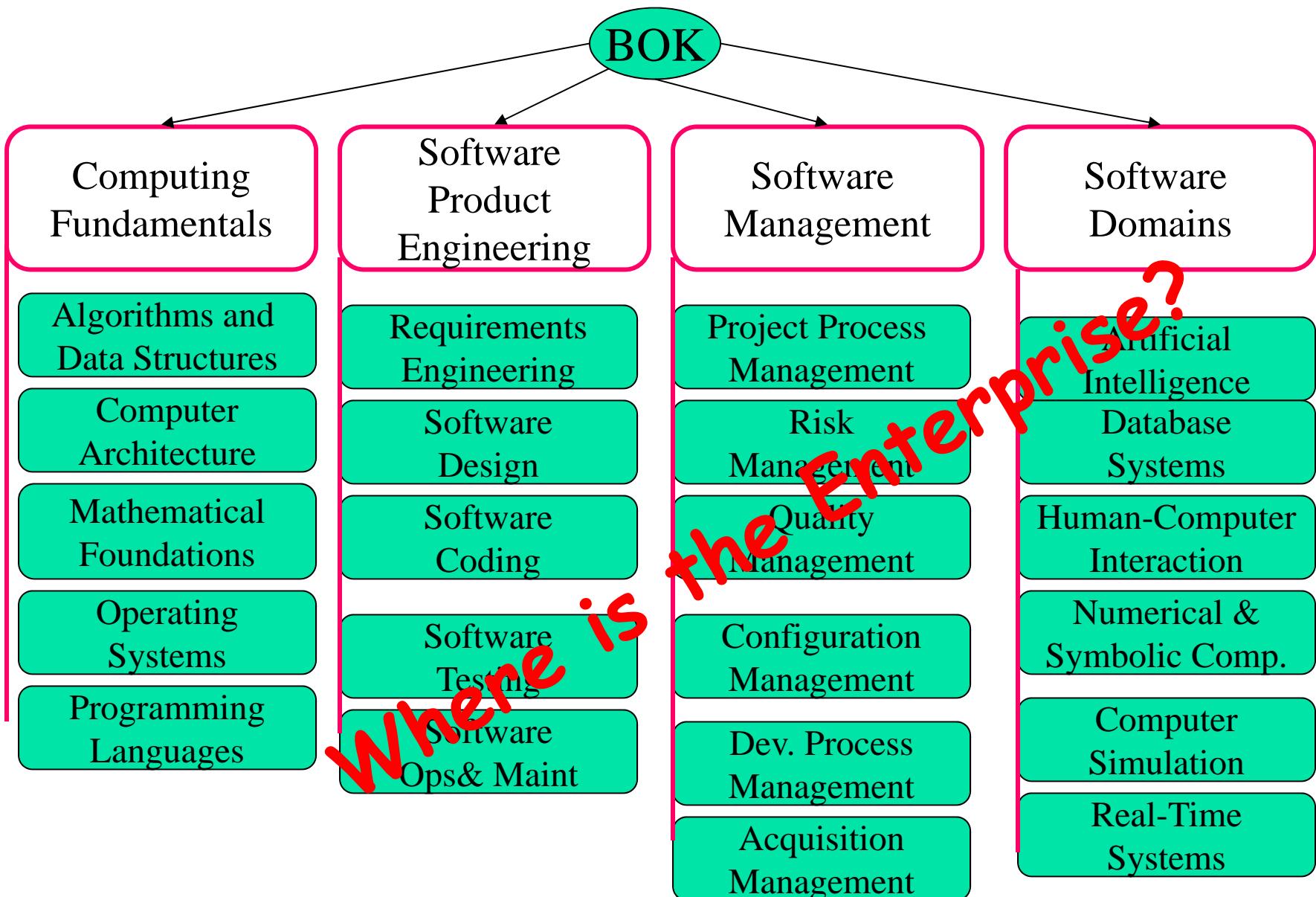


Topics that we will explore

- Web application architecture and application servers
- J2EE framework
- Java Servlets
- JavaServer Pages (JSP)
- Java Beans
- RMI
- Multithreading
- AJAX
- Java Database Connectivity (JDBC)

Software Engineering Body of Knowledge



Overview

- What is an enterprise?
- What constitutes an enterprise?
- What is enterprise computing (EC)?
- What is the nature of an EC environment?
- What makes a good enterprise application?

An Enterprise is...

any organization with set of goals

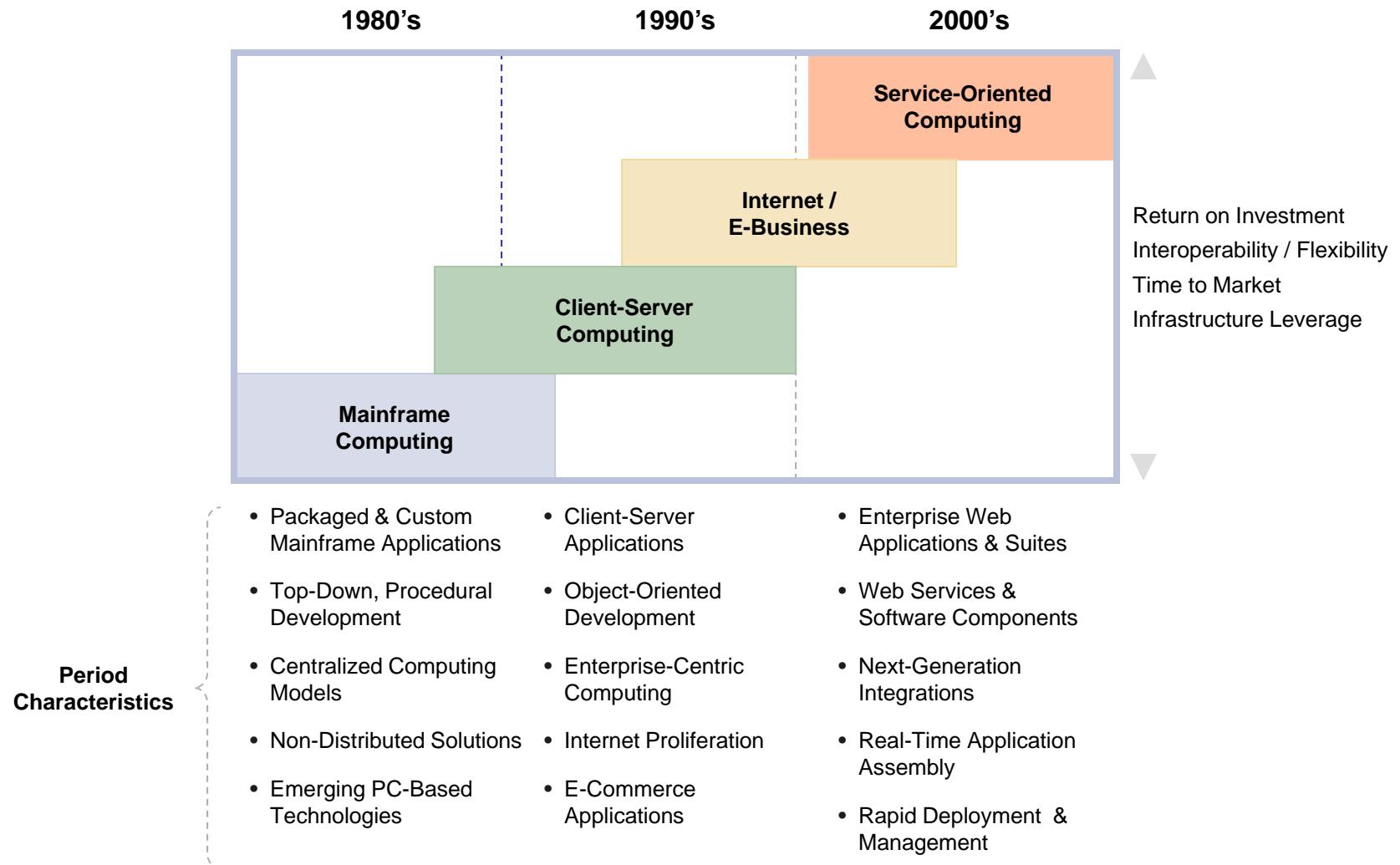


Enterprise Computing (EC)

“Enterprise computing involves the development, deployment and maintenance of the information systems required for survival and success in today’s business climate.”

Yen-Ping Shan & Ralph H. Earle,
Enterprise Computing with Objects,
Addison-Wesley, 1998.

Evolution of The Enterprise Computing



EC Environment

- computers are typically dispersed over a wide area
- old (“legacy”) applications need to be used/maintained
- very large databases
 - (e.g., tens of databases (DB), hundreds of tables per DB, thousands to millions of records per table)
- hundreds to thousands/millions of clients
- heterogeneous
 - computers, client devices, networks, applications
- constantly-changing business requirements
- mission-critical applications
 - requires reliability and performance

Enterprise Application Qualities

- extensible -- allows for future changes with minimal impact
- scalable -- gracefully handles expansion/contraction in number of clients
- usable/reliable -- functions as advertised, including error-handling
- available -- can be used whenever needed
- configurable -- can be adapted to diverse environments
- deployable -- can be easily distributed to users
- durable -- necessary information lasts/persists over time
- efficient/responsive -- uses minimal resources/perform well for clients
- unobtrusive -- doesn't get in the way of getting things done
- secure -- only authorized clients can access
- reusable -- many parts can be reused rather than re-created
- maintainable -- can be fixed with minimal impact
- timely -- is ready to use productively during window of opportunity

The Impact of Standards

Content Management	Portals Applications	Legacy Applications	Client Server Apps.	Web Services Apps.
Enterprise Web Applications Development, Deployment, & Management				
(Management of Portals, Web Services, Content, Applications & Web Initiatives)				
Oracle9iAS	WebSphere App. Server	Sun ONE App. Server	WebLogic Server	
Oracle E-Commerce	WebSphere Commerce PE	—	WebLogic Portal	
Oracle Workflow	MQSeries Workflow		WebLogic Integration	
Oracle9iAS Integration / InterConnect	CrossWorlds / WebSphere MQ Integrator Broker	Sun ONE Integration Server, EAI Edition	WebLogic Server / Tuxedo	
Oracle Advanced Queuing	IBM WebSphere MQ	Sun ONE Message Queue	—	
Oracle 9i Database	DB/2 Database	—		

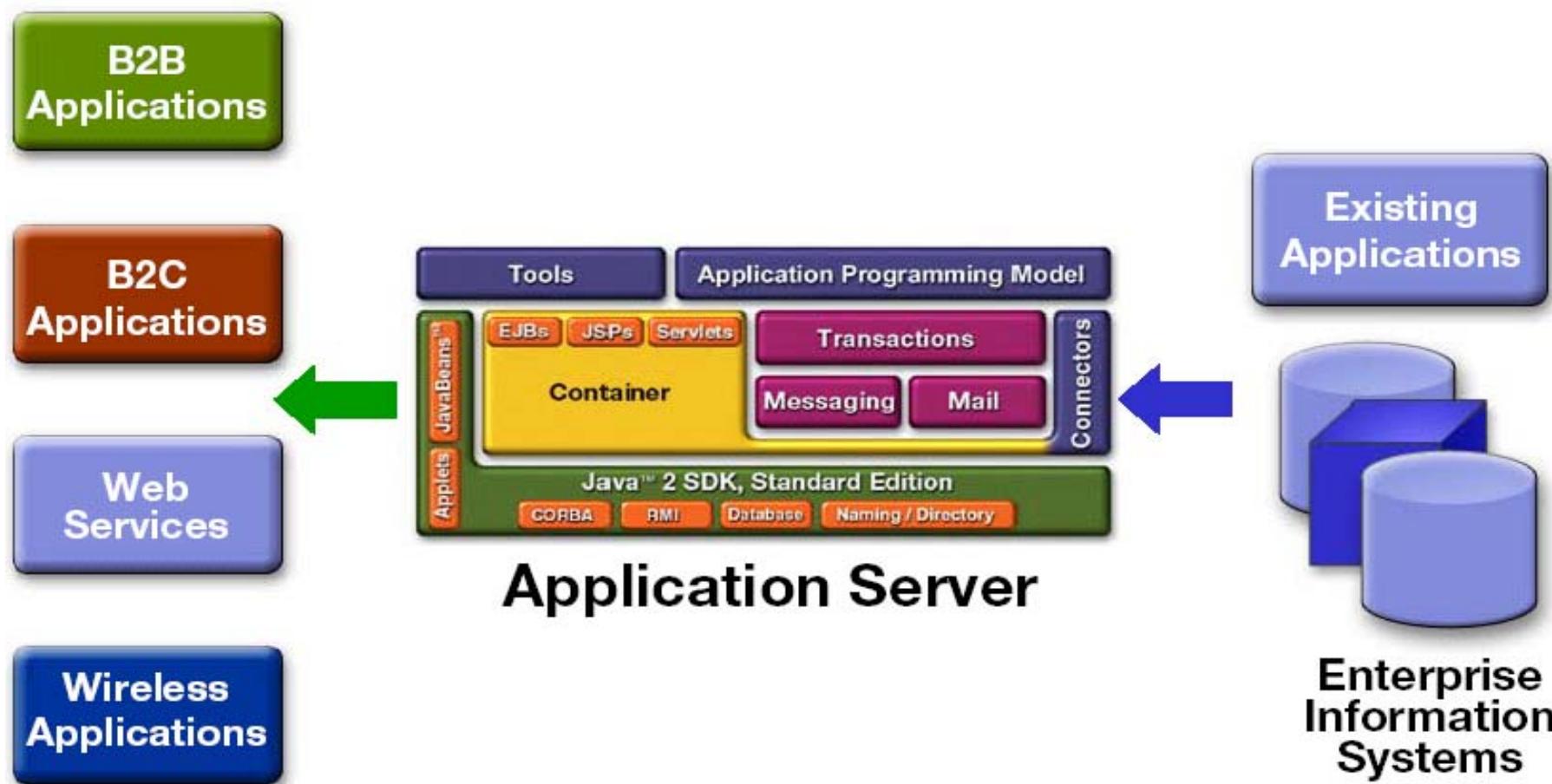
ORACLE

IBM

 *Sun*
microsystems

 *bea*™

The J2EE Platform, Architecture

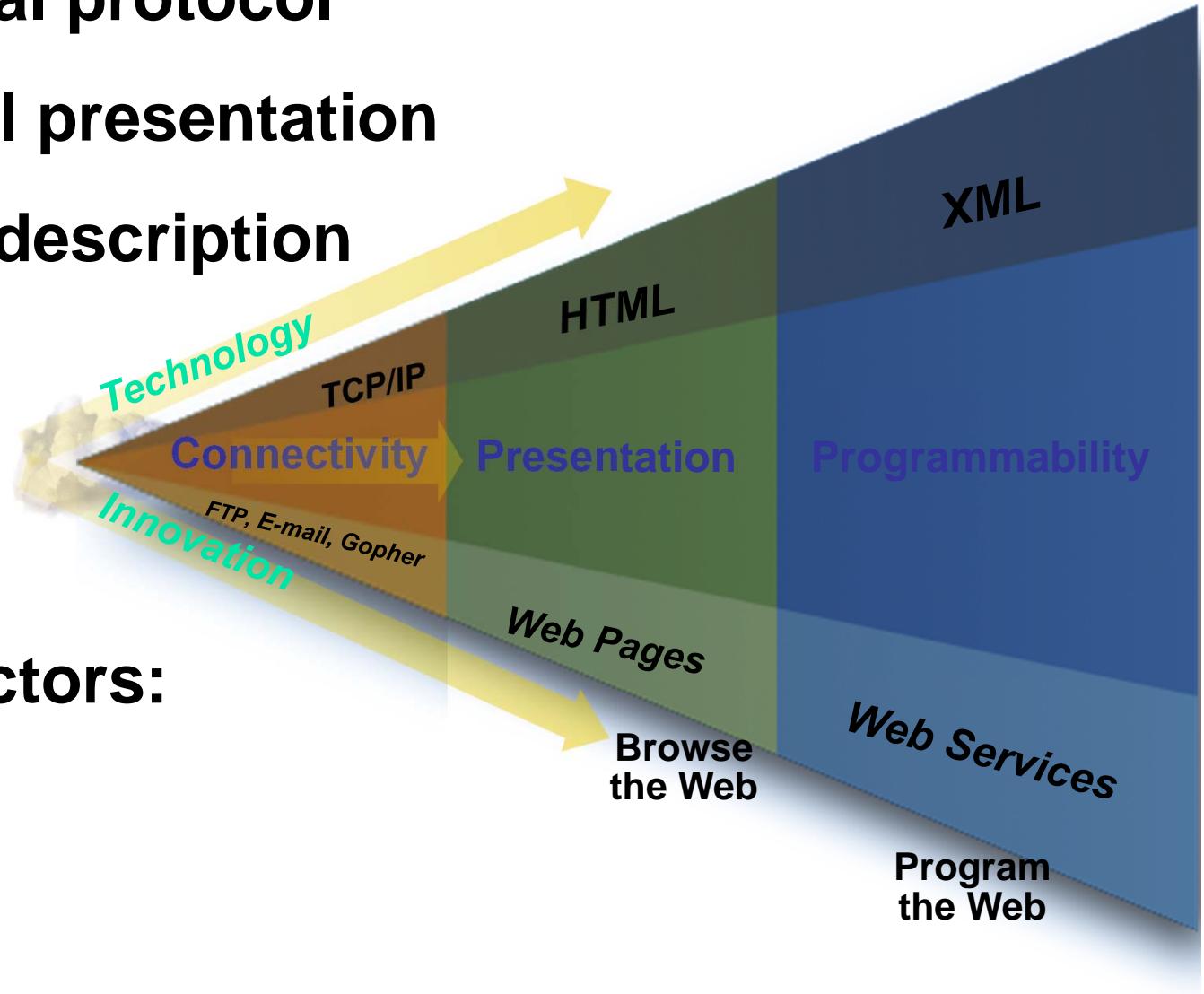


Evolution To Web Services

TCP/IP: universal protocol

HTML: universal presentation

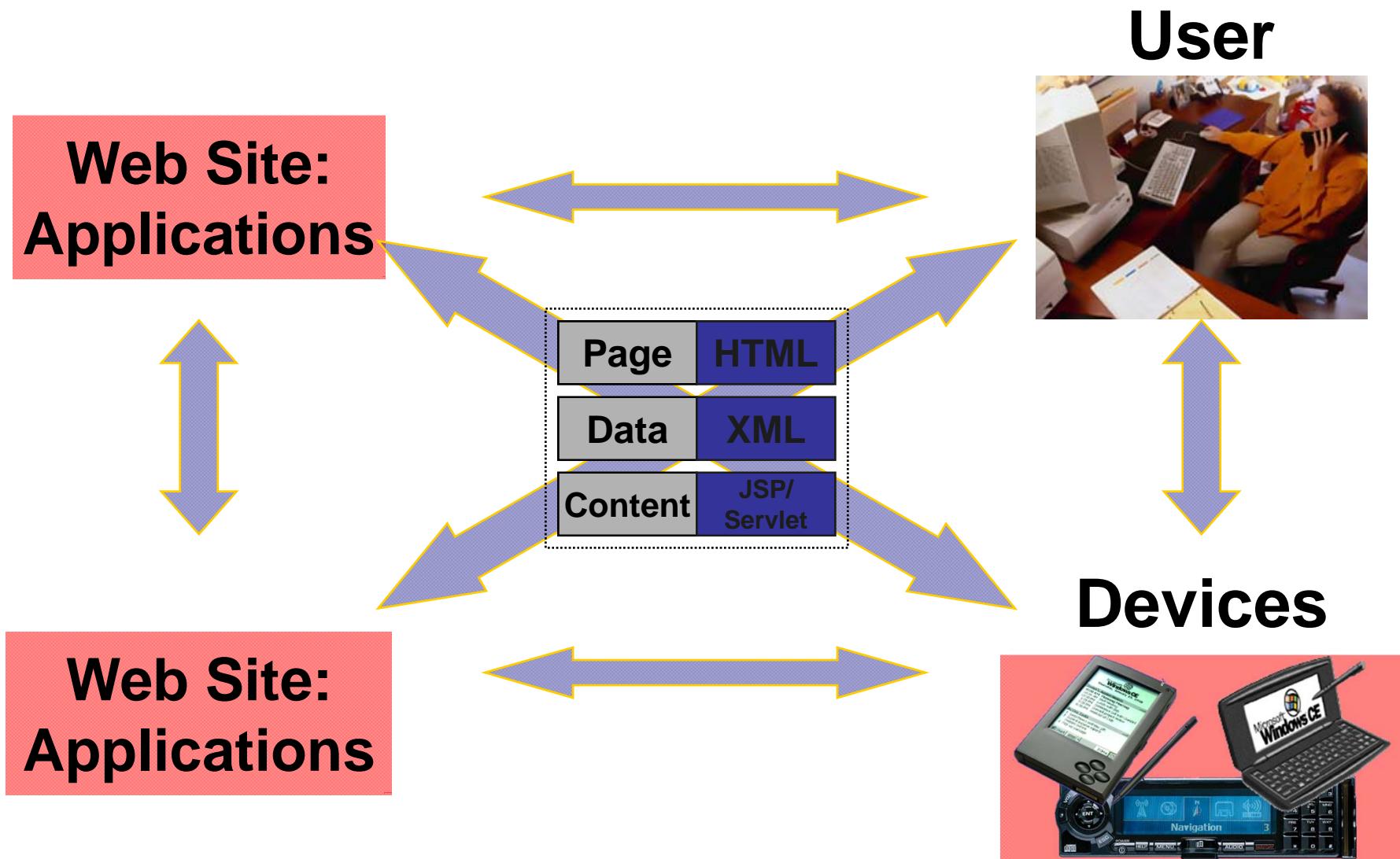
XML: universal description



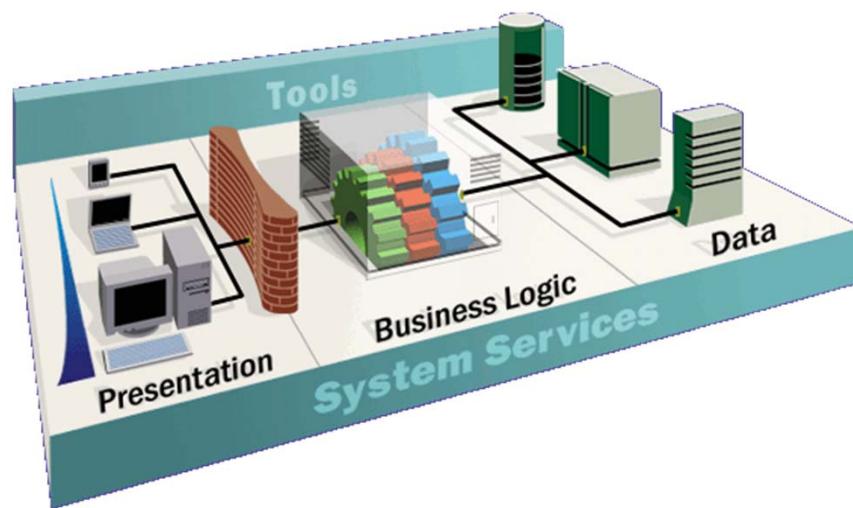
Key success factors:

