

myVox Public API

Jan 4, 2008

Overview

myVox is free service that gives web and widget developers the tools to voice enable their applications using a simple mash-up API. User call a phone number, enter a code, and leave recordings using their own phone. All API commands are simple HTTP requests and all responses are XML or JSON messages. To further simplify integration with your widgets and web apps, object-orient toolkit libraries are provided in JavaScript, ActionScript, and Java.

VoiceRecorder Applications

The first IVR application offered through myVox is a VoiceRecorder application where developer's can provide voice annotations to their widgets and web sites. The VoiceRecorder application works as follows:

1. First create an instance of the VoiceRecorder application through the myVox Developer's Website, and voice the application prompts to match the functionality of your specific widget or web application. As part of this process, a unique application instance key is created and is used to identify the VoiceRecorder application instance in all API commands.
2. Then issue a CreateRecordingList command to create a RecordingList with one or more recordings. Each recording has a sequence number, a key, an optional name, and optional metadata which is stored with the recording. You may supply your own key as long as it is unique within the recordings set or you can let the system generate one for you. The sequence number determines the order in which the recordings are voiced and played back to the user on the phone system, as well their order in the RecordingList. If the optional 'name' parameter is supplied then the phone system will identify the recording by name on the phone system. Using the slideshow widget example again, this command would be called whenever a new slide is added and the 'name' parameter could be the name the user assigned to each slide.
3. When an instance of your widget needs to create the voice recordings, issue a VoiceRecordings command to initiate a recording session for a specific RecordingList instance. This command returns the phone number to call, an access PIN, and a URL to poll real-time status of the recording session. You may also supply a call-back URL of your own to receive event-driven status of the recording session.
4. The widget user calls the phone number, enters the PIN, and is greeted with your application's welcome prompt and instructions for creating, reviewing, and re-voicing recordings. Using the slideshow example again, the caller might hear "Welcome to the TalkingPhotos recording system – please enter your PIN". After she enters the PIN she would be guided though the process of creating a recording for each recording placeholder created previously. As she walks though the process the recording status tells the slideshow widget which slide is being voiced, whether the recording is in progress or completed, etc. The slideshow widget can use this information to dynamically show the current slide being voiced and activate a "play" button as soon as a recording has been saved.
5. The order of the recordings can be changed at any time, even while a recording session is in progress, by issuing a SetRecording command with a new sequence value. The slideshow widget might use this to keep the order of the recordings in sync with the order of the photos, and lets the user manipulate the order of the photos in the middle of a recording session.
6. The CreateRecording command can also be issued at any time, even while a recording session is in progress. A slideshow widget may use this to let users interactively add new photos and voice them in the same recording session. This can provide a more intuitive and natural user experience than would be otherwise be the case if adding photos and voicing them were completely separate operations.
7. The DeleteRecording command can be called at any time to remove a recording from a RecordingList

instance, even while a recording session is in process. A slideshow widget might use this when a user deletes a photo from the slideshow.

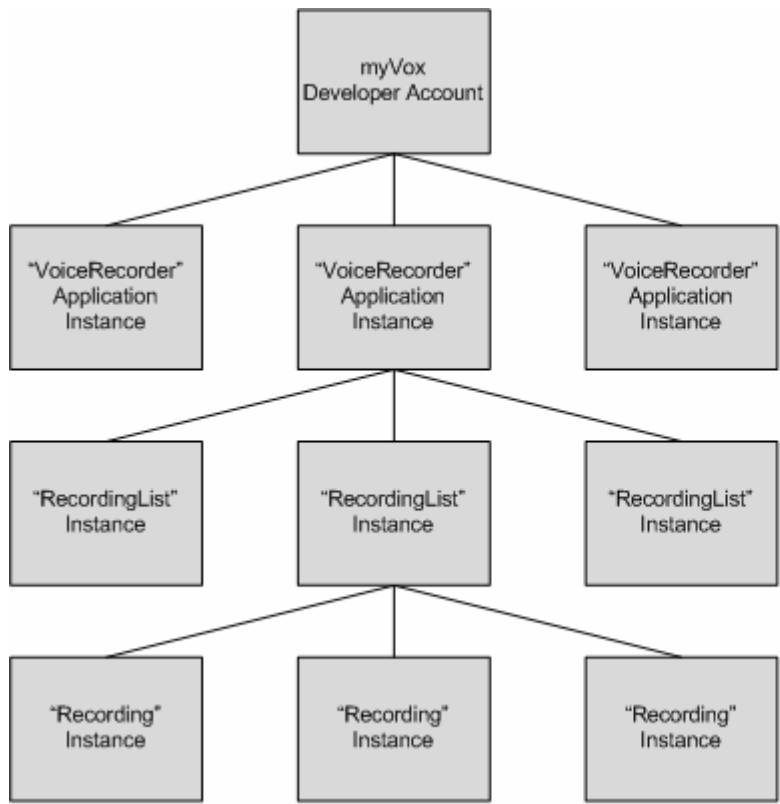
8. The caller can terminate the recording session at any time by hanging up, or by specifically selecting the “I’m done” option on the phone system. In either case the session status will indicate that the recording session has ended.
9. There is no time limit on a recording session once the caller has called in and successfully entered the PIN code. However, once the CreateRecording Session command has been issued, that session PIN must be used within a set time limit, otherwise the session will be discarded.

Protecting Assets with Passwords:

Passwords may be assigned to recording lists to control the ability to access, update or delete them and their associated recordings. If a password for a specific function is specified then it must be supplied to execute the command against that asset. To generate a random password, use “?????” as the password value. Warning: once a random password has been set it cannot be changed, since passwords cannot be read once they are assigned.

