

Overview

Tynamo in a nutshell

Tynamo is model-driven, full-stack web framework based on Tapestry 5.



A full-stack (*web application*) framework's primary goals are to provide a single set of components, covering the entire spectrum of things you'll need to build web applications, and to ensure that those components work well with one another. [1]

Tynamo's mission is to make web application development in Java simpler. We leverage existing technologies where possible and provide integrations with proven, clean and compact libraries rather than limit ourselves only to standard Java (JSRs). Tynamo is both comprehensive and modular so you are free to choose the parts you like from our full stack. Certainly, we follow standards where possible but we drive towards simplicity rather than conformance. We are not interested in reinventing the wheel, but we leverage existing projects and libraries where ever possible. And finally, we like Tapestry and our modules use Tapestry IoC extensively. We are well aware of most other front-end web technologies, but we don't use Wicket, Spring MVC, Stripes, Struts, Shale or any other and we are not planning to integrate with those technologies.

Our older sibling, [Trails framework](#), offered the same goodies for Tapestry 4. Trails was started sometime around 2005. The current committers started in the project shortly after and as Tapestry evolved, Trails and we evolved with it. The concepts in Tynamo are largely the same as in Trails, but Tynamo is more modular, simpler and easier to use. We have several stand-alone modules for Tapestry that you are free to use without the rest of Tynamo goodies, or, if you just about to start with your journey to developing Java web applications, you can start with Tynamo archetype (read the [Quick start](#)) and then gradually improve your understanding of the technologies in play while adding features and customization to your application.

We are also all about modeling the domain in an object-oriented manner. Currently, this means adopting an ORM (Object-Relational-Mapping) tool, namely Hibernate - one of the leading open-source ORM implementations. We are very interested in JPA2 (that would provide more generic, vendor-independent ORM support) but as the standard is still in the state of flux, we stick with Hibernate as we always find JPA(1) too limited. Unfortunately, while current ORM technologies are fairly advanced, there's still an [impedance mismatch](#) and they are not always easy to use, so you often need to understand the basics of relational database to be productive with ORM.

(Released) Tynamo modules

[tapestry-model](#) (see below, then the [tapestry-model guide](#))

[tapestry-conversations guide](#)

[tapestry-hibernate-seeentity guide](#)

[tapestry-exceptionpage guide](#)

[tapestry-jpa guide](#)

[tapestry-resteasy guide](#)

[tapestry-security guide](#)

[tapestry-watchdog guide](#)

What else do we have under our wings? Check out our [Roadmap](#) for more information on unreleased modules and plans.

tapestry-model

Tapestry-model is the main Tynamo module. Tapestry-model allows rapid **CRUD** application development by automatically creating list, display and edit pages for **POJOs**. Tynamo uses introspection to gather information about bean properties and creates representations and editors depending on their type, for example.

Tapestry-model itself consists of three main components: a persistence service, a descriptor service that provides meta data for the domain beans and an editor service that provides appropriate editor components based on meta data. To do the heavy lifting, Tynamo depends on several other frameworks. To understand tapestry-model, it is necessary to understand a bit about the frameworks it uses, specifically Tapestry and Hibernate.

How to use

Tynamo uses Maven(2/3). All released Tynamo modules are available through [Maven central](#). Automatically built snapshots (whenever anything changes) are available through [Codehaus CI repository](#). We run both unit and integration tests (full in-container tests) in an automated way so trunk snapshots are generally relatively safe to use, but don't blame us if something goes wrong 😊

To use the snapshot repository, you need to add the following configuration to your project's pom or modifying your internal Maven repository with the equivalent configuration (the latter is the recommended practice):

```
<repositories>
  <repository>
    <id>Codehaus Snapshots</id>
    <url>http://ci.repository.codehaus.org/</url>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
    <releases>
      <enabled>false</enabled>
    </releases>
  </repository>
</repositories>
```

Naming convention is such that the group id is always *org.tynamo* and artifact id indicates the project or the module's standalone status (such as *tapestry-conversations* that can be used without dependencies to other Tynamo modules). *tapestry-model* consists of several sub-modules.

Downloading Tynamo

If you just started with Tynamo, it's easiest to use Maven and follow the [Quick Start](#) guide. Since Tynamo is intended for implementing applications on top of it, you need a build system any way and Maven will make your life easier - just take our word for it.

If for one reason or another, you hate Maven and absolutely refuse to use it, you can download the desired [Tynamo releases from the main Maven repository at repo1.maven.org](#).

Feedback and meet the committers

Your best and fastest way to get help online: IRC

Meet the committers and even some occasional users 😊 Both Kalle and Alejandro are almost always on the Tynamo channel of Codehaus' IRC, at <irc://irc.codehaus.org:6667/#tynamo>. Feel free to join us!

Mailing list

Standard issue subscription system - [check the mailing lists we have](#)

Leave comments on the Wiki

This may work sometimes if you leave "good" comments, but I (Kalle) will mercilessly remove questions left at the end of official documentation and answered elsewhere or once the documentation has been updated. Your best bet is to comment the news if you just want to make yourself heard through the Wiki. Otherwise, feel free to edit the wiki to improve the documentation, we appreciate the effort!

-
1. full-stack framework in a nutshell by [James Bennett](#)