- Simplicity
- Standards

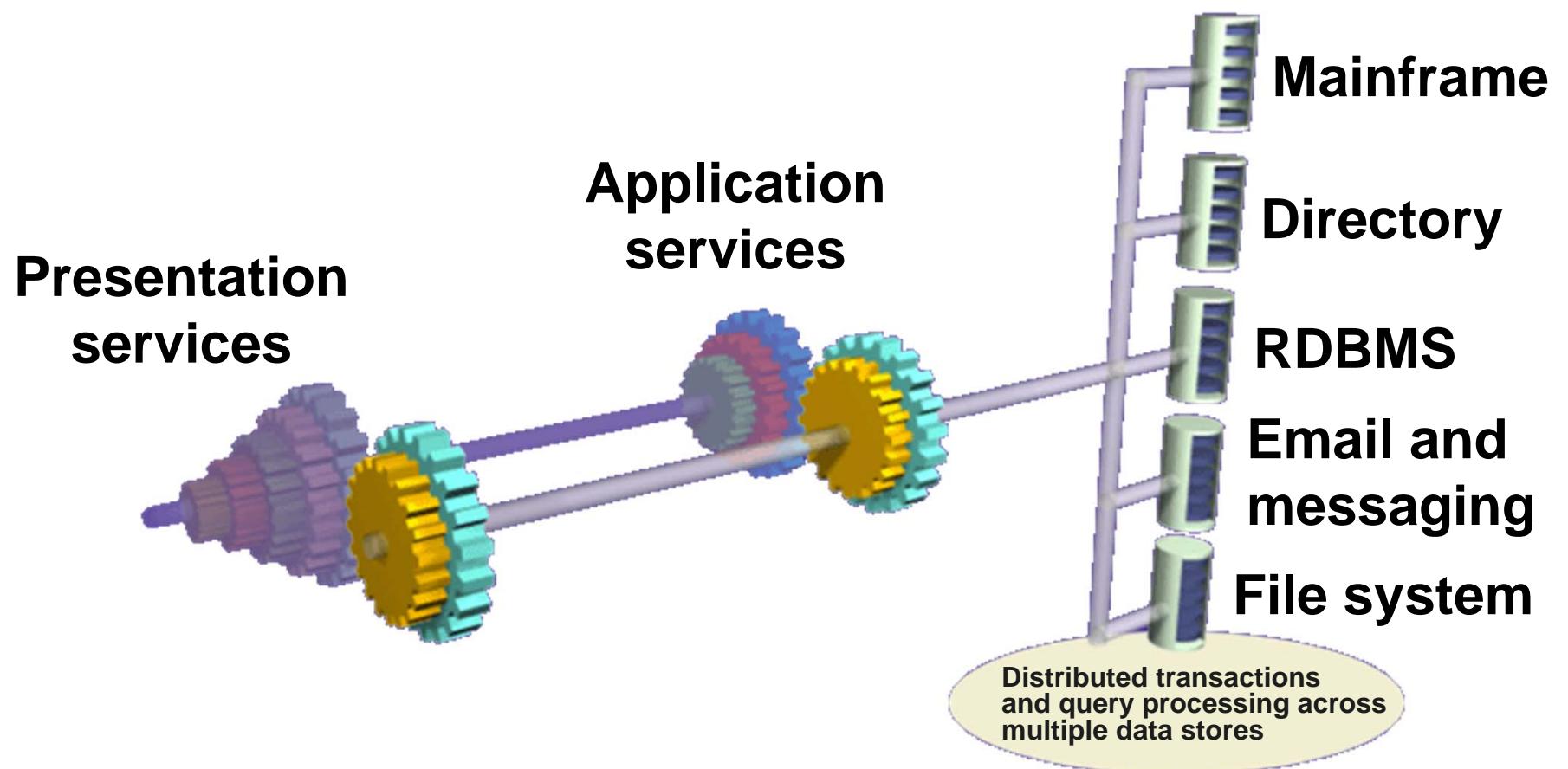
The Web Architecture



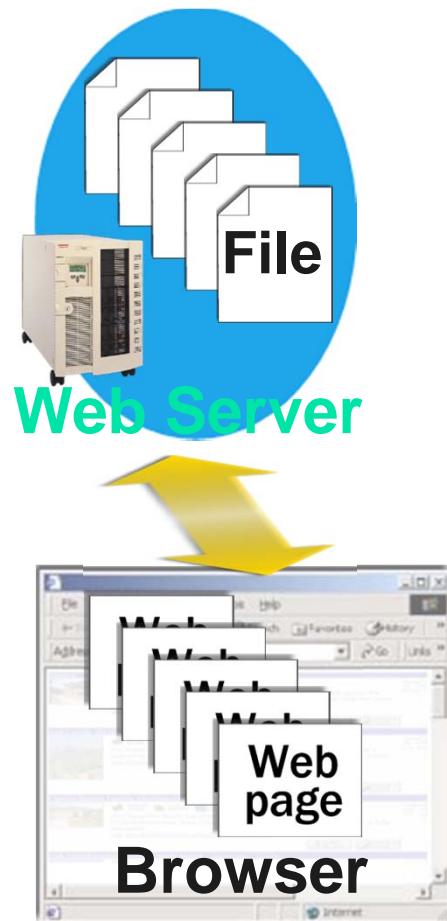
Tiers, Tools and Services



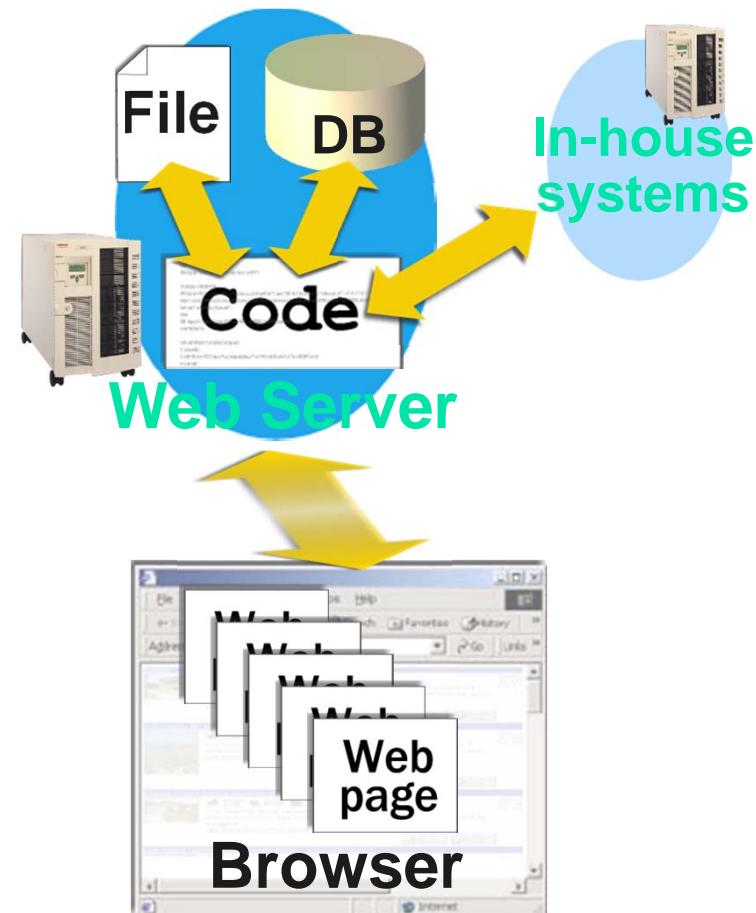
The Enterprise



1st & 2nd Generation Web Apps



1-1 correspondence
of page to file

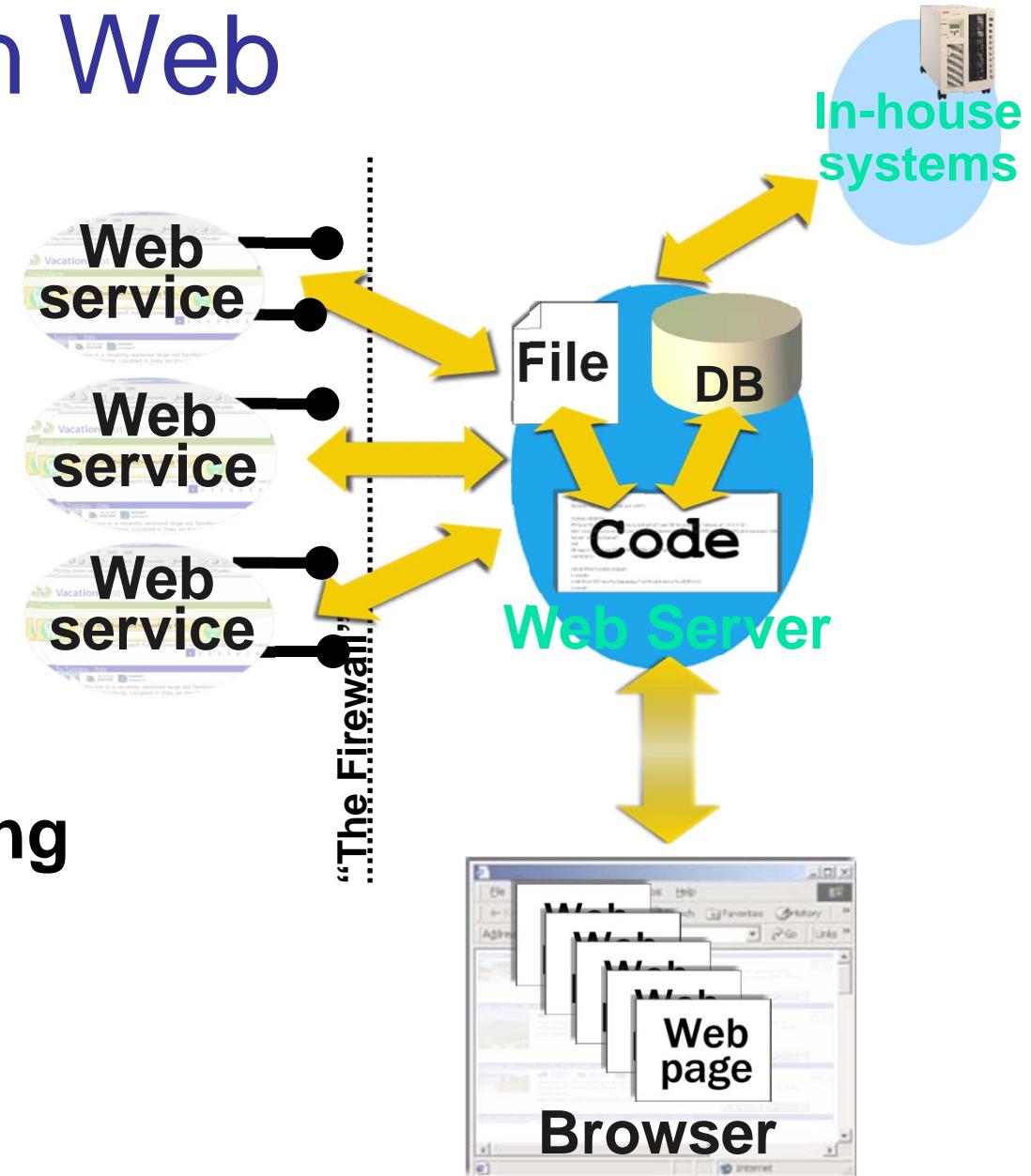


“Dynamic Pages”

3rd Generation Web

**Accessing another site today = HTML
“screen scraping”
or “your architect calls my architect”**

**Tomorrow, external
Web sites become
building blocks using
XML**
**•Web Services
•Megaservices**



“Dynamic Pages”

Examples of Web Services

Location Services

Maps, routing, nearby locations...

Shopping Services

Order tracking, supply chain, auctions, coupons...

Information Services

Headlines, weather, horoscopes, TV times...

Communication Services

Email, instant messages...

Combining Web Services



Required Tools

- JDK 6 or JDK 7
 - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- Apache Tomcat 7.0.34
 - <http://tomcat.apache.org/whichversion.html>
 - <http://www.coreservlets.com/Apache-Tomcat-Tutorial/>

Java SE Downloads - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Java SE Downloads

www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html

Java Magazine

JDK MD5 Checksum

Java SE Development Kit 7u40

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Accept License Agreement Decline License Agreement

Product / File Description	File Size	Download
Linux ARM v7 Hard Float ABI	67.62 MB	jdk-7u40-linux-arm-vfp-hflt.tar.gz
Linux ARM v6/v7 Soft Float ABI	67.62 MB	jdk-7u40-linux-arm-vfp-sflt.tar.gz
Linux x86	115.55 MB	jdk-7u40-linux-i586.rpm
Linux x86	132.83 MB	jdk-7u40-linux-i586.tar.gz
Linux x64	116.83 MB	jdk-7u40-linux-x64.rpm
Linux x64	131.63 MB	jdk-7u40-linux-x64.tar.gz
Mac OS X x64	183.35 MB	jdk-7u40-macosx-x64.dmg
Solaris x86 (SVR4 package)	139.84 MB	jdk-7u40-solaris-i586.tar.Z
Solaris x86	95.29 MB	jdk-7u40-solaris-i586.tar.gz
Solaris x64 (SVR4 package)	24.43 MB	jdk-7u40-solaris-x64.tar.Z
Solaris x64	16.17 MB	jdk-7u40-solaris-x64.tar.gz
Solaris SPARC (SVR4 package)	139.06 MB	jdk-7u40-solaris-sparc.tar.Z
Solaris SPARC	98.07 MB	jdk-7u40-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	23.74 MB	jdk-7u40-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	18.18 MB	jdk-7u40-solaris-sparcv9.tar.gz
Windows x86	123.46 MB	jdk-7u40-windows-i586.exe
Windows x64	125.25 MB	jdk-7u40-windows-x64.exe

Java SE Development Kit 7u40 Demos and Samples Downloads

Java SE Development Kit 7u40 Demos and Samples Downloads are released under the Oracle BSD License

Product / File Description	File Size	Download
Linux ARM v7 Hard Float ABI	9.89 MB	jdk-7u40-linux-arm-vfp-hflt-demos.tar.gz
Linux ARM v6/v7 Soft Float ABI	9.73 MB	jdk-7u40-linux-arm-vfp-sflt-demos.tar.gz
Linux x86	16.6 MB	jdk-7u40-linux-i586-demos.rpm
Linux x86	16.56 MB	jdk-7u40-linux-i586-demos.tar.gz

Java Magazine

Java.net

Developer Training

Tutorials

Java.com

Java magazine NEW! Get it now for FREE! Subscribe Today

Secure Search

M McAfee

Tutorial: Configuring and Using Apache Tomcat 7 with Eclipse - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Tutorial: Configuring and Using Apache ... +

www.coreservlets.com/Apache-Tomcat-Tutorial/tomcat-7-with-eclipse.html

Secure Search

CUSTOMIZED J2EE TRAINING
\${coreservlets.com}

Training Tutorials Books Consulting & Outsourcing Programming Resources Jobs home contact sitemap search

FREE TUTORIALS

- APACHE TOMCAT 6
- APACHE TOMCAT 7
- JAVASERVER FACES (JSF)
 - JSF 2
 - PRIMEFACES
 - JSF 1
- JAVA PROGRAMMING
 - GENERAL JAVA PROGRAMMING
 - JAVA 8 LAMBDA & STREAMS
- ANDROID PROGRAMMING
- HADOOP
- AJAX & GWT
- JAVASCRIPT & AJAX BASICS
 - PROTOTYPE
 - SCRIPTACULOUS
 - JQUERY
 - DOJO
 - GWT
- HTML 5
- INTERMEDIATE SERVLETS & JSP
- ADVANCED SERVLETS & JSP
 - SPRING
 - HIBERNATE & JPA
 - JAKARTA STRUTS
 - EJB3
 - WEB SERVICES WITH AXIS2
 - SCWCD
 - TUTORIALS IN CHINESE
 - TUTORIALS IN JAPANESE

TRAINING COURSES

- ABOUT THE INSTRUCTOR
- COURSE REVIEWS
- JSF 2.0 & PRIMEFACES
- AJAX (w/ JQUERY, HTML5, Etc.)
- JAVA 6 & 7 PROGRAMMING
- ANDROID PROGRAMMING
- GWT

TUTORIAL: Installing Tomcat 7 and Using it with Eclipse

This tutorial covers Tomcat 7, which supports the servlet 3.0 and JSP 2.2 specs. This means that you can also run [servlet/JSP](#) or [JSF](#) apps that support the latest versions. I recommend Tomcat 7 over Tomcat 6 for all apps, since Tomcat 7 also supports the older servlet 2.5 and JSP 2.1 specs. But, if you really need to use Tomcat 6, please see the [tutorial on Eclipse with Tomcat 6](#). It takes only a short time to download Eclipse and learn the bare bones basics of using it to build Web apps and deploy them to Tomcat, and all the information you need to do this is described in this tutorial. This time will be very quickly recouped by the savings in development, debugging, and deployment times. To get started with Web apps in Eclipse, you only need to know a very small number of Eclipse features. You can gradually learn the advanced Eclipse capabilities at your leisure. Also note that if you print this page, the entire contents (including the expanded contents of all the accordion tabs below) will be printed.

If you find these free tutorials helpful, we would appreciate it if you would [link to us](#). Send corrections or feedback on any tutorial to hall@coreservlets.com.

Quick Start

Install Java

Unzip Tomcat

Unzip [tomcat-7.0.34-preconfigured.zip](#) into the location of your choice. I use the top level of the C drive, resulting in `C:\apache-tomcat-7.0.34`. This preconfigured version of Tomcat has the following settings already in place.