Developers, Applications, & Users

myVox provides voice services for third-party developers for use in their own applications. Developers register with the myVox service by creating an myVox developer account and providing a username, password, and other required and optional data. Access to their myVox account requires authentication using the username and password they supplied during the account creation process. Developer usernames cannot be changed after the account is created, however passwords can be updated at any time. Once a developer has an myVox account she can create instances of myVox applications. Each application instance provides a specific set of functionality and is identified by an application instance key. The application instance key is used to identify the application instance in all myVox API commands. For example, a developer may create an instance of a VoiceRecorder application for the purpose of providing voice annotation to a web-based slideshow application she's created. The developer would provide the ability for third-party users to create slideshow instances, but each slideshow application instance would use the same myVox VoiceRecorder instance for the voice annotation feature. Each slideshow application instance would have a group of recordings, identified by a unique recordings group key, and each RecordingList instance can have any number of recordings. The relationship between these various entities is showing in the following diagram:



Developer creates a myVox account on myvox.com.

Developer creates one or more "VoiceRecorder" application instances through the myvox.com Developer Zone, and associates each with his or her own widgets or applications. Each can have its own custom prompts.

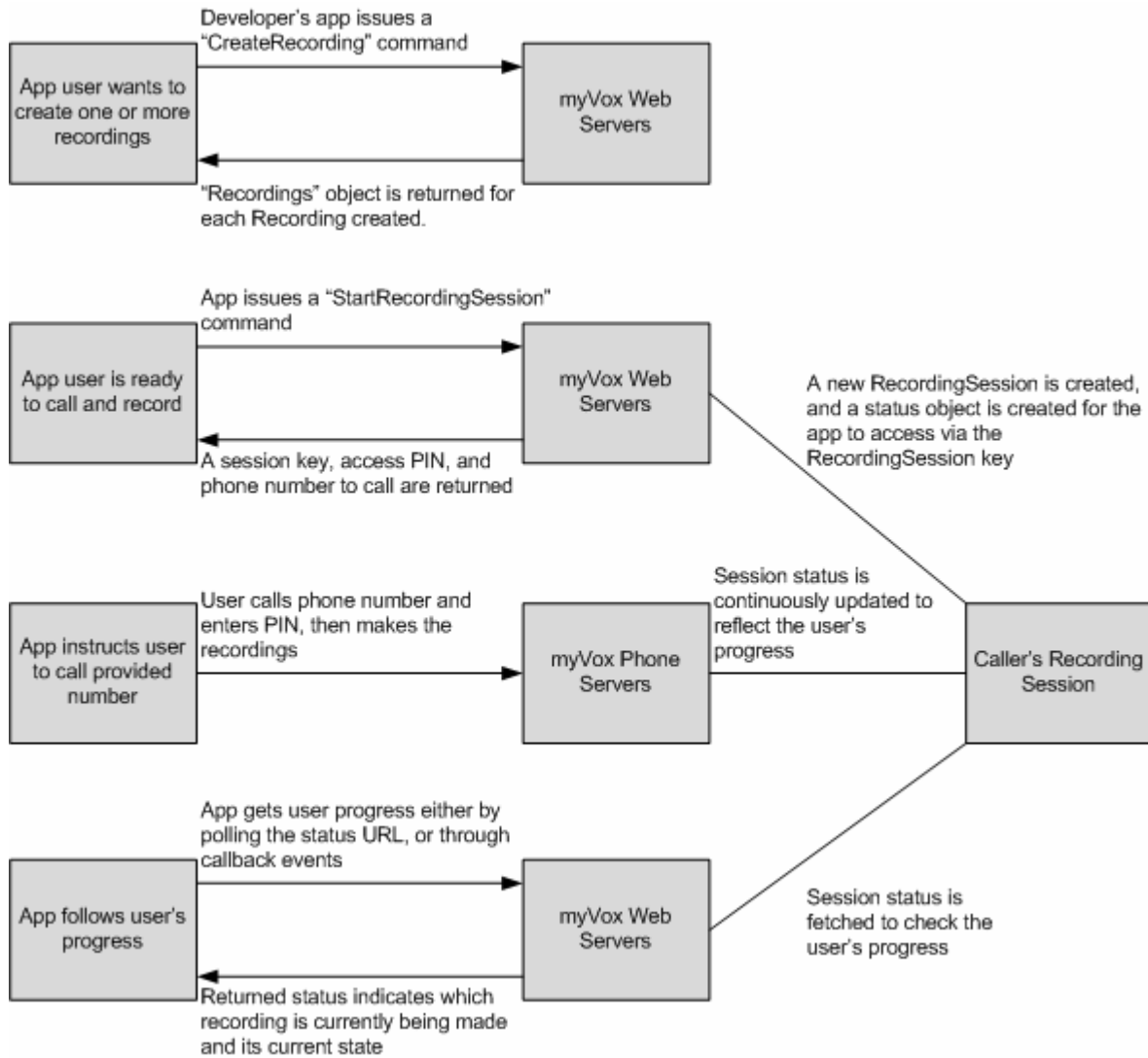
As needed, the developer or his/her application creates RecordingLists to store recordings for end users.

Recordings are created and stored within RecordingLists as needed. When it comes time to voice a recording, a RecordingSession is created, and one or more Recordings chosen to voice during that Session.

Voice

Recorder Application Process

The following diagram shows the process flow of the VoiceRecorder application. A user creates an instance of a myVox voice-enabled WebApp – a slideshow widget for example – and initiates the voice annotation process. The web app instance issues a "CreateRecordingList" command to the myVox platform to create one or more recordings to be associated with the widget and then a "VoiceRecordings" command to begin a new recording session. myVox creates and caches a new VoiceRecorder session object and returns the session key, PIN, and phone number to access the VoiceRecorder application. At this point the application begins polling for the recording status, which indicates which recording is currently being made as well as the status of all the recordings for the session. The caller calls the phone number, enters the PIN, and begins recording. When all recordings have been voiced the widget can access them by issuing a "GetRecordingList" command.



myVox API

The myVox API is implemented as a web service which provides access to all myVox functions. All API command requests are via secure HTTP POST using standard form/url name/value parameters. All API command responses are plain text strings formatted as either XML messages or JSON data structures. For the purpose of maintaining a one-to-one correlation between the XML and JSON response formats, XML responses use only nested elements rather than a mix of elements and attributes. Developer validation is performed on every API command request by comparing the supplied account username and password against the myVox database.

