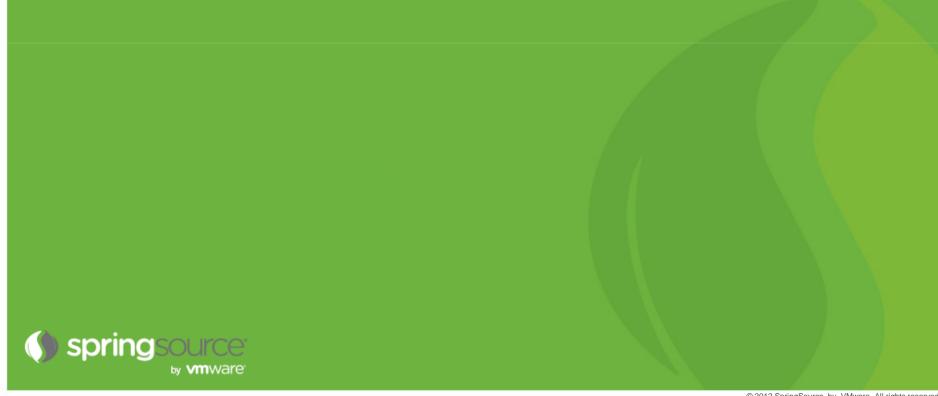
Apache Tomcat 8 Preview

Mark Thomas, Staff Engineer



Agenda

- Introductions
- Java EE 7
- Tomcat specific changes
- Timescales
- Final thoughts
- Questions

Introductions

Introductions

- Mark Thomas
- Apache Tomcat committer (markt)
- Other ASF
 - Infrastructure team
 - Security
 - Commons
 - Member
- Staff Engineer at VMware
 - Tomcat
 - Security
 - tc Server
 - support

Java EE 7

Java EE 7

- Servlet 3.1
- JSP 2.3
- Expression Language 3.0
- WebSockets 1.0
- Little / no demand for other Java EE 7 components in Tomcat
- Web container Apache TomEE
- J2EE container Apache Geronimo

Servlet 3.1

- I am part of the Expert Group
- Latest draft: June 2012
- New features
 - Non-blocking IO
 - HTTP upgrade
 - Change session ID on authentication
- Improvements
 - Clarified some ambiguities
 - Fixes some typos

Servlet 3.1

TBD features

• Protection for uncovered HTTP methods in security constraints

Dropped features

Overlays

JSP 2.3

- There is no JSP expert group
- JSP 2.3 will be a maintenance release
- JSP 2.2 arrived very late in the Java EE 6 cycle
 - Expect JSP 2.3 to be the same
- Little to do in the JSP specification
 - Updates in light of EL specification changes
- Possible changes / clarifications
 - Should JSPs respond to all HTTP methods?

EL 3.0

- I am part of the Expert Group
- Significant change
- Access to static methods and fields
- Lambda expressions
 - ((x,y)->x+y)(3,4) evaluates to 7
 - fact = n -> n==0? 1: n*fact(n-1); fact(5) evaluates to 120

EL 3.0

Collection operators

- Construction of Sets, Lists, Maps
- Was going to provide Language INtegrated Query (LINQ) support
- Will now align with Java SE 8

Other new operators

```
• String concatenation { + , cat }
```

```
Assignment =
```

Semicolon ;

A;B -> evaluate A & discard; evaluate B and return

• Class T(classname)

WebSocket 1.0

I am part of the Expert Group

- Tomcat 7 already supports WebSocket
 - Uses own API
- Tomcat 8 will support new API
 - Tomcat 7 will probably be changed to use new API
 - Current API is marked as subject to change
- WebSocket 1.0 is for clients and servers
 - Definitely implement the server side
 - Client side implementation TBD

WebSocket 1.0

Additional features

- Encoding / decoding (lots of debate here)
- Annotations

Differences

- Tomcat 7's implementation is blocking within a Frame
- WebSocket 1.0 will be non-blocking

Non-blocking

- Sort of works with the BIO connector but not really
- Fundamentally changes the API

Resources

- Aliases
- VirtualDirContext / VirtualWepappLoader
- External repositories for the WebappClassLoader
- Servlet 3.0 resource JARS
- Tomcat 7 implements each of these slightly differently
 - Very fragile
 - Servlet 3.1 overlays were going to be difficult
- New resources implementation
 - Probably not needed for overlays
 - Much cleaner implementation
- Recently moved from sandbox to trunk

NIO connector is now the default

- Additional diagnostic information in the Manager
 - SSL ciphers
 - May be back-ported to Tomcat 7
 - Ideas for additional features

API changing clean-up

- Remove duplicated functionality
- Move Manager, Loader and Resources from Container to Context
- Move Mapper from Connector to Service

Code clean-up

- Reduce warnings
- IDE, FindBugs, Javadoc, Checkstyle, etc

Timescales

Timescales

Java EE 7 final due Spring 2013

- unlikely to slip
- features may be removed

Tomcat 8

- 8.0.0 release will be after Java EE 7 is final
- no firm plan for how long after
- need to implement all the features
- need to pass the TCKs
- 8.0.0 may be preceded by some milestone releases

Progress to date

Non-blocking IO

Partially implemented by Filip Hanik

HTTP upgrade

- Implemented but not using Servlet specification API
- Servlet API implementation in progress (very early stages)

Overlays

Resource handling refactoring complete to support this

WebSocket

- Implemented but not using specification API
- Currently blocking within a WebSocket frame and needs to be non-blocking

EL 3.0

Not implemented

Final thoughts

Final thoughts

Java EE 7

- not final
- still a moving target

Tomcat 8

- depends on volunteer development effort
- needs community agreement, particularly for big changes

Help wanted

- Coding (new features, bug fixes, clean-up, de-duplication, more tests)
- documentation
- testing

Questions