

VelocityNullSupport

Here are some requests for null support in Velocity. [Here](#) is the discussion about this on the developer list.

The word "null" should be treated as null in Velocity specific content

For example:

```
#set ($list = ["one", null, "three"])
```

should make a 3 element list, with the elements "one", null and "three".

```
$foo.bar(null)
```

should call the "bar" method of the "foo" object with a null parameter.

Workarounds

1. An unreferenced reference is treated as null, so

```
#set ($list = ["one", $somevariablenamethatneverwillbeused, "three"])
```

and

```
$foo.bar($somevariablenamethatneverwillbeused)
```

will do as the above.

The "if" directive should be able to compare references with null

For example:

```
#if ($foo == null)
foo is null
#end
```

should output

```
foo is null
```

if foo was not in the Context.

```
#if ($foo != null)
foo is not null
#end
```

should output

```
foo is not null
```

if foo was in the Context.

Workarounds

1. Using [IfNullDirective](#) / [IfNotNullDirective](#)

```
#ifnull ($foo)
foo is null
#end
```

and

```
#ifnotnull ($foo)
foo is not null
#end
```

will do as the above.

2. Many others are listed at [CheckingForNull](#).

The "set" directive should accept null value as the RHS.

For example:

Assuming that \$bar wasn't in the Context,

```
#set ($foo = $bar)
#set ($foo = null)
```

should both remove "foo" from the Context, and

```
#set($map.foo = $bar)
#set($map.foo = null)
```

should both set the "foo" property of "\$map" to null.

Workarounds

1. Setting null by invoking Context#remove(), or setter methods with null. Assuming that the Context was \$ctx,

```
$ctx.remove("foo")
$map.put("foo", $bar)
$map.setFoo(null)
```

2. Using the [NullTool](#) / [ViewNullTool](#).

```
$null.setNull("foo")
```

will do as the above. Note: [NullTool](#) can't set properties to null.

3. [Bugzilla #20999](#) contains a patch to give this behaviour.
4. It is also stated in bugzilla that you can achieve this by using the [EventCartridge](#), but I can't figure out how. I'm hoping somebody will post a code snippet for us. 😊

The "foreach" directive should set null when the element is null

Currently, it preserves the last element.

For example:

```
#set ($list = ["one", null, "three"])
#foreach ($foo in $list)
$!foo
#end
```

should output:

```
one  
  
three
```

Workarounds

1. Using the [NullTool](#) / [ViewNullTool](#).

```
#foreach ($foo in $list)  
$!foo  
$null.setNull("foo")  
#end
```

will do as the above.

2. [Bugzilla #27741](#) contains a patch to give this behaviour.

null and the empty string should be totally equivalent

Comment from Nathan about the null keyword and the if directive support:

-0 i'm not totally against it, but i'm instinctively uncomfortable with using this programming concept in a template language. personally, i've always felt in a display-oriented language, null and the empty string should be totally equivalent. but that is already not the case, so i can't really be -1 on this.