Common Request and Response Elements

myVox API commands several request parameters and response elements common to all API commands. The following tables detail the requests and responses common to all myVoxAPI commands.

Common Request Parameters:

All API commands have a 'command' parameter which identifies which command is being requested, as well as

the developer's account login information ('username' and 'password'). In addition the format of the return must be specified either as XML messages or JSON constructs.

<i>Parameter Name</i>	<i>Parameter Value</i>	<i>Type</i>	<i>Parameter Description</i>
action	<string>	required	Identifies the requested command
username	<string>	required	The developer's account name
password	<string>	required	The developer's account password
reply	"xml" "json"	required	The format of the command response

Common Response Data:

All command responses are a string of ASCII text formatted as either an XML message or a JSON data construct. Each response consists of a parent OMVAPI data element containing the following child elements:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

Example XML Response:

```
<openmyVox>
  <status>failed</status>
  <error_code>100</error_code>
  <error_message>Unknown System Error</error_message>
</openmyVox>
```

Example JSON Response:

```
{
  openmyVox :
  {
    status : failed,
    error_code : 100,
    error_message : Unknown System Error
  }
}
```

VoiceRecorder Application API Commands:

The following table summarize all the API commands for creating and managing recordings within a VoiceRecorder application instance.

<i>Command Name</i>	<i>Access</i>	<i>Command Description</i>
CreateRecording	HTTP / POST	Adds a new recording placeholder to a RecordingList instance and updates the sequence numbers of any others to keep them consecutive.
CreateRecordingList	HTTP / POST	Creates a new RecordingList containing one or more recording placeholders within a VoiceRecorder application instance.
DeleteRecording	HTTP / POST	Deletes a single recording from a RecordingList instance and updates the sequence numbers of any others to keep them consecutive.
DeleteRecordingList	HTTP / POST	Deletes a RecordingList instance and all recordings associated with it

ExpireRecordingSession	HTTP / POST	Expire a recording session and unlock all recordings bound to the current session.
GetRecording	HTTP / POST	Returns data for a Recording instance associated with a RecordingList instance
GetRecordingList	HTTP / POST	Returns data for a RecordingList instance within the VoiceRecorder application instance
GetSessionStatus	HTTP / POST	Returns the status of the current recording session.
SetRecording	HTTP / POST	Updates the values of a Recording instance
SetRecordingList	HTTP / POST	Updates the values of a RecordingList instance
StartRecordingSession	HTTP / POST	Initiate a recording session to voice the recordings in a RecordingList over the phone.

CreateRecording

Adds a new recording placeholder to a RecordingList instance and sets its state to “unrecorded” and automatically selects it for recording.

Request Parameters:

<i>Parameter Name</i>	<i>Parameter Value</i>	<i>Type</i>	<i>Command Description</i>
action	“CreateRecording”	required	Identifies the “CreateRecording” command
reply	“xml” (default) “json”	optional	Specifies the format of the response messages
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList in which to create the recording
password	<string>	optional	The password required to execute this command, if one has been set.
recording_key	<string: 10 chars>	optional	A custom identification key for the new Recording instance
name	<string>	optional	The name identifier for the new Recording instance
sequence	<integer: auto-increment>	optional	The sequence of the recording within the RecordingList (first = 1)
keep_consecutive	<boolean>	optional	Adjusts the sequence values of all recordings to remain consecutive
meta_data_1	<string>	optional	Arbitrary meta-data associated with the new Recording instance
meta_data_2	<string>	optional	Arbitrary meta-data associated with the new Recording instance
meta_data_3	<string>	optional	Arbitrary meta-data associated with the new Recording instance

Response Data:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
recording_list_key	<string: 10 chars>	available	The identification key for the RecordingList instance
recording_key *	<string: 10 chars>	available	The identification key for the new Recording instance
name *	<string>	conditional	The name identifier for the recording
meta_data_1 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
meta_data_2 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
meta_data_3 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
recording_state *	“unrecorded” “recording” “recorded”	available	Indicates the current state of the recording: “unrecorded” initially, “recording” while being recorded, and “recorded” once the recording process has been completed.
sequence *	<integer: auto-increment>	available	The sequence of the recording within the RecordingList (first = 1)
recording_length *	<integer>	available	The length of the recording in seconds (0 if unrecorded)
play_key *	<string: 20 chars>	available	The unique identification key used to play the Recording

has_wav_file *	<boolean>	available	Indicates whether this recording has a WAV file
wav_url *	<string: url>	conditional	The URL to play the WAV recording file, if one is available
has_mp3_file *	<boolean>	available	Indicates whether this recording has an MP3 file
mp3_url *	<string: url>	conditional	The URL to play the MP3 recording file, if one is available
date_created *	<date-time>	available	The date and time when recording was initially created
last_updated *	<date-time>	available	The date and time when recording was last updated
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

