JAVA DAYS ´98

The Java Servlet API

James Duncan Davidson
Servlet API Project Lead
Sun Microsystems, Inc.

Sun microsystems

Guten Tag: Our Agenda

- The sales pitch
- Servlet basics
- Advanced topics
- Conclusion

The Sales Pitch: What are Servlets?

- Java objects which extend a HTTP server
- A Java Programming Language replacement for CGI, NSAPI, ISAPI, etc.
- A Java Platform Standard Extension
- Available and running on all major web servers

The Sales Pitch: Why Servlets?

- Much less overhead than the CGI model
- Safer than NSAPI & ISAPI and other native code web extension mechanisms
- HTTP is the most ubiquitous protocol on the web today
- Consistent and standard across implementations
- Write Once, Serve Anywhere!

Sun microsystems

The Sales Pitch: Servlets are Lightweight

- Can run in the same process as host HTTP Server
- Are fully threadable
- Can be deployed into distributed server environments where RAS (Reliability, Availability, Scalability) are critical

The Sales Pitch: Easy to Develop

- It’s Java!
- Extensive availability of Java Platform libraries such as JDBC, EJB, JMS, JavaMail, JavaIDL, RMI, etc.
- Extensive third party libraries
- Can develop with the smallest of servlet engines on a laptop, deploy on the most mission critical servers on enterprise class hardware

The Sales Pitch: Easy to Maintain

- Most servers allow reloading of a servlet by administrative action
- Servlets can be remotely loaded allowing several servers to share code

The Sales Pitch: Possible Servlet Applications

- Site wide document management
- Electronic Commerce
- HR Applications
- Conference and chat applications
- Anything else you can put on the Web!

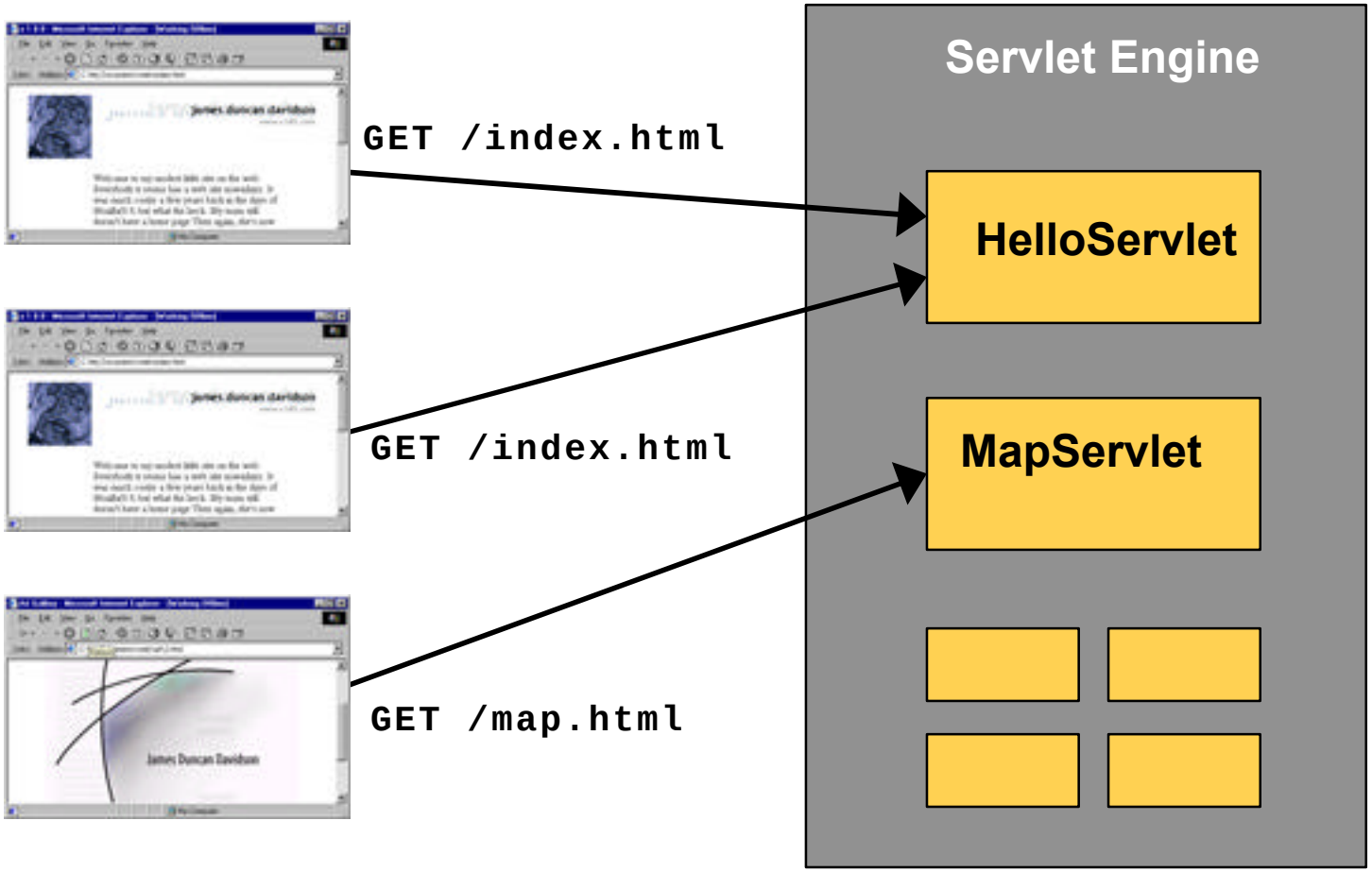
Servlet Basics: Code for a Simple Servlet


```
public class ExampleServlet extends HttpServlet {
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws ServletException, IOException
    {
        out.setContentType("text/html");
        PrintWriter out = request.getWriter();
        out.println("Guten Tag!<BR>");
        Date rightNow = new Date();
        out.println("The time is: " + rightNow);
    }
}
```

