

*Powered by Universal Speech Solutions LLC*



# Watson SS Plugin

## Usage Guide

---

Revision: 1

Created: June 28, 2018

Last updated: June 28, 2018

Author: Arsen Chaloyan

# Table of Contents

1	Overview.....	3
1.1	Installation .....	3
1.2	Applicable Versions.....	3
2	Supported Features.....	4
2.1	MRCP Methods .....	4
2.2	MRCP Events .....	4
2.3	MRCP Header Fields.....	4
2.4	Speech Data .....	4
3	Supported Voices .....	5
4	Configuration Format.....	6
4.1	Document.....	6
4.2	Synthesis Settings .....	7
4.3	Monitoring Agent .....	8
4.4	Usage Change Handler .....	8
4.5	Usage Refresh Handler .....	9
4.6	License Server.....	9
5	Configuration Steps .....	12
5.1	Using Default Configuration.....	12
5.2	Specifying Synthesis Language .....	12
5.3	Specifying Sampling Rate.....	12
5.4	Specifying Voice Parameters .....	12
5.5	Specifying Prosody Parameters .....	12
5.6	Specifying Speech Data .....	13
6	Monitoring Usage Details .....	14
7	Usage Examples.....	15
7.1	SSML.....	15
7.2	Plain Text.....	15
8	Sequence Diagram .....	17
9	References.....	18
9.1	IBM Watson.....	18
9.2	Specifications.....	18

# 1 Overview

This guide describes how to configure and use the IBM Watson Speech Synthesis (SS) plugin to the UniMRCP server. The document is intended for users having a certain knowledge of IBM Watson Text-to-Speech API and UniMRCP.



## 1.1 Installation

For installation instructions, use one of the guides below.

- RPM Package Installation (Red Hat / Cent OS)
- Deb Package Installation (Debian / Ubuntu)

## 1.2 Applicable Versions

Instructions provided in this guide are applicable to the following versions.



UniMRCP 1.5.0 and above

UniMRCP Watson SS Plugin 1.0.0 and above

# 2 Supported Features

This is a brief check list of the features currently supported by the UniMRCP server running with the Watson SS plugin.

## 2.1 MRCP Methods

- ✓ SPEAK
- ✓ STOP
- ✓ PAUSE
- ✓ RESUME
- ✓ BARGE-IN-OCCURRED
- ✓ SET-PARAMS
- ✓ GET-PARAMS

## 2.2 MRCP Events

- ✓ SPEECH-MARKER
- ✓ SPEAK-COMPLETE

## 2.3 MRCP Header Fields

- ✓ Kill-On-Barge-In
- ✓ Completion-Cause
- ✓ Voice-Gender
- ✓ Voice-Name
- ✓ Prosody-Rate
- ✓ Prosody-Volume
- ✓ Speech-Language

## 2.4 Speech Data

- ✓ Plain text (text/plain)
- ✓ SSML (application/ssml+xml or application/synthesis+ssml)

# 3 Supported Voices

All the voices supported by IBM Watson Text-to-Speech API are listed in the following page:

<https://www.ibm.com/watson/developercloud/text-to-speech/api/v1/curl.html?curl#get-voice>

# 4 Configuration Format

The configuration file of the Watson SS plugin is located in `/opt/unimrcp/conf/umswatsonss.xml`. The configuration file is written in XML.

## 4.1 Document

The root element of the XML document must be `<umswatsonss>`.

### Attributes

Name	Unit	Description
<b>license-file</b>	File path	Specifies the license file. File name may include patterns containing '*' sign. If multiple files match the pattern, the most recent one gets used.
<b>credentials-file</b>	File path	Specifies the IBM Watson credentials file to use. File name may include patterns containing '*' sign. If multiple files match the pattern, the most recent one gets used.

### Parent

None.

### Children

Name	Unit	Description
<b>&lt;synth-settings&gt;</b>	String	Specifies synthesis parameters.
<b>&lt;monitoring-agent&gt;</b>	String	Specifies parameters of the monitoring manager.
<b>&lt;license-server&gt;</b>	String	Specifies parameters used to connect to the license server. The use of the license server is optional.

