry>
                                        <!-- Request a resource for this user; that is a phone number
and perhaps an access code -->
                                        <service:reserve requestid="_data.requestId" _id="_sessionId"
_resource_group="_data._resource_group" _provide_code="provideCode"
_phone_number="_data._phone_number" />
                                        </onentry>
                                        <!--
                                        Build a response to the request; it may include access_number,
access_code and/or expiration time
                                        -->
                                        <transition event="gsgBasedServices.reserve.done"
target="waitForCall" >
                                            <script>
                                                var myResultSetObject = eval('(' +
_event.data.content + '));
                                                var result = new Object();
                                                result._id = _sessionId;
                                                if ( myResultSetObject.hasOwnProperty(
'_access_number' ) )
                                                result._access_number =
myResultSetObject._access_number;
                                                if ( myResultSetObject.hasOwnProperty( '_access_code'
) )
                                                result._access_code = myResultSetObject._access_code;
                                                if ( myResultSetObject.hasOwnProperty(
'_expiration_time' ) )
                                                result._expiration_time =
myResultSetObject._expiration_time;
                                            </script>
                                            <!-- Send a response to the client indicating the result -->
                                            <ws:response requestid="reserveId"
resultcode="JSON.stringify( result )" />
                                            </transition>
                                            </state>
                                            <!-- Waits for the call to be associated to this session by the inbound
.scxml -->
                                            <state id="waitForCall">
                                                <transition event="interaction.present" target="exit">
                                                    <!--
                                                    Once here, this script has full control over the
interaction and may route the call as needed;
                                                    For this sample, the call is simply disconnected.
                                                    -->
                                                    <ixn:terminate requestid="_data.reqid"
interactionid="_genesys.ixn.interactions[0].g_uid" reason="'finished service'"
resource="_genesys.ixn.interactions[0].voice.dnis"/>
                                                </transition>
                                                </state>
                                                <!-- This event means that the user did not match within the allowed time -->
                                                <transition event="timer" target="exit" />

```

```
    </state>
    <final id="exit"/>
    <final id="error"/>
</scxml>
```