- **The port is changed from 8080 to 80.** This lets you enter URLs of the form `http://localhost/...` instead of `http://localhost:8080/...`.
 - When you download Tomcat from the Apache site, the port is 8080 in case you already have another server running on port 80.
- **Directory listings are turned on.** If you type a URL ending in / and there is no welcome file, Tomcat shows a directory listing.
 - Directory listings were on by default in previous Tomcat versions, but are off in the current version. They are convenient during development so you can just click on files, but most developers disable them for deployed applications.
- **Tomcat monitors struts-config.xml and faces-config.xml.** Whenever either of these files changes, Tomcat reloads the Web application. This saves you from restarting the server when you change these files.
 - If you do not use Struts or JSF, this change will not be beneficial to you. But it does not hurt either way.

Alternatively, you can download any Tomcat 7 version from the [Tomcat Web site](#) and then copy `context.xml`, `server.xml`, and `web.xml` into `install_dir/conf`. These files are annotated with comments on what modifications were done to change the port to 80 and to enable directory listings and automatic server restarts.

ANNOUNCEMENTS

- Too few developers for onsite courses? Try our public courses in MD, co-sponsored by Johns Hopkins "Engineering for Professionals":
- [JSF2: Modern Web Apps in Java \(w/ PrimeFaces Intro\)](#)
August 12-16 2013
- [Hadoop: Big-Data Apps in the Cloud](#)
August 26-30 2013
- [PrimeFaces: Rich GUIs for JSF2](#)
August 20-22 2013
- [Java Programming: A Crash Course](#)
September 9-13 2013
- [Java 8: Lambdas & Streams](#)
September 16 2013
- [Spring: Simplifying Java Applications](#)
October 8-11 2013
- [RESTful & SOAP Web Services in Java](#)
October 29-31 2013
- [Java Persistence with Hibernate & JPA](#)

Secure Search

McAfee

Apache Tomcat - Apache Tomcat 7 Downloads - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Apache Tomcat - Apache Tomcat 7 Dow... +

tomcat.apache.org/download-70.cgi

Secure Search

Search the Site... Search



Apache Tomcat



Tomcat 7 Downloads

Welcome to the Tomcat 7.x download page. This page provides download links for obtaining the latest version of Tomcat 7.0.x, as well as links to the archives of older releases.

Quick Navigation

[KEYS](#) | [7.0.42](#) | [Browse](#) | [Archives](#)

Release Integrity

You **must** verify the integrity of the downloaded files. We provide OpenPGP signatures for every release file. This signature should be matched against the [KEYS](#) file which contains the OpenPGP keys of Tomcat's Release Managers. We also provide an MD5 checksum for every release file. After you download the file, you should calculate a checksum for your download, and make sure it is the same as ours.

Mirrors

You are currently using <http://www.dsgrnwld.com/am/>. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are *backup* mirrors (at the end of the mirrors list) that should be available.

Other mirrors: [Change](#)

7.0.42

Please see the [README](#) file for packaging information. It explains what every distribution contains.

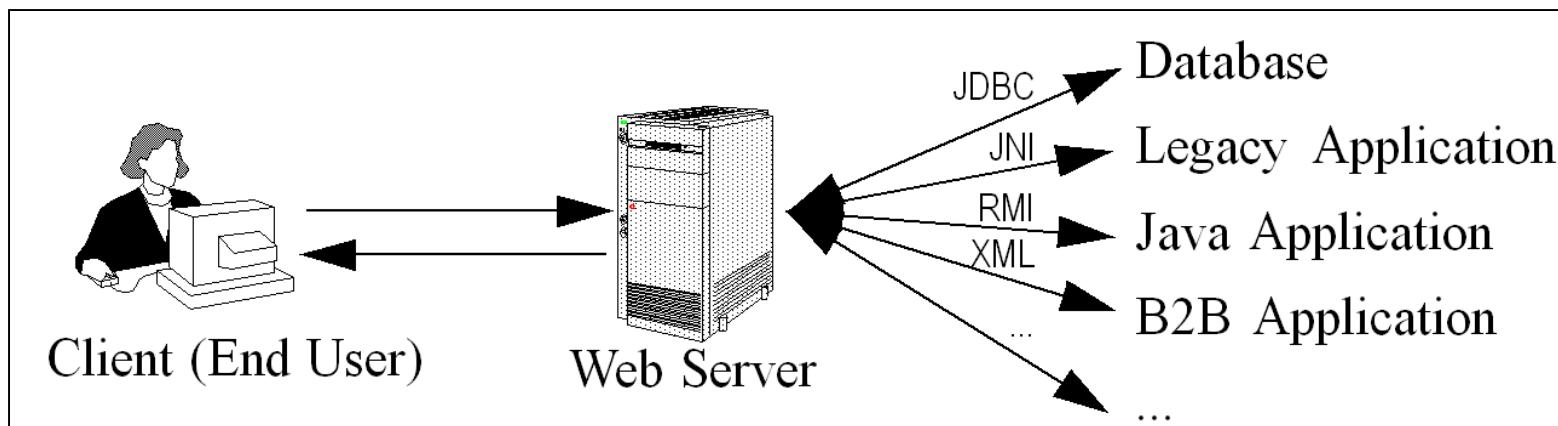
Binary Distributions

Secure Search



What is an Enterprise Application?

- Software applications that integrate and support all aspects of operations of an enterprise/ business.



Challenges for the Enterprise Application

- Challenges:
 - Heterogeneous data sources
 - Diverse user base
 - Multiple platforms/ products/ vendors/ technologies
 - High level of integration and consistency

Overview of J2EE

- “Enterprise” APIs were packaged together as Java 2, Enterprise Edition
- The standard JDK (javac, java, etc.) was called Java 2, Standard Edition
- J2EE includes the entire set of APIs, but may be used in smaller pieces.

J2EE overview (cont.)

- **Java Servlets**

- a simple, consistent mechanism for extending the functionality of a web server.

- **JavaServer Pages**

- a simplified, fast way to create dynamic web content.
 - server and platform independent.

- **Enterprise JavaBeans Architecture**

- create, deploy and manage cross- platform, component-based enterprise applications

J2EE overview (cont.)

- **JDBC**
 - a uniform interface to a wide range of relational databases.
- **JavaMail**
 - a platform independent and protocol independent framework to build Java- based mail and messaging applications.
- **Java Naming and Directory Interface (JNDI)**
 - allows unified access to multiple naming and directory services across the enterprise.

J2EE Overview (cont.)

- **Java Message Service (JMS)**
 - a standard Java API for reliable enterprise messaging services. (point-to-point and publish/subscribe)
- **Java Transaction API (JTA)**
 - a high-level transaction management specification intended for resource managers and transactional applications in distributed transaction systems.
- **Java Transaction Service (JTS)**
 - provides open, standard access to transaction resources.

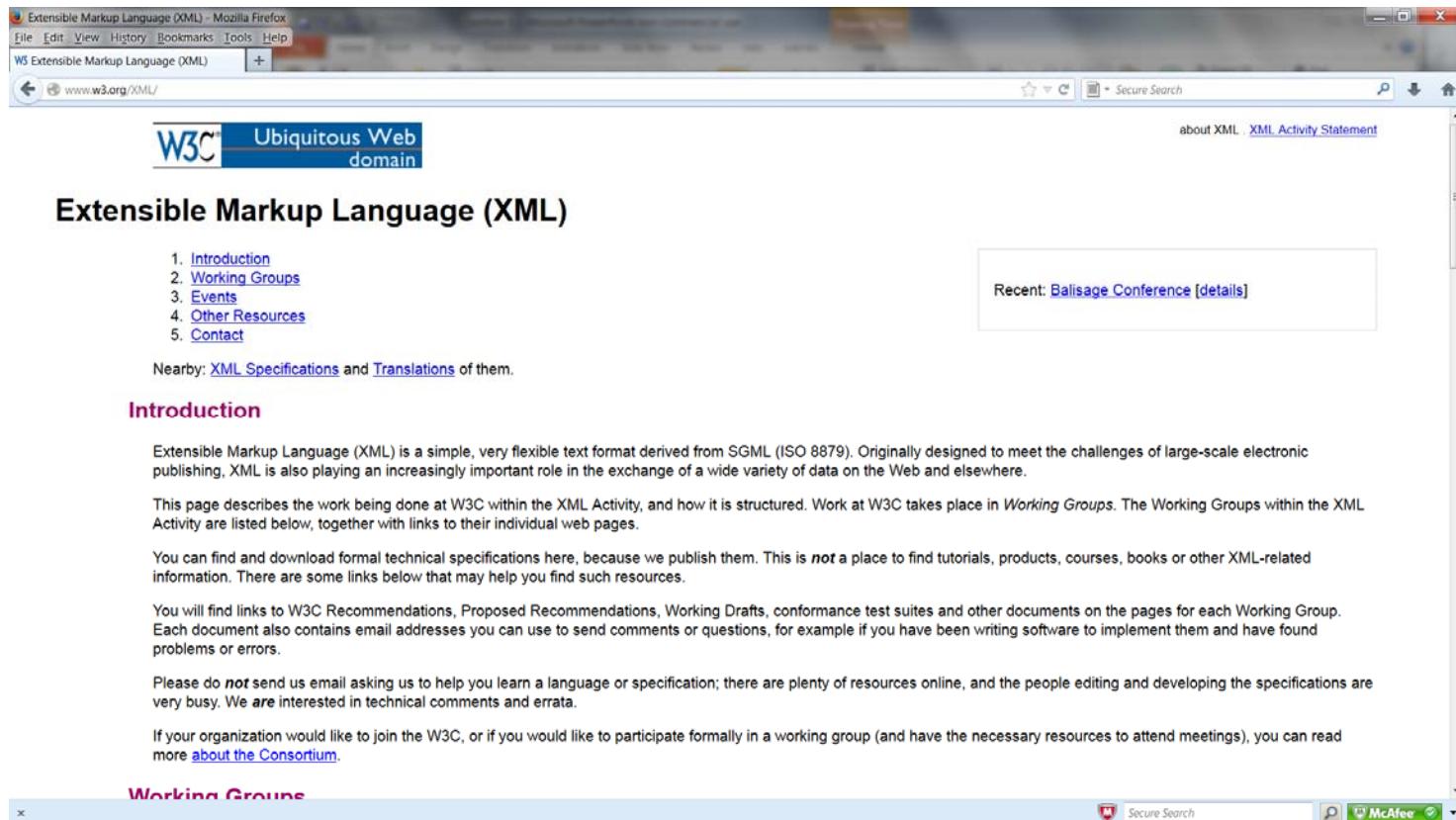
J2EE Overview (cont.)

- **Java Interface Definition Language (IDL)**
 - provides interoperability with CORBA, the industry standard for heterogeneous computing.
- **RMI-IIOP**
 - an implementation of the Java RMI API over the OMG's industry-standard Internet Inter-Orb Protocol (IIOP).
- **J2EE Connector**
 - a standard architecture for connecting the J2EE platform to heterogeneous Enterprise Information Systems.

Java and XML

- Java and XML are integrated
 - Java API for XML Messaging (JAXM)
 - Java API for XML Parsing (JAXP)
 - Java API for XML Data Binding (JAXB)
 - JavaTM API for XML-based RPC (JAX-RPC)
- Java and XML
 - <http://docs.oracle.com/javase/tutorial/jaxp/>
- XML page:
 - <http://www.w3.org/XML/>

Java and XML

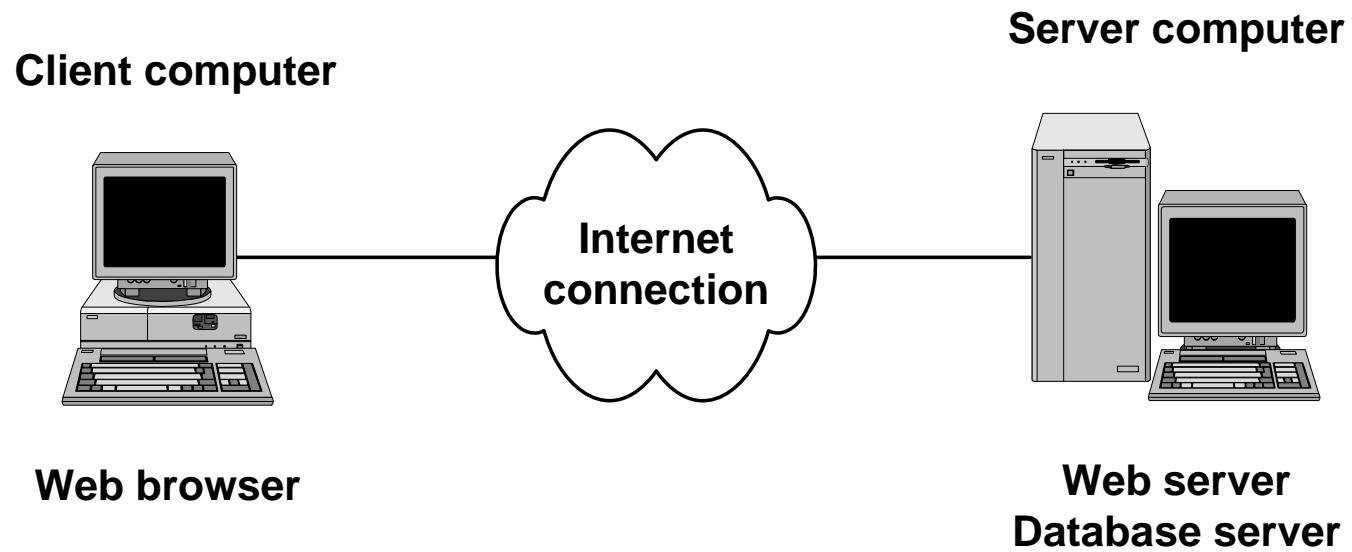


The screenshot shows a Mozilla Firefox browser window with the title bar "Extensible Markup Language (XML) - Mozilla Firefox". The address bar displays "www.w3.org/XML/". The main content area shows the W3C XML homepage. The header features the W3C logo and the text "Ubiquitous Web domain". Below the header, the page title is "Extensible Markup Language (XML)". A sidebar on the left lists navigation links: 1. Introduction, 2. Working Groups, 3. Events, 4. Other Resources, and 5. Contact. A note below the sidebar says "Nearby: [XML Specifications](#) and [Translations](#) of them.". The main content area contains several paragraphs of text about XML, its history, and its role in publishing. It also mentions the XML Activity, Working Groups, and various XML-related documents and resources. At the bottom of the page, there is a link to "about the Consortium". The browser interface includes standard toolbar icons and a status bar at the bottom.

Java and XML

The screenshot shows a Mozilla Firefox browser window displaying the "Trail: Java API for XML Processing (JAXP) (The Java™ Tutorials)" page. The URL in the address bar is docs.oracle.com/javase/tutorial/jaxp/. The page content includes the Java Tutorials logo, navigation links like "Home Page", and a red header "Trail: Java API for XML Processing (JAXP)". Below the header, it says "The Java API for XML Processing (JAXP) trail provides an introduction to Java API for XML Processing (JAXP) 1.4 technology, via examples of JAXP applications." A section titled "Before You Read This Tutorial" lists prerequisites: "The Java programming language and its development environment.", "The eXtensible Mark-up Language (XML)", "The Document Object Model (DOM), as defined by the World Wide Web Consortium (W3C) DOM Working Group.", and "Simple API for XML (SAX), as developed cooperatively by the members of the XML-DEV mailing list.". The page also lists several related topics with icons: "Introduction to JAXP", "Simple API for XML", "Document Object Model", "Extensible Stylesheet Language Transformations", "Streaming API for XML", and "JAXP 1.5 and New Properties". At the bottom, there are links for "« Previous • TOC • Next »".

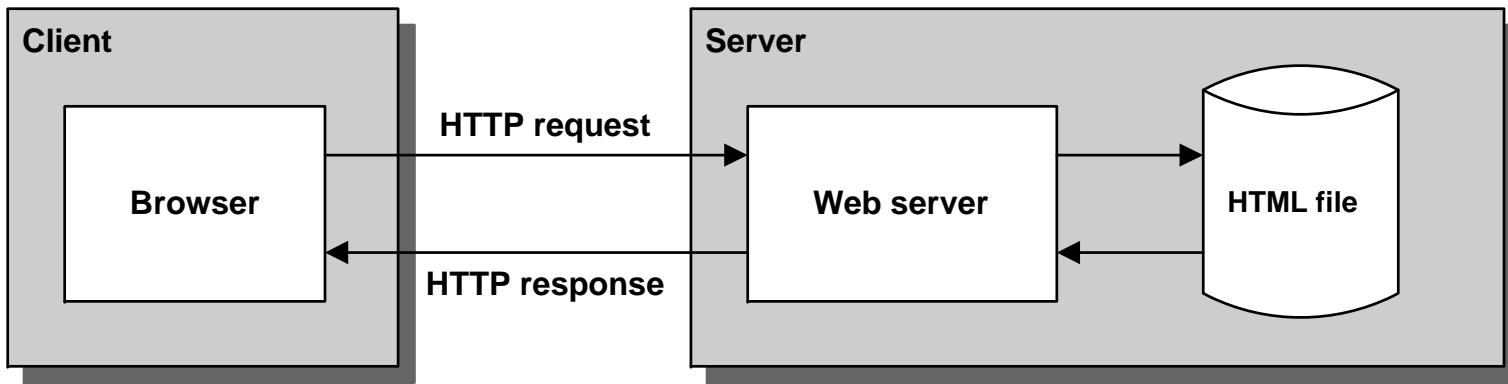
Components of a web application



The components of a web application

- Web applications are a type of *client/server application*.
- In a client/server application, a user at a *client* computer accesses an application at a *server* computer.
- For a web application, the client and server computers are connected via the Internet or an intranet.
- In a web application, the *web browser* provides the user interface for the application.
- A web application runs on the server computer under the control of *web server* software.
- For most web applications, the server computer also runs a *database management system (DBMS)*.

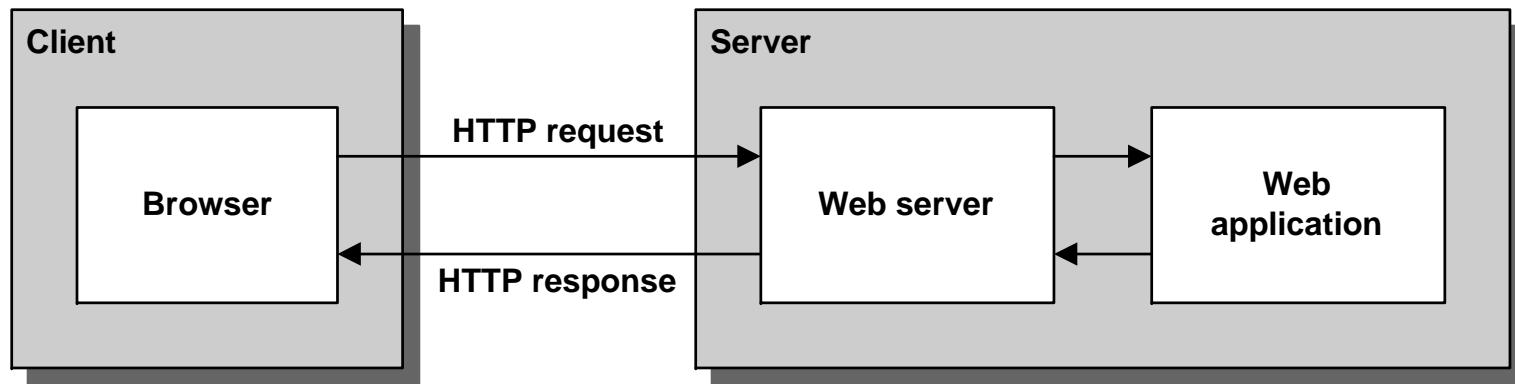
How a web server processes static web pages



How static web pages work

- *Hypertext Markup Language*, or *HTML*, is the language that the web browser converts into the web pages of a web application.
- A *static web page* is an HTML document that's stored in a file and does not change in response to user input.
- *Hypertext Transfer Protocol*, or *HTTP*, is the protocol that web browsers and web servers use to communicate.
- A web browser requests a page from a web server by sending the server a message known as an *HTTP request*. For a static web page, the HTTP request includes the name of the HTML file that's requested.
- A web server replies to an HTTP request by sending a message known as an *HTTP response* back to the browser. For a static web page, the HTTP response includes the HTML document.