### Example

This is an example of a bare document.

```
<umswatsonss license-file="umswatsonss_*.lic" credentials-file="watson.credentials">
```

```
</ umswatsonss>
```

## 4.2 Synthesis Settings

This element specifies synthesis parameters.

### Attributes

Name	Unit	Description
<b>language</b>	String	Specifies the default language to use, if not set by the client.
<b>voice-name</b>	String	Specifies the default voice name. Can be overridden by client.
<b>bypass-ssml</b>	Boolean	Specifies whether transparently bypass or normalize received SSML content before sending it to the speech service for synthesis.
<b>escape-characters</b>	Boolean	Specifies whether to escape special characters in plain text or SSML content.
<b>auth-validation-period</b>	Integer	Specifies a period in seconds used to re-validate access token based on credentials.

### Parent

```
<umswatsonss>
```

### Children

None.

### Example

This is an example of synthesis parameters.

```
<synth-settings  
  language="en-US"  
  voice-name=""  
  bypass-ssml="false"  
  escape-characters="false"  
  auth-validation-period="480"  
>
```

## 4.3 Monitoring Agent

This element specifies parameters of the monitoring agent.

### Attributes

Name	Unit	Description
<b>refresh-period</b>	Time interval [sec]	Specifies a time interval in seconds used to periodically refresh usage details. See <usage-refresh-handler>.

### Parent

<umswatsonss>

### Children

<usage-change-handler>

<usage-refresh-handler>

### Example

The example below defines a monitoring agent with usage change and refresh handlers.

```
<monitoring-agent refresh-period="60">
  <usage-change-handler>
    <log-usage enable="true" priority="NOTICE"/>
  </usage-change-handler>
  <usage-refresh-handler>
    <dump-channels enable="true" status-file="umswatsonss-channels.status"/>
  </usage-refresh-handler >
</monitoring-agent>
```

## 4.4 Usage Change Handler

This element specifies an event handler called on every usage change.

### Attributes

None.

### Parent



```
<monitoring-agent>
```

### Children

```
<log-usage>  
<update-usage>  
<dump-channels>
```

### Example

This is an example of the usage change event handler.

```
<usage-change-handler>  
  <log-usage enable="true" priority="NOTICE"/>  
  <update-usage enable="false" status-file="umswatsonss-usage.status"/>  
  <dump-channels enable="false" status-file="umswatsonss-channels.status"/>  
</usage-change-handler>
```

## 4.5 Usage Refresh Handler

This element specifies an event handler called periodically to update usage details.

### Attributes

None.

### Parent

```
<monitoring-agent>
```

### Children

```
<log-usage>  
<update-usage>  
<dump-channels>
```

### Example

This is an example of the usage change event handler.

```
<usage-refresh-handler>  
  <log-usage enable="true" priority="NOTICE"/>  
  <update-usage enable="false" status-file="umswatsonss-usage.status"/>  
  <dump-channels enable="false" status-file="umswatsonss-channels.status"/>  
</usage-refresh-handler>
```

## 4.6 License Server

This element specifies parameters used to connect to the license server.

## Attributes

Name	Unit	Description
<b>enable</b>	Boolean	Specifies whether the use of license server is enabled or not. If enabled, the license-file attribute is not honored.
<b>server-address</b>	String	Specifies the IP address or host name of the license server.
<b>certificate-file</b>	File path	Specifies the client certificate used to connect to the license server. File name may include patterns containing a '*' sign. If multiple files match the pattern, the most recent one gets used.
<b>ca-file</b>	File path	Specifies the certificate authority used to validate the license server.
<b>channel-count</b>	Integer	Specifies the number of channels to check out from the license server. If not specified or set to 0, either all available channels or a pool of channels will be checked based on the configuration of the license server.

## Parent

<umswatsonss>

## Children

None.

## Example

The example below defines a typical configuration which can be used to connect to a license server located, for example, at 10.0.0.1.

```
<license-server
  enable="true"
  server-address="10.0.0.1"
  certificate-file="unilic_client_*.cert"
  ca-file="unilic_ca.crt"
/>
```

For further reference to the license server, visit

<http://unimrcp.org/licserver>

# 5 Configuration Steps

This section outlines common configuration steps.

## 5.1 Using Default Configuration

The default configuration should be sufficient for the general use.

