

VelocityTools

[VelocityTools](#) is a sub-project which contains extensions aimed at solving common web and non-web problems. [VelocityTools](#) 1.3 is the latest stable release. [VelocityTools2](#) is also now available in alpha form.

Advantages

- Generic-Tools provide Date/Number formatting, Math operations, and much more.
- Velocity-View provides a time-tested servlet and tools for Velocity-based web applications.
- Velocity-Struts - a robust and easy-to-implement alternative to JSP for Struts Webapps.

Velocity Tools 1.3 and Velocity Tools 2.0-alpha1 are available

Both source and binary distributions are [available for download](#) through the usual Apache mirror sites. Please remember to verify the signatures of the distribution using the keys found on the main [Apache website](#) when downloading from a mirror.

Useful Links

- [Project Website](#)
- [Project Website in Japanese](#)
- [ViewCVS/SVN](#)
- [VelocityTools2Planning](#) - ideas for the next major version

Contributed Code and Examples

- [PublicFieldUberspect](#) – Uberspect implementation that exposes public fields.
- [Velosurf](#) – A database mapping tool for Velocity.
- [ServletLogChute](#) – A replacement for the ServletLogger that works with the upcoming Velocity 1.5 logging improvements.
- [CommonsLogLogChute](#) – A replacement for CommonsLogLogSystem that works with the upcoming Velocity 1.5 logging improvements.
- [LogChuteCommonsLog](#) – A replacement for LogSystemCommonsLog that works with the upcoming Velocity 1.5 logging improvements.
- [Tips for Tiles](#)
- [NullTool](#) – A generic Tool that works with null.
- [ViewNullTool](#) – A [ViewTool](#) that works with null.
- [IfNullDirective](#) – A directive (#ifnull) that renders contents if the given value is null.
- [IfNotNullDirective](#) – A directive (#ifnotnull) that renders contents if the given value is not null.
- [MonostateVelocityEngine](#) – A nasty [VelocityEngine](#) for people who want to be polymorphic with the Singleton model. See [Bug 21258](#).
- [TemplateNameFilter](#) – A servlet filter that adds ".vtl" to URIs when they point towards templates.
- [StreamLayoutServlet](#) – Layout revisited: This is another layout sevllet that will preserve the direct streaming of the page towards the response stream without caching it into memory.

Old Contributed Code (for use with old versions of [VelocityTools](#))

- [VelocityValidationToolHack](#) for Struts 1.0 – A quick hack to enable client-side javascript validation, not Struts 1.1 compatible.
- [ArrayTool](#) – A generic Tool that works with arrays. Now, use the [ListTool](#) that also works transparently with lists.
- [BrowserSniffer](#) – A browser sniffer tool. Depends on jdk 1.4 for regexps.
- [EscapeTool](#) – A generic Tool that works with escaping. Depends on commons-lang 2.0.

Velocimacro Library

The mailing list often has questions/suggestions for for common macros. We'd like to gather the good ones in our [VelocimacroLibrary](#). 😊 And this is a Wiki, so contributions are very welcome!

- [VelocityGenericMacros](#) – non-web macros, applicable in **any** Velocity setting, these might make use of the Generic Tools.
- [VelocityViewMacros](#) – web-oriented macro and/or make use of VelocityView tools (using the recommended keys!)
- [VelocityStrutsMacros](#) – applicable for VelocityStruts applications, and will generally but optionally use the Velocity-Struts tools (using recommended keys!).

TODO:

- document recommended keys for each standard tool
- write plan/instructions for using the \$XHTML setting for VelocityView and VelocityStruts macros
- develop/document standard coding style for macros
- develop/document standard way of documenting macros (maybe using <http://www.chickenshick.com/velocidoc/>)