

Managing the UI

Posted At : December 6, 2006 3:36 PM | Posted By : Steve

Related Categories: Layout Components

Peter Bell has an interesting series on "Architecting Your Views". As usual, his thoughts are very interesting and inspired me to share my own views on the matter.

Peter starts by referencing my [article on layout components](#) and then [describes his requirements for a CMS](#). In brief, they are:

- Page Templates
- Multiple Dynamic Content Areas
- Reusable Screens
- Extensible UI Components
- Snippets
- Separation of Concerns
- Declarative Specification

I think that the distinction between a CMS and a site layout is an important one. Layout components can help with some (but not all) of these requirements. Many of them, however, are outside the scope of what your layout mechanism should provide.

Layout components can help with page templating by providing an easy mechanism to switch to a different layout.

In order to use my "Home" layout (as opposed to, perhaps, my default one), I add the following code above my call to the layout component.

```
<cfset layout = layout.switchLayout("Home")>
```

No reason why that couldn't be a dynamic value from the CMS instead of a hard-coded "Home" as well.

Most of the remaining functionality is in the domain of the CMS, but I do want to touch on a few other points that I will combine in one basic topic.

Snippets and Extensible UI Components are both reusable visual elements. Peter uses a Render component while many other people use custom tags. I think this brings up an important distinction. Some UI components are inherently general in purpose while others are intrinsically tied to the layout.

For example, every administrative section I have worked on needs tables and forms with similar functionality. As such, I have custom tags to manage each (available on my site, but still need to be cleaned up a bit). On the other hand, a given site may have a small calendar widget that will be used in various places throughout the site. It won't be used on other sites, but will need to be used in various places on this one.

In that case, I often add a method to my layout component. In this example, I would add a calendar() method which would render the calendar component. This also allows the calendar method to be extended for different layouts. For example, it might show up in the Home or Default layout, but not the layout used for outputting to MS Word.

I think that the Render component idea has some promise, but also may be more complex than is often needed. Most significantly, however, I think the distinction between general UI components and components that are essentially a part of the layout is an important distinction.

Incidentally, if you would like to see a full-fledged example of layout components at work, check out [CodeCop](#) (which will also review your code and which - thanks to layout components - looks right in CF6 admin, CF7 admin, or on its own). It is still in beta, but makes for a pretty good example of layout components (as well as the previously mentioned custom tags).