* Attributes of a "recording" parent attribute, used to describe a single Recording instance.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="New recording 62B3S1lLBB created" status="success">
  <voice_recorder_key>AOAIG2UUVO</voice_recorder_key>
  <recording_list_key>WWJKFT0A5V</recording_list_key>
  <recording>
    <recording_key>62B3S1lLBB</recording_key>
    <recording_state>recorded</recording_state>
    <name>My Recording</name>
    <sequence>1</sequence>
    <is_in_session>>false</is_in_session>
    <play_key>HJ4SF86FE666EL9U73GE</play_key>
    <has_wav_file>>false</has_wav_file>
    <has_mp3_file>>true</has_mp3_file>
    <mp3_url>http://api.myVox.com/HJ4SF86FE666EL9U73GE/recording.mp3</mp3_url>
    <date_created>2007-12-12 22:32:59 EST</date_created>
    <last_updated>2007-12-12 22:32:59 EST</last_updated>
    <last_accessed>2007-12-12 22:32:59 EST</last_accessed>
  </recording>
</myVoxapi>
```

CreateRecordingList

Creates a new "RecordingList" containing one or more recording placeholders within a VoiceRecorder application instance. "Recording" placeholders are initialized to an "unrecorded" state.

Request Parameters:

Parameter Name	Parameter Value	Type	Command Description
action	"CreateRecordingList"	required	Identifies the "CreateRecordingList" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
password	<string>	optional	The password required to execute this command, if one has been set.
recording_list_key	<string: 10 chars>	optional	A custom identification key for the new RecordingList instance
name	<string>	optional	The name identifier for the RecordingsList instance
meta_data_1	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_2	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_3	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance

access_password	<string>	optional	Password to protect access to the RecordingList instance (??? = random)
update_password	<string>	optional	Password to protect updates to the RecordingList instance (??? = random)
delete_password	<string>	optional	Password to protect deleting the RecordingList instance (??? = random)
inital_recordings	<integer: default = 0>	optional	The number of initial unrecorded Recording instances with the RecordingList
max_recordings	<integer>	optional	The maximum number of Recordings allowed for this RecordingList

Response Data:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
voice_recorder_key	<string: 10 chars>	available	The identification key for the VoiceRecorder application instance
recording_list_key *	<string: 10 chars>	available	The identification key for the RecordingList instance
name *	<string>	conditional	The name identifier for the RecordingList instance
meta_data_1 *	<string>	available	Arbitrary meta-data associated with the new RecordingList instance
meta_data_2 *	<string>	available	Arbitrary meta-data associated with the new RecordingList instance
meta_data_3 *	<string>	available	Arbitrary meta-data associated with the new RecordingList instance
recording_count	<integer>	available	The number of recordings currently associated with this RecordingList
max_recordings *	<integer>	conditional	The maximum number of Recordings allowed for this RecordingList
created *	<date-time>	available	The date and time when RecordingList instance was initially created
access_protected *	<boolean>	available	Indicates whether RecordingList access is password protected
update_protected *	<boolean>	available	Indicates whether RecordingList updates is password protected
delete_protected *	<boolean>	available	Indicates whether RecordingList deleting is password protected
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

* Attributes of an "recording_list" parent attribute, used to describe a single RecordingList instance.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="New recording list MyCustomKey created" status="success">
  <voice_recorder_key>AOAIG2UUVO</voice_recorder_key>
  <recording_list>
    <recording_list_key>MyCustomKey</recording_list_key>
    <name>Test Recording List</name>
    <meta_data_1>xxx</meta_data_1>
    <meta_data_2>yyy</meta_data_2>
    <meta_data_3>zzz</meta_data_3>
    <access_protected>>false</access_protected>
    <update_protected>>false</update_protected>
    <delete_protected>>false</delete_protected>
    <recording_count>3</recording_count>
    <max_recordings>5</max_recordings>
    <date_created>2007-12-12 20:21:43 EST</date_created>
    <last_updated>2007-12-12 20:21:43 EST</last_updated>
    <last_accessed>2007-12-12 20:21:43 EST</last_accessed>
  </recording_list>
</myVoxapi>
```

DeleteRecording

Deletes a single recording from a RecordingList instance and updates the sequence numbers of any others to keep them consecutive.

Request Parameters:

<i>Parameter Name</i>	<i>Parameter Value</i>	<i>Type</i>	<i>Command Description</i>
action	"DeleteRecording"	required	Identifies the "DeleteRecording" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList instance
recording_key	<string: 10 chars>	required	The identification key for the Recording instance
keep_consecutive	<boolean>	optional	Adjusts the sequence values of all recordings to remain consecutive

Response Data:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

Sample Response:

```
<myVoxapi message="Recording list KRYQUKAU0G deleted" status="success"/>
```

DeleteRecordingList

Deletes a RecordingList instance and all recordings associated with it.

Request Parameters:

<i>Parameter Name</i>	<i>Parameter Value</i>	<i>Type</i>	<i>Command Description</i>
action	"DeleteRecordingList"	required	Identifies the "DeleteRecordingList" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList to be deleted

Response Data:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

Sample Response:

```
<myVoxapi message="Recording list BBBB deleted" status="success"/>
```

ExpireRecordingSession

Used to expire a recording session and release all recordings bound to the current session.

Request Parameters:

<i>Parameter Name</i>	<i>Parameter Value</i>	<i>Type</i>	<i>Command Description</i>
action	"ExpireRecordingSession"	required	Identifies the "ExpireRecordingSession" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
recording_session_key	<string: 10 chars>	required	Identifies a recording session tied to a specific RecordingList instance.

