The SSML Specification of ReadSpeaker VTAPI

The software described in this manual is furnished under a valid license agreement or nondisclosure agreement with ReadSpeaker. The software may be used only in accordance with the terms of the agreement.

DISCLAIMER.,. THIS MANUAL IS DISTRIBUTED AS-IS. ReadSpeaker SHALL NOT BE LIABLE FOR TECHNICAL OR
EDITORIAL ERRORS OR OMISSIONS CONTAINED HEREIN; NOR FOR INCIDENTAL OR CONSEQUENTIAL
DAMAGES RESULTING FROM THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL. THE
INFORMATION IN THIS MANUAL IS SUBJECT TO CHANGE WITHOUT ANY PRIOR NOTICE.

Revision history

date	description
2017-07-01	created
2020-06-10	Updated template format.

Table Of Contents

1. Overview

ReadSpeaker text-to-speech product by ReadSpeaker is the leading text-to-speech solution, which synthesizes an arbitrary input text to produce synthetic speech that is natural, human-like and clear by accurately analyzing and processing linguistic information such as phonetic content, sentence structure, prosody and pauses

As the proud leader of high quality, high performance voice engines in Korean, English, Chinese , Japanese and EU languages that are able to synthesize variety of text and symbols in best quality voice, ReadSpeaker is widely used in industries including telecommunications, finance service and public welfare. With the precise pronunciation, natural human-like synthesized voice, variety of supported platforms, high quality service, stability, fast synthesis speed and variety of voice types and language, ReadSpeaker proudly guarantees that ReadSpeaker is the best speech synthesis product in the world.

This document explains about SSML (Speech Synthesis Markup Language) with a wide range of examples on how to use it. The language is one of ways to use ReadSpeaker VTAPI (VT Application Program Interface) necessary to develop a program with multiple ReadSpeaker engines in use.

2. References

Some functions described in the below documents are also stated in this document as available.

```
Speech Synthesis Markup Language (SSML) Version 1.1 W3C Recommendation 7
September 2010
Pronunciation Lexicon Specification (PLS) Version 1.0 W3C Recommendation 14
October 2008
```

3. Elements and Attributes

3.1 speak

<speak> supports the below attributes as it is the root element in the SSML.

```
version : As a mandatory attribute which defines a SSML spec, an allowed value
is 1.1 (But, 1.0 also
available no matter what the spec is.)
xml:lang : As an optional attribute to the first language being synthesized, the
language and voice would
be changed if they are not identical to the other ones set to
VTAPI_SetEngineHandle(). It is expressed with
'en', or 'en-US' for example.
xml:base : It is an optional attribute to representing a document base URI
related to audio or lexicon.
(Available on Windows and Linux only)
startmark : It is an optional attribute to representing a starting position to
render in the SSML sentence.
endmark : It is an optional attribute to representing a end position of the
rendering in the SSML
sentence.
Other elements are left out.
```

```
<speak version="1.1">
This is a sample.
</speak>
```

3.2 lexicon

<lexicon> element supports the two document formats, PLS (Pronunciation Lexicon Specification 1.0) and CSV
(User-Dictionary of ReadSpeaker).

```
uri : It is a mandatory attribute to designate a document to support. An URI is
designated if there is the
xml:base which is a child element of <speak>. What <lookup> element holds
determines what part of the
PLS document would be referred while it ranges all over a sentence in case of
the CSV document without
<lookup> element. It is not recommendable that those two are used at the same
time.
xml:id : It is a mandatory attribute but, it is not available on a CSV document.
As an id to the PLS
document, <lookup> element is referred to render.
Other elements are left out.
```

3.3 lookup

<lookup> element must have a ref attribute. With it, it renders the PLS document designated with xml:id from
<lexicon>.

```
ref : It is a mandatory attribute and refers an id value to the PLS document for
rendering.
```

```
<speak version="1.1" xml:lang="en-US">
This is a sample.
</speak>
```

```
<speak version="1.1" xml:base="http://128.0.0.1">
<audio src="sample.wav">This is a sample.</audio>
</speak>
```

```
<speak version="1.1" startmark="mark1" endmark="mark2">
This is a <mark name="mark1"/>sample<mark name="mark2"/> for testing.
</speak>
```

```
<speak version="1.1" xml:base="http://128.0.0.1/lexicons">
<lexicon uri="lexicon1.pls" xml:id="pls1"/>
<lookup ref="pls1">readspeaker demo application.</lookup>
readspeaker demo application.
</speak>
```

```
<speak version="1.1">
<lexicon uri="C:\Users\tester\userdictionarie\userdict.csv" xml:id="pls1"/>
readspeaker demo application.
readspeaker demo application.
</speak>
```

