

Web Technology

Web Clients and Web Servers

R S Milton

February 16, 2011

Web clients

Web clients

- Send a HTTP request message to web server
- Process the HTTP response message from the web server
- User agents — directly used by human users
- Software robots — crawl web collecting information



Basic functions of browser

Browser areas

- Menu bar
- Tool bar (within it, location bar)
- Client area — document displayed
- Status bar

Basic functions of browser

Basic functions (Scheme URL)

- Create HTTP request message
- DNS — Host name → IP address of server
- TCP connection with server
- Send HTTP request message
- Process document in the HTTP response message

Status message	Meaning
Resolving	Looking up DNS with host name, waiting for IP address
Connecting	Creating TCP connection with server
Waiting	Sent HTTP request to server, waiting for HTTP response
Transferring data	Started receiving HTTP response, not yet completed
Done	Received HTTP response completely, further processing may be needed before display

- TCP connection with server machine — based on port, deliver it to server
- Document processing
 - text/html — *render*
 - font parameters, layout parameters

URL

Scheme	://	Host:Port	Path	?	Query string	#	Fragment
Scheme	://	Authority	Request-URI			#	Fragment

Example: http:// www.example.org:8080 /a/b/c.html ? t=win # here

Additional functionality

- Automatic URL completion
- Script execution
- Event handling
- Management of form
- Secure communication
- Plug-in execution

Cache control

Cache:

A repository for copies of information that originate elsewhere

- Automatically cache on the client computer copies of the resources returned by the server
- Improves system performance

- Quicker display
- Reduced network traffic
- Reduced load on the server

Cache control — Invalid information

- *Resource in the server is modified*
- Cached copy becomes old — *invalid*
- Ask the server whether or not the client's copy is valid
- Send HTTP request using HEAD method
- Server returns the status line and the header fields
 - ★ Last-modified
 - ★ Expires
 - ★ ETag
- Little network traffic

Features

- File → Save Page As
- Edit → Find
- View → Page Style
- View → Page Source
- History
- Bookmarks

Features

Edit → Preferences

Customize browser behavior

- Content → Languages

Preferred languages in order of preference

- Content → Fonts and Colors

Character encoding

- Advanced → Offline storage (Cache)
- Main → Add-ons → Themes

Web servers

Server features

- Accept HTTP request from web clients
- Return requested resources in the HTTP response

Web browser

Web server

- | | |
|---|---|
| • Send a HTTP request to web server | • Accept HTTP request from web clients |
| • Process the HTTP response from the web server | • Return requested resources in the HTTP response |

Server salient features

- Multiple concurrent threads (subtasks)
 - Server can (should?) handle multiple requests concurrently
 - One server *copy* (subtask) per client
- Virtual hosts

- Several Host names = one IP address (one computer)
- Host header determines virtual host
- Each virtual host has separately configured software
- Server side scripting
 - On receiving a request, response body is generated by executing programs
 - Support for executing programs

Sever features

- Open TCP connection to one or more ports
- Spawn one thread to handle this connection
- Establish TCP connection with the client, receive an HTTP request
- Host header field → Virtual host
- Map the Request-URI to a resource on the server
- Create a HTTP response

- Resource
 - File → body contains the requested file
 - Program → run the program with the request body as the input data and put the output data in the response body
- Log — IP address of the request, status of the response
- Connection = keep-alive, thread monitors the connection, until
 - Client sends another request, or
 - Initiates a connection close

Server history

Web servers

- Apache
- IIS

Servlet container

- Web servers call a *Servlet container* to run server side Java programs (*servlets*)
- Servlet container provides
 - JVM
 - Communication between the servlet and web server

Tomcat

- Servlet container
- Standalone web server
- Runs non-Java programs also!
- Configure web server

Server configuration and tuning

- Configuration parameters
 - External communication
 - Internal processing
- Tomcat
 - Coyote — HTTP/1.1 communication
 - Catalina — Servlet container

External communication parameters

- IP addresses and ports used
- Optimal number of threads
- Max number of threads
- Max number of TCP connection requests in the queue
- After serving a request, how long to wait before closing the TCP connection if another request is not received