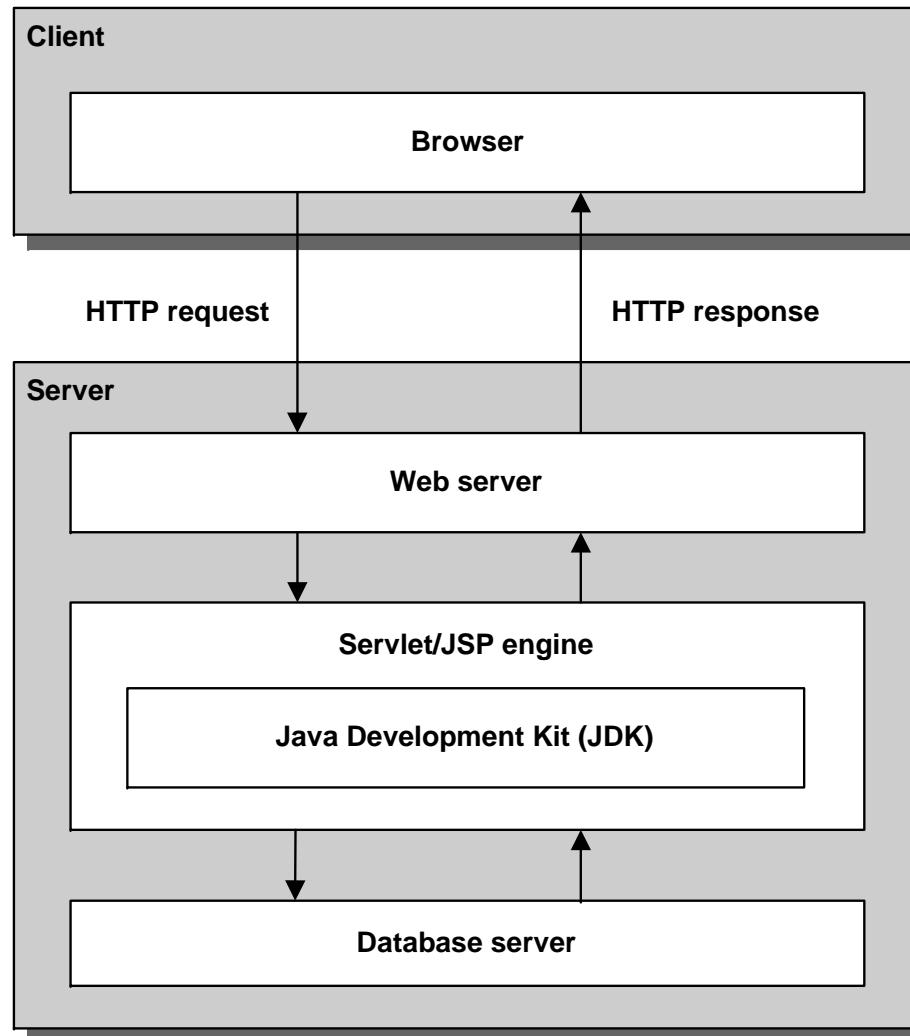
How a web server processes dynamic web pages



How dynamic web pages work

- A *dynamic web page* is an HTML document that's generated by a web application. Often, the web page changes according to parameters that are sent to the web application by the web browser.
- When a web server receives a request for a dynamic web page, the server passes the request to the web application. Then, the application generates a response, which is usually an HTML document, and returns it to the web server.
- The web server, in turn, wraps the generated HTML document in an HTTP response and sends it back to the browser.
- The browser doesn't know or care whether the HTML was retrieved from a static HTML file or was dynamically generated by the web application. Either way, the browser displays the HTML document that is returned.

The components of a Java web application



Components needed for Java web application

- Java web applications consist of JavaServer Pages and servlets.
- A *servlet/JSP engine*, or *servlet/JSP container*, is the software that allows the web server to work with servlets and JSPs.
- The *Java Enterprise Edition (Java EE)*, specification describes how web servers can interact with servlet/JSP engines.
- For a servlet/JSP engine to work, it must have access to Java's *Java Development Kit (JDK)*, which comes as part of the *Java Standard Edition (Java SE)*.
- Among other things, the JDK contains the core Java class libraries, the Java compiler, and the *Java Runtime Environment (JRE)*.
- Java web applications that use *Enterprise JavaBeans (EJBs)* require an additional server component known as an *EJB server*, or *EJB container*.

■ Who use the Java-based technologies for their websites?

The screenshot shows a Mozilla Firefox browser window displaying the Home Depot homepage. The address bar shows the URL: www.homedepot.com/webapp/wcs/stores/servlet/HomePageView?langId=-1&storeId=10051&catalogId=10053. The page features a sidebar with a 'SHOP ALL DEPARTMENTS' menu containing links like 'Everything for Winter', 'Appliances', 'Bath', etc. A search bar at the top right includes a 'GO' button and a 'CART' button. A large promotional banner in the center offers '10% OFF ALL APPLIANCES \$297 OR MORE' with a 'Shop Now >' link. Below the banner are images of various appliances and home improvement products. At the bottom, there are four promotional boxes: 'DREAM & REDEEM', 'AVOID THE CROWDS BUY ONLINE, PICK UP IN STORE', 'SPECIAL BUY OF THE DAY', and 'TODAY ONLY SAVINGS! Save up to 50%'. The background of the page has a wood-grain texture.

■ Who use the Java-based technologies for their websites?

The screenshot shows a Mozilla Firefox browser window displaying the Sony VAIO Laptops & Desktops page. The address bar shows the URL: store.sony.com/webapp/wcs/stores/servlet/CategoryDisplay?catalogId=10551&storeId=10151&langId=-1&identifier=S_Notebooks&SR=nav:electronic. The page features the Sony logo and navigation links for Products, Savings, Media, and Support. A search bar is at the top right. The main content area is titled "VAIO® Computers" and includes a welcome message, a "Save up to \$400" offer for laptops, a "Free sheet battery" offer for S Series notebooks, and a "VAIO Clearance Event". The right sidebar has a "FEEDBACK" section with social media icons for Facebook, Twitter, and YouTube. The bottom of the page includes a "Secure Search" bar and a McAfee security badge.

Sony VAIO Laptops Desktops Computers PCs | Sony Store USA - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Sony VAIO Laptops Desktops Computers... +

store.sony.com/webapp/wcs/stores/servlet/CategoryDisplay?catalogId=10551&storeId=10151&langId=-1&identifier=S_Notebooks&SR=nav:electronic

Search Results

Most Visited Getting Started Latest Headlines

SONY make.believe

Login My Account My Cart Find a store Weekly Deals

Free ground shipping on all orders. USA

PRODUCTS SAVINGS MEDIA SUPPORT Search Products

COMPUTERS AND TABLETS TV AND HOME ENTERTAINMENT CAMERAS AND CAMCORDERS PERSONAL AUDIO AND PHONES NEW FROM SONY

VAIO® Computers

Welcome to the official home of custom VAIO computers.

Sony recommends Windows® 7.

Intel® CORE™ i7 Visibly Smart Learn more

Computers & Tablets

Laptops and Desktops
Tablets
E-Readers

Laptops & Desktops

Compare all PCs
Laptop buying guide

Screen Size

11" - 13"

Save up to \$400.

Build select S and F Series laptops and save on processor, solid state drive and battery upgrades.

SHOP NOW

Free sheet battery.

Buy a 15.5" S Series, double your battery life for free.

Shop now

VAIO Clearance Event

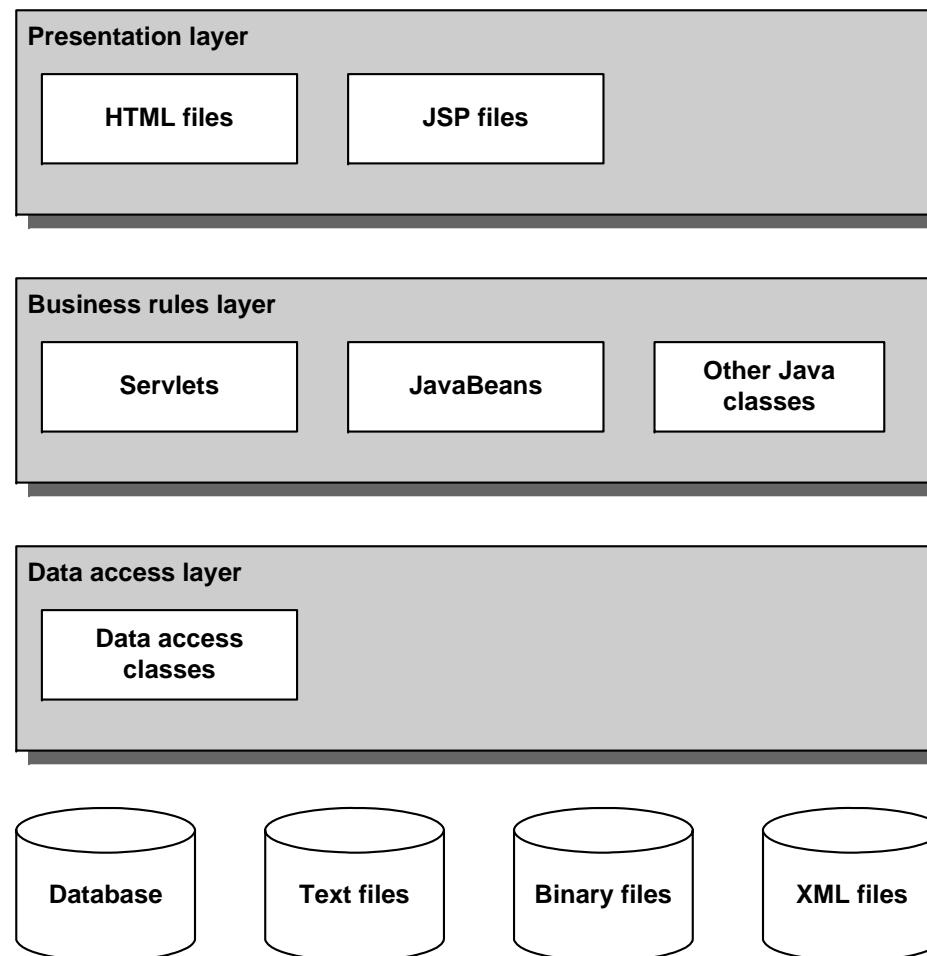
Up to \$350 instant savings, while supplies last.

Shop now

Secure Search

McAfee

The architecture for a typical Java web application



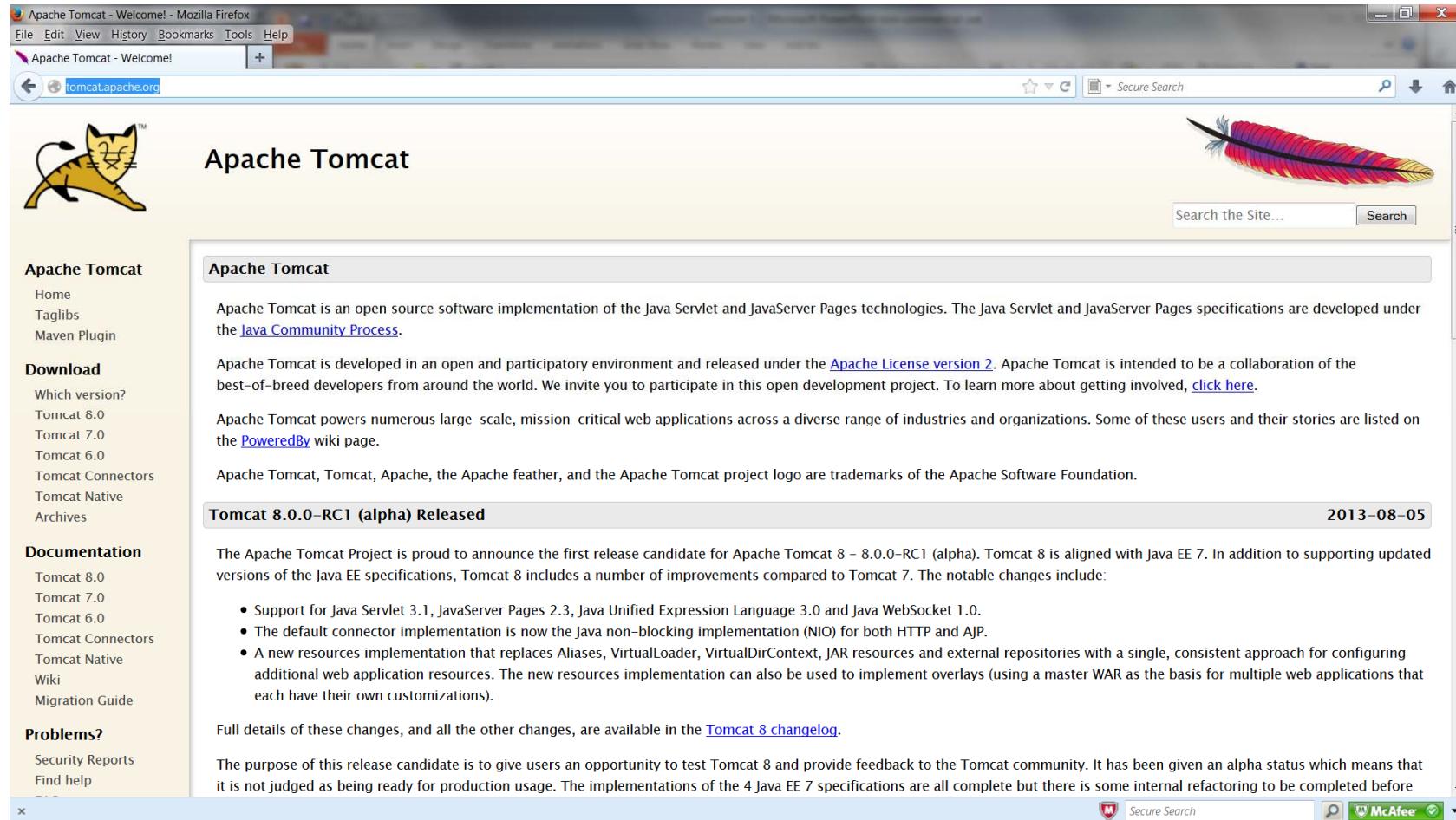
The architecture for a typical Java web application (continued)

- The *presentation layer* for a typical Java web application consists of HTML pages and JSPs.
- The *business rules layer* for a typical Java web application consists of servlets. These servlets may call other Java classes including a special type of Java class known as a *JavaBean*. In chapters 9 and 10, you'll learn how to use several special types of tags within a JSP to work with JavaBeans.
- The *data access layer* for a typical Java web application consists of classes that read and write data that's stored on the server's disk drive.
- For a serious web application, the data is usually stored in a relational database. However, it may also be stored in binary files, in text files, or in *Extensible Markup Language* (or *XML*) files.

Free Servlet and JSP Engines

- Apache Tomcat
 - <http://tomcat.apache.org/>

Free Servlet and JSP Engines



The screenshot shows the Apache Tomcat homepage as it would appear in Mozilla Firefox. The title bar reads "Apache Tomcat - Welcome! - Mozilla Firefox". The address bar shows "tomcat.apache.org". The main content area features the Apache feather logo and a search bar. On the left, there's a sidebar with links for "Apache Tomcat", "Download", "Documentation", and "Problems?". A central section discusses the Apache Tomcat project, its history, and its impact on web development. A specific release candidate for Tomcat 8 is highlighted, along with a list of notable improvements. The bottom of the page includes a footer with security information and a McAfee logo.

Apache Tomcat

Apache Tomcat

Apache Tomcat is an open source software implementation of the Java Servlet and JavaServer Pages technologies. The Java Servlet and JavaServer Pages specifications are developed under the [Java Community Process](#).

Apache Tomcat is developed in an open and participatory environment and released under the [Apache License version 2](#). Apache Tomcat is intended to be a collaboration of the best-of-breed developers from around the world. We invite you to participate in this open development project. To learn more about getting involved, [click here](#).

Apache Tomcat powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations. Some of these users and their stories are listed on the [PoweredBy](#) wiki page.

Apache Tomcat, Tomcat, Apache, the Apache feather, and the Apache Tomcat project logo are trademarks of the Apache Software Foundation.

Tomcat 8.0.0-RC1 (alpha) Released 2013-08-05

The Apache Tomcat Project is proud to announce the first release candidate for Apache Tomcat 8 – 8.0.0-RC1 (alpha). Tomcat 8 is aligned with Java EE 7. In addition to supporting updated versions of the Java EE specifications, Tomcat 8 includes a number of improvements compared to Tomcat 7. The notable changes include:

- Support for Java Servlet 3.1, JavaServer Pages 2.3, Java Unified Expression Language 3.0 and Java WebSocket 1.0.
- The default connector implementation is now the Java non-blocking implementation (NIO) for both HTTP and AJP.
- A new resources implementation that replaces Aliases, VirtualLoader, VirtualDirContext, JAR resources and external repositories with a single, consistent approach for configuring additional web application resources. The new resources implementation can also be used to implement overlays (using a master WAR as the basis for multiple web applications that each have their own customizations).

Full details of these changes, and all the other changes, are available in the [Tomcat 8 changelog](#).

The purpose of this release candidate is to give users an opportunity to test Tomcat 8 and provide feedback to the Tomcat community. It has been given an alpha status which means that it is not judged as being ready for production usage. The implementations of the 4 Java EE 7 specifications are all complete but there is some internal refactoring to be completed before

Apache Tomcat 7.0.34

Tutorial: Configuring and Using Apache Tomcat 7 with Eclipse - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Tutorial: Configuring and Using Apache ... +
www.coreservlets.com/Apache-Tomcat-Tutorial/tomcat-7-with-eclipse.html
Secure Search

CUSTOMIZED J2EE TRAINING
\${coreservlets.com}
Training Tutorials Books Consulting & Outsourcing Programming Resources Jobs home contact sitemap

Tutorial: Installing Tomcat 7 and Using it with Eclipse

This tutorial covers Tomcat 7, which supports the servlet 3.0 and JSP 2.2 specs. This means that you can also run [servlet/JSP](#) or [JSF](#) apps that support the latest versions. I recommend Tomcat 7 over Tomcat 6 for all apps, since Tomcat 7 also supports the older servlet 2.5 and JSP 2.1 specs. But, if you really need to use Tomcat 6, please see the [tutorial on Eclipse with Tomcat 6](#). It takes only a short time to download Eclipse and learn the bare bones basics of using it to build Web apps and deploy them to Tomcat, and all the information you need to do this is described in this tutorial. This time will be very quickly recouped by the savings in development, debugging, and deployment times. To get started with Web apps in Eclipse, you only need to know a very small number of Eclipse features. You can gradually learn the advanced Eclipse capabilities at your leisure. Also note that if you print this page, the entire contents (including the expanded contents of all the accordion tabs below) will be printed.

If you find these free tutorials helpful, we would appreciate it if you would [link to us](#). Send corrections or feedback on any tutorial to hall@coreservlets.com.

Quick Start

Install Java

Unzip Tomcat

Unzip [tomcat-7.0.34-preconfigured.zip](#) into the location of your choice. I use the top level of the C drive, resulting in `C:\apache-tomcat-7.0.34`. This preconfigured version of Tomcat has the following settings already in place.

- **The port is changed from 8080 to 80.** This lets you enter URLs of the form `http://localhost/...` instead of `http://localhost:8080/...`.
 - When you download Tomcat from the Apache site, the port is 8080 in case you already have another server running on port 80.
- **Directory listings are turned on.** If you type a URL ending in / and there is no welcome file, Tomcat shows a directory listing.
 - Directory listings were on by default in previous Tomcat versions, but are off in the current version. They are convenient during development so you can just click on files, but most developers disable them for deployed applications.
- **Tomcat monitors struts-config.xml and faces-config.xml.** Whenever either of these files changes, Tomcat reloads the Web application. This saves you from restarting the server when you change these files.
 - If you do not use Struts or JSF, this change will not be beneficial to you. But it does not hurt either way.

Alternatively, you can download any Tomcat 7 version from the [Tomcat Web site](#) and then copy `context.xml`, `server.xml`, and `web.xml` into `install_dir/conf`. These files are annotated with comments on what modifications were done to change the port to 80 and to enable directory listings and automatic server restarts.