Response Data:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
session_state	"expired"	available	Indicates the current state of the session is "expired".
voice_recorder_key	<string: 10 chars>	available	Identifies the voice recorder that owns the recording_list instance
recording_list_key	<string: 10 chars>	available	The identification key for the recording_list instance
recording_session_key	<string: 10 chars>	available	Identifies a recording session tied to a specific RecordingList instance
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="Recording session status has been retrieved" status="success">
  <session_state>called</session_state>
  <voice_recorder_key>ZV2ZEXHJ4P</voice_recorder_key>
  <recording_list_key>36PONP0XTR</recording_list_key>
  <recording_session_key>V6GUN79S7T</recording_session_key>
</myVoxapi>
```

GetRecording

Returns data for a Recording instance associated with a RecordingList instance. If "recording_key" is specified then the matching recording is returned. If one or more meta-data fields are specified then the recording that matches the meta-data is returned.

Request Parameters:

<i>Parameter Name</i>	<i>Parameter Value</i>	<i>Type</i>	<i>Command Description</i>
-----------------------	------------------------	-------------	----------------------------

action	"GetRecording"	required	Identifies the "GetRecording" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList instance
recording_key	<string: 10 chars>	optional	The identification key for the Recording instance
meta_data_1	<string>	optional	Arbitrary meta-data associated with the Recording instance
meta_data_2	<string>	optional	Arbitrary meta-data associated with the Recording instance
meta_data_3	<string>	optional	Arbitrary meta-data associated with the Recording instance

Response Data:

Attribute Name	Attribute Value	Type	Parameter Description
recording_list_key	<string: 10 chars>	available	The identification key for the RecordingList instance
recording_key *	<string: 10 chars>	available	The identification key for the new Recording instance
name *	<string>	conditional	The name identifier for the recording
meta_data_1 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
meta_data_2 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
meta_data_3 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
recording_state *	"unrecorded" "recording" "recorded"	available	Indicates the current state of the recording: "unrecorded" initially, "recordings" while being recorded, while being reviewed, and "recorded" once the recording process has been completed.
sequence *	<integer: auto-increment>	available	The sequence of the recording within the RecordingList (first = 1)
length *	<integer>	available	The length of the recording in seconds (0 if unrecorded)
play_key *	<string: 20 chars>	available	The unique identification key used to play the Recording
has_wav_file *	<boolean>	available	Indicates whether this recording has a WAV file
wav_url *	<string: url>	conditional	The URL to play the WAV recording file, if one is available
has_mp3_file *	<boolean>	available	Indicates whether this recording has an MP3 file
mp3_url *	<string: url>	conditional	The URL to play the MP3 recording file, if one is available
date_created *	<date-time>	available	The date and time when recording was initially created
last_updated *	<date-time>	available	The date and time when recording was last updated
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

- Attributes of a "recording" parent attribute, used to describe a single Recording instance.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="Recording 62B3S1l1LBB returned" status="success">
  <voice_recorder_key>AOAIG2UUVO</voice_recorder_key>
  <recording_list_key>WWJKFT0A5V</recording_list_key>
  <recording>
    <recording_key>62B3S1l1LBB</recording_key>
    <recording_state>recorded</recording_state>
    <name>My Recording</name>
    <sequence>1</sequence>
    <is_in_session>>false</is_in_session>
  </recording>
</myVoxapi>
```

```

    <play_key>HJ4SF86FE666EL9U73GE</play_key>
    <has_wav_file>>false</has_wav_file>
    <has_mp3_file>>true</has_mp3_file>
    <mp3_url>http://api.myVox.com/HJ4SF86FE666EL9U73GE/recording.mp3</mp3_url>
    <date_created>2007-12-12 22:32:59 EST</date_created>
    <last_updated>2007-12-12 22:32:59 EST</last_updated>
    <last_accessed>2007-12-12 22:32:59 EST</last_accessed>
  </recording>
</myVoxapi>

```

GetRecordingList

Returns data for a RecordingList instance associated with the specified VoiceRecorder application instance. If “recording_list_key” is specified then the matching RecordingList is returned. If one or more meta-data fields are specified then the RecordingList instance matching the meta-data is returned. A maximum of 100 entries can be returned in a single command to prevent excessive database load.

Request Parameters:

Parameter Name	Parameter Value	Type	Command Description
action	“GetRecordingList”	required	Identifies the “GetRecordingList” command
reply	“xml” (default) “json”	optional	Specifies the format of the response messages
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	optional	The identification key for the RecordingList instance
meta_data_1	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_2	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_3	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance

Response Data:

Attribute Name	Attribute Value	Type	Parameter Description
voice_recorder_key	<string: 10 chars>	available	The identification key for the VoiceRecorder application instance
recording_list_key *	<string: 10 chars>	available	The identification key for the RecordingList instance
name *	<string>	conditional	The name identifier for the RecordingList instance
meta_data_1 *	<string>	conditional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_2 *	<string>	conditional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_3 *	<string>	conditional	Arbitrary meta-data associated with the new RecordingList instance
recording_count	<integer>	available	The number of recordings currently associated with this RecordingList
max_recordings *	<integer>	conditional	The maximum number of Recordings allowed for this RecordingList
created *	<date-time>	available	The date and time when RecordingList was initially created
access_protected *	<boolean>	available	Indicates whether RecordingList access is password protected
update_protected *	<boolean>	available	Indicates whether RecordingList updates is password protected
delete_protected *	<boolean>	available	Indicates whether RecordingList deleting is password protected
status	“success” “failed”	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see “Error Codes” list)

error_message <string> conditional Describes the reason for a failure in text (see "Error Codes" list)

