Introducing Apache Tomcat 7

Mark Thomas - SpringSource
Agenda

- Introduction
- Overview
- Servlet 3.0
- JSP 2.2
- EL 2.2
- Other new features
- Current status
- Useful resources
- Questions
Introduction

• Tomcat committer & PMC member
• Wrote a large proportion of the updates for Tomcat 7
• Tomcat 7 release manager

• Apache Commons (DBCP & Pool) committer
• ASF member
• ASF Infrastructure team
• ASF security team member
• Lead the SpringSource security team
• tc Server developer
## Overview

<table>
<thead>
<tr>
<th></th>
<th>Tomcat 4</th>
<th>Tomcat 5</th>
<th>Tomcat 6</th>
<th>Tomcat 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Servlet</td>
<td>2.3</td>
<td>2.4</td>
<td>2.5</td>
<td>3.0</td>
</tr>
<tr>
<td>JSP</td>
<td>1.2</td>
<td>2.0</td>
<td>2.1</td>
<td>2.2</td>
</tr>
<tr>
<td>EL</td>
<td>(2.0)</td>
<td></td>
<td>2.1</td>
<td>2.2</td>
</tr>
<tr>
<td>Java</td>
<td>1.2?</td>
<td>1.4</td>
<td>1.5</td>
<td>1.6</td>
</tr>
</tbody>
</table>
Servlet 3.0
Servlet 3.0

- Asynchronous processing
- Pluggability
- Annotations
- Session management
- Miscellaneous
Servlet 3.0 – Asynchronous processing

• Response processing can now be asynchronous
  – Request processing is still synchronous
• Outline
  – start asynchronous processing
  – request/response passed to a new thread
  – container thread returns to the pool
  – new thread does its work
  – new thread completes response or forwards to another resource (e.g. JSP)
Servlet 3.0 – Asynchronous processing

• Allows more efficient use of container threads
  – when waiting for external resources
  – when rationing access to a resource
  – or any other time when the container thread would be blocking

• Allows separation of request and response
  – chat applications
  – stock tickers

• All filters, servlets and valves in the processing chain must support asynchronous processing
Servlet 3.0 - Pluggability

- Annotations
- Web fragments
- Static resources in JARs
- Programmatic configuration

- Purpose
  - Improve productivity
  - Remove the need to web.xml
Servlet 3.0 - Pluggability

• Pros
  – development can be faster
  – applications can be more modular

• Cons
  – fault diagnostics are significantly hampered
  – might end up enabling things you don’t want or need

• Overall, I don’t recommend using it for production

• Instead:
  – get Tomcat to generate the equivalent web.xml
  – use the equivalent web.xml instead
Servlet 3.0 - Pluggability

• JARs can contain their own web.xml
  – /META-INF/web-fragment.xml
  – essentially same content as web.xml
  – rules for determining fragment ordering

• Allows JARs to be self-contained

• JARs can also contain static resources
  – always used, cannot be excluded by fragment ordering
  – non-deterministic if there are duplicate resources in multiple JARs
Servlet 3.0 - Annotations

- Servlets, Filters, Listeners
  - Can be placed on any class
  - Tomcat has to scan every class on application start
- JARs scanned if included in fragment ordering

- Security, File Upload
  - Placed on Servlets
  - Processed when class is loaded

- File Upload has almost – but not quite – the same API as Commons File Upload
Servlet 3.0 – Programmatic configuration

• Allows a sub-set of what is possible in web.xml
  – Add Servlets, Filters and Listeners
  – Change session tracking
  – Configure session cookies
  – Configure security
  – Set initialisation parameters

• Allows greater control / optional configuration
Servlet 3.0 – Session tracking

- Adds tracking via SSL session ID
  - Must be used on its own
- Allows selecting of supported tracking methods
  - URL, cookie, SSL
- URL based tracking is viewed as a security risk
Servlet 3.0 – Session cookies

- Can control default parameters for session cookies
  - Name – may be overridden by Tomcat
  - Domain – may be overridden by Tomcat
  - Path – may be overridden by Tomcat
  - MaxAge
  - Comment
  - Secure – may be overridden by Tomcat
  - HttpOnly – may be overridden by Tomcat
Servlet 3.0 - Miscellaneous

- **httpOnly**
  - Not in any of the specifications
  - However, widely supported
  - Prevents scripts accessing the cookie content
  - Provide a degree of XSS protection

- **Programmatic login**
  - Useful when creating a new user account
  - Can log the user in without redirecting them to the login page
  - Allows the application to trigger a login
JSP 2.2
<jsp-config>
    <jsp-property-group>
        <url-pattern>*.jsp</url-pattern>
        <default-content-type>text/html</default-content-type>
    </jsp-property-group>
    <jsp-property-group>
        <url-pattern>*.jsp</url-pattern>
        <buffer>4096</buffer>
    </jsp-property-group>
    <jsp-property-group>
        <url-pattern>*.jsp</url-pattern>
        <error-on-undeclared-namespace>
            true
        </error-on-undeclared-namespace>
    </jsp-property-group>
</jsp-config>
JSP 2.2 - `<jsp:attribute .../>`

- Added support for the `omit` attribute
  - Use with EL to control the presence of an attribute
EL 2.2
EL 2.2 – Method invocations

