Tapestry4Final

NOTE: This is outdated information that applies only to Tapestry 4. For similar information on current versions, see

http://tapestry.apache.org/release-notes.html

Tapestry 4.0's final release is nearly here; this page is a collective editting space to finalize the release notice that will go out.

–

Tapestry 4.0 final release now available.

After nearly two years of work, the next major release of the Tapestry web application framework is now available.

Tapestry is an open-source framework for creating dynamic, robust, highly scalable web applications in Java. Tapestry complements and builds upon the standard Java Servlet API, and so it works in any servlet container or application server.

Tapestry divides a web application into a set of pages, each constructed from components. This provides a consistent structure, allowing the Tapestry framework to assume responsibility for key concerns such as URL construction and dispatch, persistent state storage on the client or on the server, user input validation, localization/internationalization, and exception reporting. Developing Tapestry applications involves creating HTML templates using plain HTML, and combining the templates with small amounts of Java code using (optional) XML descriptor files. In Tapestry, you create your application in terms of objects, and the methods and properties of those objects – and specifically not in terms of URLs and query parameters. Tapestry brings true object oriented development to Java web applications.

Tapestry is specifically designed to make creating new components very easy, as this is a routine approach when building applications. The distribution includes over fifty components, ranging from simple output components all the way up to complex data grids and tree navigators.

Tapestry is architected to scale from tiny applications all the way up to massive applications consisting of hundreds of individual pages, developed by large, diverse teams. Tapestry easily integrates with any kind of backend, including J2EE, HiveMind and Spring.

Tapestry 4.0 represents a significant advance over Tapestry 3.0. The following are the *most significant* changes between the two releases:

- The new 4.0 specification DTDs have been simplified.
- The syntax used for binding parameters inside an HTML template and inside an XML specification is now consistent. Both make use of the binding prefixes.
- "Friendly" URLs (that is, URLs that pack more information into the path and less into query parameters) are built in. This makes it easy to divide your application across many folders (reducing clutter), and leverage J2EE declarative security along the way.
- Listener methods are much easier and more flexible; listener parameters in the URL are automatically mapped to listener method parameters, and listener methods can return the page name or page instance to activate.
- Component parameters now just work, without having to worry about "direction".
- Applications can now have a global message catalog, in addition to per-page and per-component message catalogs. Messages not found in the component message catalog are searched for in the application catalog.
- Full, native support for developing JSR-168 Portlets has been added.
- Tapestry 4.0 makes much less use of reflection and OGNL than Tapestry 3.0; partly because there are many new binding prefixes and largely because of how parameters are now implemented.
- HiveMind services and Spring beans can be directly injected into page and component classes.
- Tapestry 4.0 includes optional JDK 1.5 annotation support (but Tapestry still works with JDK 1.3).
- Tapestry 4.0 debuts a new and much more sophisticated user input validation subsystem.
- Line precise error reporting can now display the contents of files containing errors.
- Forms can now be canceled, bypassing client-side validation logic, and invoking an alternate listener on the server-side.
- You are no longer limited to just Global and Visit; you can have as many application state objects as you like.
- The use of HiveMind under the covers means that Tapestry can be easily customized to fit your needs.
- Page properties can now be persisted on the client, as well as in the session.
- Components and component parameters can now be marked as deprecated. Component parameters may have aliases (used when renaming a parameter).

Tapestry is released under the Apache Software Licence 2.0.

Tapestry is distributed as a combined binary/source distribution, and an additional documentation distribution. Tapestry may be downloaded from the Apache Mirrors.