* Attributes of an "recording_list" parent attribute, used to describe a single RecordingList instance.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="Recording list MyCustomKey returned" status="success">
  <voice_recorder_key>AOAIG2UUVO</voice_recorder_key>
  <recording_list>
    <recording_list_key>MyCustomKey</recording_list_key>
    <name>Test Recording List</name>
    <meta_data_1>xxx</meta_data_1>
    <meta_data_2>yyy</meta_data_2>
    <meta_data_3>zzz</meta_data_3>
    <access_protected>>false</access_protected>
    <update_protected>>false</update_protected>
    <delete_protected>>false</delete_protected>
    <recording_count>3</recording_count>
    <max_recordings>5</max_recordings>
    <date_created>2007-12-12 20:21:43 EST</date_created>
    <last_updated>2007-12-12 20:21:43 EST</last_updated>
    <last_accessed>2007-12-12 20:21:43 EST</last_accessed>
  </recording_list>
</myVoxapi>
```

GetSessionStatus

Returns the status of the current recording session, including whether a user has called into the VoiceRecorder IVR, the caller's current state, the state of the current recording being voiced or reviewed.

Request Parameters:

Parameter Name	Parameter Value	Type	Command Description
action	"GetSessionStatus"	required	Identifies the "GetSessionStatus" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
recording_session_key	<string: 10 chars>	required	Identifies a recording session tied to a specific RecordingList instance.

Response Data:

Attribute Name	Attribute Value	Type	Parameter Description
session_state	"created" "called" "recording" "ended" "expired"	available	Indicates the current state of the session. "created" indicates a session has been created and is active, but no phone call has been made. "called" indicates that the user has called into the VoiceRecorder IVR and entered the PIN for the session. "recording" indicates a recording is being voiced. "ended" indicates that the call has terminated. "expired" indicates that the session expired before a user called in.
voice_recorder_key	<string: 10 chars>	available	Identifies the voice recorder that owns the recording_list instance
recording_list_key	<string: 10 chars>	available	The identification key for the recording_list instance
recording_session_key	<string: 10 chars>	available	Identifies a recording session tied to a specific RecordingList instance
current_recording *1	<object>	available	The Recording currently being voiced (see note 1)
session_recordings *2	<object>	available	A list of <recording> tags for the recordings selected to be voiced.
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)

error_message <string> conditional Describes the reason for a failure in text (see "Error Codes" list)

Notes:



The details of the recording currently being voiced. Only available if the session state is "recording".

A list of <recording> elements containing the details of all the recordings that are currently selected to be voiced in the current session. Refer to "GetRecording" for the details of a <recording> element. **Sample Response:**

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="Recording session status has been retrieved" status="success">
  <session_state>called</session_state>
  <voice_recorder_key>ZV2ZEXHJ4P</voice_recorder_key>
  <recording_list_key>36PONP0XTR</recording_list_key>
  <recording_session_key>V6GUN79S7T</recording_session_key>
</myVoxapi>
```

SetRecording

Updates the values of a Recording instance.

Request Parameters:

Parameter Name	Parameter Value	Type	Command Description
action	"SetRecording"	required	Identifies the "SetRecording" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList instance
recording_key	<string: 10 chars>	required	The identification key for the Recording instance
name	<string>	optional	The name identifier for the new Recording instance
sequence	<integer: auto-increment>	optional	The sequence of the recording within the RecordingList (first = 1)
keep_consecutive	<boolean>	optional	Adjusts the sequence values of all recordings to remain consecutive
meta_data_1	<string>	optional	Arbitrary meta-data associated with the Recording instance
meta_data_2	<string>	optional	Arbitrary meta-data associated with the Recording instance
meta_data_3	<string>	optional	Arbitrary meta-data associated with the Recording instance
access_password	<string>	optional	Password to protect access to the Recording instance (??? = random)
update_password	<string>	optional	Password to protect updates to the Recording instance (??? = random)
delete_password	<string>	optional	Password to protect deleting the Recording instance (??? = random)

Response Data:

Attribute Name	Attribute Value	Type	Parameter Description
----------------	-----------------	------	-----------------------

recording_list_key	<string: 10 chars>	available	The identification key for the RecordingList in which to create the recording
recording_key *	<string: 10 chars>	available	The identification key for the new Recording instance
name *	<string>	conditional	The name identifier for the recording
meta_data_1 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
meta_data_2 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
meta_data_3 *	<string>	conditional	Arbitrary meta-data associated with the new Recording instance
recording_state *	"unrecorded" "recording" "recorded"	available	Indicates the current state of the recording: "unrecorded" initially, "recordings" while being recorded, and "recorded" once the recording process has been completed.
sequence *	<integer: auto-increment>	available	The sequence of the recording within the RecordingList (first = 1)
length *	<integer>	available	The length of the recording in seconds (0 if unrecorded)
play_key *	<string: 20 chars>	available	The unique identification key used to play the Recording
has_wav_file *	<boolean>	available	Indicates whether this recording has a WAV file
wav_url *	<string: url>	conditional	The URL to play the WAV recording file, if one is available
has_mp3_file *	<boolean>	available	Indicates whether this recording has an MP3 file
mp3_url *	<string: url>	conditional	The URL to play the MP3 recording file, if one is available
date_created *	<date-time>	available	The date and time when recording was initially created
last_updated *	<date-time>	available	The date and time when recording was last updated
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