• Now possible to invoke methods on a bean

```java
<%  
    TesterBeanA beanA = new TesterBeanA();  
    TesterBeanB beanB = new TesterBeanB();  
    beanB.setName("Tomcat");  
    beanA.setBean(beanB);  
    pageContext.setAttribute("testBeanA", beanA);  
    pageContext.setAttribute("testBeanB", beanB);  
%>
<tags:echo echo="00- 
    ${testBeanA["bean"].sayHello('JUnit')}" />
<tags:echo echo="01- ${testBeanA.bean.sayHello('JUnit')}" />
<tags:echo echo="02- ${testBeanB.sayHello('JUnit')}" />
```
EL 2.2 – Method invocations

• Correctly identifying the intended method is tricky
  – inheritance
  – varargs
  – narrowing and widening conversions
  – EL rules on coercion of Strings to types

• Likely to be some differences between containers

• Tomcat tries to do what the Java compiler does
Other Tomcat 7 changes
Tomcat 7 – Other changes

- Management
- Performance
- Security
- Code clean-up
- Extensibility
- Stability
- Flexibility
Tomcat 7 - Management

• Add the ability to fix the remote JMX ports
• Single line log formatter
• Manager app can distinguish between primary, backup and proxy sessions (for clusters)
• Aligned Mbeans with reality
  – GSoC 2010
• General improvements to JMX support
  – Can now have a server.xml with just a <Server .../> element and create a fully working Tomcat instance (Hosts, Contexts etc. all via JMX
Tomcat 7 - Performance

• Unlikely to see a step change
• Can limit the number of JSPs loaded at any one time
Tomcat 7 - Security

- Generic CSRF protection
- Separate roles for Manager and Host Manager apps
- Session fixation protection
  - changes session ID on authentication
- Enable the LockOutRealm by default
- Enable an access log by default
- Added ability to disable exec command for SSI
Tomcat 7 – Code clean-up

- Use generics throughout
- Remove deprecated code
- Remove unused code
- Reduce duplication, particularly in the connectors
- Better definition of Lifecycle interface
- Added Checkstyle to the build process
Tomcat 7 - Extensibility

- Added hooks for RFC66 – used by Virgo
- Re-factored to simplify Geronimo integration
- Significantly simpler embedding
Tomcat 7 - Stability

• Builds on Tomcat 6
• Tomcat 6 is already very stable
• Significant reductions in the open bug count
  – 6 open bugs without patches when I wrote this slide
  – for Tomcat 5.5.x, 6.0.x and 7.0.x combined
• Added unit tests
  – CI using BIO, NIO and APR/native on every commit
• Memory leak detection and prevention
  – Back-ported to Tomcat 6
Tomcat 7 - Flexibility

• Copying of /META-INF/context.xml is now configurable
• Alias support for Contexts
  – Map external content into a web application
• Shutdown address is now configurable
• Tomcat equivalent of some httpd modules
  – mod_expires
  – mod_remoteIP
Tomcat 7 status
Tomcat 7 - Status

- Passes Servlet 3.0 TCK
  - HTTP BIO
  - HTTP NIO
  - HTTP APR/native
  - AJP BIO (with httpd 2.2.15 & mod_jk)
  - AJP APR/native (with httpd 2.2.15 & mod_jk)
- Passes JSP 2.2 TCK
- Passes EL 2.2 TCK
- All with the security manager enabled
Tomcat 7 - Status

• 7.0.0 beta 29 June 2010
• 7.0.1 not released (failed to start with security manager)
• 7.0.2 beta 11 August 2010
• 7.0.3 not released (missing license header in source file)
• 7.0.4 in progress as I write the slides

• 7.0.4 is likely to be beta
Tomcat 7 - Status

• What is needed for Tomcat 7 to be stable?
• Three +1 votes from committers 😊

• In practice, the committers each have their own criteria
• I’m looking for two (maybe three) releases with
  – no major code changes that might cause regressions
  – TCKs all pass (already have this)
  – no major bugs reported
  – good levels of adoption (already have this)
Tomcat 7 plans
Tomcat 7 - Plans

• One release every month
  – bug 49884 put a spanner in the works
  – required a major refactoring of all of the connectors and the asynchronous handling
  – took about a month so delayed 7.0.3
  – hopefully back on track now
• Stable by the end of this year?
• Keep on top of the open bugs
• Work on bringing the open enhancement requests down
Tomcat 7 - Plans

• JSP 196 implementation?
  – The Java Authentication SPI for Containers (Servlet Container Profile)
  – Geronimo has most (all?) of this already

• Windows authentication
  – Looking unlikely – too much baggage
  – Waffle

• Simpler JNDI configuration for shared resources
  – No more <ResourceLink .../>

• More JMX improvements
Tomcat 7 - Plans

- Further improvements to the memory leak protection
- Continue migration from Valves to Filters
- Java EE 6 web profile
  - No interest (so far) from user community
  - Had more questions from journalists than users
  - No plans at present
Useful resources
Useful resources

• http://tomcat.apache.org
• http://blogs.apache.org/tomcat
• Mailing lists
  – users
  – dev
  – announce
• http://tomcat.markmail.org
• http://svn.apache.org/repos/asf/tomcat/trunk/webapps/docs/changelog.xml
• http://ci.apache.org/projects/tomcat/tomcat7/docs/
• http://s.apache.org/tomcat-7-open-issues
Questions