3.4 meta

<meta> not available

3.5 metadata

<metadata> not available

3.6 p, s, token, w

<p> element represents a paragraph. <s> element represents a sentence. <token> element is the same as
<w> element to express segmentation of a word. These elements support the below optional attributes.

```
xml:lang : It changes a language if it is either included or the language is
different from what is previously
defined.
role : It is available only in <token> or <w>. the rendering is in progress in
association with a <lexeme>
role which is defined when <lexicon> is used.
Other elements are left out.
```

```
<speak version="1.1" xml:base="C:\Users\tester\lexicons">
<lexicon uri="lexicon1.pls" xml:id="pls1"/>
<lexicon uri="lexicon2.pls" xml:id="pls2"/>
<lookup ref="pls1">readspeaker demo application.</lookup>
readspeaker demo application.
<lookup ref="pls2">readspeaker demo application.</lookup>
</speak>
```

```
<speak version="1.1" xml:base="http://128.0.0.1/userdictionaries">
<lexicon uri="userdict.csv" xml:id="pls1"/>
readspeaker demo application.
readspeaker demo application.
</speak>
```

```
<speak version="1.1">
<lexicon uri="http://128.0.0.1/lexicons/lexicon1.pls" xml:id="pls1"/>
<lookup ref="pls1">readspeaker demo application.</lookup>
readspeaker demo application.
</speak>
```

```
<speak version="1.1">
<p>
<s>Text within a sentence element.</s>
<s>Text within a <w>sentence</w> element.</s>
<s>More text in <token>another</token> sentence.</s>
</p>
</speak>
```

3.7 say-as

<say-as> element allows to indicate information on the type of text construct contained within the element
and specifies the level of detail for rendering the contained text. To this end, it supports the below attributes.
When used, either interpret-as attribute or type attribute should be there.

```
interpret-as : There are multiple options available: (vxml:)boolean,
(vxml:)digits, (vxml:)currency,
(vxml:)number, (vxml:)phone, (vxml:)date, (vxml:)time, characters, cardinal,
ordinal, telephone to select.
format : As an optional attribute, a value changes, depending on the interpret-
as.
detail : As an optional attribute, a value changes, depending on the interpret-
as.
type : As a custom attribute, the interpret-as can be bypassed. it renders by
defining a duration format.
(duration(:hms), duration:hm, duration:ms, duration:h, duration:m, duration:s
are available.)
```

```
<speak version="1.1">
<say-as interpret-as="boolean">true</say-as>
<say-as interpret-as="vxml:boolean">false</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="digits">12345</say-as>
<say-as interpret-as="vxml:digits">12345</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="currency">USD45.30</say-as>
<say-as interpret-as="vxml:currency">USD45.30</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="number">12345</say-as>
<say-as interpret-as="vxml:number">12345</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="phone">9998881000</say-as>
<say-as interpret-as="vxml:phone">9998881000</say-as>
<say-as interpret-as="telephone">9998881000</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="date" format="ymd">2005/07/20</say-as>
<say-as interpret-as="vxml:date">20050720</say-as>
<say-as interpret-as="vxml:date">200507??</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="time" format="hms24">18:10:53</say-as>
<say-as interpret-as="vxml:time">0600a</say-as>
<say-as interpret-as="vxml:time">0600p</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="characters">Hello</say-as>
</speak>
```

```
<speak version="1.1">
Super Bowl <say-as interpret-as="cardinal">49</say-as>
</speak>
```

```
<speak version="1.1">
<say-as interpret-as="ordinal">2</say-as>
</speak>
```

3.8 phoneme

<phoneme> supports the below attributes as it renders a phonemic/phonetic pronunciation for the contained text.

```
ph : As a mandatory attribute, it provides an alphabet value for pronunciation.
(Either IPA format or the
other one including Unicode symbols is available.)
alphabet : Multiple values are available: ipa, x-worldbet, x-sampa, x-tasampa,
x-sapi, x-cmu, x-ntsampa,
sapi, cmu, x-pentax, x-pinyin to select. IPA is generally used as default if not
selected.
type : It is provided as an optional attribute for some engines.
```

```
<speak version="1.1" xml:lang="en-US">
<phoneme alphabet="ipa" ph="t&#x259;mei&#x325;&#x27E;ou&#x325;">tomato</phoneme>
<!-- This is an example of IPA using character entities -->
<!-- Because many platform/browser/text editor combinations do not
correctly cut and paste Unicode text, this example uses the entity
escape versions of the IPA characters. Normally, one would directly
use the UTF-8 representation of these symbols: "t?mei??ou?". -->
</speak>
```

3.9 sub

<sub> element is employed to indicate that the text in the alias attribute value replaces the contained text for
pronunciation to render.

```
alias : As a mandatory attribute, it replaces the contained text of <sub>
element.
```

```
<speak version="1.1">
<sub alias="World Wide Web Consortium">W3C</sub>
<!-- World Wide Web Consortium -->
</speak>
```