Tuning parameters

- Adjust parameters for performance — *Tuning*
- Interplay and trade-off
 - Max # of threads $\uparrow \Rightarrow$ memory, CPU \Rightarrow slower response
 - Max # of threads $\downarrow \Rightarrow$ clients refused connection
 - Simulate before “going alive” — load generation and stress test

Internal processing parameters

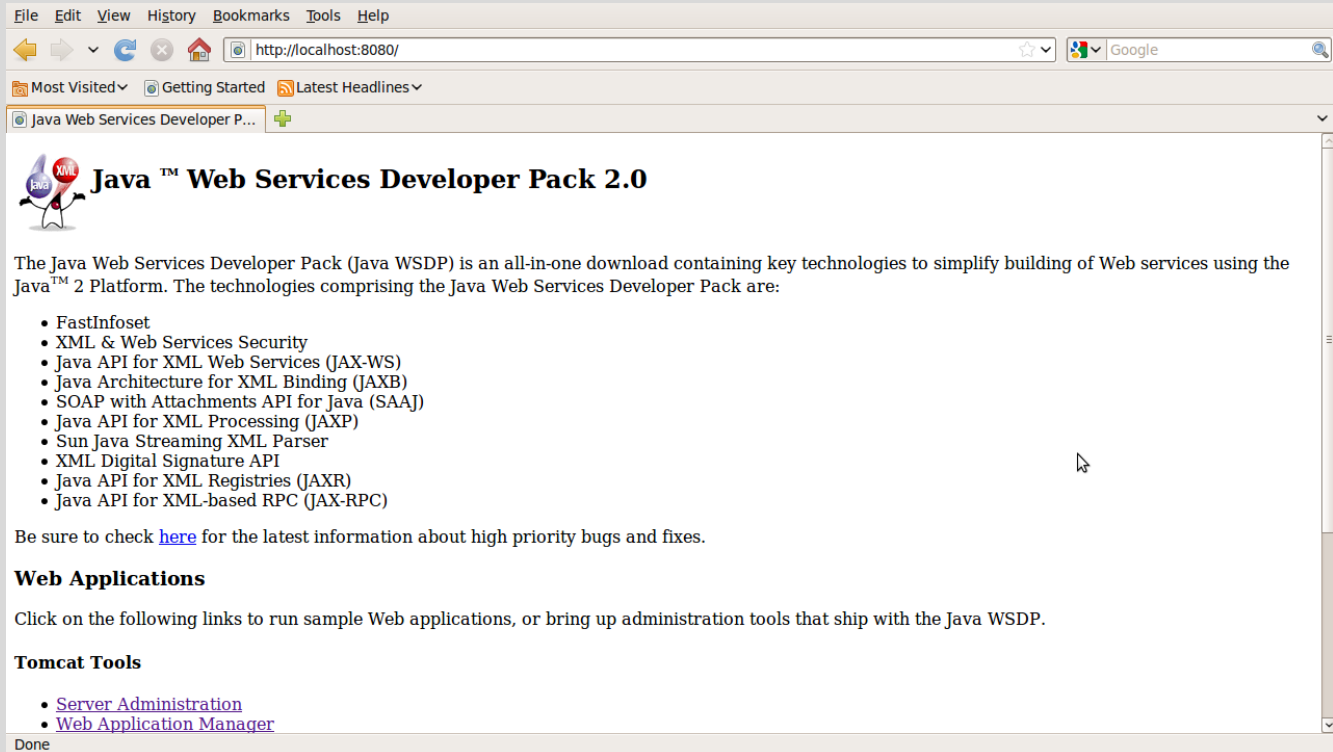
- Client machines allowed
- Virtual hosts listening on a port
- Logging
- Request-URI → server's file system
- Resources password-protected?
- Resources cached in server's memory?

