

FreeBSD DevSummit @ BSDCan 2014

Documentation Translation System Working Group

Agenda

1. Presentation/Demo on how translation is currently done using the FreeBSD German Documentation Project as an example
2. Demonstration of how gettext tools can be used to help support the translation process
3. List of things that are needed (like scripts, solutions to edge cases) and people willing to help out
4. Open Discussion

Goals

- Develop a feeling of how tedious the current translation process is, yet how important translations are
- Discovering new tools and how they could help us
- Discuss building a translation memory from translated strings to be shared within the project to translate other documents not currently being covered (like marketing materials, man pages (Insert bikeshed here), etc.)
- Automate as much as possible, connect the tools together
- Discuss the use of Pootle like PC-BSD does
- Requirements for such a translation system (server, infrastructure, people)

German Translation Project Demo

Pain points of the current approach

- Tedious, little automation, almost no tool support
- Usually only 2 translators trying to catch up to changes by multiple en_US doc committers
- Almost no outside help from contributors

Is there a better way
to do this?

Look at the source again and think about what actually needs translating...

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE article PUBLIC "-//FreeBSD//DTD DocBook XML V5.0-Based Extension//EN"
    "http://www.FreeBSD.org/XML/share/xml/freebsd50.dtd">
<article xmlns="http://docbook.org/ns/docbook" xmlns:xlink="http://www.w3.org/1999/xlink" version="5.0" xml:lang="en">
  <info><title>Port Mentor Guidelines</title>
  <authorgroup>
    <author><orgname>The &os; Ports Management Team</orgname></author>
  </authorgroup>
</info>
<sect1 xml:id="port-mentor.guidelines">
  <title>Guideline for Mentor/Mentee relationships</title>
  <para>This section is intended to [...].</para>
  ...
</sect1>
</article>
```

The gettext workflow



