

Article

What's New in Java SE 6

By Danny Coward, Java SE Platform Lead

Approved Feature List Table: Mark Reinhold, Java SE Chief Engineer, Updated February 2007

[Articles Index](#)

Version 6 of the Java Platform, Standard Edition (Java SE), was released for general availability in December 2006. So here are the top 10 things you need to know about the release, if you're still hovering hesitantly over that [Download Page](#) link. Subsequent articles will provide more in-depth information.

Contents

- [Top 10 Things You Need to Know](#)
- [Approved Features](#)

Top 10 Things You Need to Know

1. Web Services

All developers get first-class support for writing XML web service client applications. No messing with the plumbing: You can expose your APIs as .NET interoperable web services with a [simple annotation](#). Not your style? Want to handle the XML directly? Knock yourself out: Java SE 6 adds new parsing and XML to Java object-mapping APIs, previously only available in [Java EE](#) platform implementations or the [Java Web Services Pack](#).

2. Scripting

You can now mix in [JavaScript](#) technology source code, useful for prototyping. Also useful when you have teams with a variety of skill sets. More advanced developers can plug in their own scripting engines and mix their favorite scripting language in with Java code as they see fit.

Perhaps You Thought you couldN't program with a scripting language and Java together. Which will you Be trying ?

3. Database

For a great out-of-the-box development experience with database applications, the Java SE 6 development kit – though not the Java Runtime Environment (JRE) – co-bundles the all-Java JDBC database, [Java DB](#) based on [Apache Derby](#). No more need to find and configure your own JDBC database when developing a database application! Developers will also get the updated JDBC 4.0, a well-used API with many [important improvements](#), such as special support for XML as an SQL datatype and better integration of Binary Large Objects (BLOBs) and Character Large Objects (CLOBs) into the APIs.

4. More Desktop APIs

Much has been said about this [spoonful of sugar](#) (to go with the Desktop team's [cake](#)), so we will only skim a little. GUI developers get a large number of new tricks to play like the ever popular yet newly incorporated SwingWorker utility to help you with threading in GUI apps, [JTable](#) sorting and filtering, and a new facility for quick splash screens to quiet impatient users.

5. Monitoring and Management

The really big deal here is that you don't need do anything special to the startup to be able to attach on demand with any of the [monitoring and management tools](#) in the Java SE platform. Java SE 6 adds yet more diagnostic information, and we co-bundled the infamous memory-heap analysis tool [Jhat](#) for forensic explorations of those core dumps.

6. Compiler Access

Really aimed at people who create tools for Java development and for frameworks like JavaServer Pages (JSP) or Personal Home Page construction kit (PHP) engines that need to generate a bunch of classes on demand, the compiler API opens up programmatic access to `javac` for in-process compilation of dynamically generated Java code. The compiler API is not directly intended for the everyday developer, but for those of you deafened by your screaming inner geek, [roll up your sleeves](#) and give it a try. And the rest of us will happily benefit from the [tools](#) and the improved Java frameworks that use this.

"For those of you deafened by your screaming inner geek, roll up your sleeves and give it a try."

Danny Coward
Java SE Platform Lead

7. Pluggable Annotations

It is becoming a running joke in Java technology circles, at least some that contain us, that for every [wished-for feature](#) missing in Java technology, there's a budding [annotation](#) that will solve the problem. Joke no more, because Java tool and framework vendors can put a different smile on your face, defining their own annotations and have core support for plugging in and executing the processors that do the heavy lifting that can make custom annotations so cool.

8. Desktop Deployment

Those of you deploying applications to the desktop will soon discover that it's a tale of a large number of [smaller changes](#) that add up to

a big difference to existing applications: better platform look-and-feel in Swing technology, LCD text rendering, and snappier GUI performance overall. Java applications can integrate better with the native platform with things like new access to the platform's System Tray and Start menu. At long last, Java SE 6 unifies the Java Plug-in technology and Java WebStart engines, which just makes sense. Installation of the Java WebStart application got a much needed [makeover](#).