Servlet Basics

- Servlets have a well defined lifecycle
- Servlets are managed objects that are loaded on demand and can be unloaded by the server at any time
- Servlets can be mapped to any part of the URL namespace that a servlet engine has control over
- Multiple threads of execution can run through a servlet unless otherwise specified

Servlet Basics: An Illustration



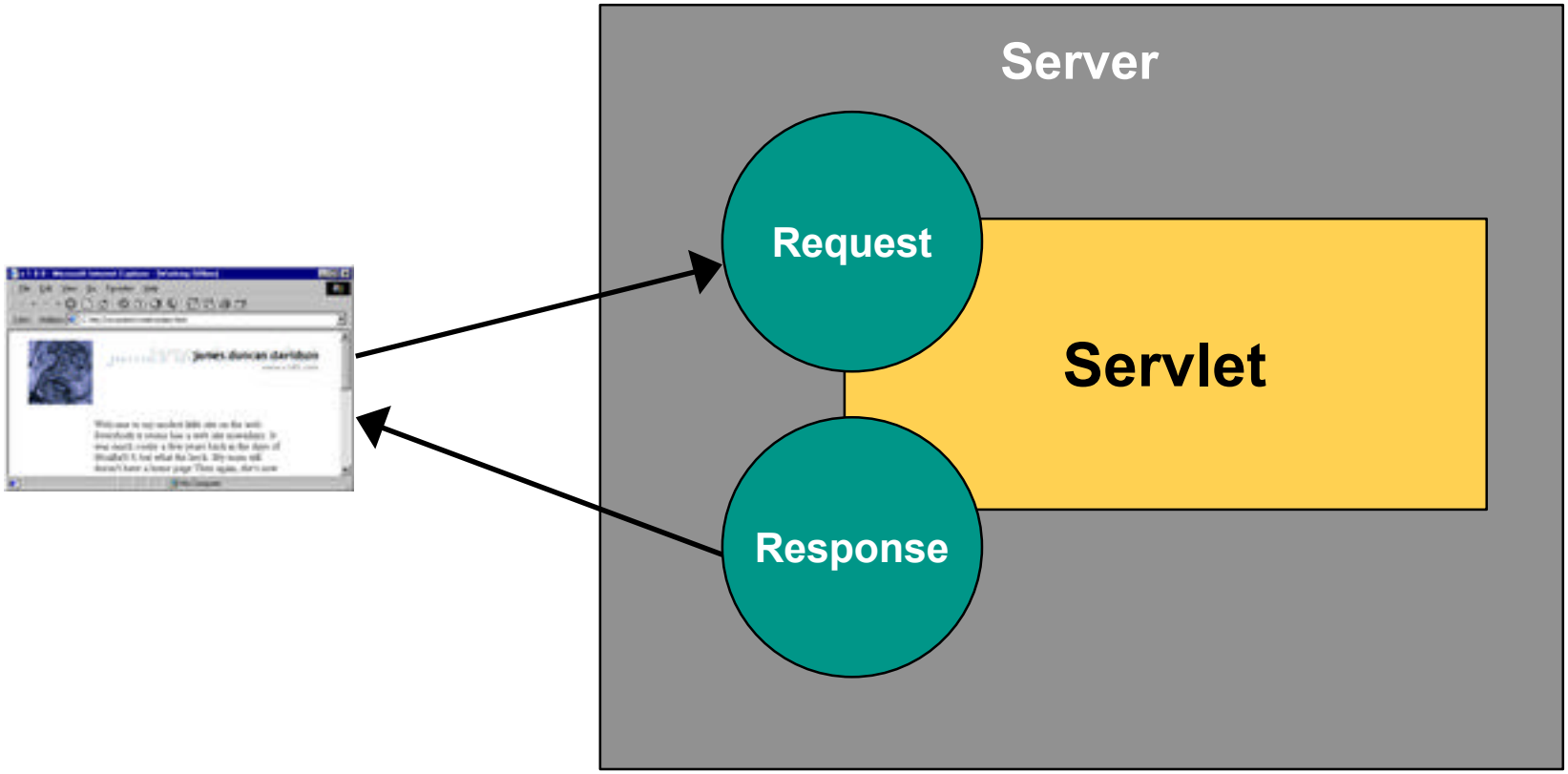
Servlet Basics: The Servlet Lifecycle

- Servet is instantiated by the server
- It is initialized via the init method
- The service method is called 0-multiple times
- Servlet is given a chance to clean up when being unloaded via a destroy method

Servlet Basics: Anatomy of a Request

- A client makes a request on a server
- The request is resolved to a servlet by the server
- The servlet is called with via the service method with a Request object and Response object
- The servlet provides a response to the request

Servlet Basics: Illustration of a Client Request



Servlet Basics: The Request Object

- Encapsulates all information from the client
- Access to request headers
- Access to an InputStream or Reader containing data from the client
- Access to CGI Like information
- Access to form data and query parameters
- Access to server specific parameters such as SSL information

Servlet Basics: Frequently Used Request Methods


```
javax.servlet.ServletRequest {
    Enumeration getParameterNames();
    String getParameter(String parameterName);
    String getRemoteAddr();
}

javax.servlet.http.HttpServletRequest {
    String getRequestURI();
    Enumeration getHeaderNames();
    String getHeader(String headerName);
    HttpSession getSessiOn();
    Cookie[] getCookies();
}
```

Servlet Basics: The Response Object

- Encapsulates all communication back to the client
- Access to response headers
- Access to an OutputStream to write data to the client
- Access to setting cookies
- Convenience method for sending redirects, error pages, etc.

