

# LOST IN THE DOCS

Felix Schumacher

Tomcat & JMeter

[fschumacher@apache.org](mailto:fschumacher@apache.org)

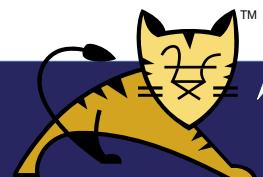


**APACHECON 2019**

EUROPE Oct. 22<sup>nd</sup> - 24<sup>th</sup>

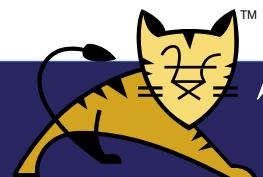
# AGENDA

- Where to find documentation
- Interesting bits found in the deep labyrinth of documentation



# CONVENTIONS USED ON THE SLIDES

- Shorten long package names from org.apache.tomcat to o.a.t (or c for catalina)



# LOST IN THE DOCS

- Possible reasons
  - Not reading the docs
  - Too much docs
  - Too many features
  - Source-Documented



# HOW TO READ THE DOCS

Four main sources for documentation

- User Guide
- Reference
- Wiki
- Source



# DOCS: TOMCAT USER GUIDE

Common use cases with sample code

<https://tomcat.apache.org/tomcat-9.0-doc>

When in doubt **Start here**



# DOCS: REFERENCE

List of all configurable elements

Grouped by function

<https://tomcat.apache.org/tomcat-9.0-doc/config/index.html>

Good when you **know what to look for**



# DOCS: WIKI

Community generated content

<https://cwiki.apache.org/confluence/display/TOMCAT>

Helpful when you are looking for broader  
**information about general usage**



# DOCS: SOURCE

Who needs documentation anyway

<https://github.com/apache/tomcat/>

Right place when you think documentation  
is **wrong or missing**



# DOCS: OTHER SOURCES

Have a look around on the internet

E.g. by searching “site:tomcat.apache.org stuck thread” or use the mailing list

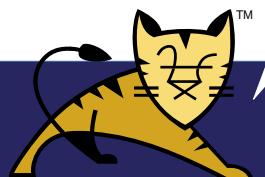
<http://tomcat.apache.org/lists.html>

For hard problems **mailing list is really helpful**



# LOST IN THE DOCS

- Randomly picked interesting topics
- Subjective choice



# PARALLEL DEPLOYMENT

- Enables deploying of new versions of your software without interruption
- Routes sessions to the correct version of your app
- Removes old versions when they are not used anymore (`undeployOldVersions`)



# PARALLEL DEPLOYMENT

## MORE DOCS

Mentioned in Tomcat Reference and User Guide:

<https://tomcat.apache.org/tomcat-9.0-doc/config/context.html>

<https://tomcat.apache.org/tomcat-9.0-doc/manager-howto.html>



# PARALLEL DEPLOYMENT

## USAGE

Deploy a new WAR with a name ending on  
**##VERSION\_STRING**

Example: R00T##001.war

Look out for warnings in the logs. You may hold on to your objects too tightly.



# LOST IN SYSTEM PROPS

Tomcat can be fine tuned using quite a few system properties

They are listed at the reference section

<http://tomcat.apache.org/tomcat-9.0-doc/config/systemprops.html>



# LOST IN TOO MANY SYSTEM PROPS

- System props can be defined in `conf/catalina.properties`
- System props are set for the whole JVM
- `CATALINA_OPTS` is used on startup only



# PROPERTY\_SOURCE

Inserts values into Tomcat's configuration files (context.xml, server.xml, web.xml)

**New** PropertySource that reads values from environment variables

Mentioned in Tomcat Reference:

<https://tomcat.apache.org/tomcat-9.0-doc/config/systemprops.html>



# PROPERTY\_SOURCE USAGE

In config file reference a value

```
<Context>
  <Resource name="jdbc/database"
            port="${db.port}"
```

```
</Context>
```

that is set by Java system property

```
CATALINA_OPTS="-Ddb.port=4223"
```



# PROPERTY\_SOURCE ADVANCED

o.a.t.util.digester.

PROPERTY\_SOURCE=my.PropertySource

Create **my.PropertySource** that implements  
o.a.t.util.IntrospectionUtils.PropertySource  
and place it in \${CATALINA\_BASE}/lib



# RECYCLE\_FACADES

Tomcat is re-using request/response objects

If you are **holding** onto those objects, you will get in **trouble**

This option gives you a new one every time



# RECYCLE\_FACADES

Set  
org.apache.catalina.connector.RECYCLE  
\_FACADES to **true**

Mentioned in Tomcat Reference

<https://tomcat.apache.org/tomcat-9.0-doc/config/systemprops.html>



# DEFAULTS FOR VIRTUAL HOSTS

Tomcat can set defaults for web.xml and context.xml based on virtual hosts

Useful to customize webapps for different parties on the same instance



# DEFAULTS FOR VIRTUAL HOST PREPS

Add a virtual host to server.xml inside Engine tag

```
<Host name="customerone"  
      appBase="portal-webapps"  
      unpackWars="true"  
      autoDeploy="true" />
```

Create the webapps and conf folder

```
$ mkdir -p conf/Catalina/customerone  
$ mkdir portal-webapps
```



# DEFAULTS FOR VIRTUAL HOSTS

## USAGE

- Put **web.xml.default** and/or **context.xml.default** files in conf/Catalina/customerone
- Possible content is same as `web.xml` and `context.xml`
- Order of element lookup in the files is
  - Files deployed with the webapp
  - Files found in the virtual host config dir
  - Files found in the default host  
(which is called `localhost` in default setup)