* Attributes of a "recording" parent attribute, used to describe a single Recording instance.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="Recording 62B3S1lLBB updated" status="success">
  <voice_recorder_key>AOAIG2UUVO</voice_recorder_key>
  <recording_list_key>WWJKFT0A5V</recording_list_key>
  <recording>
    <recording_key>62B3S1lLBB</recording_key>
    <recording_state>unrecorded</recording_state>
    <name>My Recording</name>
    <sequence>1</sequence>
    <is_in_session>>false</is_in_session>
    <play_key>HJ4SF86FE666EL9U73GE</play_key>
    <has_wav_file>>false</has_wav_file>
    <has_mp3_file>>true</has_mp3_file>
    <mp3_url>http://api.myVox.com/HJ4SF86FE666EL9U73GE/recording.mp3</mp3_url>
    <date_created>2007-12-12 22:32:59 EST</date_created>
    <last_updated>2007-12-12 22:32:59 EST</last_updated>
    <last_accessed>2007-12-12 22:32:59 EST</last_accessed>
  </recording>
</myVoxapi>
```

SetRecordingList

Updates the values of a RecordingList instance.

Request Parameters:

Parameter Name	Parameter Value	Type	Command Description
----------------	-----------------	------	---------------------

action	"SetRecordingList"	required	Identifies the "SetRecordingList" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList instance
name	<string>	optional	The name identifier for the RecordingsList instance
meta_data_1	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_2	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
meta_data_3	<string>	optional	Arbitrary meta-data associated with the new RecordingList instance
access_password	<string>	optional	Password to protect access to the RecordingList instance (??? = random)
update_password	<string>	optional	Password to protect updates to the RecordingList instance (??? = random)
delete_password	<string>	optional	Password to protect deleting the RecordingList instance (??? = random)
max_recordings	<integer>	optional	The maximum number of Recordings allowed for this RecordingList

Response Data:

<i>Attribute Name</i>	<i>Attribute Value</i>	<i>Type</i>	<i>Parameter Description</i>
voice_recorder_key	<string: 10 chars>	available	The identification key for the VoiceRecorder application instance
recording_list_key *	<string: 10 chars>	available	The identification key for the RecordingList instance
name *	<string>	conditional	The name identifier for the RecordingList instance
meta_data_1 *	<string>	available	Arbitrary meta-data associated with the new RecordingList instance
meta_data_2 *	<string>	available	Arbitrary meta-data associated with the new RecordingList instance
meta_data_3 *	<string>	available	Arbitrary meta-data associated with the new RecordingList instance
recording_count	<integer>	available	The number of recordings currently associated with this RecordingList
max_recordings *	<integer>	conditional	The maximum number of Recordings allowed for this RecordingList
created *	<date-time>	available	The date and time when RecordingList was initially created
access_protected *	<boolean>	available	Indicates whether RecordingList access is password protected
update_protected *	<boolean>	available	Indicates whether RecordingList updates is password protected
delete_protected *	<boolean>	available	Indicates whether RecordingList deleting is password protected
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

* Attributes of an "recording_list" parent attribute, used to describe a single RecordingList instance.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="Recording list MyCustomKey updated" status="success">
  <voice_recorder_key>AOAIG2UUV0</voice_recorder_key>
  <recording_list>
    <recording_list_key>MyCustomKey</recording_list_key>
    <name>Test Recording List</name>
    <meta_data_1>xxx</meta_data_1>
    <meta_data_2>yyy</meta_data_2>
    <meta_data_3>zzz</meta_data_3>
    <access_protected>>false</access_protected>
    <update_protected>>false</update_protected>
  </recording_list>
</myVoxapi>
```

```

    <delete_protected>>false</delete_protected>
    <recording_count>3</recording_count>
    <max_recordings>5</max_recordings>
    <date_created>2007-12-12 20:21:43 EST</date_created>
    <last_updated>2007-12-12 20:21:43 EST</last_updated>
    <last_accessed>2007-12-12 20:21:43 EST</last_accessed>
  </recording_list>
</myVoxapi>

```

StartRecordingSession

Used to initiate a new recording session. The parameters identify the application instance which the recordings are to be associated as well as the maximum number of recordings that can be created in the session. The return values indicate the phone number to call, a short PIN to access the recording session, and the URL to poll for status. If a 'caller_id' value is specified it is used as the PIN value caller's enter to identify their session and if 'Caller ID' info is available, their session will be looked up automatically and the prompted PIN entry step is skipped. The 'overwrite' parameter is used with the 'caller_id' to delete an existing session keyed under the Caller ID value, else an error is returned indicating that a session keyed to the Caller ID is currently in use.

Request Parameters:

Parameter Name	Parameter Value	Type	Command Description
action	"StartRecordingSession"	required	Identifies the "StartRecordingSession" command
reply	"xml" (default) "json"	optional	Specifies the format of the response messages
caller_id	<string: 10 digits>	optional	Used to automatically identify the caller's session by their Caller ID
overwrite	"true" "false" (efault)	optional	Used with 'caller_id' to overwrite an existing session with the same Caller ID
password	<string>	optional	The password required to execute this command, if one has been set.
voice_recorder_key	<string: 10 chars>	required	The identification key for the VoiceRecorder application instance
recording_list_key	<string: 10 chars>	required	The identification key for the RecordingList instance
recording_keys	<string>	required	A delimited list of recording_key values to select the recordings to be voiced

Response Data:

Attribute Name	Attribute Value	Type	Parameter Description
recording_session_key	<string: 10 chars>	available	Identifies a new recording session tied to the RecordingList instance.
ivr_pin	<string: 4 digits>	available	The code caller's enter on the phone system to identify the session
phone_number *1	<string: 10 digits>	available	The phone number the user calls to voice the recordings
session_recordings *2	<object>	available	A list of <recording> tags for the recordings selected to be voiced.
excluded_recordings *3	<object>	optional	A list of <recording> tags for the recordings excluded from the session.
status	"success" "failed"	available	Indicates whether the requested command succeeded or failed
error_code	<integer>	conditional	Identifies the reason for a failure by error code (see "Error Codes" list)
error_message	<string>	conditional	Describes the reason for a failure in text (see "Error Codes" list)

Notes:

1. The "phone_number" tag may be repeated if there is more than one phone number associated with the Voice Recorder application. Each one includes "phone_number_key", "phone_number", "display_number", and "country" values.
2. A list of the recordings that are currently selected to be voiced. This may include fewer recordings than those requested in the StartRecordingSession command if one or more requested recordings are

already in use by another recording session. Refer to the "GetRecording" command for the details of a <recording> element.

3. A list of recordings excluded from the current session because they are currently being voiced in a different recording session. Refer to "GetRecording" for the details of a <recording> element.

