



A STRONG HERITAGE

While the name may be new, WildFly is the eighth version of the project previously known as JBoss AS. Red Hat is committed to Java, and WildFly 8 will focus on quickly delivering an open-source implementation of Java EE 7.

BLAZING FAST START-UP

In the highly optimized boot process of WildFly 8*, services are started concurrently to eliminate unnecessary waits and to tap into the power of multi-core processors. Non-critical services are kept on ice until first use. Subsequent boots save additional time by leveraging cached or indexed metadata.

As a result, WildFly offers a 10-fold reduction in startup time over previous versions and even gives Jetty and Tomcat a run for their money.

MEMORY DIET

WildFly takes an aggressive approach to memory management to minimize garbage collector pauses, only loads JAR files it needs and uses indexed metadata over full parses. All this keeps the memory footprint exceptionally small.

These optimizations enable AS 7 to run with stock JVM settings and on small devices. It also leaves more headroom for application data and supports higher scalability.

Server profiles can be used to trim unneeded capabilities, allowing you to make the server as lean as possible.

MODULAR DESIGN

WildFly does classloading right. It uses JBoss Modules to provide true application isolation, hiding server implementation classes from the application and only loading the classes your application needs. Modules, packaged as collections of classes, are peers that remain isolated unless explicitly defined as a dependency of another module. Visibility rules have sensible defaults, yet can be customized.

ELEGANT ADMINISTRATION

WildFly administration is centralized, simple and user-focused. The configuration file is based on a straightforward domain model that you can easily comprehend and no internal wiring is exposed.

The same configuration file can be used for multiple servers in domain mode. In contrast, standalone mode controls a single instance, similar to how previous versions of JBoss AS were launched.

COMMAND CENTRAL

Consistent and powerful management is available out of the box, including a polished, user-friendly web console, Java and HTTP APIs and a command line tool (CLI). The web console provides a clear perspective into the server runtime, while the programmatic APIs can be leveraged by tools, scripts and geeks.

What makes the interfaces especially useful is that changes are persistent, so you have the option to avoid fiddling with the configuration file directly.

PATCHING MADE EASY

Patching has never been easier with WildFly. Use the command line to deploy, stage, apply and even rollback system wide patches.

STRICT COMPLIANCE

JBoss has always been committed to application portability to prevent vendor lock-in. That's why we are aiming to make WildFly Java EE 7 certified. But we didn't stop there!

WildFly boasts an internal test suite based on Arquillian that achieves even more strict compliance to the portability expectations of our community of users.

PROJECT UNDERTOW

WildFly also includes Project Undertow. Project Undertow is based on the concept of lightweight async handlers. These handlers are chained together to form a complete HTTP server. The handlers can also hand off to blocking handlers backed by a thread pool.

Undertow architecture is designed to give the end user complete flexibility when configuring the server. If the user simply wants to serve up static files, then they can configure a server with only the handlers that are required for that task, and multiple security mechanisms can be coordinated without being tied to servlets.

For more on WildFly, please visit the WildFly forums at JBoss.org

<https://community.jboss.org/en/wildfly>



Scan this and find out more about the WildFly project.
<http://wildfly.org>



JBoss Community

* WildFly is in the upstream for JBoss Enterprise Application Platform. For a free product download go to: <http://jboss.org/products/eap>