Components of service

- Connector
- Host
- Logger
- Realm
- Valve

Edit, Save, Commit, Restart

JWSDP



The screenshot shows a web browser window with the address bar set to `http://localhost:8080/`. The browser's menu bar includes File, Edit, View, History, Bookmarks, Tools, and Help. Below the address bar, there are tabs for 'Most Visited', 'Getting Started', and 'Latest Headlines'. The active tab is titled 'Java Web Services Developer P...'. The main content area displays the 'Java™ Web Services Developer Pack 2.0' page. It features the Java logo and a list of technologies included in the pack. A mouse cursor is visible on the right side of the page. At the bottom of the browser window, the status bar shows 'Done'.

File Edit View History Bookmarks Tools Help

http://localhost:8080/ Google

Most Visited Getting Started Latest Headlines

Java Web Services Developer P...

Java™ Web Services Developer Pack 2.0

The Java Web Services Developer Pack (Java WSDP) is an all-in-one download containing key technologies to simplify building of Web services using the Java™ 2 Platform. The technologies comprising the Java Web Services Developer Pack are:

- FastInfoset
- XML & Web Services Security
- Java API for XML Web Services (JAX-WS)
- Java Architecture for XML Binding (JAXB)
- SOAP with Attachments API for Java (SAAJ)
- Java API for XML Processing (JAXP)
- Sun Java Streaming XML Parser
- XML Digital Signature API
- Java API for XML Registries (JAXR)
- Java API for XML-based RPC (JAX-RPC)

Be sure to check [here](#) for the latest information about high priority bugs and fixes.

Web Applications

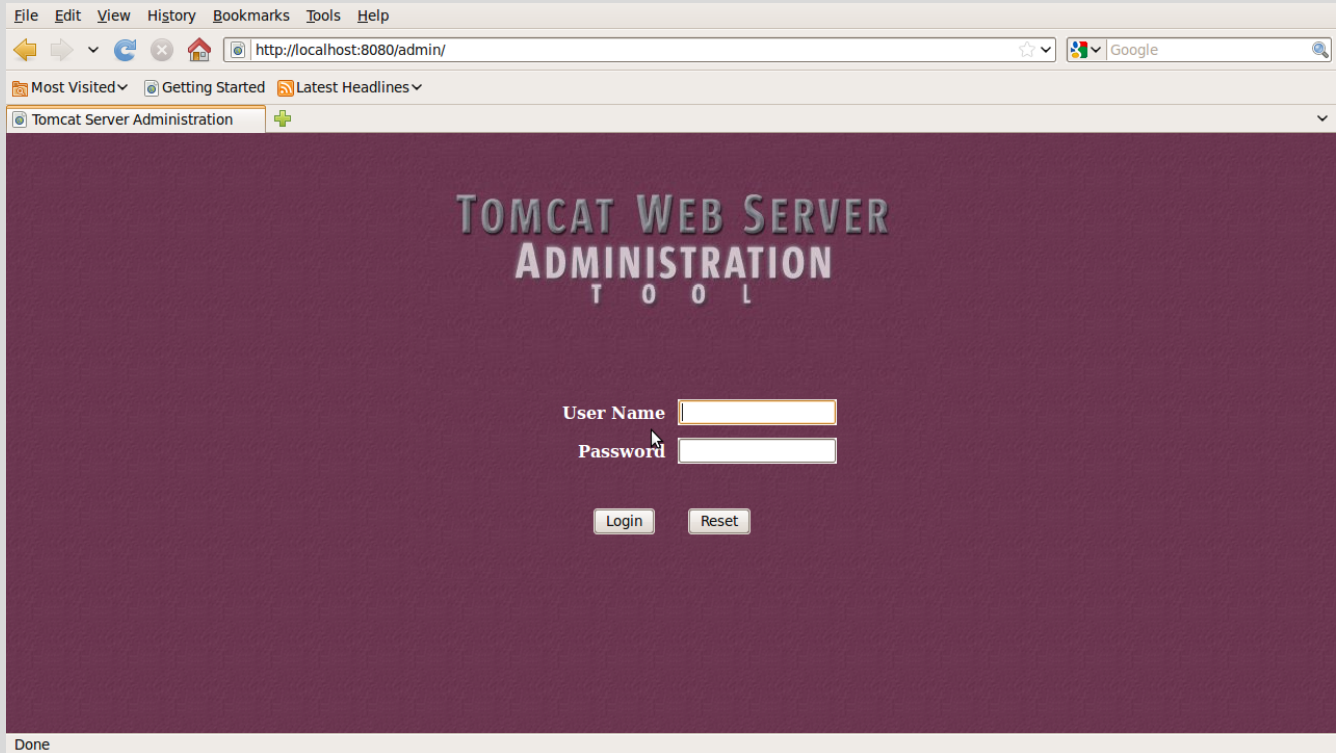
Click on the following links to run sample Web applications, or bring up administration tools that ship with the Java WSDP.

