YmtdHosting

Hosting

JSP is great for hosted environments (ie: shared development environments, ISP's and large companies) where many different kinds of people are putting pages on the servlet engine may not know much more than HTML. Part of the reason it is so nice is that with the HTML'ish syntax JSP is so easy to learn!

However, if one really looks at that statement in more detail it becomes apparent that this might not be such a good thing. Consider the following snippet of Java code:

```
    Hashtable strings = new Hashtable();
    int i=0;
    while (true)
    {
        strings.put (&amp;quot;dead&amp;quot;+i, new StringBuffer(999999));
    }
```

What this does is that it creates a Hashtable and then goes into a tight loop. At the same time, it creates a new StringBuffer object that has a default size of 999999. Therefore, creating what amounts to a memory leak in the Java Virtual Machine (which all of the hosted applications share).

As soon as all of the memory is gone, every single hosted application will start to receive the dreaded "OutOfMemoryError". The reason why this is so bad has already been explained earlier (YmtdErrorHandling). Essentially JSP pages themselves cannot catch OOME errors and the entire servlet engine will suddenly become useless, all because of what amounts to a few lines of badly written Java code.

Remember, it is a bad idea to put Java code into a JSP page. Tell that to the 14 year old kid who is being hosted on your ISP's servlet engine who really does not care that others might be affected by these actions.

Velocity does not have this issue because there is no while loop in the Velocity Template Language. The only looping construct in Velocity is a #foreach and that loops over an array that has a finite amount of elements. It is also possible to disable the #foreach directive and limit the amount of recursive nesting that is possible. Note: it is possible to modify Velocity to add a #while directive to the template language.

There is nothing in JSP or Struts that prevents people from embedding Java code in the page.

You make the decision.

YmtdImplementation <- Previous | Next -> YmtdConclusion