## 5.2 Specifying Synthesis Language

Synthesis language can be specified by the client per MRCP session by means of the header field *Speech-Language* set in a *SET-PARAMS* or *SPEAK* request, or inline in the SSML data. Otherwise, the parameter *language* set in the configuration file *umswatsonss.xml* is used. The parameter defaults to *en-US*.

## 5.3 Specifying Sampling Rate

Sampling rate is determined based on the SDP negotiation. Refer to the configuration guide of the UniMRCP server on how to specify supported encodings and sampling rates to be used in communication between the client and server. Either 8 or 16 kHz can be used by IBM Watson Text-to-Speech API.

## 5.4 Specifying Voice Parameters

### Global Settings

The default voice name can be specified from the configuration file *umswatsonss.xml* using the *voice-name* attribute of the *synth-settings* element.

### MRCP Header Fields

The voice name and gender can be specified by the MRCP client in *SET-PARAMS* and *SPEAK* requests.

- Voice-Name

This is an optional parameter indicating the name of the voice to use for synthesis.

### SSML Content

The voice name can also be specified using the corresponding attributes of the *voice* element in SSML content. In order to parse and determine the parameter and pass it forward to IBM Watson Text-to-Speech API accordingly, the *bypass-ssml* attribute of the *synth-settings* element must be set to *false* in the configuration file *umswatson.xml*.

## 5.5 Specifying Prosody Parameters

The following prosody parameters can be specified by the MRCP client in *SET-PARAMS* and *SPEAK* requests.

- Prosody-Rate

This is an optional parameter indicating the speaking rate, which can be set to one of the following labels: *x-slow*, *slow*, *medium*, *fast*, *x-fast*, *default*.

- Prosody-Volume

This is an optional parameter indicating the speaking volume, which can be set to one of the following labels: *silent*, *x-soft*, *soft*, *medium*, *loud*, *x-loud*, *default*.

## 5.6 Specifying Speech Data

Speech data can be specified by the MRCP client in *SPEAK* requests using one of the following content types:

- plain/text
- application/ssml+xml (or application/synthesis+ssml)

## 6 Monitoring Usage Details

The number of in-use and total licensed channels can be monitored in several alternate ways. There is a set of actions which can take place on certain events. The behavior is configurable via the element *monitoring-agent*, which contains two event handlers: *usage-change-handler* and *usage-refresh-handler*.

While the *usage-change-handler* is invoked on every acquisition and release of a licensed channel, the *usage-refresh-handler* is invoked periodically on expiration of a timeout specified by the attribute *refresh-period*.

The following actions can be specified for either of the two handlers.

- log-usage

This action logs the number of in-use and total licensed channels. The following is a sample log statement, indicating 0 in-use and 2 total channels.

```
[NOTICE] WSS Usage: 0/2
```

- update-usage

This action writes the number of in-use and total licensed channels to a status file *umswatsonss-usage.status*, located by default in the directory `${UniMRCPIInstallDir}/var/status`. The following is a sample content of the status file.

```
in-use channels: 0  
total channels: 2
```

- dump-channels

This action writes the identifiers of in-use channels to a status file *umswatsonss-channels.status*, located by default in the directory `${UniMRCPIInstallDir}/var/status`.

# 7 Usage Examples

## 7.1 SSML

This examples demonstrates how to perform speech synthesis by using a SPEAK request with an SSML content.

C->S:

```
MRCP/2.0 309 SPEAK 1
Channel-Identifier: 4dde51f37d1a9546@speechsynth
Content-Type: application/ssml+xml
Voice-Age: 28
Content-Length: 163

<?xml version="1.0"?>
<speak version="1.0" xml:lang="en-US" xmlns="http://www.w3.org/2001/10/synthesis">
  <p>
    <s>Welcome to Uni MRCP.</s>
  </p>
</speak>
```

S->C:

```
MRCP/2.0 83 1 200 IN-PROGRESS
Channel-Identifier: 4dde51f37d1a9546@speechsynth
```

S->C:

```
MRCP/2.0 122 SPEAK-COMplete 1 COMPLETE
Channel-Identifier: 4dde51f37d1a9546@speechsynth
Completion-Cause: 000 normal
```

## 7.2 Plain Text

This examples demonstrates how to perform speech synthesis by using a SPEAK request with a plain text content.