Tomcat Tools

- [Server Administration](#)
- [Web Application Manager](#)

Done

Login



The image shows a web browser window with the address bar displaying `http://localhost:8080/admin/`. The browser's menu bar includes File, Edit, View, History, Bookmarks, Tools, and Help. Below the menu bar are navigation buttons (back, forward, home, etc.) and a search bar with the Google logo. The browser's tab bar shows a single tab titled "Tomcat Server Administration". The main content area of the browser displays the Tomcat Web Server Administration login page. The page has a dark purple background with the text "TOMCAT WEB SERVER ADMINISTRATION" in a large, stylized font. Below this text, there are two input fields: "User Name" and "Password". A mouse cursor is pointing at the "User Name" input field. Below the input fields are two buttons: "Login" and "Reset". The browser's status bar at the bottom shows the word "Done".

File Edit View History Bookmarks Tools Help

http://localhost:8080/admin/ Google

Most Visited Getting Started Latest Headlines

Tomcat Server Administration

**TOMCAT WEB SERVER
ADMINISTRATION**
T O O L

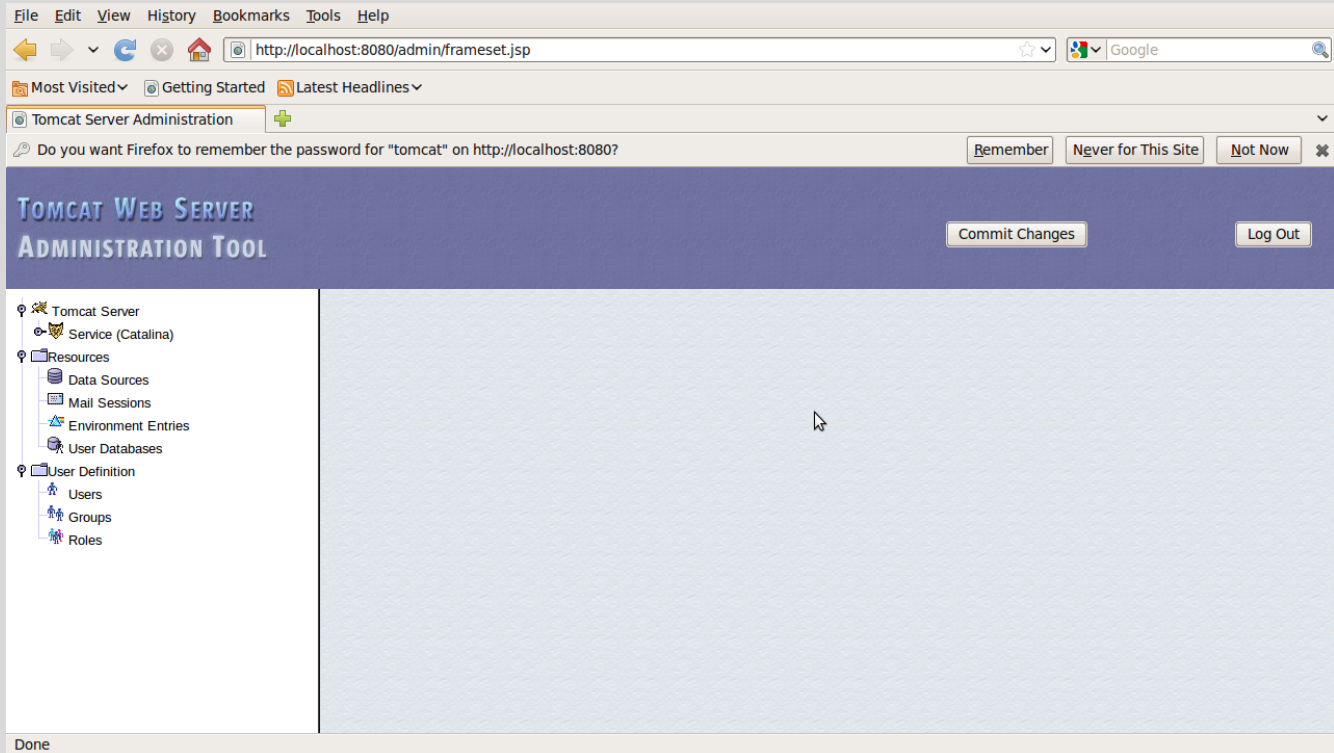
User Name

Password

Login Reset

Done

Admin Tool



Connector component

File Edit View History Bookmarks Tools Help

http://localhost:8080/admin/frameset.jsp

Most Visited Getting Started Latest Headlines

Tomcat Server Administration

TOMCAT WEB SERVER ADMINISTRATION TOOL

Commit Changes Log Out

- Tomcat Server
 - Service (Catalina)
 - Connector (8080)**
 - Host (localhost)
 - Context (/)
 - Resources
 - Data Sources
 - Mail Sessions
 - Resource Links
 - Environment Entries
 - Resources
 - Data Sources
 - Mail Sessions
 - Environment Entries
 - User Databases
 - User Definition
 - Users
 - Groups
 - Roles

Connection Linger (milliseconds) :	<input type="text" value="-1"/>
Connection Timeout (milliseconds) :	<input type="text" value="20000"/>
Connection Upload Timeout (milliseconds) :	<input type="text" value="300000"/>
Default Buffer Size:	<input type="text" value="2048"/>
Disable Upload Timeout:	<input type="text" value="True"/>
Max KeepAlive Requests:	<input type="text" value="100"/>
Max Spare Threads:	<input type="text" value="75"/>
Max Threads:	<input type="text" value="150"/>