ANNOUNCEMENTS

Too few developers for onsite courses? Try our public courses in MD, co-sponsored by Johns Hopkins "Engineering for Professionals":
[JSF2: Modern Web Apps in Java \(w/ PrimeFaces Intro\)](#)
August 12-16 2013

[Hadoop: Big-Data Apps in the Cloud](#)
August 26-30 2013

[PrimeFaces: Rich GUIs for JSF2](#)
August 20-22 2013

[Java Programming: A Crash Course](#)
September 9-13 2013

[Java 8: Lambdas & Streams](#)
September 16 2013

[Spring: Simplifying Java Applications](#)
October 8-11 2013

[RESTful & SOAP Web Services in Java](#)
October 29-31 2013

[Java Persistence with Hibernate & JPA](#)

Secure Search McAfee

Apache Tomcat 7.0.34

Tutorial: Configuring and Using Apache Tomcat 7 with Eclipse - Mozilla Firefox

Tutorial: Configuring and Using Apache ... www.coreservlets.com/Apache-Tomcat-Tutorial/tomcat-7-with-eclipse.html

CUSTOMIZED J2EE TRAINING \${coreservlets.com}

Training Tutorials Books Consulting & Outsourcing Programming Resources Jobs home contact sitemap search

FREE TUTORIALS

- APACHE TOMCAT 6
- APACHE TOMCAT 7
- JAVASERVER FACES (JSF)
 - JSF 2
 - PRIMEFACES
 - JSF 1
- JAVA PROGRAMMING
 - GENERAL JAVA PROGRAMMING
 - JAVA 8 LAMBDA & STREAMS
 - ANDROID PROGRAMMING
- HADOOP
- AJAX & GWT
- JAVASCRIPT & AJAX BASICS
 - PROTOTYPE
 - SCRIPTACULOUS
 - JQUERY
 - DOJO
 - GWT
- HTML 5
- INTERMEDIATE SERVLETS & JSP
- ADVANCED SERVLETS & JSP
- SPRING
- HIBERNATE & JPA
- JAKARTA STRUTS
- EJB3
- WEB SERVICES WITH AXIS2
- SCWCD
- TUTORIALS IN CHINESE
- TUTORIALS IN JAPANESE

TRAINING COURSES

- ABOUT THE INSTRUCTOR
- COURSE REVIEWS
- JSF 2.0 & PRIMEFACES
- AJAX (w/ JQUERY, HTML5, Etc.)
- JAVA 6 & 7 PROGRAMMING
- ANDROID PROGRAMMING
- GWT

Tutorial: Installing Tomcat 7 and Using it with Eclipse

The Tomcat 7.0.34-preconfigured.zip file has been successfully downloaded. This means that you can also run [servlet/JSP](#) or [JSF](#) apps that support the latest versions. I recommend using Java 7, Servlet 2.5 and JSP 2.1 specs. But, if you really need to use Tomcat 6, please see the [tutorial on Eclipse with Tomcat 6](#). This tutorial covers the basics of using it to build Web apps and deploy them to Tomcat, and all the information you need to do so. You will learn how to use Eclipse's advanced features for saving time in development, debugging, and deployment times. To get started with Web apps in Eclipse, you only need to download the advanced Eclipse capabilities at your leisure. Also note that if you print this page, the entire contents of the page will be included.

If you have any questions or comments, please feel free to contact us. Send corrections or feedback on any tutorial to hall@coreservlets.com.

What should Firefox do with this file?

Open with Windows Explorer (default)

Save File

Do this automatically for files like this from now on.

OK Cancel

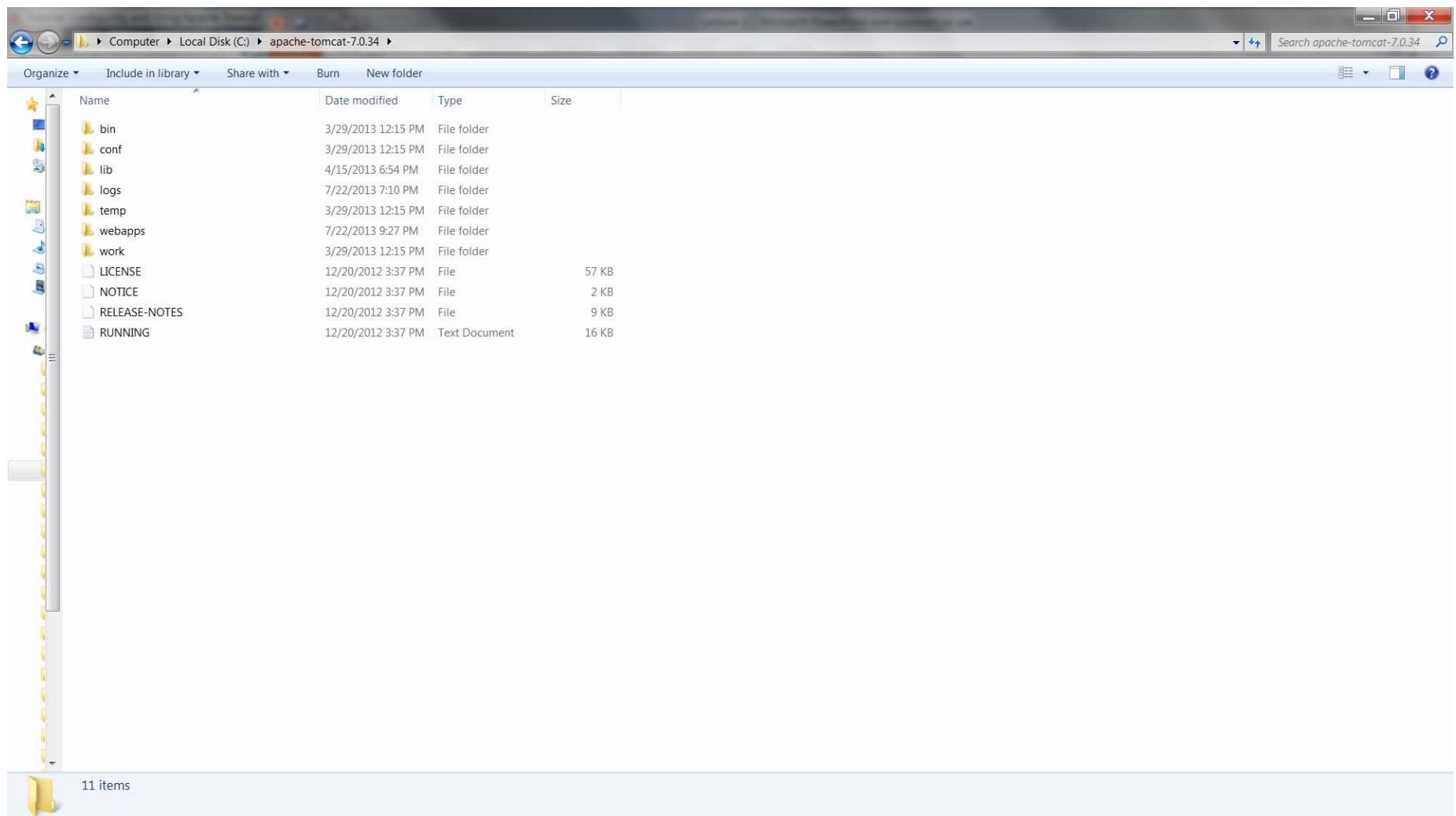
ANNOUNCEMENTS

Too few developers for onsite courses? Try our public courses in MD, co-sponsored by Johns Hopkins "Engineering for Professionals":

- JSF2: Modern Web Apps in Java (w/ PrimeFaces Intro)** August 12-16 2013
- Hadoop: Big-Data Apps in the Cloud** August 26-30 2013
- PrimeFaces: Rich GUIs for JSF2** August 20-22 2013
- Java Programming: A Crash Course** September 9-13 2013
- Java 8: Lambdas & Streams** September 16 2013
- Spring: Simplifying Java Applications** October 8-11 2013
- RESTful & SOAP Web Services in Java** October 29-31 2013
- Java Persistence with Hibernate & JPA**

Secure Search McAfee

Apache Tomcat 7.0.34



The subdirectories

Directory	Description
bin	Files for working with Tomcat such as the startup and shutdown batch files.
conf	Files for configuring Tomcat such as server.xml, context.xml, and web.xml.
lib	JAR files that contain classes that are available to all web applications. As a result, you can put any JAR files you want to make available to all web applications in this directory.
logs	Log files.
temp	Temporary files used by the JVM.
webapps	The directories and files for the web applications.
work	The source code and class files for the servlets that Tomcat generates for the JSPs.

The files

File	Description
RELEASE-NOTES	General information about the current release of Tomcat.
running.txt	Instructions for installing, starting, and stopping Tomcat.

The JAR files that need to be available to the JRE

`servlet-api.jar`
`jsp-api.jar`
`el-api.jar`
`tomcat-dbcp.jar`

Description

- The directory that holds the files for Tomcat is known as the *Tomcat home directory*. By default, this directory is named apache-tomcat-7.0.X, but you can rename it to tomcat.
- The Java Archive (JAR) files shown above contain the Java classes that need to be available to the JDK and JRE when you develop servlets and JSPs. By copying these JAR files from Tomcat's lib subdirectory to the JDK's jre\lib\ext subdirectory, you make the classes available to the JDK and JRE.

Apache Tomcat 7.0.34

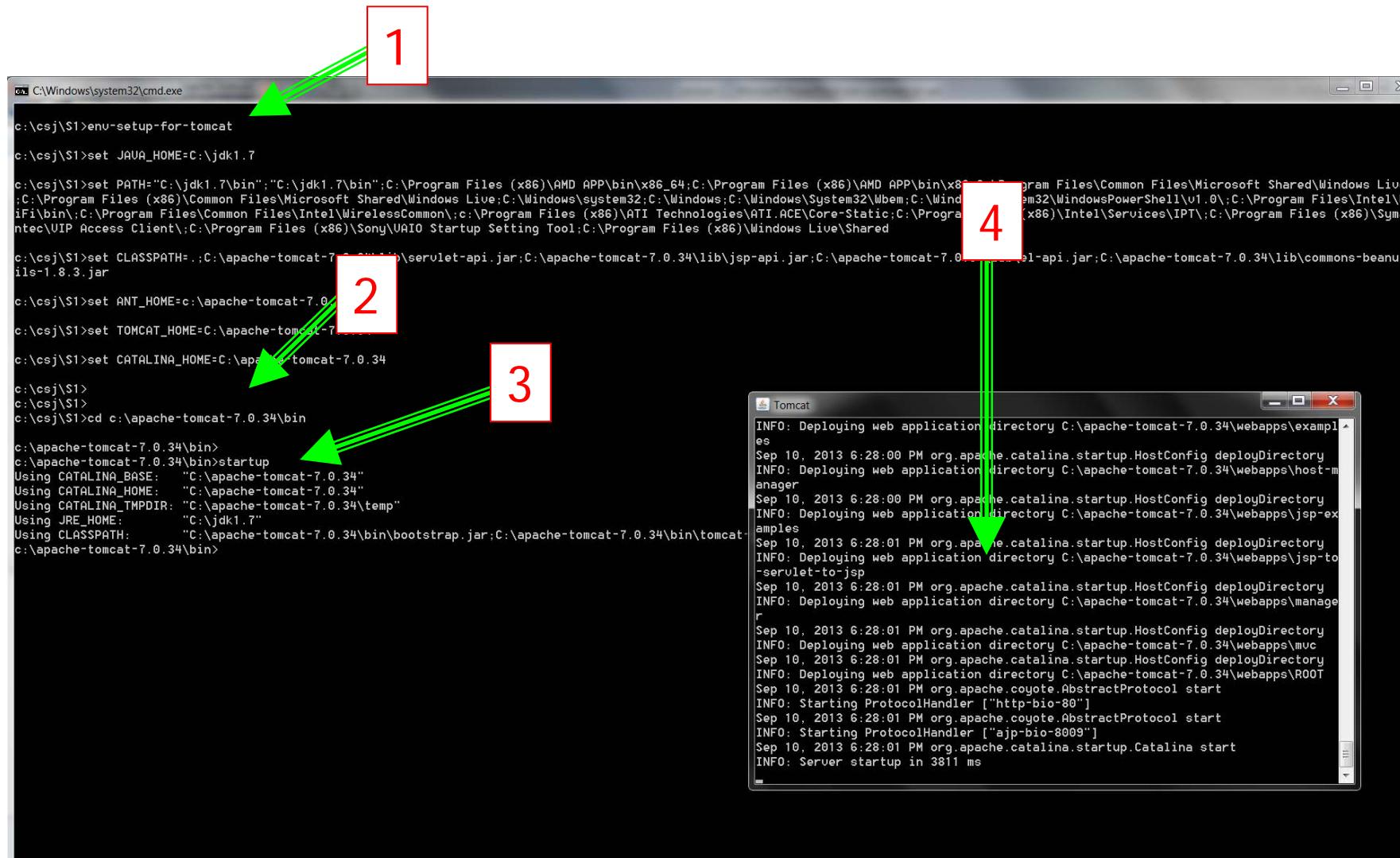


The screenshot shows a Windows Notepad window with the title "env-setup-for-tomcat - Notepad". The window contains the following text:

```
File Edit Format View Help
set JAVA_HOME=C:\jdk1.7
set PATH="C:\jdk1.7\bin";%PATH%
set CLASSPATH=.;C:\apache-tomcat-7.0.34\lib\servlet-api.jar;C:\apache-tomcat-7.0.34\lib\jsp-api.jar;C:\apache-tomcat-7.0.34\lib\el-api.jar;C:\apache-tomcat-7.0.34\lib\commons-beanutils-1.8.3.jar
set ANT_HOME=c:\apache-tomcat-7.0.34
set TOMCAT_HOME=C:\apache-tomcat-7.0.34
set CATALINA_HOME=C:\apache-tomcat-7.0.34
```

The Notepad window has a standard Windows interface with a menu bar, toolbar, and status bar at the bottom.

Apache Tomcat 7.0.34



Apache Tomcat 7.0.34

Apache Tomcat/7.0.34 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Apache Tomcat/7.0.34 +
localhost Secure Search Find Help

Home Documentation Configuration Examples Wiki Email Lists Java EE Tutorials

Apache Tomcat/7.0.34 The Apache Software Foundation http://www.apache.org/

If you're seeing this, you've successfully installed Tomcat 7. Congratulations!
Pre-configured version from Marty Hall and coreservlets.com. See <http://www.coreervlets.com/Apache-Tomcat-Tutorial/>.

 Recommended Reading:
[Integrating Tomcat 7 with Eclipse](#)
[Security Considerations HOW-TO](#)
[Manager Application HOW-TO](#)
[Clustering/Session Replication HOW-TO](#)

Server Status
Manager App
Host Manager

Developer Quick Start

Tomcat Setup	Realms & AAA	JSF 2 Tutorial	Servlet Specifications
First Web Application	JDBC DataSources	Servlet 3.0 & JSP 2.2 API	Tomcat Versions
Servlet & JSP Tutorial	Servlet & JSP Training	PrimeFaces Tutorial	JSF 2 & PrimeFaces Training

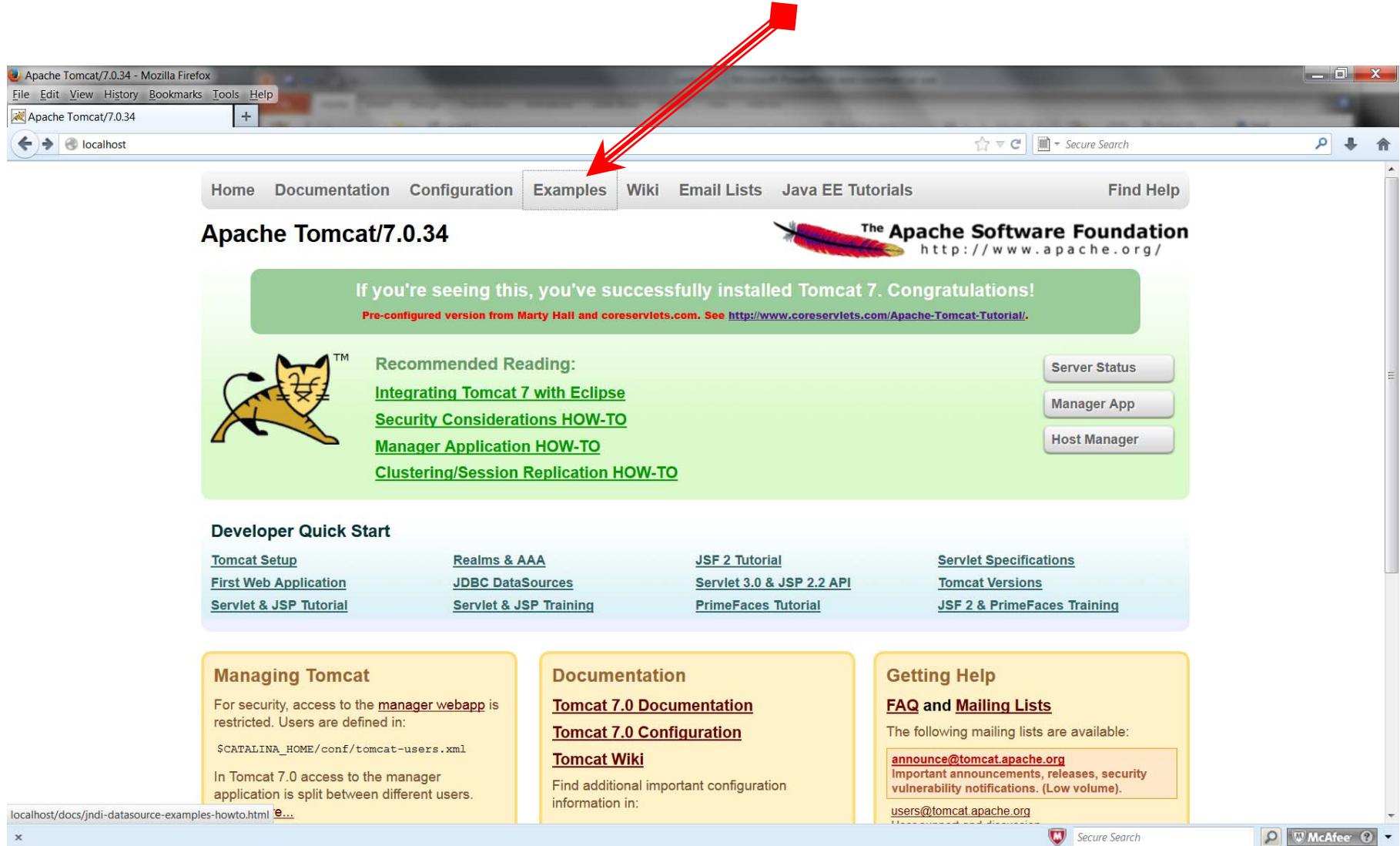
Managing Tomcat
For security, access to the [manager webapp](#) is restricted. Users are defined in:
SCATALINA_HOME/conf/tomcat-users.xml
In Tomcat 7.0 access to the manager application is split between different users.
[Read more...](#)

Documentation
[Tomcat 7.0 Documentation](#)
[Tomcat 7.0 Configuration](#)
[Tomcat Wiki](#)
Find additional important configuration information in:

Getting Help
[FAQ and Mailing Lists](#)
The following mailing lists are available:
announce@tomcat.apache.org
Important announcements, releases, security vulnerability notifications. (Low volume).
users@tomcat.apache.org

localhost/examples/ Secure Search McAfee

Apache Tomcat 7.0.34



The screenshot shows the Apache Tomcat 7.0.34 homepage in Mozilla Firefox. The title bar reads "Apache Tomcat/7.0.34 - Mozilla Firefox". The top navigation bar includes links for Home, Documentation, Configuration, Examples (which is highlighted with a red arrow), Wiki, Email Lists, Java EE Tutorials, and Find Help. Below the navigation bar, the page title is "Apache Tomcat/7.0.34". To the right is the "The Apache Software Foundation" logo and the URL "http://www.apache.org/". A green banner at the top says "If you're seeing this, you've successfully installed Tomcat 7. Congratulations!" and provides a link to a pre-configured version from Marty Hall and coreservlets.com. The main content area features a cartoon cat icon and a section titled "Recommended Reading" with links to "Integrating Tomcat 7 with Eclipse", "Security Considerations HOW-TO", "Manager Application HOW-TO", and "Clustering/Session Replication HOW-TO". To the right are buttons for "Server Status", "Manager App", and "Host Manager". Below this is a "Developer Quick Start" section with links to "Tomcat Setup", "First Web Application", "Servlet & JSP Tutorial", "Realms & AAA", "JDBC DataSources", "Servlet & JSP Training", "JSF 2 Tutorial", "Servlet 3.0 & JSP 2.2 API", "PrimeFaces Tutorial", "Servlet Specifications", "Tomcat Versions", and "JSF 2 & PrimeFaces Training". The bottom section contains three boxes: "Managing Tomcat", "Documentation", and "Getting Help". The "Managing Tomcat" box discusses security and user management. The "Documentation" box links to the Tomcat 7.0 Documentation, Configuration, and Wiki. The "Getting Help" box lists mailing lists like announce@tomcat.apache.org and users@tomcat.apache.org.

