

# Getting All Your Web Apps To Wear The Company Brand

**Tim Knapp**

*Emerge Technology*

tim@emergetec.com

## **Abstract**

Cross-platform theming! No matter what the platform/language, you can seamlessly apply one theme across all your disparate web applications.

## **1. Introduction**

The technology is called Deliverance and Tim Knapp will outline how to use this powerful tool to theme your web applications (presentation link: <http://tinyurl.com/deliverance-pres0>)

## **2. The Problem**

One common problem web developers face is having one company theme but many different web applications with different theming approaches, for example: mailing list software, issue tracking software, Content Management Software, and blogging software.

This problem can be exacerbated by your company or client wishing to use a best-of-breed web application rather than whatever comes native with their preferred web platform, for example WordPress to manage their blog.

Then to add to your woes, Management may decide to move to a new web platform altogether and you're faced with the momentous task of retheming the website yet again.

Upgrades too tend to also mean rewriting applications to work on the new version and this can be a long drawn out process. And sadly by the time you've completed it, more often than not, the new version of the web application has already been released.

So what's the solution to all these varying problems? Deliverance.

### 3. A Deliverance Example

So how difficult is it to use Deliverance to theme your web applications? Let's say you wanted to replace the navigation and logo in your company theme with the actual logo and navigation from your content management system software. To achieve this in Deliverance you could simply do the following:

```
<replace content='#portal-logo img' theme='#logo a img' />
<replace content='children:.portletNavigationTree .navTree'
theme='children:#submenu ul' />
```

### 3. Why Use Deliverance?

One very compelling reason to use Deliverance is because it uses standard HTML and CSS, so web designers don't need to learn yet another theming technology. Therefore there's no theme integration work that needs to be done - the theme can be used as is. And you guessed it, this means it can be used with any web framework.

Deliverance has also been designed with performance and flexibility in mind. In its slowest incarnation it can be run as a separate proxy server, or for a bit more speed as either WSGI middleware or an XSLT inline transformation.

### 4. How Does It Work?

So how does Deliverance work? Simply put, Deliverance moves elements from your content into placeholders in your theme without modifying either of them.

If you want the gory technical details, an XML rules file defines the actions you wish to perform on the content and theme. There are 4 available actions that you can perform on the content or theme and you select the items you wish to operate on via CSS selectors or XPath expressions.

### 5. 4 Deliverance Actions

There are 4 different Deliverance actions that can be performed on the content and theme. The 4 actions and associated definitions sourced from the Deliverance documentation website[1] are below:

1. replace: “replaces something in the theme with something in the content”
2. append: “append puts things from the content after things in the theme”
3. prepend: “prepend puts things from the content before things in the theme”
4. drop: “remove problematic elements from a theme or content”

## 6. Deployment Options

The standard way people usually use Deliverance is as a standalone proxy server as this is generally the simplest way to get Deliverance up-and-running. A buildout environment[2] that could be used to set up Deliverance as a proxy is available in the References section.

As mentioned earlier, Deliverance can also be deployed as WSGI middleware using repoze. The scope of this document doesn't allow for us to discuss this here but again there are links[3] to sites containing further information in the References section.

And finally if you're specifically deploying Deliverance for a Plone CMS site, you can use collective.xdv[4], which pre-compiles a XSLT transformation file that is applied to the HTTP request when serving up pages in the site.

## 7. References

[1] <http://deliverance.openplans.org>

[2] <http://tinyurl.com/deliverance-pres0-code>

[3] <http://www.sixfeetup.com/blog/2009/4/27/deploying-plone-and-zine-together-with-deliverance-using-repoze>

<http://www.martinaspeli.net/articles/rolling-out-repoze>

[4] <http://plone.org/theming>