Servlet Basics: Frequently Used Response Methods


```
javax.servlet.ServletResponse {
    PrintWriter getWriter();
    ServletOutputStream getOutputStream();
    void setContentType(String type);
    void setContentLength(int length);
}

javax.servlet.http.HttpServletResponse {
    void addCookie(Cookie cookie);
    void setStatus(int statusCode);
    void sendError(int statusCode);
    void sendRedirect(String url);
}
```

Servlet Basics: Session Tracking

- Sessions are a series of requests made by a specific client during a specific time period
- There have been many approaches to session tracking
- Unified in the Servlet API - Sessions are completely managed by the server
- Can use cookies, URL rewriting, or some other future mechanism

Servlet Basics: Session Tracking Code Sample


```
Public void doGet(HttpServletRequest request,
                  HttpServletResponse response)
    throws ServletException, IOException
{
    HttpSession session = request.getSession();
    Cart cart = (Cart)session.getValue("cart");

    // ...
    // do logic to get the inventory number and
    // quantity of goods that user wants to add to cart


    cart.addItem(inventoryNumber, quanity);
}
```

Advanced Topics: Not just for HTML

- Many programmers use servlets to assemble HTML pages, but…
- Servlets can generate images using AWT, Java 2D
- Servlets can generate custom data formats for use by applets and other web-aware software components
- Servlets can send serialized objects to and read serialized objects from applets and other Java Platform applications

Advanced Topics: Image Generation Code


```
public void doGet(HttpServletRequest request,
                  HttpServletResponse response)
    throws ServletException, IOException
{
    // do logic to build image
    Image image = buildImage(request);
    response.setContentType("image/gif");
    ServletOutputStream out = response.getOutputStream();

    // use acme.com's Gif Encoder
    GifEncoder encoder = new GifEncoder(image, out);
    encoder.encode();
}
```

Advanced Topics: Client / Servlet Communication

- Two primary methods of applet/servlet communication:
  - Text based using java.util.Properties
  - Binary based using object serialization
- The benefit of using Properties is that non Java Platform based clients can interoperate
- The benefit of using serialization is that complex data structures can be transmitted easily

Advanced Topics: Using Properties


```
// servlet side
{
    Properties props = new Properties();
    props.put("result", "true");
    ServletOutputStream out = response.getOutputStream();
    props.save(out, "Servlet Property Stream");
}
// client side
{
    URL url = new URL("http://myserver/resultservlet");
    InputStream in = url.openStream();
    Properties props = new Properties();
    props.load(in);
    String result = props.getProperty("result");
}
```