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<myVoxapi message="A new recording session has been started" status="success">
  <recording_session_key>V6GUN79S7T</recording_session_key>
  <ivr_pin>785754</ivr_pin>
  <phone_number_list>
    <phone_number_info>
      <phone_number_key>IG2UUAOAVO</phone_number_key>
      <phone_number>5125381945</phone_number>
      <display_number>512-538-1945</display_number>
      <country>US</country>
    </phone_number_info>
  </phone_number_list>
  <session_recordings>
    <recording>
      <recording_key>26069U4CAN</recording_key>
      <recording_state>unrecorded</recording_state>
      <name>Recording 1</name>
      <sequence>1</sequence>
      <is_in_session>true</is_in_session>
      <has_wav_file>>false</has_wav_file>
      <has_mp3_file>>false</has_mp3_file>
      <date_created>2007-12-13 01:06:47 EST</date_created>
      <last_updated>2007-12-13 01:11:26 EST</last_updated>
      <last_accessed>2007-12-13 01:06:47 EST</last_accessed>
    </recording>
    <recording>
      <recording_key>Z4JK68NSE1</recording_key>
      <recording_state>unrecorded</recording_state>
      <name>Recording 2</name>
      <sequence>2</sequence>
      <is_in_session>true</is_in_session>
      <has_wav_file>>false</has_wav_file>
      <has_mp3_file>>false</has_mp3_file>
      <date_created>2007-12-13 01:07:09 EST</date_created>
      <last_updated>2007-12-13 01:11:27 EST</last_updated>
      <last_accessed>2007-12-13 01:07:09 EST</last_accessed>
    </recording>
  </session_recordings>
</myVoxapi>
```

API Error Codes

The myVox API provides extensive error reporting on all API commands. If a command error is detected an error response is generated with four attributes: a "status" value set to "failed", an "error_code" value set to the numerical code for the error, an "error_message" value with a general text description of the error, and an

“error_details” value containing additional information specific to the error instance. The following tables lists all the errors that can be generated by the API.

General System Errors

<i>Error Code</i>	<i>Error Message</i>	<i>Error Description</i>
1000	Unknown system error	Generated when an unexpected error occurs
1001	Authentication failed	Attempt to execute a protected command without being logged in
1002	Unrecognized API command	The 'action' parameter is not a recognized API command
1003	Parameter error	Generated if a required parameter is missing or if a specified parameter value does not meet the constraint requirements for the command. Examples of constraint errors are alpha characters where only digits are expected, string length that are not within the required length range, numbers not within the required numeric range, etc.

Developer Account errors

<i>Error Code</i>	<i>Error Message</i>	<i>Error Description</i>
1100	Unknown developer error	An unexpected error occurred while processing a Developer command
1101	Create developer error	CreateDeveloper command failed. Error details indicates the specific reason.
1102	Get developer error	GetDeveloper command failed. Error details indicates the specific reason.
1103	Set developer error	SetDeveloper command failed. Error details indicates the specific reason.
1104	Delete developer error	DeleteDeveloper command failed. Error details indicates the specific reason.
1105	Username is not available	CreateDeveloper command failed because the requested username is already taken.
1106	Changed password error	ChangePassword command failed. Error details indicates the specific reason.
1107	Email password error	EmailPassword command failed. Error details indicates the specific reason.
1108	Login error	LoginDeveloper command failed. Error details indicates the specific reason.

Voice Recorder errors

<i>Error Code</i>	<i>Error Message</i>	<i>Error Description</i>
1200	Unknown voice recorder error	An unexpected error occurred while processing a VoiceRecorder command
1201	Create voice recorder error	CreateVoiceRecorder command failed. Error details indicates the specific reason.
1202	Get voice recorder error	GetVoiceRecorder command failed. Error details indicates the specific reason.
1203	Set voice recorder error	SetVoiceRecorder command failed. Error details indicates the specific reason.
1204	Delete voice recorder error	DeleteVoiceRecorder command failed. Error details indicates the specific reason.
1205	Play prompt error	PlayPrompt command failed. Error details indicates the specific reason.
1206	Set prompt error	SetPrompt command failed. Error details indicates the specific reason.
1207	Upload prompt error	UploadPRompt command failed. Error details indicates the specific reason.
1208	Unauthorized voice recorder error	Generated when one developer attempts to modify another developer's voice recorder.

Recording List errors

<i>Error Code</i>	<i>Error Message</i>	<i>Error Description</i>
1300	Unknown recording list error	An unexpected error occurred while processing a recordingList command
1301	Create recording list error	CreateRecordingList command failed. Error details indicates the specific reason.
1302	Get recording list error	GetRecordingList command failed. Error details indicates the specific reason.

1303	Set recording list error	SetRecordingList command failed. Error details indicates the specific reason.
1304	Delete recording list error	DeleteRecordingList command failed. Error details indicates the specific reason.
1305	Recording list key error	CreateRecordingList failed because the custom key specified is already in use.

Recording errors

<i>Error Code</i>	<i>Error Message</i>	<i>Error Description</i>
1400	Unknown recording error	An unexpected error occurred while processing a Recording command
1401	Create recording error	CreateRecording command failed. Error details indicates the specific reason.
1402	Get recording error	GetRecording command failed. Error details indicates the specific reason.
1403	Set recording error	SetRecording command failed. Error details indicates the specific reason.
1404	Delete recording error	DeleteRecording command failed. Error details indicates the specific reason.
1405	Play recording error	PlayRecording command failed. Error details indicates the specific reason.
1406	Recording key error	CreateRecording command failed because the custom key specified is already in use.

Misc errors

<i>Error Code</i>	<i>Error Message</i>	<i>Error Description</i>
1500	Unknown recording session error	An unexpected error occurred while processing a RecordingSession command
1501	Start recording session error	StartRecordingSession command failed. Error details indicates the specific reason.
1502	Get session status error	GetSessionStatus command failed. Error details indicates the specific reason.
1503	Expire recording session error	ExpireRecordingSession command failed. Error details indicates the specific reason.