VelocityWhitespaceGobbling

The Great Whitespace Debate

Since ancient times, men have fought over many trifling and trivial things. Among these, few have been more fiercely contested than the dreadful matter of **whitespace handling**. Should it be gobbled in whole, ignored, randomly eliminated, or dealt with according to a complex yet carefully chosen set of guidelines? As with any other religious concern, the variety of answers runs the gamut. Some men hold tightly to one answer as the true answer for all men in all situations, whilst others freely mix and match, admiring each tactic in different situations. Some men have grown weary of debating such trivial concerns, whilst others remain blissfully ignorant that such matters matter at all. To those reading this today, I put the question yet once more, and I beg you to remain civil in response...

How then, shall we handle whitespace in Velocity ???

Possible (competing?) Objectives

- Make Velocity whitespace handling more intuitive
- Source text should become easier to read
- Backwards compatible: Results from existing templates should be the same

The Four Options of the Apocalypse

- VelocityWhitespaceGobbleAsUsual
- VelocityWhitespaceGobbleAll
- VelocityWhitespaceGobbleNone
- VelocityWhitespaceGobbleStructuredTemplates
- VelocityWhitespaceGobbleSetindentDirective

The History of the Debate

- Bugzilla #27413 - Enhance Whitespace Gobbling
- One of many threads

Alternatives and Workarounds

- JTidy
- VelocityWhitespaceTruncatedByLineComment (Use a Velocity comment to artificially end a line, but doesn't remove leading or trailing spaces, tabs, etc.)
- Create a lightweight whitespace gobbling filter to pass the velocity output through

Discussion of JTidy as a Workaround

I have looked into using JTidy as a workaround and it's quite promising. But we have to remember a few things before we all embrace the JTidyness.

- JTidy is only for HTML and XHTML (up to XHTML 1.0, not yet XHTML 1.1) so it can't be used for other documents and, in particular, text documents where whitespace does not get gobbled by the parser.
- JTidy is buggy and at times does not create W3C compliant code even if your original code is W3C compliant.
- As JTidy can add, remove or move elements in your HTML you don't have direct control over the end result. So you can't just roll this out to all your HTML output in one go. This is the same argument as the one against whitespace gobbling in Velocity – you would have to test evey single page (template) to make sure that nothing funny has happened.
- The more steps you have before you output adds more complexity, more maintaining and more debugging.

All these points are quite compelling reasons not to use JTidy.

Democracy in Action (aka Democracy Inaction)

Add your name below to vote for your preferred **DEFAULT** behavior.

- For Gobbling As Usual
  - everyone who has read this and not voted! (yes, this is a guilt trip)
  - Paul Andrew Loy (paul ART keteracel DOT com) – for consistency with current templates but would want the option for new templates to use gobble all or structured as long as it was consistent
- For Gobbling All Whitespace
  - (no votes yet)
- For Gobbling No Whitespace
  - Nathan Bubna (nathan AT esha DOT com)
  - Tim Colson (tcolson AT cisco DOT com)
  - Mike Kienenberger (mkienenb AT alaska DOT net) – but I'd go with "Structured Templates" as a close second.
  - Shinobu Kawai Yoshida (shinobu AT ieee DOT org)
- For Gobbling Structured Templates
  - Christoph Reck (apache AT recks DOT org)
  - Will Glass-Husain (wglass AT forio DOT com)
  - Marino Jonsson (marinoj AT centrum DOT is)
  - Peter Urban (urban AT jaist DOT ac DOT jp)
  - Claude Brisson (claude AT renegat DOT net) – bkoz it's almost identical to "Globbing No Whitespace" for people that use few indented directives

    - Marc Novakowski (marc AT pandora DOT com)
    - Michael Hall (mhall AT lakeland DOT net) – I like the
    - Alden Taylor (ataylor AT atech DOT com) – but would also see value in having SetindentDirective with +4 -4, would be excellent to make gobbling configurable
    - Tom Corcoran (boardtc AT gmail DOT com) – working on a per macro basis would be nice, e.g. have a #macrotrim

Add any write-in candidates below:

Secondary Issues

- Adding support for non-default behavior
- Different ways to configure/support non-default behavior
- Treatment of whitespace within VTL (all the above is about whitespace within schmoo)