

Open APIs
for Open
Minds

All the Cool Kids are using the Cloud

Jason Fox
FIWARE Foundation



Text-to-Speech Plug-In for DITA-OT

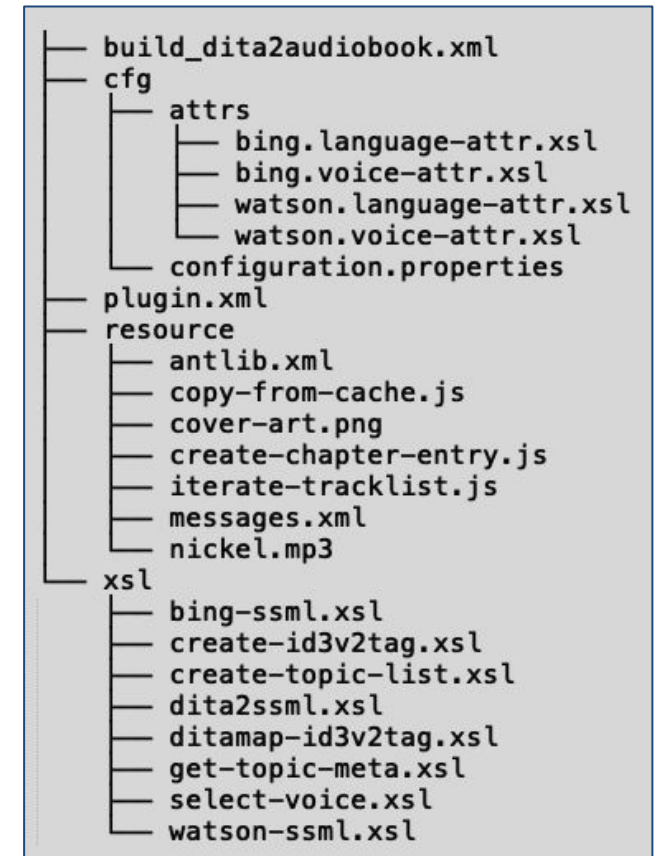
● XSLT 95.3%

● JavaScript 4.7%

- **Transform:** build-init plus custom XSL Transform
- Five Steps:
 - a. DITA to **SSML** + MetaData - Standard XSLT transform
 - b. SSML to MP3 files using a Web Service
 - c. Add MetaData to each MP3 - ffmpeg
 - d. Generate Chapter Listing - Standard XSLT transform
 - e. Merge MP3 to AudioBook - ffmpeg
- Switch Web Services (Watson/Bing/Dummy) using Params

```
<dummy-text-to-speech if:set="audiobook.dummy" out="@{out}"/>
<watson-text-to-speech if:set="audiobook.watson" file="@{src}"
  out="@{out}" voice="@{voice}" mp3="@{mp3}"/>
<bing-text-to-speech if:set="audiobook.bing" file="@{src}"
  out="@{out}" voice="@{voice}" mp3="@{mp3}"/>
```

- **GitHub:** fox.jason.audiobook



Text-to-Speech Plug-In for DITA-OT

● XSLT 95.3%

● JavaScript 4.7%

- For full flexibility when calling a webservice use cUrl

```
<exec taskname="mp3-watson" executable="curl" osfamily="unix" |
  resultproperty="watson.result" outputproperty="watson.out" failOnError="false">
  <arg value="-X" />
  <arg value="POST" />
  <arg value="-u" />
  <arg value="apikey:${watson.apikey}" />
  <arg value="-H" />
  <arg value="Content-Type: application/json" />
  <arg value="-H" />
  <arg value="Accept: audio/mp3" />
  <arg value="--data" />
  <arg value="{&quot;text&quot;: &quot;${watson.ssml}&quot;}" />
  <arg value="--output" />
  <arg value="@{out}" />
  <arg value="-s" />
  <arg value="${watson.text-to-speech.url}?voice=@{voice}" />
</exec>
```

- **JavaScript** - Dynamic Properties

```
var oldDuration = project.getProperty("audiobook.duration");
project.setProperty("audiobook.duration", Number(oldDuration) + duration);
```

Text-to-Speech Plug-In for DITA-OT

● XSLT 95.3%

● JavaScript 4.7%

- Use XSLT to extract metadata

```
<xslt taskname="get-meta" in="{ssml.srcfile}" out="{meta.temp.file}" style="{dita.plugin.fox.jason.audiobook.dir}/xsl/create-id3v2tag.xsl">
  <param expression="{audiobook.voice}" name="VOICE"/>
  <param expression="@{track}" name="TRACK" />
  <xmlcatalog refid="dita.catalog"/>
</xslt>
```

