

# How can I refactor HTML markup out of my property files?

Asked 16 years, 4 months ago   Modified 11 years, 6 months ago   Viewed 3k times



I've recently inherited a internationalized and text-heavy Struts 1.1 web application. Many of the JSP files look like:

9



```
<p>
  <bean:message key="alert" />
</p>
```



and the properties files look like:

```
messages.properties
alert=Please update your <a href="/address.do">address</a> and <a
href="/contact.do">contact information</a>.
```

with the appropriate translations in N other languages (messages\_fr.properties, etc).

Problems:

1. **DRY violation** - I have N references to my Struts action URLs instead of 1, which makes refactoring action URLs error-prone.
2. **Mixed concerns** - My application's markup is now in more than just my JSP files, making it difficult for a web specialist to tweak the markup (using CSS, etc).
3. **Post-translation markup** - Anytime I receive newly-translated text, I must decide what to surround with the `<a>...</a>` markup. Easy for English but less so for unfamiliar languages.

I've considered adding placeholders in the messages file, like:

```
alert=Please update your {0} and {1}.
```

but then the words "address" and "contact information" would somehow need to be localized, wrapped with markup, and passed to my message tag - and I can't see an easy way to do it.

What can I do to improve this?

Share

Improve this question

Follow

edited Jun 11, 2013 at 14:18



Linus Caldwell

11k ● 12 ● 47 ● 60

asked Aug 16, 2008 at 16:42



Brian Laframboise

5,514 ● 2 ● 32 ● 32

## 4 Answers

Sorted by: Highest score (default)



Avoid creating links within long blocks of text. Prefer shorter text that can act as a logically complete and independent link.



Generally, it will lead to fewer problems. Sometimes you have to compromise your UI design to accommodate localization; sometimes you need to compromise your localization process to accommodate the UI.

Any time a developer manually manipulates post-translation strings is a source of potentially expensive bugs. Cutting/pasting or string editing can result in character corruption, misplaced strings, etc. A translation defect needs the participation of outside parties to fix which involves cost and takes time.

Thinking on it, something like this might be less ugly:

```
<p>Please update your address and contact information.
<br />
<a href="/address.do">update address</a>
<br />
<a href="/contact.do">update contact information</a></p>
```

...but I'm no UI designer.

Share

Improve this answer

Follow

edited Jun 11, 2013 at 14:20



Linus Caldwell

11k ● 12 ● 47 ● 60

answered Aug 16, 2008 at 21:59



McDowell

109k ● 31 ● 206 ● 270



One approach that comes to mind is that you could store the translated replacement parameters i.e. "address" and "contact information" in a separate properties file, one per locale. Then have your Action class (or probably some helper class) look up the values from the correct ResourceBundle for the current locale and pass them to the message tag.



Share Improve this answer Follow

answered Aug 16, 2008 at 17:40



**John Topley**  
115k ● 47 ● 199 ● 240



Perhaps:

0

```
#
alert=Please update your {0}address{1} and {2}contact information{3}.
```



Share Improve this answer Follow

answered Aug 16, 2008 at 16:55



**McDowell**  
109k ● 31 ● 206 ● 270



The message message tag API allows only 5 parametric arguments

0

Ah! I blame my complete ignorance of the Struts API.



To quote the [manual](#):



Some of the features in this taglib are also available in the JavaServer Pages Standard Tag Library (JSTL). The Struts team encourages the use of the standard tags over the Struts specific tags when possible.

You could probably do this with the <http://java.sun.com/jsp/jstl/fmt> taglib.

```
<fmt:bundle basename="messages">
  <fmt:message key="alert">
    <fmt:param value='<a href="/">' />
    <fmt:param value="</a>" />
    <fmt:param value='<a href="/">' />
    <fmt:param value="</a>" />
  </fmt:message>
</fmt:bundle>
```

The downside is that this isn't valid XML and yanking the values to variables involves more indirection, lookups and verbosity. This is not a good solution.

I don't know Struts, but if it is anything like JavaServer Faces (same architect), then there is probably support for configuring a replacement control. I would either replace the existing control with a more flexible one or add a new one.

Anytime I receive newly-translated text, I must decide what to surround with the `<a>...</a>` markup.

There is no way you should be doing this and I see this as a fault in your translation process (I am an ex-localization engineer and ex-developer of localization tools). The `{0}` characters should be included in the files that are sent to the translators. The localization guidelines should explain the string's context and the meaning of any variables.

You can programmatically validate the property bundles on return. String-specific regex's might do the trick. It isn't outside the realms of possibility that "address" and "contact information" would swap order during translation.

The simplest solution is to redesign the messages to render:

```
<a href="/address.do">Please update your address.</a>  
<a href="/contact.do">Please update your contact information.</a>
```

I accept that this might not be a solution in all cases and may have your UI designer spitting teeth.

Share

Improve this answer

Follow

edited Jun 11, 2013 at 14:23



Linus Caldwell

11k ● 12 ● 47 ● 60

answered Aug 16, 2008 at 20:56



McDowell

109k ● 31 ● 206 ● 270

---