3.10 lang

<lang> element supports the below attribute as it changes a language applied to the text for rendering.

```
xml:lang : As a mandatory attribute, the language changes if either the element
included or the value is
different from the language of the previously defined element.
Other elements are left out.
```

```
<speak version="1.1" xml:lang="en-US">
The French word for cat is <w xml:lang="fr">chat</w>.
He prefers to eat pasta that is <lang xml:lang="it-IT">al dente</lang>.
</speak>
```

3.11 voice

<voice> supports the below attributes as it changes the language and voice applied to the text for
rendering.

```
name : An attribute to change a speaker
gender : An attribute to change the speaker gender
languages : An attribute to change the language
Other elements are left out.
```

```
<speak version="1.1" xml:lang="en-US">
<voice name="James">James voice here.</voice>.
<voice gender="female">A female voice here.</voice>.
<voice languages="ja-JP">これは日本語の音色です。</voice>.
</speak>
```

3.12 emphasis

<emphasis> element supports the below attribute as it emphasizes the text for rendering.

```
level : There are multiple options available: strong, moderate, none, reduced to
select. It is an optional so
that 'moderate' is applied as default if there is no value to the attribute.
```

```
<speak version="1.1">
That is a <emphasis> big </emphasis> car!
That is a <emphasis level="strong"> huge </emphasis> bank account!
</speak>
```

3.13 break

<break> element supports the below attributes as it puts a break between the text.

```
strength : There are multiple options available: none, x-weak, weak, medium,
strong, x-strong to select.
time : An integer value is available for 's' and 'ms'.
```

```
<speak version="1.1">
Take a deep breath <break/>
then continue.
Press 1 or wait for the tone. <break time="3s"/>
I didn't hear you! <break strength="weak"/> Please repeat.
</speak>
```

3.14 prosody

<prosody> element supports the below attributes as it permits control of the pitch, speaking rate and volume
of the speech output.

```
pitch : It controls the level of the pitch with the range from 50 to 200(%).
+10(%), -10(%) or other levels
relative to the default (100) are available. Also, multiple options are
available: x-high, high, medium, low,
x-low, default to select.
```

```
rate : It controls the speaking rate with the range from 50 to 400(%). +10(%),
-10(%) or other levels
relative to the default (100) are available. Also, multiple options are
available: x-fast, fast, medium, slow,
x-slow, default to select.
volume : It controls the text volume with the range from 0 to 500. +10(db),
-10(db) or other levels relative
to the default (100) are available. Also, multiple options are available: x-
loud, loud, medium, soft, x-soft,
silent, default to select.
Other elements are left out.
```

```
<speak version="1.1">
<s>I am speaking this at the default pitch, rate, and volume for this voice.</s>
```

```
<s><prosody pitch="200%">I am speaking this at approximately twice
pitch</prosody></s>
<s><prosody pitch="+100">I am speaking this at approximately twice
pitch</prosody></s>
```

```
<s>The price of XYZ is <prosody rate="50%">$45</prosody></s>
<s>The price of XYZ is <prosody rate="-50%">$45</prosody></s>
```

```
<s><prosody volume="200%">
I am speaking this at approximately twice the original signal amplitude.
```

```
</prosody></s>
<s><prosody volume="+100">
I am speaking this at approximately twice the original signal amplitude.
</prosody></s>
<s><prosody volume="+6dB">
I am speaking this at approximately twice the original signal amplitude.
</prosody></s>
```

```
<s><prosody volume="50%">
I am speaking this at approximately half the original signal amplitude.
</prosody></s>
<s><prosody volume="-50">
I am speaking this at approximately half the original signal amplitude.
</prosody></s>
<s><prosody volume="-6dB">
I am speaking this at approximately half the original signal amplitude.
</prosody></s>
</speak>
```

3.15 audio

<**audio**> element supports the below attributes as it makes it possible to insert the recorded audio files (wav or pcm format)

```
src : As a mandatory attribute to designate what audio to insert, it designates the URI information related
to the attribute for xml:base of <speak> if the attribute is there.
fetchtimeout : It is the timeout for fetches. it is set with 10000(ms) as default.
mode : It is a custom attribute. If it is set as background, the audio can be mixed up with the text inside
<audio> element.
Other elements are left out.
```

```
<speak version="1.1" xml:base="http://128.0.0.1/audio">
<audio src="fisrt.wav" mode="background">This audio is mixed with this text.</audio>
<audio src="second.wav">This audio is second audio.</audio>
<audio src="third.pcm" fetchtimeout="3000">This audio is not exist.</audio>
</speak>
```

```
<speak version="1.1">
<audio src="http://128.0.0.1/audio/fisrt.wav">This audio is first audio.</audio>
</speak>
```

3.16 mark

<**mark**> is an empty element and supports one attribute: name only.

```
name: It represents the name of mark.
```

```
<speak version="1.1">
Hello <mark name="here"/> world.
</speak>
```

3.17 desc

<**desc**> not available.