9. Security

You can have all the security features you like in the platform — and this release adds a few more, like the [XML-Digital Signature \(XML-DSIG\) APIs](#) for creating and manipulating digital signatures — but if you don't have well supported security administrators, your security may be at risk. So Java SE 6 has simplified the job of its security administrators by providing various new ways to access platform-native security services, such as native Public Key Infrastructure (PKI) and cryptographic services on Microsoft Windows for secure authentication and communication, Java Generic Security Services (Java GSS) and Kerberos services for authentication, and access to LDAP servers for authenticating users.

0. The -lities: Quality, Compatibility, Stability

You probably knew that Sun has done regular feature releases of the Java SE platform over the last 10 years. So we certainly feel like we've built up some expertise in this area, such as the ever growing 80,000 test cases and several million lines of code testing [conformance](#) (being just one aspect of our testing activity). You probably noticed that, unlike the last release, people have been downloading binary [snapshots](#) for the last 20 (not just 6) months. And what's more, they've been filing bugs. So, before we even got to [beta](#), we'd fixed a number of quality and regression issues. Doesn't that add up to a better product? Oh, and by the way, [performance](#) is looking better than J2SE 5.0.

So now you know all you need to know. Go on, try it! If you need more persuasion, peruse the following table of approved features.

Features

The following summary of the approved feature list is sorted by area, component, and feature name. The first part of the table highlights platform specification features; the second part lists features co-bundled with the Java Development Kit (JDK). For more details, please see the [Release Contents specification](#). You can check out the initial implementations of many – though not all – of these new features in the [weekly snapshot builds](#) of the reference implementation.

Client (Desktop)

2D - GIF image writer

AWT

- Access to Desktop helper applications
- Fast splash screens
- Improved modal dialogs
- System-tray support

Internationalization

- Pluggable locale data
- Resource-bundle enhancements
- Unicode string normalization

Swing

- Baseline/gap APIs
- Improve Swing drag-and-drop
- `JTabbedPane`: Tabs as components
- `JTable` sorting, filtering, and highlighting
- `SwingWorker`
- Text-component printing

Core

JSR 223: Scripting for the Java Platform

Debug

- Access to heap contents
- Attach-on-demand
- Multiple simultaneous agents

Libs

- Array reallocation
- Collections: Deques
- Collections: Sorted sets and maps with bidirectional navigation
- Critical file-I/O enhancements
- Floating point: Add core IEEE 754 recommended functions
- `java.util.concurrent` updates

JSR 202: Java Class-File Specification Update

 - Password prompting
 - Reflective access to parameter names
 - Service-provider lookup

Management & Monitoring

- Generalized lock monitoring
- Generalized MBean descriptors
- Generic annotations for MBean descriptor contents
- MXBeans

Net

- Internationalized domain names
- Internationalized resource identifiers
- Programmatic access to network parameters
- Simple HTTP cookie manager

Security

- **JSR 105**: XML Digital-Signature APIs

Tools

- **JSR 199**: Java Compiler API
- **JSR 269**: Pluggable Annotation-Processing API

Enterprise

JSR 250: Common annotations

JDBC

- **JSR 221**: JDBC 4.0

XML

- JavaBeans Activation Framework (JAF) 1.1
- **JSR 173**: Streaming API for XML (StAX)
- **JSR 181**: Web Services Metadata
- **JSR 222**: Java Architecture for XML Binding (JAXB) 2.0
- **JSR 224**: Java API for XML Web Services (JAX-WS) 2.0

Features Co-Bundled With the JDK

Java DB, based on Apache Derby

Rhino JavaScript engine

For More Information

- [Java SE 6 Launch Page](#)
- [Java SE 6 Download Page](#)
- [Reference implementation](#)
- [Rhino JavaScript engine](#)
- [Java DB, based on Apache Derby](#)
- [Exciting Enhancements Coming for Desktop Java](#)
- [Demos Dazzle at the Java Platform Roadmaps Session](#)
- [Core Java Technology Features](#)
- [More Enhancements in Java SE 6](#)
- [Desktop Features](#)
- [Using the Desktop API](#)
- [The New Modality API](#)

Rate and Review

Tell us what you think of the content of this page.

Excellent Good Fair Poor

Comments:

Your email address (no reply is possible without an address):

[Sun Privacy Policy](#)

Note: We are not able to respond to all submitted comments.

Submit >