# DEFAULTS FOR VIRTUAL HOSTS

Mentioned in Tomcat Reference

<https://tomcat.apache.org/tomcat-9.0-doc/config/context.html>

Mentioned in Tomcat User Guide

<https://tomcat.apache.org/tomcat-9.0-doc/virtual-hosting-howto.html>



# SOURCE CODE DOCS

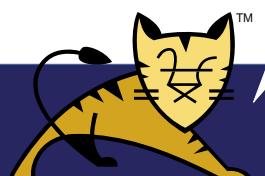
A new way of Tomcat's cluster setup using  
Kubernetes is documented in the source  
only

*OK, it can be found in the wiki, too*



# CLUSTER SETUP FOR KUBERNETES

- Tomcat's builtin session replication uses multicast to find cluster nodes
- Kubernetes doesn't like multicast
- Configure Membership Provider for Kubernetes
- Configure your application on Kubernetes



# CLUSTER SETUP FOR KUBERNETES (TOMCAT SIDE)

Add a cluster definition in server.xml to your Engine or Host tag

```
<Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster">
  <Channel
    className="org.apache.catalina.tribes.group.GroupChannel">
      <Membership
        className="o.a.c.t.membership.cloud.CloudMembershipService"/>
    </Channel>
</Cluster>
```



# CLUSTER SETUP DOCS

- <https://www.slideshare.net/jfclere/from-a-cluster-to-the-cloud>
- <https://github.com/jfclere/tomcat-kubernetes>
- <https://cwiki.apache.org/confluence/display/tomcat/ClusteringCloud>



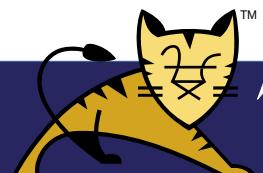
# VALVES

- A small selection of built-in valves
  - StuckThreadDetectionValve
  - SemaphoreValve
  - RewriteValve



# VALVES EXAMPLE 1

- **StuckThreadDetectionValve** detects requests that take too long
- Information about those requests is available through JMX and will be logged
- Optionally interrupts stuck threads



# VALVES EXAMPLE 2

- **SemaphoreValve** guards a resource from too much concurrency
- Most useful for synchronous servlets
- Guards all or nothing :(
- Subclasses can control concurrency by overwriting `SemaphoreValve#controlConcurrency(request, response)`



# VALVES EXAMPLE 3

- **RewriteValve** mimics Apache httpd rewrite module
- Use it for simple rewrite rules that are read from a file named **rewrite.config**



# LOST IN TOO MANY FEATURES

You may be wondering why Tomcat provides **two** different pool implementations for JDBC



# WHICH JDBC POOL

- **jdbc-pool**

fixed problems found in old dbcp based pool  
capable of advanced stuff  
development has stalled

- **tomcat-dbcp**

dbcp2 based pool (fixed the problems with old  
dbcp based pool)  
is actively maintained



# LOST ON THE RUN

An often overlooked gem is the file  
**RUNNING.txt** in the root directory of a fresh  
Tomcat installation



# **BIN/SETENV.SH**

Customize environment settings

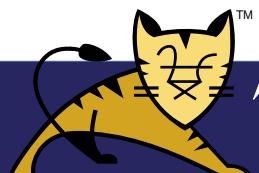
## **Don't edit catalina.sh or startup.sh**

- You don't need to
- Makes updates easier



# MULTI-INSTANCE SETUP

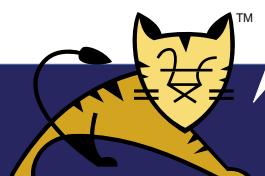
- easy updates and lightweight instances
- CATALINA\_HOME
  - points to the extracted binaries downloaded from Tomcat homepage, can be read only
- CATALINA\_BASE
  - points to a stripped down installation that gets laid over CATALINA\_HOME



# MULTI-INSTANCE SETUP

## USAGE

```
$ TCV=apache-tomcat-9.0.27
$ mkdir -p /srv/tomcat && cd /srv/tomcat
$ tar xf "/tmp/${TCV}.tar.gz"
$ mkdir -p
tc-instance/{bin,logs,webapps,temp}
$ cp -r "${TCV}/conf" tc-instance
$ export CATALINA_HOME="$PWD/${TCV}"
$ export CATALINA_BASE="$PWD/tc-instance"
$ "${CATALINA_HOME}/bin/startup.sh"
```



# RESOURCE FRAMEWORK

- Every “file” – classes, jars, static resources – in Tomcat is read through the resource framework. (Tomcat 8.5 and up)
- Replaces VirtualDirContext and VirtualWebappClassLoader from Tomcat 7.0 and below



# RESOURCES EXAMPLE

```
<Context>
  <Resources>
    <PreResources
      base=
        "${catalina.base}/special-config.jar"
      className=
        "o.a.c.webresources.JarResourceSet"
      webAppMount=
        "/WEB-INF/classes" />
    </Resources>
  </Context>
```



# RESOURCES TYPES

- WebResourceSet (Interface)
- DirResourceSet (files as files)
- FileResourceSet (one file as a file)
- JarResourceSet (files inside jar as files)



# RESOURCES ORDERING

- PreResources
- MainResources
- ClassResources
- JarResources
- PostResources



# LOST IN THE DOCS

- Thanks for listening
- Questions?



# LOST IN THE DOCS

- Thanks for listening
- Find missing docs and add them