- Use exec to call ffmpeg (Mp3 to mp3 + metadata, mp3 to m4a)

```
<exec taskname="mp3-meta" executable="ffmpeg" osfamily="unix">
  <arg value="-i"/>
  <arg value="{mp3.temp.file}"/>
  <arg value="-i"/>
  <arg value="{meta.temp.file}"/>
  <arg value="-map_metadata"/>
  <arg value="1"/>
  <arg value="-codec"/>
  <arg value="copy"/>
  <arg value="{mp3.output.file}"/>
  <arg value="-y"/>
</exec>
```



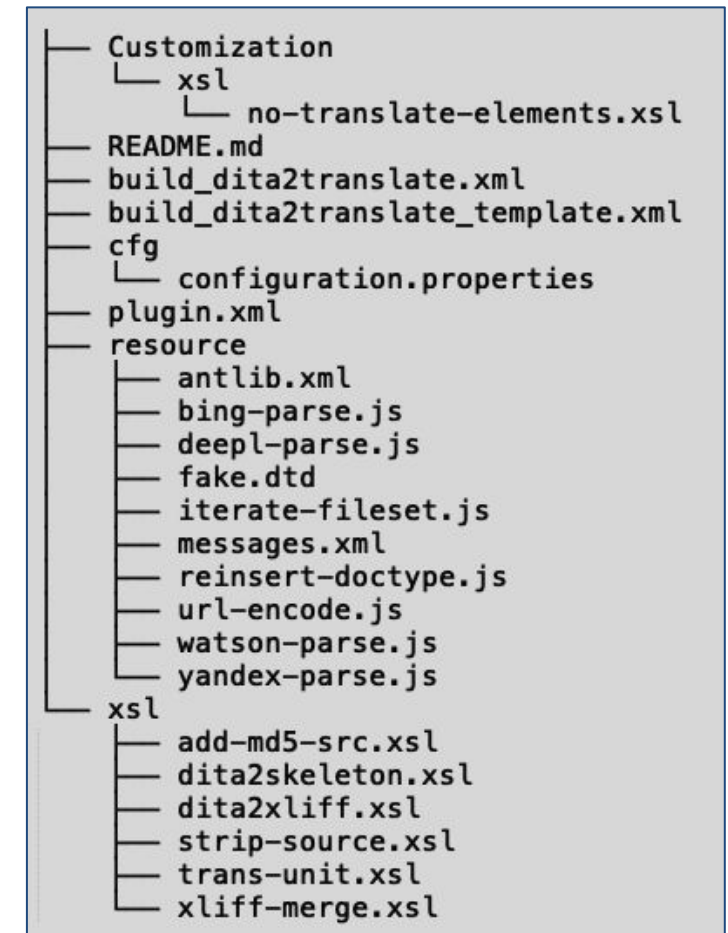
A FEW CENTURIES FROM NOW, ALL THE ENGLISH OF THE PAST 400 YEARS WILL SOUND EQUALLY OLD-TIMEY AND INTERCHANGEABLE.

Language Translation Plug-In for DITA-OT

● XSLT 72.1%

● JavaScript 27.9%

- **Transform:** Three custom XSL Transforms
 - DITA to XLIFF
 - Identity Transform plus Fletcher's Checksum
 - Create Skeleton Files
 - Create **XLIFF 1.2** Files
 - XLIFF to translated XLIFF
 - Iterate using XMLTask
 - Send to translate service and interpret result
 - **no-translate** post process ??
 - XLIFF to DITA
 - Create DITA and DITAMAP
 - Iterate using XMLTask
- **GitHub:** [fox.jason.translate.xliff](https://github.com/fox-jason/translate.xliff)



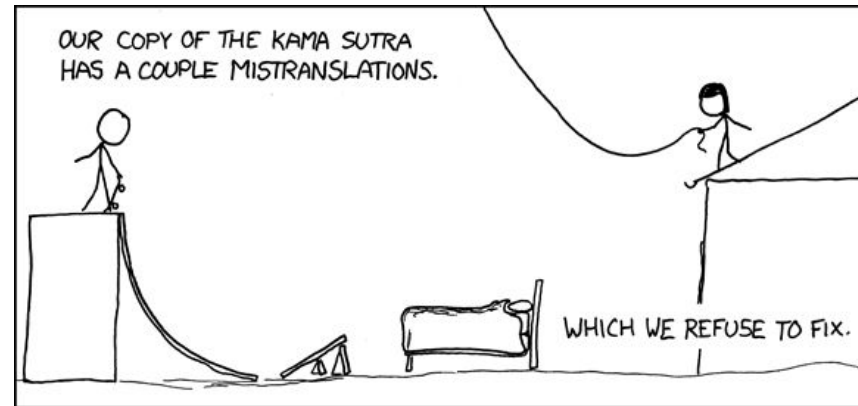
Language Translation Plug-In for DITA-OT

● XSLT 72.1%

● JavaScript 27.9%

■ Caveats:

- Don't rely on auto-translate - Check with a native speaker



- Don't overuse keyrefs - use normalized DITA transform

Thank you!

<http://fiware.org>

Follow @FIWARE on Twitter