C->S:

MRCP/2.0 155 SPEAK 1  
Channel-Identifier: 85667d0efbf95345@speechsynth  
Content-Type: text/plain  
Voice-Age: 28  
Content-Length: 20

Welcome to Uni MRCP.

S->C:

MRCP/2.0 83 1 200 IN-PROGRESS  
Channel-Identifier: 85667d0efbf95345@speechsynth

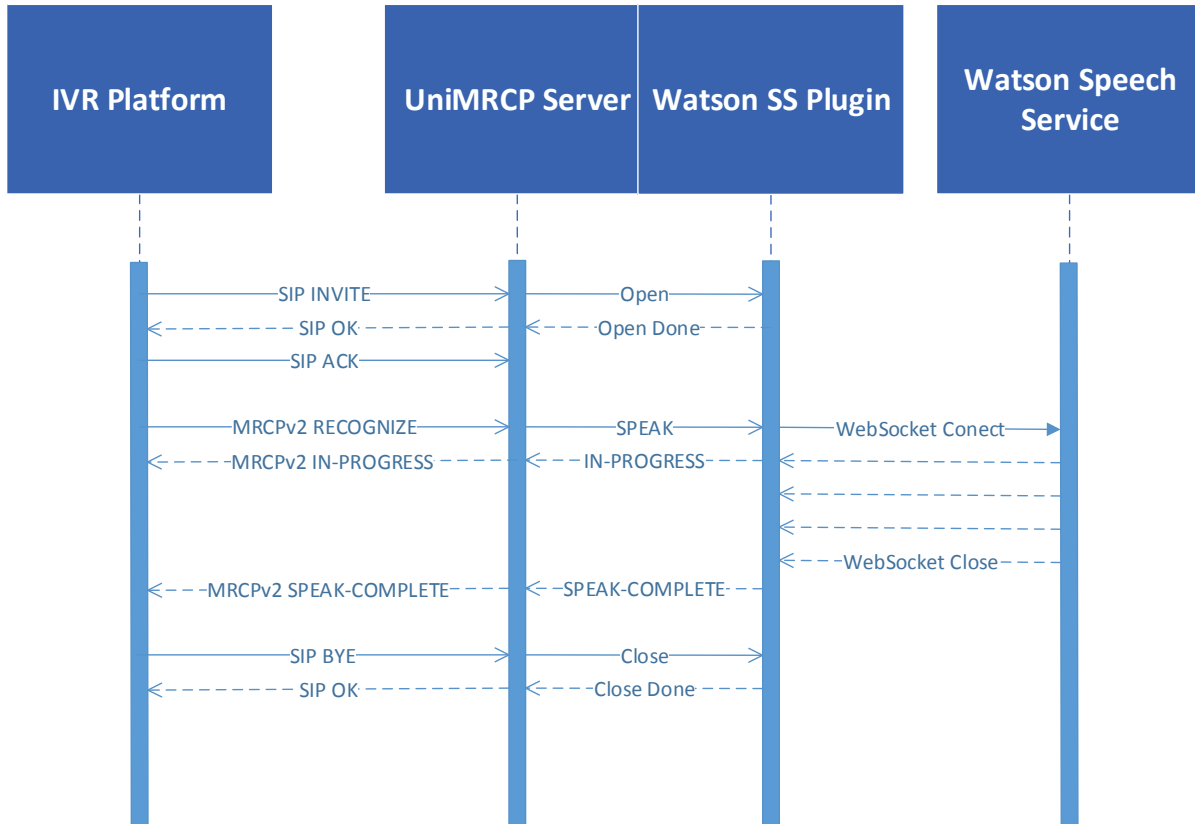
S->C:

MRCP/2.0 122 SPEAK-COMplete 1 COMPLETE  
Channel-Identifier: 85667d0efbf95345@speechsynth  
Completion-Cause: 000 normal



# 8 Sequence Diagram

The following sequence diagram outlines common interactions between all the main components involved in a typical synthesis session performed over MRCPv2.



# 9 References

## 9.1 IBM Watson

- [Text to Speech API](#)
- [Credentials](#)

## 9.2 Specifications

- [Speech Synthesizer Resource](#)
- [SSML](#)