If you're seeing this, you've successfully installed Tomcat 7. Congratulations!

Pre-configured version from Marty Hall and coreservlets.com. See <http://wwwcoreservletscom/Apache-Tomcat-Tutorial>.

Recommended Reading:

- [Integrating Tomcat 7 with Eclipse](#)
- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Developer Quick Start

Tomcat Setup	Realms & AAA	JSF 2 Tutorial	Servlet Specifications
First Web Application	JDBC DataSources	Servlet 3.0 & JSP 2.2 API	Tomcat Versions
Servlet & JSP Tutorial	Servlet & JSP Training	PrimeFaces Tutorial	JSF 2 & PrimeFaces Training

Managing Tomcat

For security, access to the [manager webapp](#) is restricted. Users are defined in: `$CATALINA_HOME/conf/tomcat-users.xml`

In Tomcat 7.0 access to the manager application is split between different users.

<localhost/docs/jndi-datasource-examples-howto.html>

Documentation

[Tomcat 7.0 Documentation](#)

[Tomcat 7.0 Configuration](#)

[Tomcat Wiki](#)

Find additional important configuration information in:

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

announce@tomcat.apache.org
Important announcements, releases, security vulnerability notifications. (Low volume).

users@tomcat.apache.org

Apache Tomcat 7.0.34

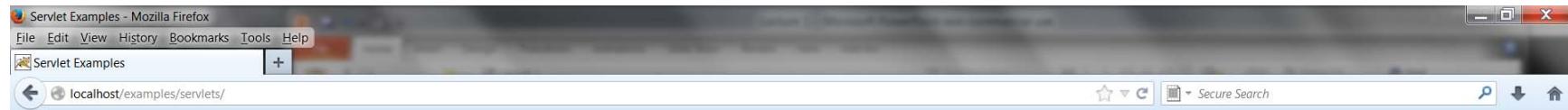


Apache Tomcat Examples

- [Servlets examples](#)
- [JSP Examples](#)
- [WebSocket Examples](#)



Apache Tomcat 7.0.34



Servlet Examples with Code

This is a collection of examples which demonstrate some of the more frequently used parts of the Servlet API. Familiarity with the Java(tm) Programming Language is assumed.

These examples will only work when viewed via an http URL. They will not work if you are viewing these pages via a "file://..." URL. Please refer to the *README* file provided with this Tomcat release regarding how to configure and start the provided web server.

Wherever you see a form, enter some data and see how the servlet reacts. When playing with the Cookie and Session Examples, jump back to the Headers Example to see exactly what your browser is sending the server.

To navigate your way through the examples, the following icons will help:

-  Execute the example
-  Look at the source code for the example
-  Return to this screen

Tip: To see the cookie interactions with your browser, try turning on the "notify when setting a cookie" option in your browser preferences. This will let you see when a session is created and give some feedback when looking at the cookie demo.

Hello World
Request Info
Request Headers
Request Parameters
Cookies
Sessions

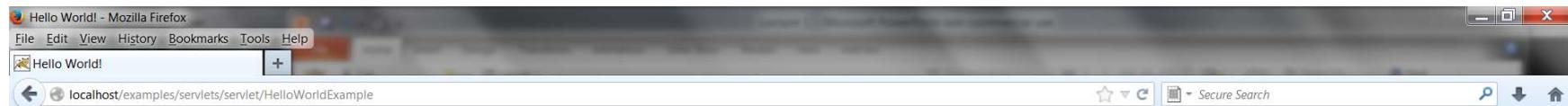
 [Execute](#)
 [Execute](#)
 [Execute](#)
 [Execute](#)
 [Execute](#)
 [Execute](#)

 [Source](#)
 [Source](#)
 [Source](#)
 [Source](#)
 [Source](#)
 [Source](#)

Note: The source code for these examples does not contain all of the source code that is actually in the example, only the important sections of code. Code not important to understand the example has been removed for clarity.

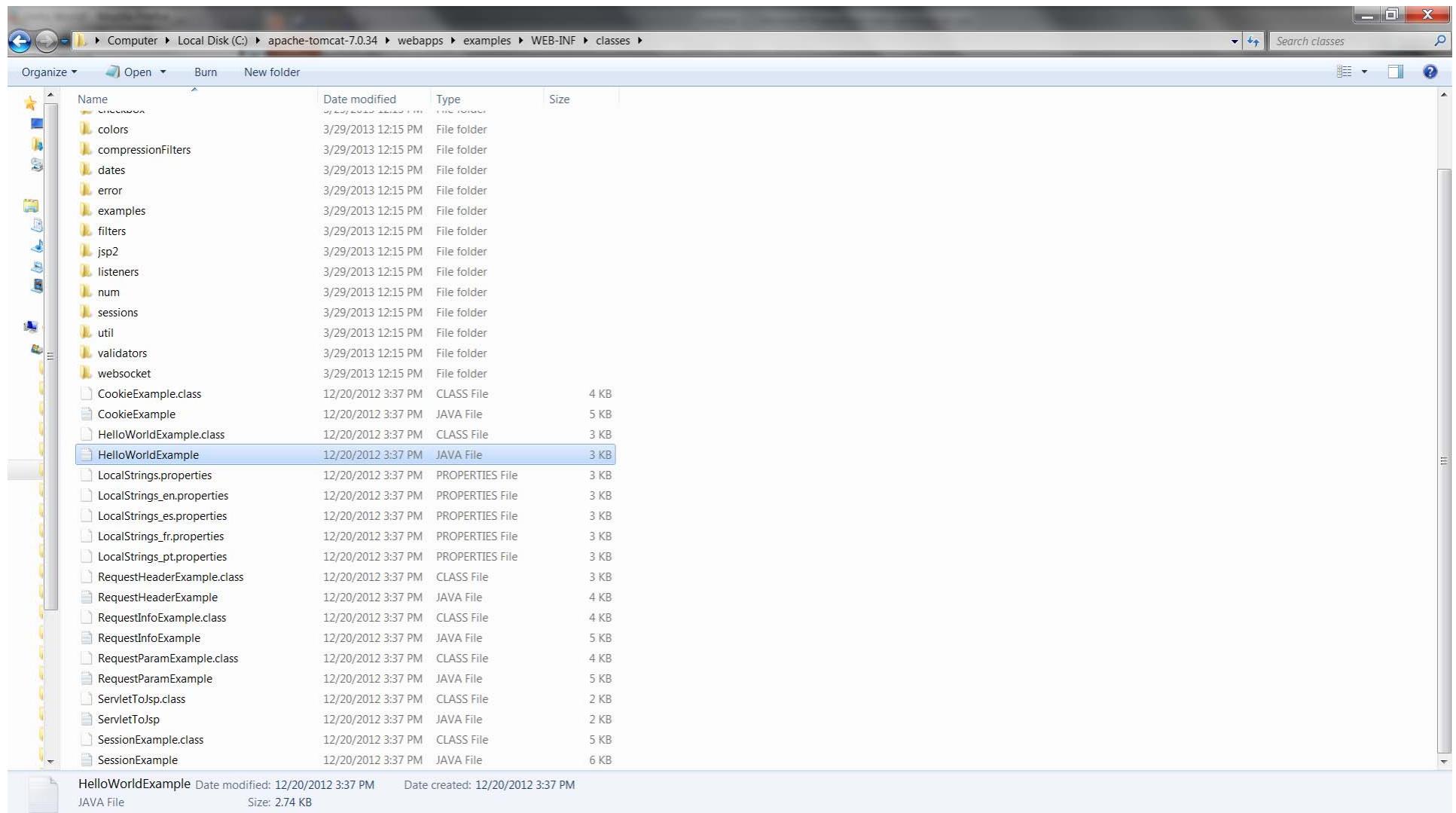


Apache Tomcat 7.0.34



Hello World!

Apache Tomcat 7.0.34



The components of an HTTP URL

`http://localhost:8080/examples/servlets/index.html`

protocol host port path filename

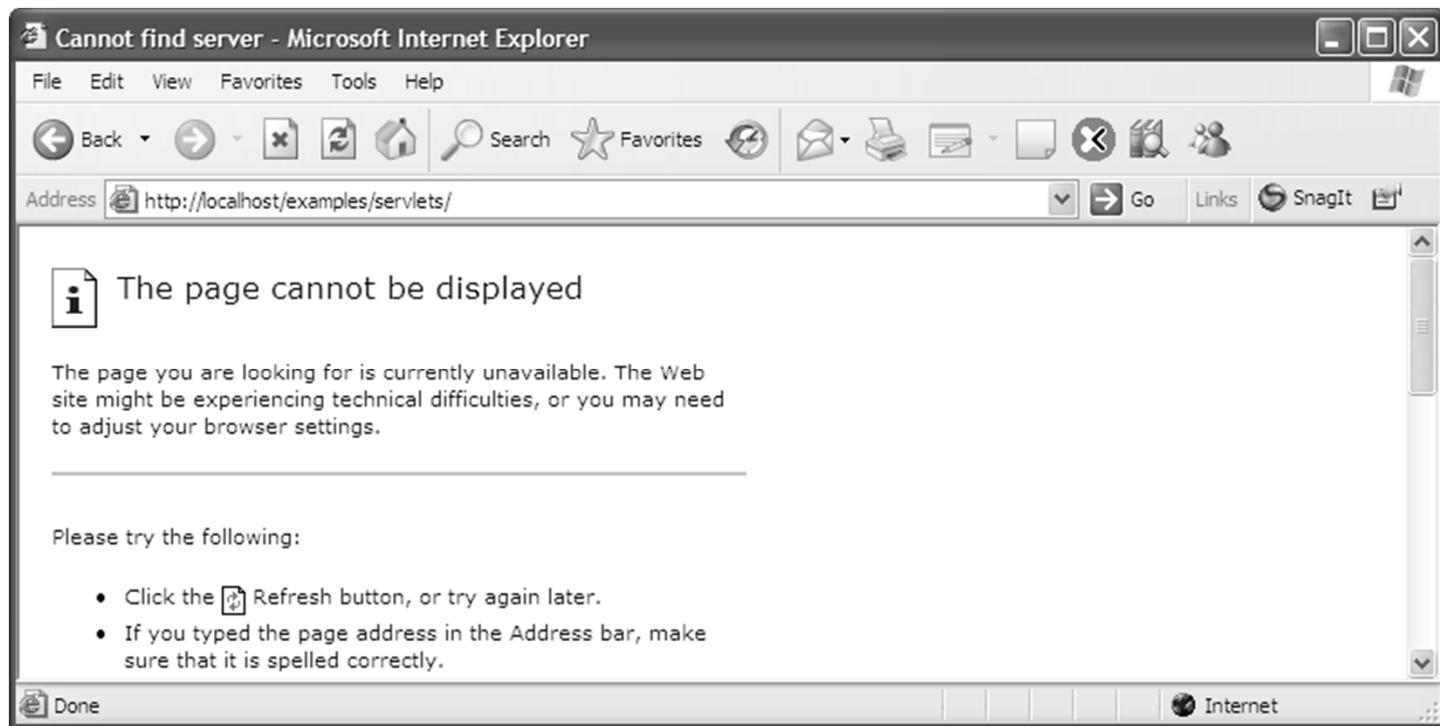
HTTP URLs that you can use to test Tomcat

`http://localhost:8080/`
`http://localhost:8080/examples/servlets/`
`http://localhost:8080/examples/jsp/`

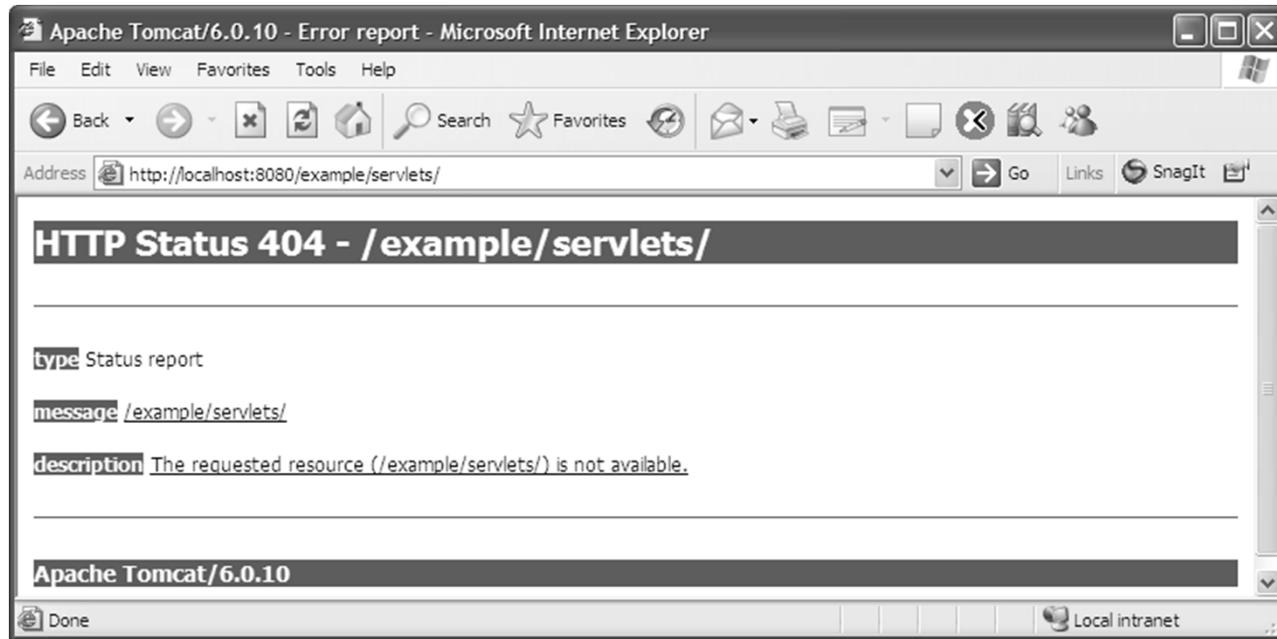
How to view a web page via an HTTP URL

1. Start Tomcat.
2. Start your web browser.
3. Type an HTTP URL into your web browser and press Enter.

The Internet Explorer's error page



Tomcat's default 404 error page



How to solve common Tomcat problems

- If the browser displays an error page that says “The page cannot be displayed,” the HTTP request isn’t connecting with a web server. To solve this problem, make sure that the Tomcat engine is running, and make sure that you’ve entered a valid host name and port number.
- If the browser displays a 404 error page, Tomcat is receiving the HTTP request, but it can’t find the requested resource. To solve this problem, make sure that you’ve entered the path and filename of the URL correctly.

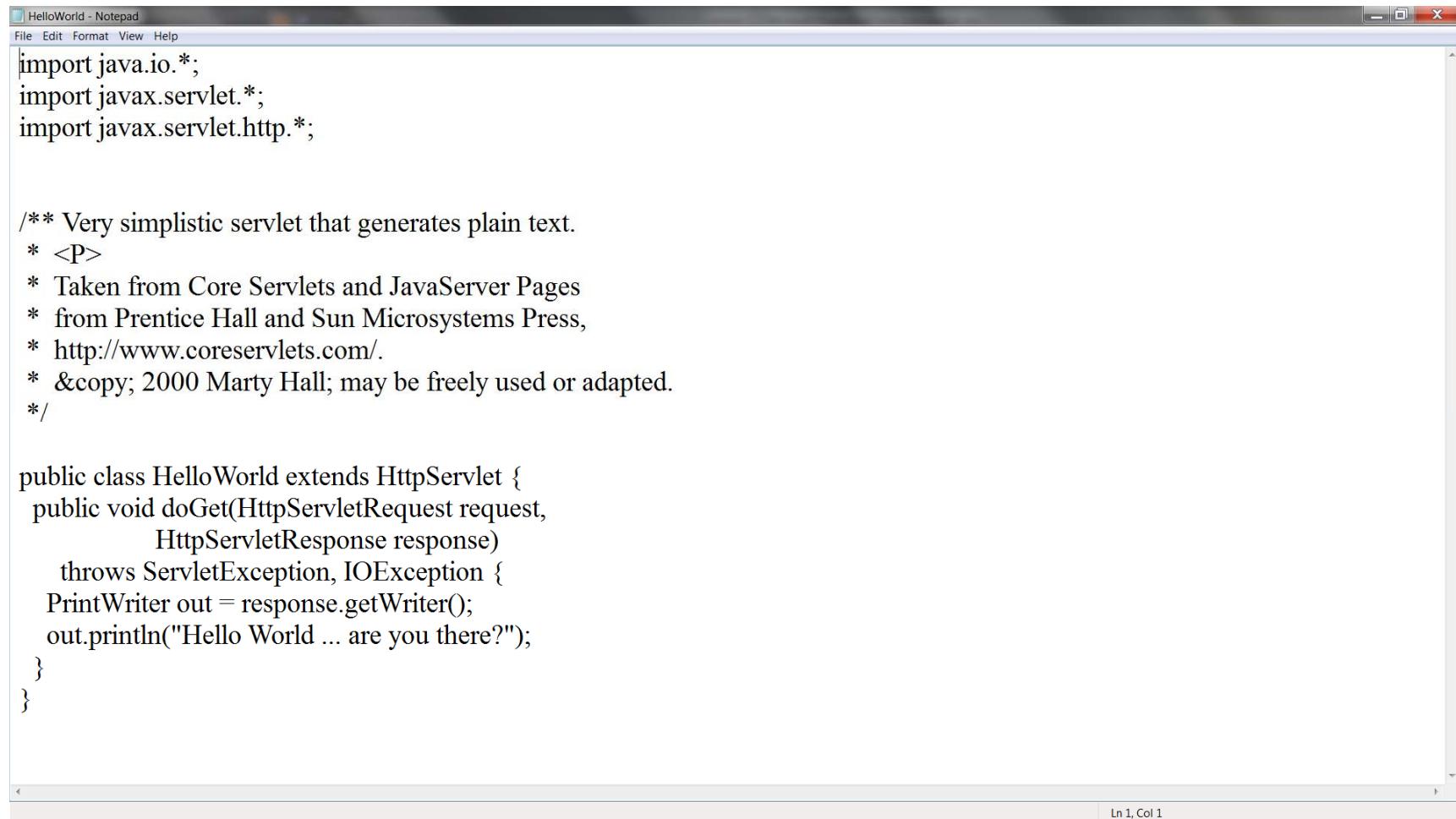
Compiling and Invoking Servlets

- Set your CLASSPATH
 - Servlet JAR file (e.g., *install_dir/lib/servlet-api.jar*).
 - Top of your package hierarchy
- Put your servlet classes in proper location
 - Locations vary from server to server. E.g.,
 - *tomcat_install_dir/webapps/ROOT/WEB-INF/classes*
See <http://archive.coreservlets.com/Using-Tomcat.html>
- Invoke your servlets
 - *http://host/servlet/ServletName*
 - Custom URL-to-servlet mapping (via *web.xml*)

Compiling and Invoking Servlet (Tomcat 7.0.34; Class Setup)

- Place your source code in
 - C:\csj
- Or
 - C:\apache-tomcat-7.0.34\webapps\csj
- CLASSPATH already set
- Start DOS; type "javac HelloWorld.java"
- Place HelloWorld.class in servlet directory
 - **C:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes**
- **OR**
- **C:\apache-tomcat-7.0.34\webapps\ROOT\WEB-INF\classes**

Compiling and Invoking Servlet (Tomcat 7.0.34; Class Setup)