Min Spare Threads:	<input type="text" value="25"/>
TCP No Delay :	<input type="text" value="True"/>
X Powered By :	<input type="text" value="False"/>
Ports	
Port Number:	<input type="text" value="8080"/>

Done

Connector component

- Accept count — TCP connectionqueue length
- Connection timeout
- IP address
- Port number
- Min spare threads
- Max threads
- Max spare threads

Host component

File Edit View History Bookmarks Tools Help

http://localhost:8080/admin/frameset.jsp

Most Visited Getting Started Latest Headlines

Tomcat Server Administration

TOMCAT WEB SERVER ADMINISTRATION TOOL

Commit Changes Log Out

- Tomcat Server
 - Service (Catalina)
 - Connector (8080)**
 - Host (localhost)
 - Context (/)
 - Resources
 - Data Sources
 - Mail Sessions
 - Resource Links
 - Environment Entries
 - Resources
 - Data Sources
 - Mail Sessions
 - Environment Entries
 - User Databases
 - User Definition
 - Users
 - Groups
 - Roles

Connection Linger (milliseconds) :	<input type="text" value="-1"/>
Connection Timeout (milliseconds) :	<input type="text" value="20000"/>
Connection Upload Timeout (milliseconds) :	<input type="text" value="300000"/>
Default Buffer Size:	<input type="text" value="2048"/>
Disable Upload Timeout:	<input checked="" type="checkbox" value="True"/>
Max KeepAlive Requests:	<input type="text" value="100"/>
Max Spare Threads:	<input type="text" value="75"/>
Max Threads:	<input type="text" value="150"/>
Min Spare Threads:	<input type="text" value="25"/>
TCP No Delay :	<input checked="" type="checkbox" value="True"/>
X Powered By :	<input type="checkbox" value="False"/>
Ports	
Port Number:	<input type="text" value="8080"/>

Done

Host component — Virtual hosts

- URLs with fully qualified domain name
- localhost → DNS → IP address 127.0.0.1, web server on the local machine
 - From local machine web browser requests localhost
 - From any machine web browser requests port 8080 (Service's Connector)
- New Host Items

Host → Context component

- Web applications: Collection of programs and files — provide a function to web users

Field name	Description
Name	Virtual host = FQ domain name or localhost
Application base	Directory containing web application for the virtual host
Deploy on startup	Should the web application be initialized when the server starts?