en_US
document

?



local_lang
document

Step 1:

Extract strings to be translated



en_US
document



POT file

Step 2:

Translate Strings in POT file



en_US
document



POT file



PO file

Step 3:

Generate translated document



en_US
document



POT file



PO file



local_lang
document

Using `textproc/po4a`

- Generate POT file from en_US XML source file:
`po4a-gettextize -f xml -m en_US.ISO8859-1/
<path>/chapter.xml -p strings.pot`
- Copy POT file to PO:
`cp strings.pot translation.po`
- Translate PO file using an editor like `poedit`
- Generate translated document:
`po4a-translate -k 0 -w 70 -f xml -m
en_US.ISO8859-1/<path>/chapter.xml -p
translation.po -l result.xml`

Open Questions

- What happens when updates are made?
 - New lines added
 - Updated sentences
 - Deleted sentences
- How do we keep the translation memory (TM) updated?
 - Document specific or global TM?

Scaling up the solution

- Idea from this TED talk by Luis von Ahn about reCAPTCHA:
[http://www.ted.com/talks/
luis_von_ahn_massive_scale_online_collaboration](http://www.ted.com/talks/luis_von_ahn_massive_scale_online_collaboration)

Involve more outside contributors
by providing strings/paragraphs to
translate from a website

Involve more outside contributors by providing strings/paragraphs to translate from a website

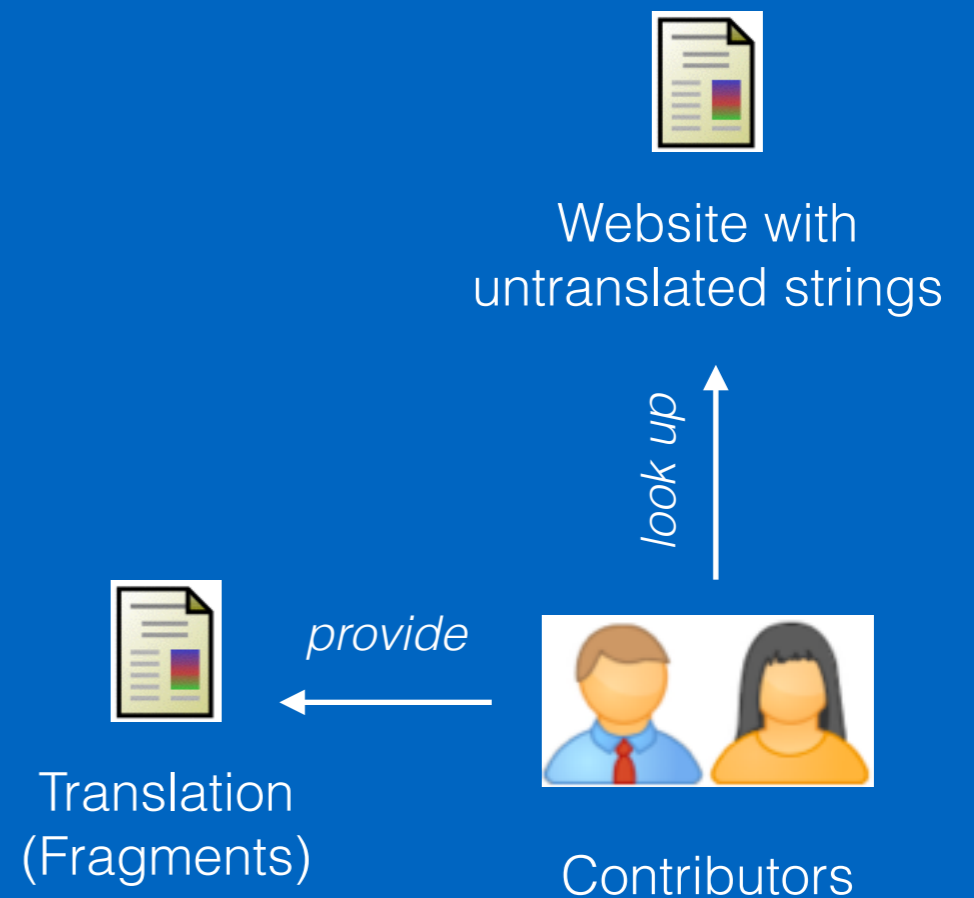


Website with
untranslated strings

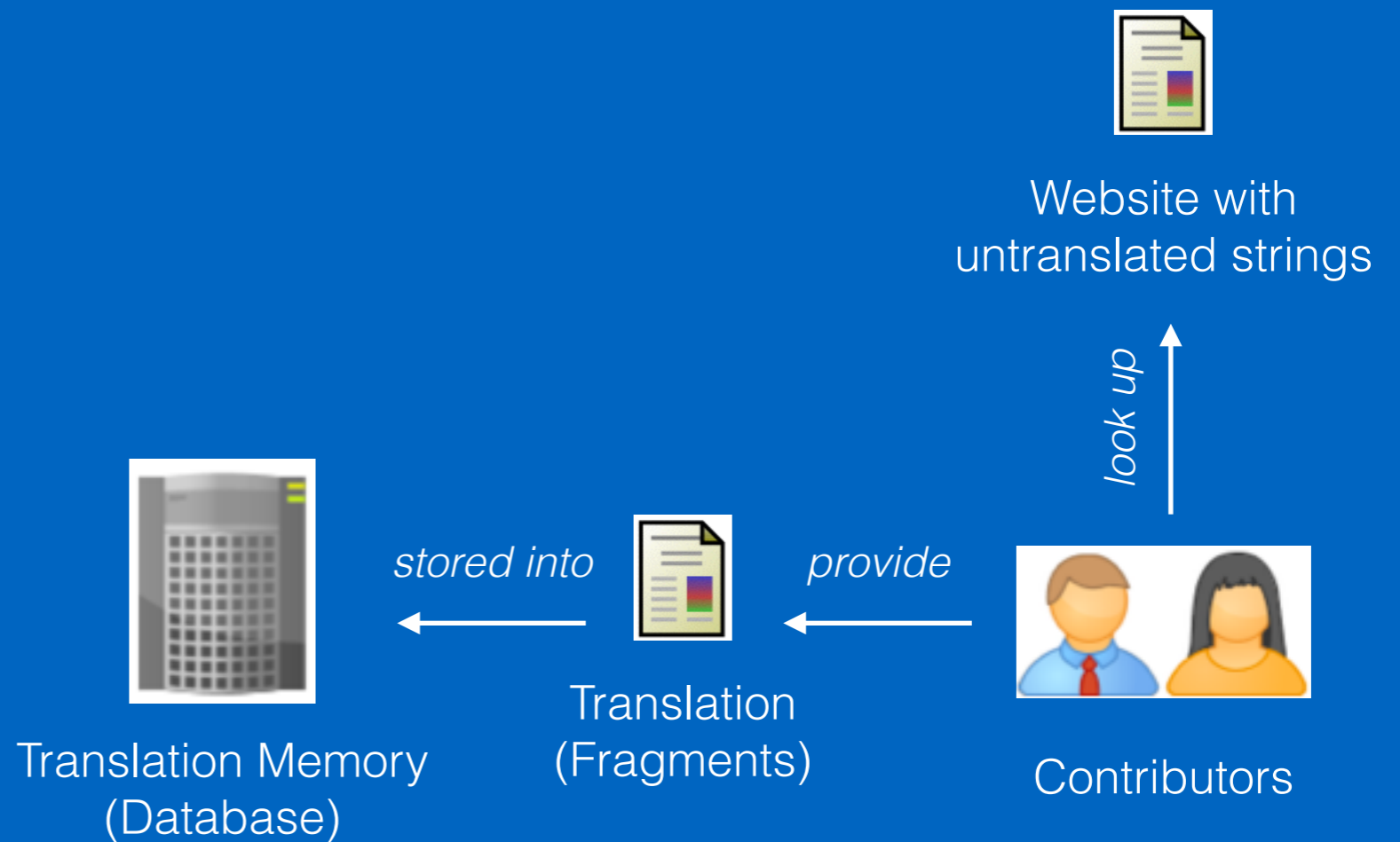


Contributors

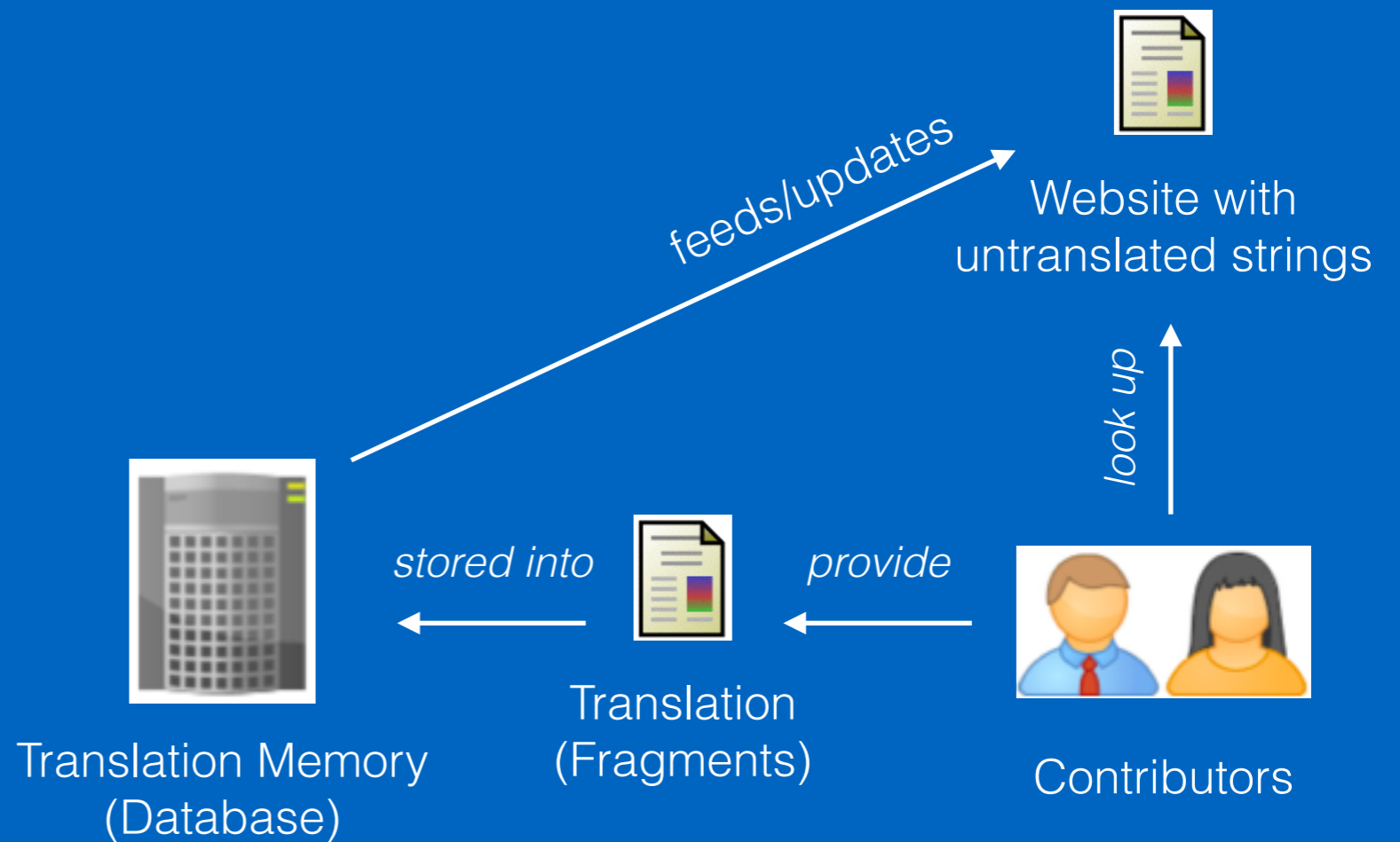
Involve more outside contributors by providing strings/paragraphs to translate from a website



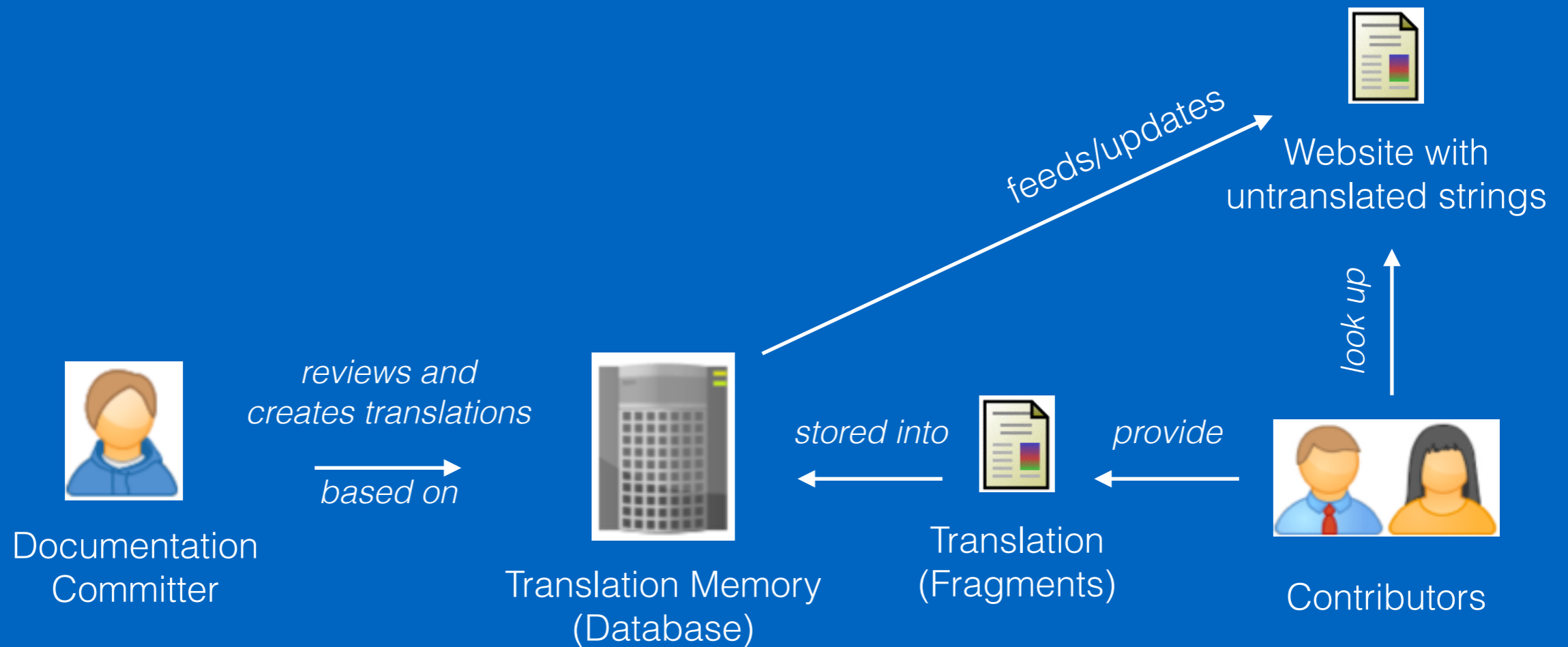
Involve more outside contributors by providing strings/paragraphs to translate from a website



Involve more outside contributors by providing strings/paragraphs to translate from a website



Involve more outside contributors by providing strings/paragraphs to translate from a website



Involve more outside contributors by providing strings/paragraphs to translate from a website