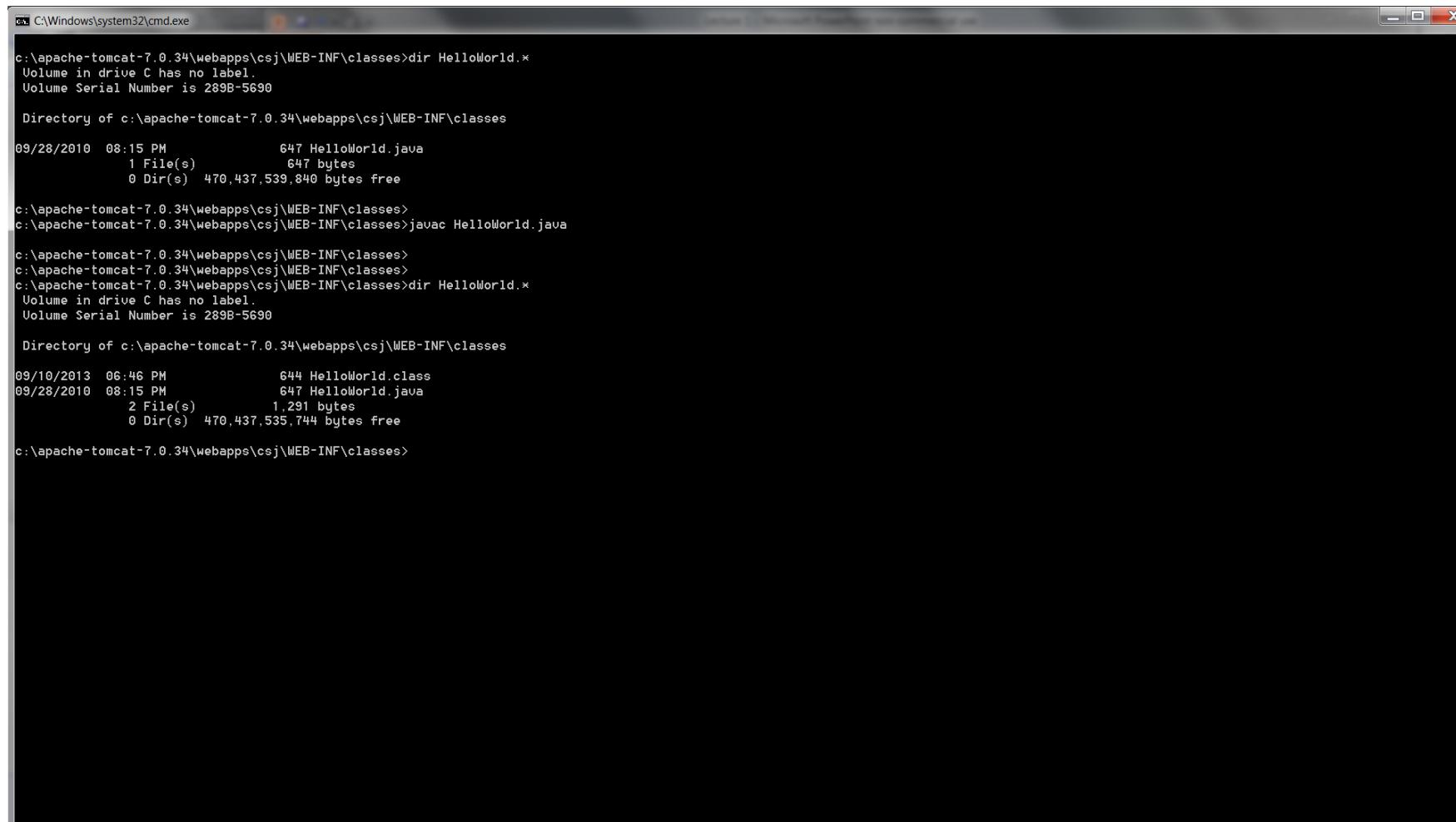
The screenshot shows a Windows Notepad window titled "HelloWorld - Notepad". The window contains Java code for a servlet named "HelloWorld". The code includes imports for java.io, javax.servlet, and javax.servlet.http. It features a multi-line comment at the top providing copyright and source information. The main class definition follows, extending HttpServlet and overriding the doGet method to print "Hello World ... are you there?" to the response writer.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

/** Very simplistic servlet that generates plain text.
 * <P>
 * Taken from Core Servlets and JavaServer Pages
 * from Prentice Hall and Sun Microsystems Press,
 * http://www.coreservlets.com/.
 * © 2000 Marty Hall; may be freely used or adapted.
 */

public class HelloWorld extends HttpServlet {
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        out.println("Hello World ... are you there?");
    }
}
```

Compiling and Invoking Servlet (Tomcat 7.0.34; Class Setup)



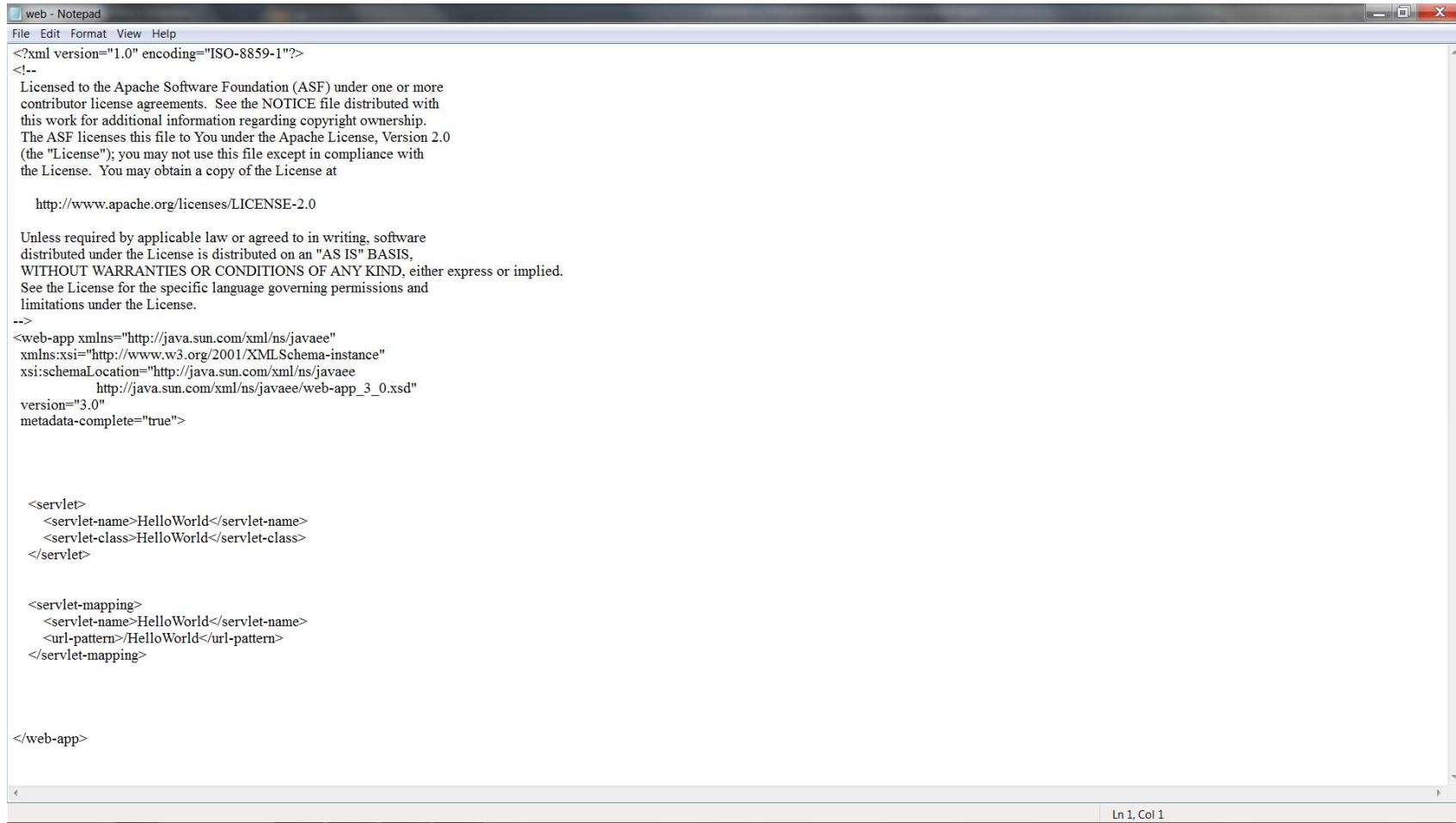
A screenshot of a Windows Command Prompt window titled "cmd C:\Windows\system32\cmd.exe". The window displays a command-line session for compiling a Java servlet named "HelloWorld".

```
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>dir HelloWorld.*  
Volume in drive C has no label.  
Volume Serial Number is 289B-5690  
  
Directory of c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes  
  
09/28/2010 08:15 PM      647 HelloWorld.java  
               1 File(s)      647 bytes  
               0 Dir(s) 470,437,539,840 bytes free  
  
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>  
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>javac HelloWorld.java  
  
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>  
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>  
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>dir HelloWorld.*  
Volume in drive C has no label.  
Volume Serial Number is 289B-5690  
  
Directory of c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes  
  
09/10/2013 06:46 PM      644 HelloWorld.class  
09/28/2010 08:15 PM      647 HelloWorld.java  
               2 File(s)      1,291 bytes  
               0 Dir(s) 470,437,535,744 bytes free  
  
c:\apache-tomcat-7.0.34\webapps\csj\WEB-INF\classes>
```

Compiling and Invoking Servlet (Tomcat 7.0.34; Class Setup)

- You can create your own context “subdirectory” under the root directory for tomcat:
 - C:\apache-tomcat-7.0.34\webapps\csj
- Keep all of your homeworks, projects, etc. under
 - C:\apache-tomcat-7.0.34\webapps\csj
- Start server
 - From MS-DOS cmd type startup.bat and hit enter
- Invoke servlet
 - <http://localhost/csj>HelloWorld>

Web.xml to Invoke Servlet



The screenshot shows a Windows Notepad window titled "web - Notepad". The content of the file is the JavaServer Faces configuration file (Web.xml) for a simple "HelloWorld" application.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<web-app xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
    http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
  version="3.0"
  metadata-complete="true">

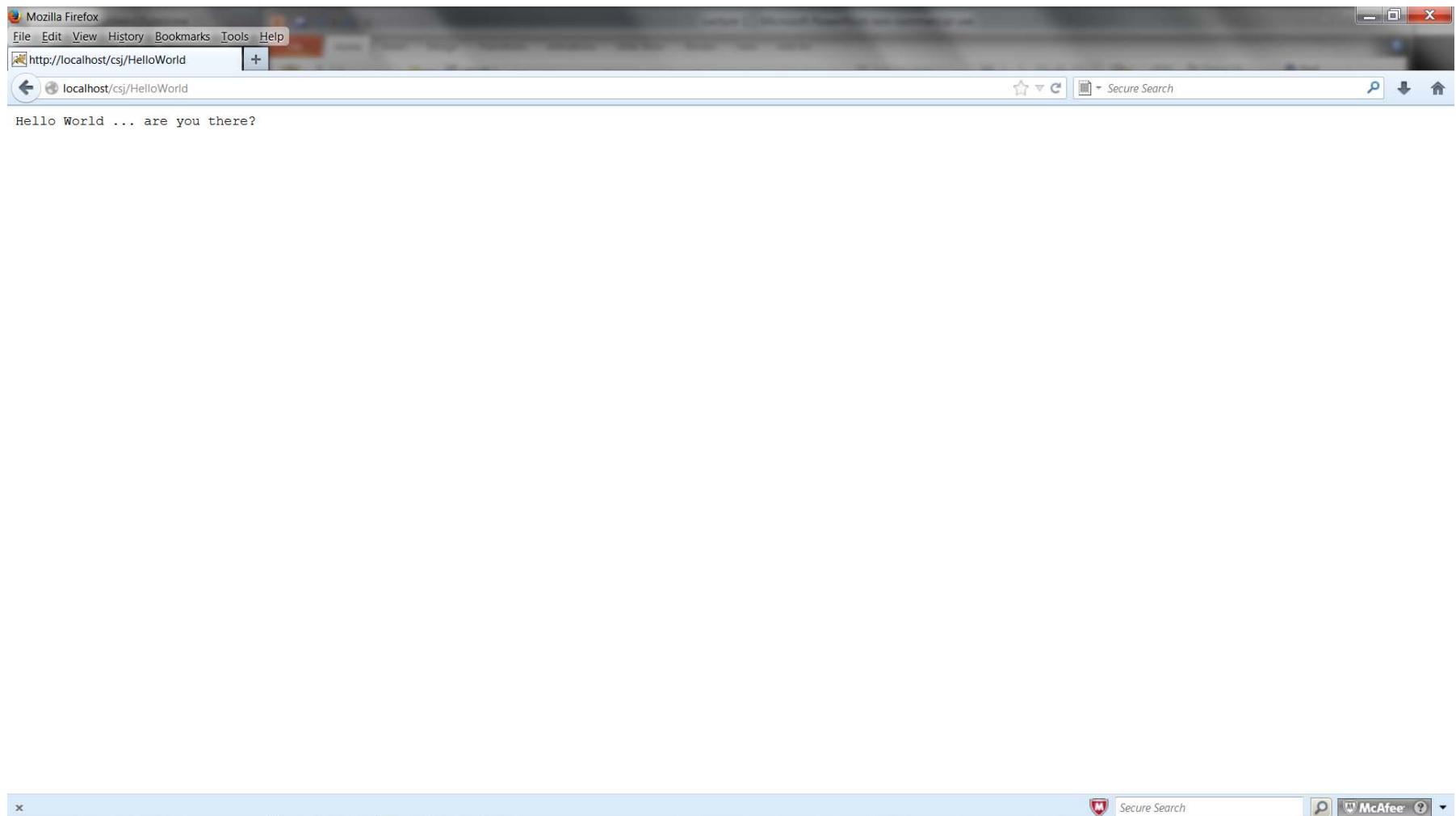
  <servlet>
    <servlet-name>HelloWorld</servlet-name>
    <servlet-class>HelloWorld</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>HelloWorld</servlet-name>
    <url-pattern>/HelloWorld</url-pattern>
  </servlet-mapping>

</web-app>
```

The status bar at the bottom of the Notepad window displays "Ln 1, Col 1".

Compiling and Invoking Servlet



How to add a username and password for the manager role

- Open the tomcat-users.xml file that's in Tomcat's conf directory in a text editor.
- Add a role element that defines the manager role.
- Add a user element that provides a username and password for the manager role.
- If Tomcat is running when you add users to the tomcat-users.xml file, you need to restart Tomcat after you close the file so Tomcat will recognize the changes.

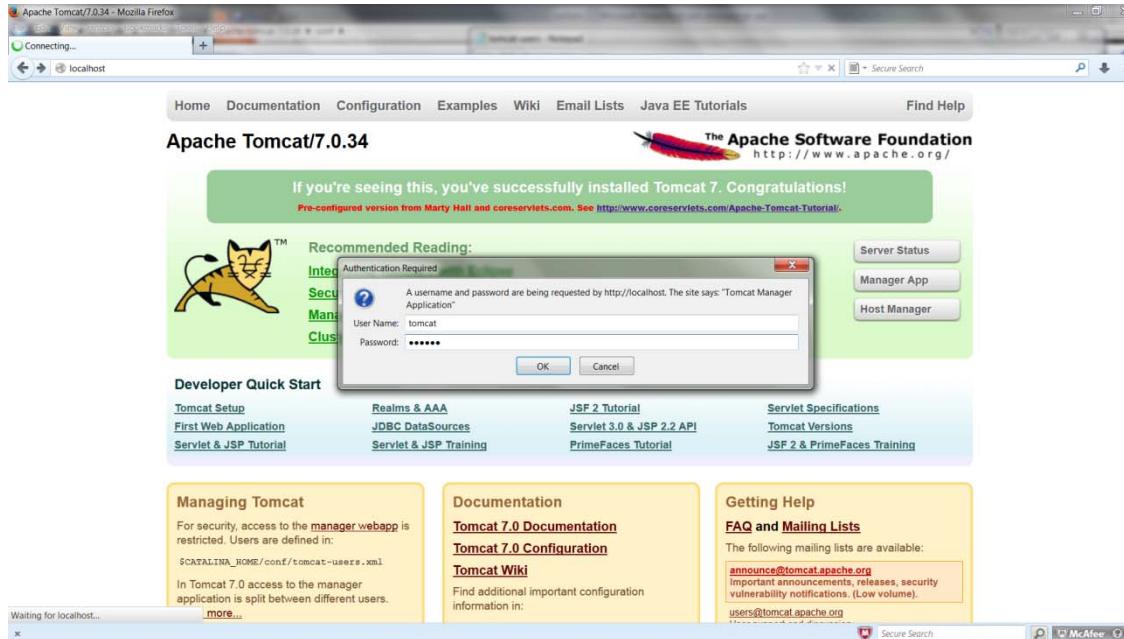
The tomcat-users.xml file

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
    <role rolename="manager"/>
    <user username="admin" password="sesame" roles="manager"/>
</tomcat-users>
```

How to start Tomcat's Web Application Manager

- To start the Web Application Manager, start a web browser and go to <http://localhost/manager/html>.
- Tomcat will prompt you for a username and password. If you supply a valid username and password for the manager role, you will be able to view the Web Application Manager.

The Authentication Required dialog box



/manager - Mozilla Firefox

File Edit View History Bookmarks Tools Help

/manager

localhost/manager/html

Secure Search

The Apache Software Foundation <http://www.apache.org/>

TM

Tomcat Web Application Manager

Message: OK

Manager

List Applications [HTML Manager Help](#) [Manager Help](#) [Server Status](#)

Applications

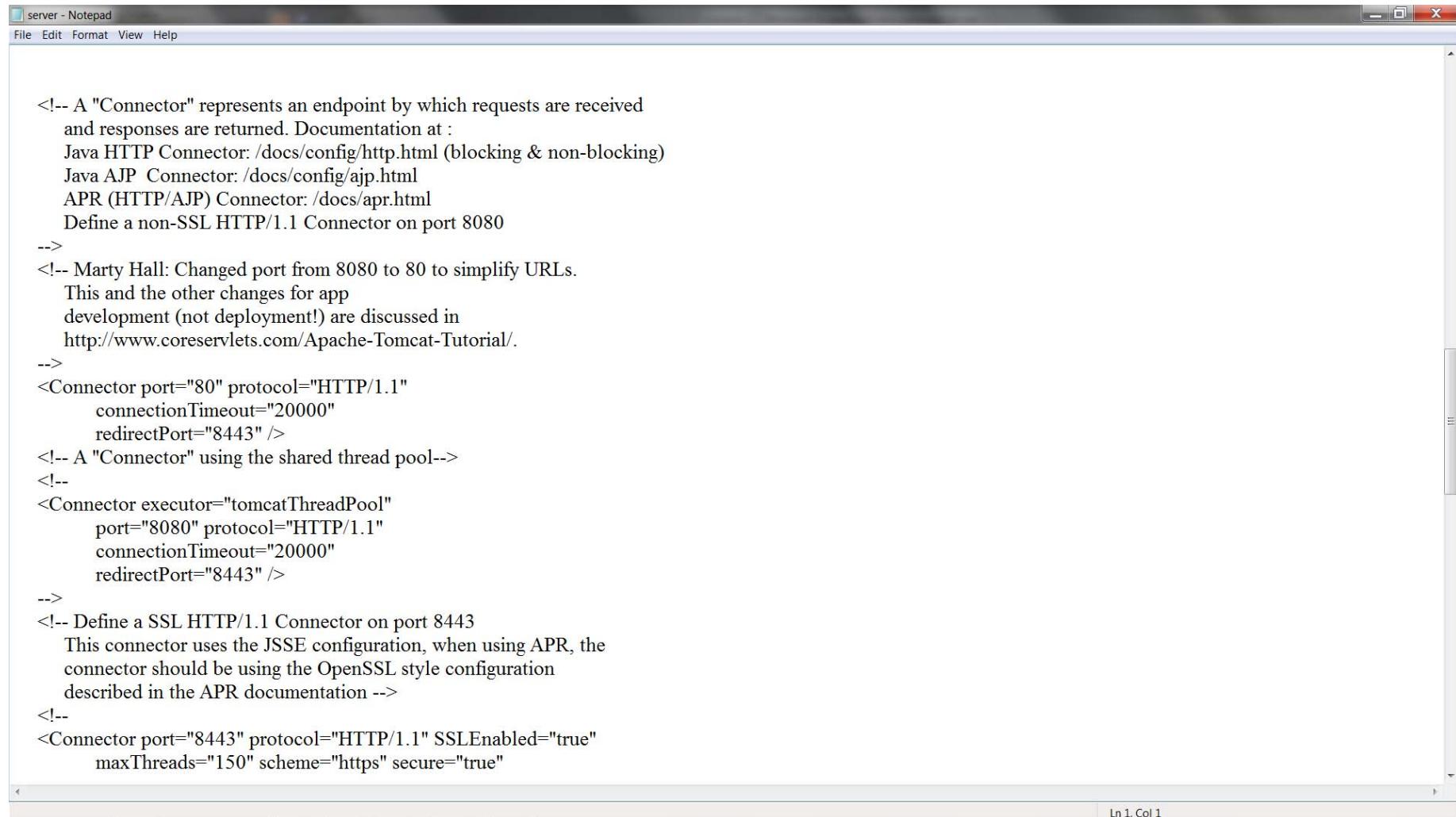
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/ajax-basics-1	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/ajax-basics-2	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/ajax-my-app	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/csi	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

x Secure Search McAfee ?

How to use Tomcat's Web Application Manager

- To reload all of the classes for an application, click on the Reload link for the application.
- To stop an application, click on the Stop link for the application.
- To start an application, click on the Start link for the application.
- To undeploy an application, click on the Undeploy link for the application. This deletes all files for the web application from Tomcat's server.

The server.xml file



server - Notepad

File Edit Format View Help

