VelocityFAQ

The archives are littered with snippets of wisdom that are not captured in the User or Developer guides. Find something that isn't in the manual, and folks ask a lot on the list? Please, add it here. Eventually, the info may move into the offical guides. If you have a specific question that is not answered here or in the mailing list archives, then ask please the user list.

Velocity

- Q: Why isn't foo() method on Foobar object working? The object is in the template ($foobar works!) but the method is not resolved ($foobar.foo() doesn't work!)? Why can't Velocity find my method?
- A: Make sure that both the Foobar class and the foo() method are declared as "public". Velocity will only allow public methods on public classes to be called for security reasons. Many people will declare the method public, but leave the class as package private. Declare both the class and the method as public and Velocity should cooperate nicely.

- Q: How can conditionals be nested? For example, in the equivalent of "if($a){ if($b){ b1; } else{ b2; } if($c){ c1; } else { c2; } }", where do the '#end's go?
- A: Exactly where you put the closing braces in java. So your example would be "#if($a)#if($b) b1 #else b2 #end #if($c) c1 #else c2 #end#end". Pretty easy.

- Q: Adding ## at the end of a template causes a parsing exception, what do I do?
- A: A workaround is: add a newline to the end of the file. This issue has been fixed in Velocity 2.x.

- Q: Can velocity render 'dynamic properties' like #set ($bar = "name") $foo.$bar ?
- A: You can do so using the #evaluate() directive:

```
#set($bar = 'name')
#set($var = '$foo') ## a string in single quotes is not interpolated
#evaluate("${var}.$bar")
```

You can also check the RenderTool or ViewRenderTool in VelocityTools, which offers some flexibility on which context to use in the evaluation, or write a custom tool using Velocity.evaluate(...).

- Q: How do I output values from a hash when I do not know the keys?
- A: Easy - remember, the context has real java objects, all public methods can be used.

Here is a solution provided by Shinobu Kawai:

```
#foreach($key in $myMap.keySet())
 #set($value = $myMap.get($key))
 $key = $value
#end
```

or you can do

```
#foreach($entry in $myMap.entrySet())
 $entry.key = $entry.value
#end
```

- Q: I want to check for null, something like this:

```
#if ($car.fuel == null)
```

- A: See CheckingForNull

- Q: I want to do a "silent" #parse/#include
- A: You can try the solution provided by Fred Toth using a combination of a tool and a macro. See the mail archive for details.
- Q: Fred Toth's solution doesn't work for VelocityTools 1.2. 😞
- A: From 1.2, VelocityViewServlet uses the SeparateInstanceModel. You will have to set the VelocityEngine instead of the Velocity singleton class. Something like SilentParseInclude.

- Q: Why are Velocity templates not working from JavaScript?
- A: Velocity runs on the **server**, whereas JavaScript runs on the **client** in the browser. Examples: a user requests a page, on the server Velocity renders templates into HTML which is sent to the browser to display. All of the Velocity template processing happens **before** the page is sent to the browser. There is no way for JavaScript to render Velocity template directives without going BACK to the server. Note also that Velocity **can** be used to render custom JavaScript. (Tim Colson)

- Q: How can i lower memory usage by Velocity?

- A: If you have no concerns about speed, the easiest thing is to just turn off template caching. This can be done by setting `<resource loader name>.resource.loader.cache = false` (see Resource Management). If you still want to maintain some level of caching, since Velocity uses an LRUMap for template caching, you can limit caching to any specified number of your most used templates. Just set `resource.manager.defaultcache.size=<some number>`. It is also possible to create your own custom ResourceCache and plug that in instead of the default cache.

If none of these tweaks work for you here are a few more tips:

- try rewriting a few velocimacros in java (the ones that are called most often)

- try rewriting a few pages, using loops instead of repeatedly calling velocimacros

Remember, Velocity doesn't compile your templates. They are parsed into an AST (abstract syntax tree) and that is what gets cached. So, reducing template complexity will shrink the tree and produce immediate memory savings. For loops and (java) "tools" are simpler than velocimacros. (Nathan Bubna & Bruno Carle)

- Q: How can i access my object's public fields in my templates?
- A: Currently, you have three options:

- Wrap your object with a FieldMethodizer
- Configure your VelocityEngine to use a custom uberspector like the PublicFieldUberspect
- Lobby the velocity-dev list to add public field introspection as a default fallback if no matching method is found 🙂

Velocity Tools

- Q: The MessageTool is throwing a NullPointerException, why doesn't VelocityTools test for NPE's?
- A: This is because Struts' ActionServlet needs to be initialized before trying to use the Struts tools. Have a look at your web.xml file and make sure it starts up first. (Nathan Bubna)

- Q: Why is the VelocityServlet deprecated in favor of Tools VelocityViewServlet?
- A: Some time back, the Velocity Developers decided it might be better to separate web functionality from the core Velocity engine. They decided that VelocityTools would be a better place to drop this functionality. Nobody has had the guts to remove VelocityServlet from the Core, maybe in 2.0? <grin> (Tim Colson) Deprecate in 1.5, remove in 1.6 / 2.0 (whatever is next)? (Will Glass-Husain).

- Q: (since VelocityTools 1.2) Why doesn't my non-web use of Velocity no longer work? For example, I have code that allows the user to send email (based on a Velocity template) from a web app and it no longer works.
- A: One possibility is that it might not be sharing a configuration with VelocityViewServlet. More specifically, VelocityViewServlet was previously based on a singleton Velocity class. If you had other uses of Velocity within the same webapp that also used the singleton Velocity class, they would share configuration data. Now VelocityViewServlet uses a VelocityEngine instance to process templates. You can get this instance for other uses of velocity by calling getVelocityEngine from a subclass. Or, create and configure a second instance of VelocityEngine. (while slightly more involved, this allows different configuration options which are often desired). (Will Glass-Husain)

- Q: VelocityTools doesn't work with Struts 1.2!
- A: For Struts 1.2, you need VelocityTools 1.2, which is now available for download.

DVSL & Anakia & Texen & Misc

- Q: Is DVSL stable enough for use in production environment?
- A: It seems to work for most folks. Try it out and if there are problems, report them to velocity-dev.