Development-Backtracking

Backtracking

When Component Actions are used with WebObjects, to be able to preserve state across requests, the server must maintain a cache of previously visited pages. This page cache has a limited size. As a result, if the user presses the back button too many times to the point of exhausting this cache, it will result in a "User has backtracked too far" error.

Prevention with Direct Actions

Because direct actions don't require page state like component actions, the best way to avoid backtracking problems is to use Development-Direct Actions in your app.

Old Sessions and Backtracking

What is the simplest way to prevent a user from backtracking (using the browser back button) to the pages previously used by another user who has logged out?

To keep users from accessing someone else's session when they use the back button, you should call the session().terminate() in your logout action. In order for this to work, you have to issue a redirect, after calling terminate(), to page with no session, because if you return another page using pageWithName(), that page will reference the session you just terminated, and it won't work. My logout action looks something like this:

```
public WOComponent logout() {
   WORedirect redirect = (WORedirect)pageWithName("WORedirect";);
   redirect.setUrl("/cgi-bin/WebObjects/MyApp";); // entry point
   session().terminate();
   return redirect;
}
```

This will prevent the session from working if someone uses the back button, but it doesn't prevent them from seeing the content on the pages. If sensitive content is the issue, the best thing to do is to close the window when the user logs out. This would assume that when they loged in, you opened a window for the sensitive part of the app to run in. That way, when you use javascript to close it, you won't trigger a warning message that the user can intercept. With a window that you opened, you can do the redirect like above, to an html page that looks like:

```
<HTML>
<HEAD>
  <TITLE>Closing Window...</TITLE>
</HEAD>
<BODY onLoad="window.close()">
</BODY>
</HTML>
```

Then there won't be any pages to go back to.