Auto deploy

Web applications added while the server is running — should they be initialized?

Host directory — Application Base

- Absolute pathname
- Relative pathname
 - Relative to Tomcat's directory
 - Advantages

Context

File Edit View History Bookmarks Tools Help

http://localhost:8080/admin/frameset.jsp

Most Visited Getting Started Latest Headlines

Tomcat Server Administration

TOMCAT WEB SERVER ADMINISTRATION TOOL

Commit Changes Log Out

- Tomcat Server
 - Service (Catalina)
 - Connector (8080)
 - Host (localhost)
 - Context (/)**
 - Resources
 - Data Sources
 - Mail Sessions
 - Resource Links
 - Environment Entries
 - Resources
 - Data Sources
 - Mail Sessions
 - Environment Entries
 - User Databases
 - User Definition
 - Users
 - Groups
 - Roles

Context Properties

Property	Value
Cookies:	True
Cross Context:	False
Debug Level:	0
Document Base:	/usr/local/tomcat50-jwsdp/webapps/ROOT
Override:	False
Path:	
Reloadable:	False
Swallow Output:	False
Use Naming:	True

Loader Properties

Property	Value
Debug Level:	0

Done

Context directory — Document Base

- Absolute pathname
- Relative pathname
 - Relative to Application Base

Logging

- Record of server activity
- *Access log*: information about every HTTP request
- *Message log*: debugging information, by web applications

Access control

The screenshot shows the Tomcat Web Server Administration Tool interface. The browser address bar displays `http://localhost:8080/admin/frameset.jsp`. The page title is "TOMCAT WEB SERVER ADMINISTRATION TOOL". The left sidebar shows a tree view of the administration tool's structure, with "Users" selected under "User Definition". The main content area displays the "Users List" page, which includes a table with columns "User Name" and "Full Name". The table lists three users: "both", "role1", and "tomcat". The "User Actions" dropdown menu is set to "----Available Actions----".

Tomcat Server Administration

TOMCAT WEB SERVER
ADMINISTRATION TOOL

Commit Changes Log Out

Tomcat Server

- Service (Catalina)
- Connector (8080)
- Host (localhost)
- Context (/)
- Resources
 - Data Sources
 - Mail Sessions
 - Resource Links
 - Environment Entries
- Resources
 - Data Sources
 - Mail Sessions
 - Environment Entries
 - User Databases
- User Definition
 - Users**
 - Groups
 - Roles

Users List User Actions ----Available Actions----

User Name	Full Name
both	
role1	
tomcat	

`http://localhost:8080/admin/users/listUsers.do?databaseName=Users%3Atype%3DUserDatabase%2Cdatabase%3DUserDatabase&forward=Users+List+Setup`

Access control

File Edit View History Bookmarks Tools Help

http://localhost:8080/admin/frameset.jsp

Most Visited Getting Started Latest Headlines

Tomcat Server Administration

TOMCAT WEB SERVER ADMINISTRATION TOOL

Commit Changes Log Out

- Tomcat Server
 - Service (Catalina)
 - Connector (8080)
 - Host (localhost)
 - Context (/)
 - Resources
 - Data Sources
 - Mail Sessions
 - Resource Links
 - Environment Entries
 - Resources
 - Data Sources
 - Mail Sessions
 - Environment Entries
 - User Databases
 - User Definition
 - Users**
 - Groups
 - Roles

Edit Existing User Properties

User Actions: ----Available Actions----

Save Reset

User Properties	
User Name:	tomcat
Password:	••••••
Full Name:	

	Group Name	Description
<input checked="" type="checkbox"/>	admin	
<input type="checkbox"/>	role1	
<input checked="" type="checkbox"/>	tomcat	

	Role Name	Description
<input checked="" type="checkbox"/>	admin	
<input type="checkbox"/>	role1	
<input checked="" type="checkbox"/>	tomcat	

Save Reset

Done