

Introduction

What is Apache Tapestry?

Apache 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 adding a small java class for each. 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.

Tapestry is architected to scale from tiny, single-page 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 JEE, Spring and Hibernate.

It's more than what you can do with Tapestry ... it's also how you do it! Tapestry is a vastly productive environment. Java developers love it because they can make Java code changes and see them immediately ... no redeploy, no restart! And it's blazingly fast to boot (even when files have changed). Designers love it because Tapestry templates are so close to ordinary HTML, without all the cruft and confusion seen in [JavaServer Pages](#). Managers love it because it makes it easy for large teams to work together productively, and because they know important features (including localization) are baked right in. Once you work in Tapestry there's no going back!

Tapestry is released under the Apache Software License 2.0.

Third Party Libraries, Tutorials and Resources

A number of third party libraries, tutorials and resources are listed on the [Modules](#) page.

About Releases and Snapshots

Most users will want to use the latest stable release of Tapestry, and for that your best bet for new projects is to use the Quickstart Maven archetype to create your initial Tapestry project, as described on the [Getting Started](#) page. The Quickstart archetype generates a full, working project directory. For upgrading existing projects, just use the Maven dependency listed on the [Download](#) page.

You can also pull down Tapestry modules in the form of binary and source JARs from the [Maven Central repository](#).

Tapestry itself is built using Gradle, which makes it really easy to download the source and build it yourself, either the whole project, or just one single module.

The use of Maven and Gradle has let us move with great speed, providing preview releases and snapshots.

Snapshots are intermediate versions of releases, with "-SNAPSHOT" at the end of the version number. Maven notices that -SNAPSHOT suffix and handles the dependency specially. It knows that snapshot releases can change frequently, so it will keep checking (at least once a day, maybe more often) to see if there's an updated version of the snapshot.

A nightly build process on Tapestry's continuous integration server creates new snapshots every night.

Snapshots don't go in the Maven central repository (that's reserved for full releases). Instead, they go into the Tapestry snapshots repository at <https://repository.apache.org/content/groups/snapshots/org/apache/tapestry/>.

To access the snapshot repository, just add `-DremoteRepositories=http://repository.apache.org/snapshots/` to the command line when running Maven.

Documentation on this site sometimes refers to the latest snapshot ... that is, it is often ahead of the last official release, with version-specific differences clearly marked. In some cases, it is written as if the snapshot release is stable. For example, if documentation refers to version 5.7.x and that hasn't been released yet, you can try 5.7.x-SNAPSHOT.

Related Articles

- [Introduction](#)
- [Tapestry for JSF Users](#)
- [Tapestry Tutorial](#)
- [Principles](#)
- [Getting Started](#)