```
<!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Documentation at :
Java HTTP Connector: /docs/config/http.html (blocking & non-blocking)
Java AJP Connector: /docs/config/ajp.html
APR (HTTP/AJP) Connector: /docs/apr.html
Define a non-SSL HTTP/1.1 Connector on port 8080
-->
<!-- Marty Hall: Changed port from 8080 to 80 to simplify URLs.
This and the other changes for app
development (not deployment!) are discussed in
http://www.coreservlets.com/Apache-Tomcat-Tutorial/.
-->
<Connector port="80" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
           port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
-->
<!-- Define a SSL HTTP/1.1 Connector on port 8443
This connector uses the JSSE configuration, when using APR, the
connector should be using the OpenSSL style configuration
described in the APR documentation -->
<!--
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
           maxThreads="150" scheme="https" secure="true"
```

Ln 1, Col 1

How to change the port that's used by Tomcat

1. Use the Windows Explorer to navigate to Tomcat's conf directory and open the server.xml file in a text editor.
2. Replace all instances of the current port, which is 8080 by default, to a four-digit number that's greater than 1024 or to 80. To do this, you may want to use the Find and Replace feature of your text editor.
3. Save the changes to the server.xml file.
4. Stop and restart Tomcat.

Notes

- If you have a port conflict with another application, you can change the 8080 default to a 4-digit number greater than 1024.
- If you don't enter a port when you specify a URL, your browser will use port 80. So if you change Tomcat's default port to 80, you don't need to specify a port when entering a URL.

The standard directories and files for a web application

- The top-level directory for a web application is known as its *root directory*.
- A Java web application is a hierarchy of directories and files in a standard layout defined by the Java EE specification. All Java web applications must use the root, \WEB-INF, and \WEB-INF\classes directories.
- To make classes within a JAR file available to more than one web application, you can put the JAR file in Tomcat's lib directory.

A web.xml file

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<web-app xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
        http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
    version="2.5">

    <display-name>Murach's Servlets and JSP:
        Music Store site</display-name>

    <description>The Music Store web site that's described
        in Murach's Java Servlets and JSP (second
        edition)</description>

    <!-- Enable servlet mapping -->
    <servlet>
        <servlet-name>AddToEmailListServlet</servlet-name>
        <servlet-class>music.email.AddToEmailListServlet
            </servlet-class>
    </servlet>
```

A web.xml file (continued)

```
<!-- Map servlets to URL patterns -->
<servlet-mapping>
    <servlet-name>AddToEmailListServlet</servlet-name>
    <url-pattern>/email/addToEmailList</url-pattern>
</servlet-mapping>

<!-- Specify index pages -->
<welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
</welcome-file-list>

</web-app>
```

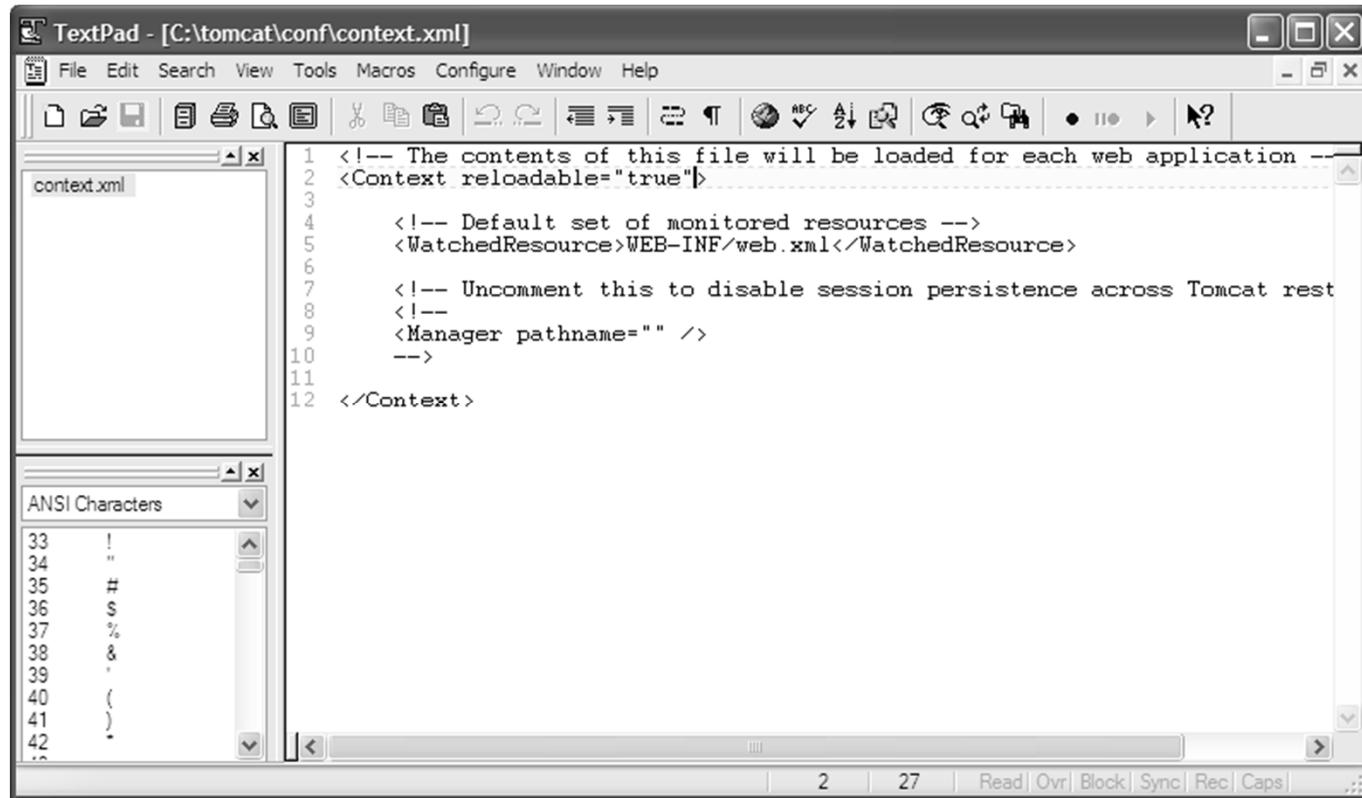
Description

- Every web application requires a web.xml file in the WEB-INF directory. This file is known as the *deployment descriptor* for the web application. The highlighted code is required in the file.

What the web.xml file can do

- Enable servlet mapping so you can call a servlet using any URL or URL pattern.
- Define initialization parameters for a servlet or the entire application.
- Define error pages for an entire application.
- Provide security constraints to restrict access to certain web pages and servlets.

The context.xml file



A screenshot of the TPad text editor window. The title bar reads "TextPad - [C:\tomcat\conf\context.xml]". The menu bar includes File, Edit, Search, View, Tools, Macros, Configure, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Find, and Print. The main editor area displays the XML code for context.xml:

```
1 <!-- The contents of this file will be loaded for each web application -->
2 <Context reloadable="true">
3
4     <!-- Default set of monitored resources -->
5     <WatchedResource>WEB-INF/web.xml</WatchedResource>
6
7     <!-- Uncomment this to disable session persistence across Tomcat restart
8     <!--
9         <Manager pathname="" />
10    -->
11
12 </Context>
```

To the left of the main editor is a floating character palette titled "ANSI Characters" with a scroll bar. It lists characters from 33 to 42 with their corresponding symbols: !, ", #, \$, %, &, :, (,), .

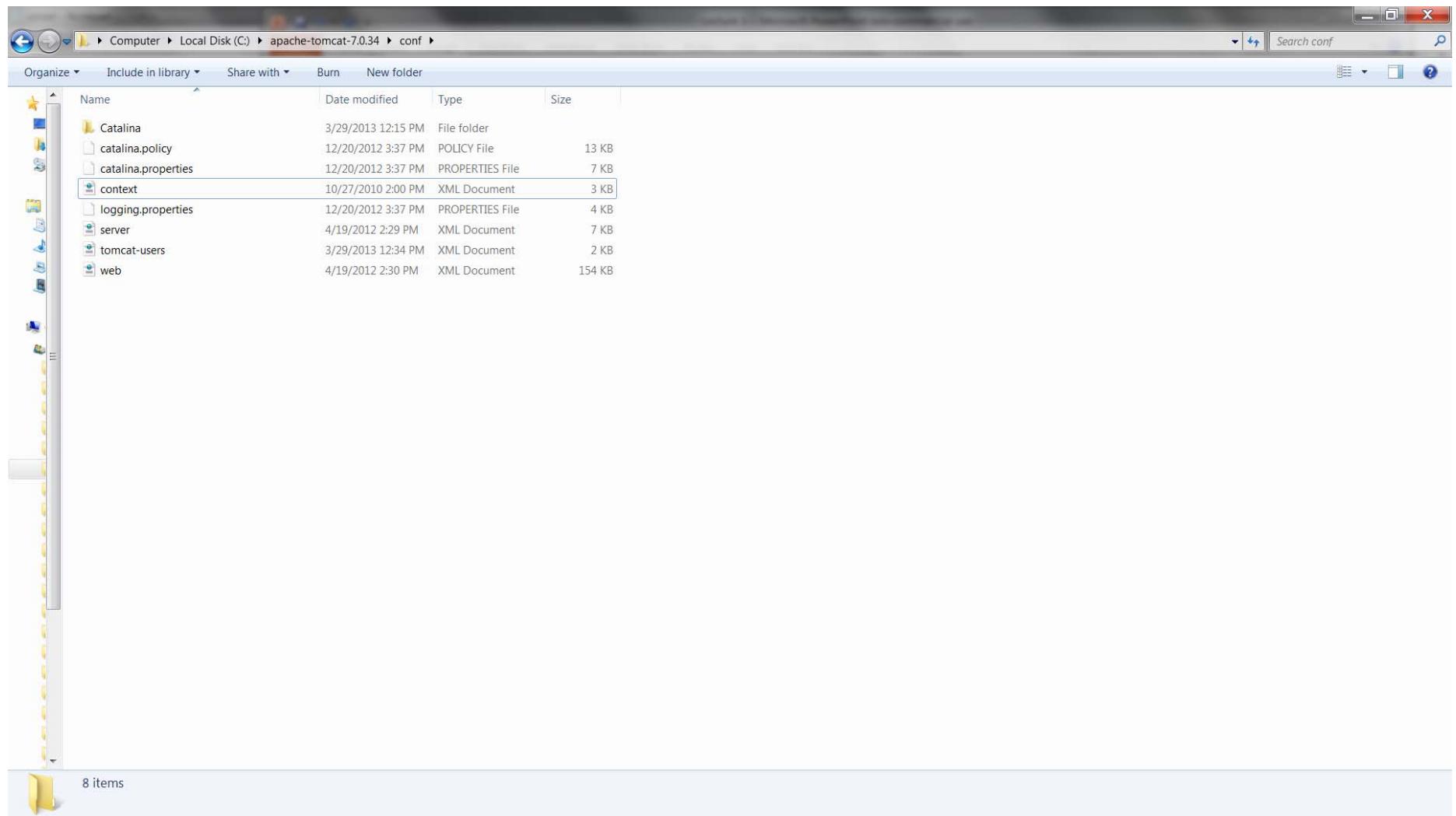
How to turn on servlet reloading

1. Use a text editor to open the context.xml file in Tomcat's conf directory.
2. Add the reloadable attribute to the Context element and set this attribute to true like this:
`<Context reloadable="true">`
3. Save the changes to the context.xml file
4. If necessary, stop and restart Tomcat.

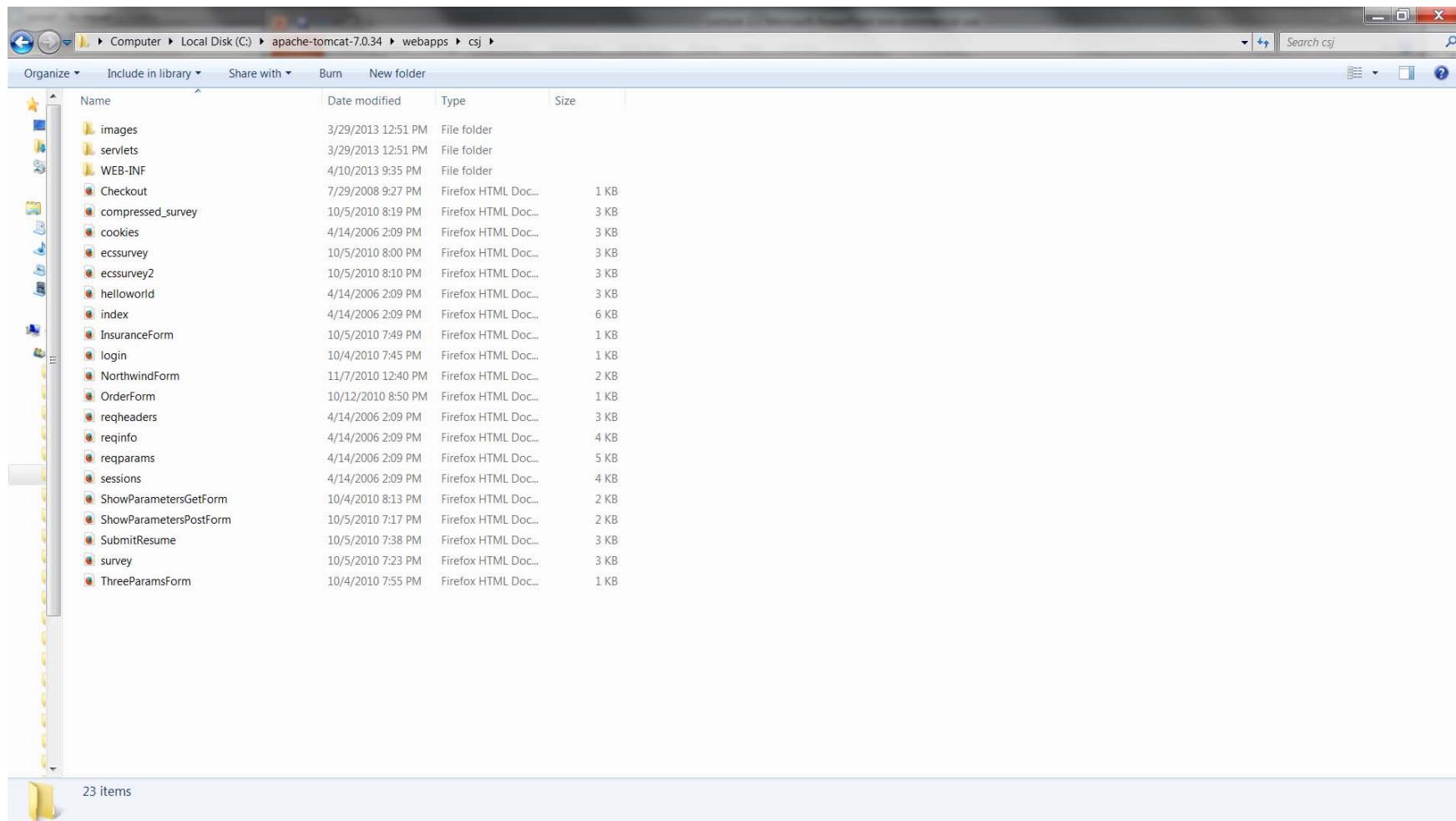
Servlet reloading

- If *servlet reloading* isn't on, which is the default setting, you have to stop and restart Tomcat each time that you change one of the classes that's in memory.
- If you turn servlet reloading on, Tomcat checks the modification dates of the classes in memory and automatically reloads the ones that have changed. Although this is useful in development, it can cause performance problems in a production environment.
- The context.xml file is an XML file that controls how the Tomcat engine is configured. Tomcat reads this file every time it starts to configure itself. You can use a text editor to edit this file. Then, you can stop and restart Tomcat to put the changes into effect.
- This is a global setting that affects all web applications running on this instance of Tomcat.

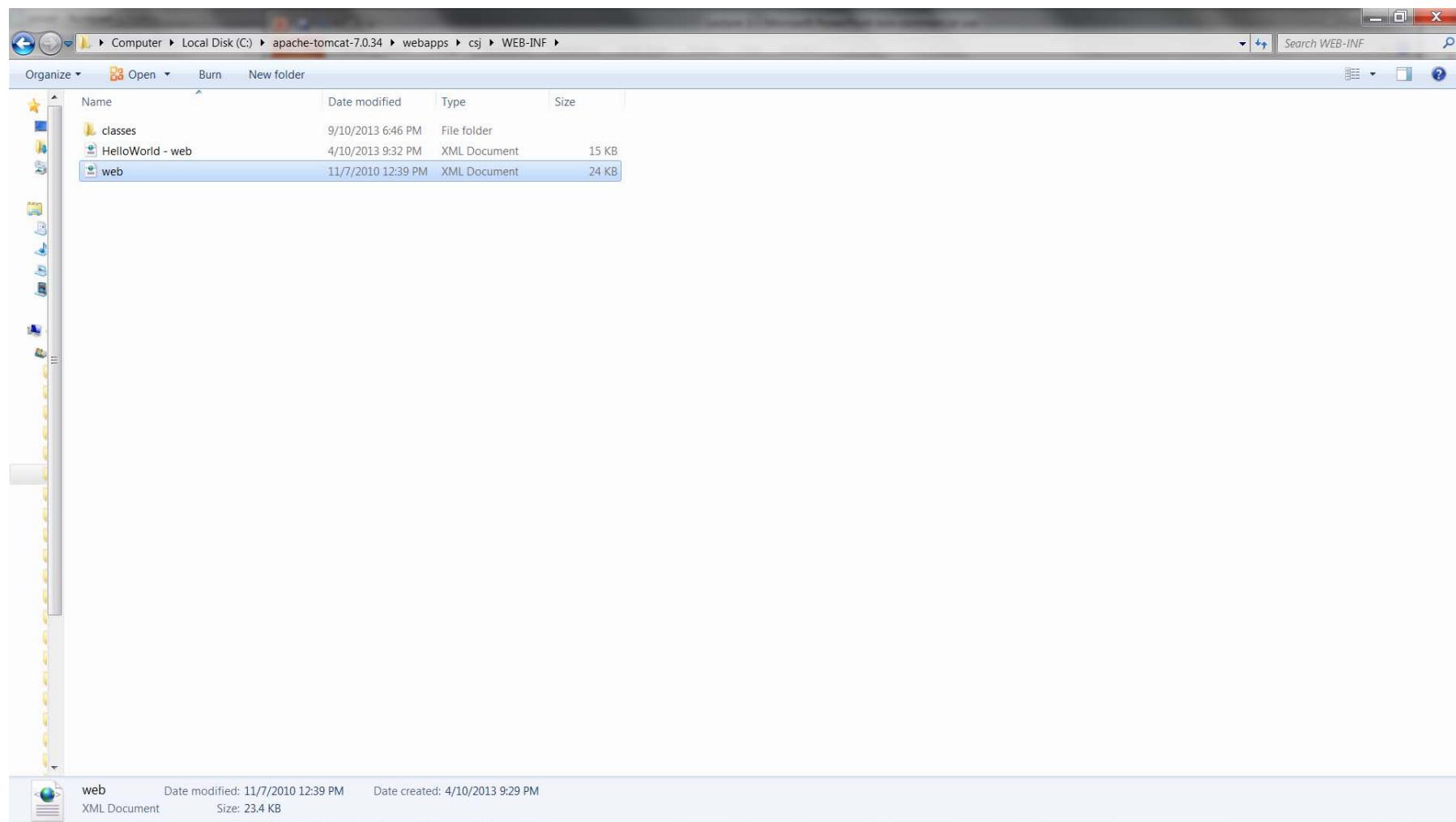
■ What is in the configuration directory?



■ What is in the csj directory?



■ What is in the csj/WEB-INF directory?



- Finally:

- Please finish the installation and testing for Tomcat
- Please compile and run HelloWorld servlet example
- Make sure env-setup-for-tomcat is executed before you start running tomcat and compiling your servelts