

# Tapestry5Caveats

This page is about little issues that come up when using Tapestry (or web apps in general), usually in the form of an error message. These are NOT Tapestry bugs, merely differences between the your approach and Tapestry. Tapestry5 bugs should be logged in the JIRA.

Some of these issues are more likely to be encountered by Tapestry 5 early adopters, for example those who worked with 5.0.5 and are now using 5.0.10.

## NoClassDefFoundError

appears in T5.0.10

```
Render queue error in BeginRender[Portal:layouterp.menuerp.pagelink_2]:
Could not convert 'something' into a component parameter binding:
java.lang.NoClassDefFoundError:
com/mycompany/myapplication/pages/MyPage
```

Here, 'something' is some element in your application.

### Cause & Solution

Per HLS:

You'll see an explanation in the console. I suspect you are using a slightly older Tapestry preview release; more recently, this situation is an error. Because of how Tapestry instruments your code, instance variables must be private.

In this example, start looking in 'MyPage.java' for a variable that is not private. Make it private and use getters/setters for each attribute.

This also applies when you are subclassing components. Even though Inheritance will let you access public/private variables in the superclass, you will need to make the variables private and use getters/setters instead.

## Underscores in Prefixes for Fields, but not in Accessors

Use of underscore prefixes for fields, but not in accessors

Some people just grew into doing it this way. By default, Eclipse's getter/setter autogeneration keeps the underscore.

### Cause & Solution

Josh says:

This is configurable in Eclipse. Under Window | Preferences look at Java |

Code Style there is a "Conventions for variable names" section where you can tell eclipse what your prefix/suffix preferences are... Then it will make nice getter/setter names for you.

## Non case sensitivity of page names in URL

This is just a nice feature in Tapestry, documented in the code

## URL Shortening of page names under a similarly named directory

A documented feature intended to make things nice for users. For example:

A page named "Report" under a directory named "Report" will not resolve. (Fixed in 5.0.7+)

A page named "StatusReport" under a directory named "Report" will resolve to "Status" (i.e. report/status in the url)

A page named "ReportStatus" under a directory named "Report" will resolve to "Status" (i.e. report/status in the url)

A page named "StatusReport1" under a directory named "Report" will resolve to "StatusReport1" (i.e. report/statusreport1 in the url)

### Cause & Solution

For now, just pay attention to how you want your site laid out. If you get a Tapestry error telling you it can't find the page or component you specified, it may be because the URL is being shortened.

Per HLS:

"When a package name is replicated as a prefix or suffix of the unqualified class name, it is stripped out (unless that would reduce the name to the empty string)."

## Case sensitivity for @OnEvent and similar annotations

This is just a Java case sensitivity rule. Those unfamiliar with annotations before working with Tapestry (as I was) may forget that case matters in annotations! @onEvent is not the same, and your IDE should tell you this.

Non case sensitive for value of [OnEvent](#)

```
@OnEvent(value="SUBMIT")
void doSomething(){}

// is the same as

@OnEvent(value="submit")
void doSomething(){}

```

## Case sensitivity for event method names

Form event methods without annotations:

Besides typos in the name of the method, there is a case sensitivity issue to be aware of: the "on" prefix must be lower case. i.e. "OnSubmit" will not get called, but "onSubmit" will. T5's extensive case insensitivity can cause us to be lazy sometimes.

## DTD Case Sensitivity

Tapestry XML DTD's and their keys are case sensitive

This is in accordance with XML rules.

## Injected Asset Case Sensitivity

Injected Assets are case sensitive.

This is a Servlet API /Java runtime rule.

## Property File Case Sensitivity

Names of \*.properties files are case sensitive.

Names of properties inside the \*.properties files are not case sensitive

This is intended.

## Tomcat [OutOfMemoryError](#)

For now I've only encountered this in Tomcat 5.5 under Windows, and it really has nothing to do with Tapestry 5.

### Cause & Solution

With T5 you may find yourself being so productive that you create packages that are too deep. This happened to me when I added Yahoo's YUI library to a component library I was building. I'm guessing because the depth of characters in the package went beyond some magic number (maybe something like "255" characters), Tomcat could not load the application. What is distressing is that the error I received was "OutOfMemoryError" which made me *think* it may be Tapestry.

I fixed the problem by refactoring my components to a separate project, which had a shorter package depth. Now my main project references the new component library in its POM. When I make changes to the component library, I run

```
c:\projects\wtpworkspace\T5MyCompany\mvn install
```

And the component library is copied to my local repository. Make sure the component library is being packaged as a **JAR** not a **WAR**. Then I stop the main application if it was running, and restart it. (Tomcat doesn't reload the library for me, so this forces it to use the new jar.)