Pivotal

A NEW PLATFORM FOR A NEW ERA
Migrating to Apache Tomcat 8

Mark Thomas, Staff Engineer, Pivotal

9 April 2014
Agenda

• Introductions
• Specification changes
• Tomcat changes
• Summary
• Questions
Introductions
Introductions

• markt@apache.org
  – Apache Tomcat committer since December 2003
  – Apache Tomcat PMC member from the beginning

• Tomcat 8 release manager
• Member of the Apache Tomcat security team
• Apache Commons PMC member
• Member of the Apache Infrastructure team
Introductions

• Staff Engineer at Pivotal
• Primary role is to work on Apache Tomcat
• Pivotal tc Server
  – Based on Tomcat
  – Keep tc Server updated as new Tomcat versions are released
• 3rd line support for Tomcat and tc Server
• Lead the Pivotal security team
Why This Presentation?

- Lots of information available about what is new in Tomcat 8
- J2EE specifications are meant to be backwards compatible
  - That isn’t always the case
- Tomcat configuration often changes between versions
- Not so much information on what might break when you switch to Tomcat 8
Specification changes
Specification Changes

• Servlet 3.1 (JSR 340)
• JSP 2.3 (JSR 245 maintenance release)
• EL 3.0 (JSR 341)
• WebSocket 1.0 (JSR 356)
Specification Changes: EL 3.0

- Coercion of nulls to Number, Character or Boolean
- EL 2.2 and earlier (0, 0, false)
- EL 3.0 and later (null, null, null)
- System property
  - org.apache.el.parser.COERCE_TO_ZERO
  - Set to true for EL 2.2 behaviour
Specification Changes: JSP 2.3

- Minor changes to reflect the changes in EL 3.0

- JSP 2.3
  - Supported: GET, POST and HEAD
  - Undefined: Everything else

- JSP 2.2 and earlier
  - Undefined: Most implementations assumed all

- Tomcat only permits GET, POST and HEAD
  - Protection against verb tampering
Specification Changes: WebSocket 1.0

• Tomcat 7 initially shipped with a proprietary WebSocket API
• Tomcat 8 ships with a JSR 356 WebSocket implementation
  – Also back-ported to Tomcat 7
• The proprietary WebSocket API is deprecated in Tomcat 7
• Applications using the proprietary API need to migrate
  – Relatively simple
  – https://svn.apache.org/r1424733
Specification Changes: Servlet 3.1

- Session ID changes by default on authentication
  - Prevents session fixation
Specification Clarifications: Servlet 3.1

• Annotation scanning
  – Independent of version declared in web.xml
  – Declare version 2.5, all annotations (@WebServlet etc.) are still processed
  – Applications may change behaviour moving from Tomcat <=6 to Tomcat >=7

• Absolute ordering
  – Excluded JARs are not scanned for SCIs
  – Excluded JARs are not scanned for matches for SCIs
  – Excluded JARs are not scanned for annotations
Specification Clarifications: Servlet 3.1

- **metadata-complete**
  - Does not control scanning of SCIs
  - Does not control scanning of matches to SCI

- **Container provided SCIs are always loaded**

- **Async processing**
  - `AsyncContext.getRequest|Response]()` throws ISE outside of async processing
  - Default async timeout is 30 seconds
  - Listener exceptions are logged but do not affect other listeners
Tomcat Changes
Tomcat Changes: Connectors

- Default connector has changed from BIO to NIO
  - BIO is likely to be dropped for Tomcat 9
- Only BIO option not supported by NIO is irrelevant for NIO
  - disableKeepAlivePercentage
- May notice different network / CPU loads
  - More established, idle connections
  - Marginally higher CPU load
Tomcat Changes: Static Resources

• Tomcat 7
  – Aliases
  – VirtualLoader
  – VirtualDirContext
  – JAR resources
  – External repositories

• All variations on a theme

• Each implemented differently
Tomcat Changes: Static Resources

• Tomcat 8
  – New WebResources implementation
    ▪ JAR resources
  – External resources for class loader

• Completely new configuration

• Caching attributes removed from Context
Tomcat Changes: Static Resources

• Resources now defined by as:
  – Pre-resources
  – Main resources
  – JAR resources
  – Post-resources
Tomcat Changes: Static Resources

- Resources attributes:
  - base
  - internalPath
  - webappMount
  - readOnly
Tomcat Changes: Internal APIs

• Lots of changes
  – Manager, Loader and Resources are now Context only
  – Mapper moved from Connector to Service
  – WebResources

• If you extend a Tomcat class, review the API docs
Tomcat Changes: Database Connection Pools

- Tomcat 7 and earlier based on DBCP 1
- Tomcat 8 based on DBCP 2
- Better performance in concurrent environments
  - Comparable to Tomcat’s JDBC pool
- There are some changes to configuration attributes
  - Reflect consistency changes made in Commons Pool 2
Tomcat Changes: Database Connection Pools

- `maxActive` -> `maxTotal`
- `maxWait` -> `maxWaitMillis`
- Validation no longer requires a validation query
  - Uses `Connection.isValid()`
Tomcat Changes: Cookies

- The Cookie specifications are a mess
- Netscape Cookie spec (version 0) is full of holes
- RFC2109 (version 1) is generally OK but few browsers get even close to implementing it correctly
- RFC2965 (version 2) isn’t really implemented at all
- Yet, generally, cookies work
- How come?
Tomcat Changes: Cookies

- RFC6265 pragmatic documentation of what has actually been implemented (mostly)
- Tomcat currently tries to correctly implement RFC2109
  - Has various workarounds for ‘broken’ clients
  - Has a number of bugs as identified by jboynes
  - Isn’t going to like it when UTF-8 data starts appearing in cookies
- Jeremy Boynes has done a very through review
Tomcat Changes: Cookies

- Only changes so far have been code clean-up
- Changes need to be completed before Tomcat 8 is declared stable
  - Likely to be my next focus for Tomcat development
- Should be backwards compatible
- Cookie changes nearly always break something for someone
- Review the wiki page for full details
Summary
Summary

- Unlikely to be affected by most of the issues discussed here
  - Cookies?

- The ASF’s JIRA instance switched to Tomcat 8 without hitting any of these issues
  - We did uncover a few Jira bugs though

- Help is always available on the Apache Tomcat user’s mailing list
Questions