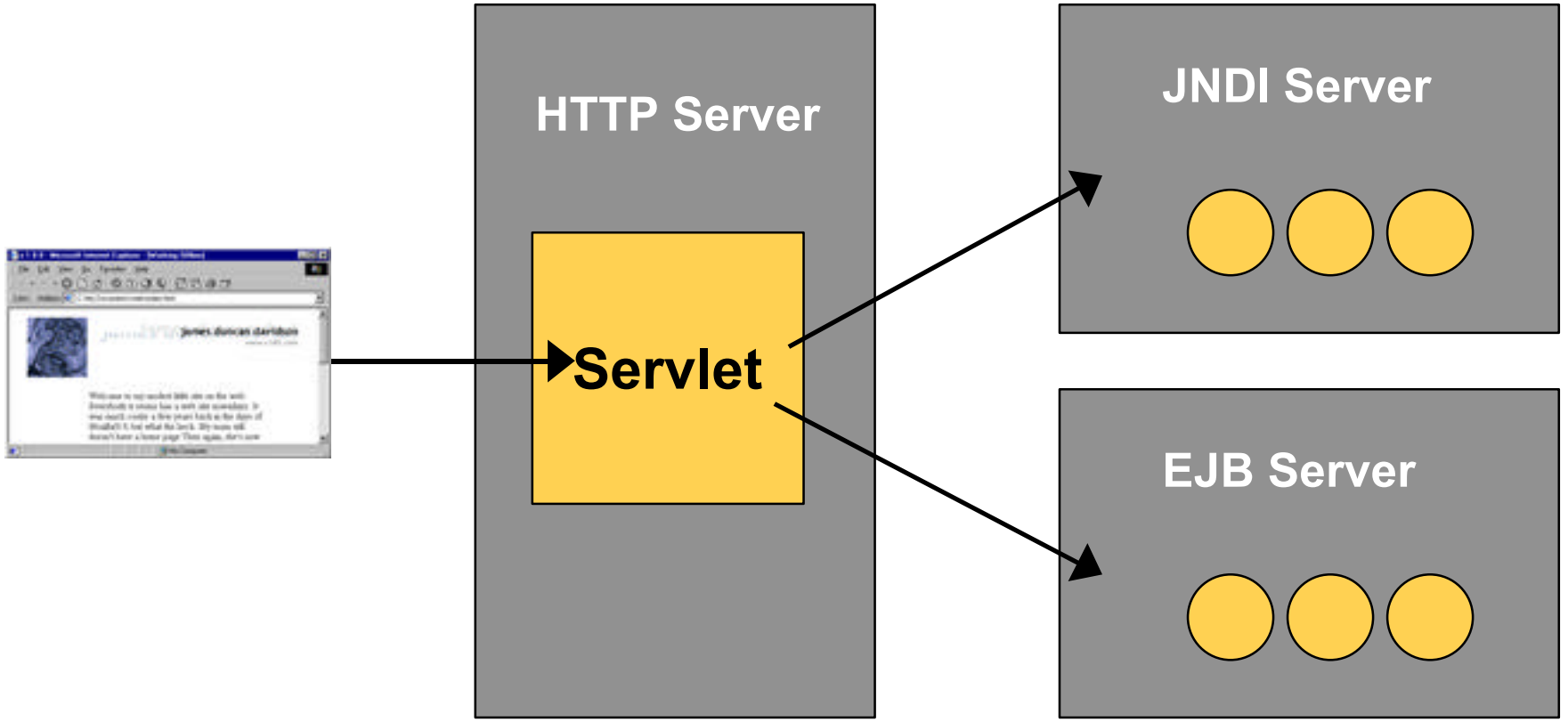
Advanced Topics: Using Serialization


```
// servlet side
{
    Result result = getResult(); // generate object
    ServletOutputStream out = response.getOutputStream();
    ObjectOutputStream oOut = new ObjectOutputStream(out);
    oOut.writeObject(result);
}
// client side
{
    URL url = new URL("http://myserver/resultservlet");
    InputStream in = url.openStream();
    ObjectInputStream oIn = new ObjectInputStream(in);
    Result result = (Result)oIn.readObject();
}
```

Advanced Topics: Enterprise Role of Servlets

- EJBs can only be accessed in environments where RMI and IIOP work
- Unfortunately RMI and IIOP don’t go over firewalls well
- Every client has a browser, not every client has the ability to run your application code
- Servlets serve as the HTTP speaking middle tier between any kind of thin client and large enterprise services made available via EJB

Servlet Basics: Illustration of Servlets in an Enterprise System



Conclusion: Current Servlet API Status

- Servlet API 2.1 Specification released November 4th
- Servlet Development Kit early access later this month
- Servlet Development Kit FCS by January 1999
- Third party support for 2.1 arriving soon!

Conclusion: Servers Supporting Servlets

- Java Web Server
- Sun Web Server
- Netscape Application Server
- The Apache Project
- IBM WebSphere
- Zues Web Server
- BEA Weblogic Tengah
- KonaSoft Enterprise Server
- ATG Dynamo Application Server
- Novacode NetForge
- W3C Jigsaw
- More!

Sun microsystems

Conclusion: Servlet Engines for IIS & Netscape Enterprise

- NewAtlanta ServletExec
- LiveSoftware Jrun
- WAI Cool Runner

Conclusion: Servlet Partners

- Sun Microsystems
- IBM
- Netscape
- BEA Weblogic
- Art Technology Group
- The Apache Group
- Live Software
- New Atlanta Communications
- Other Individual Industry Leaders

Sun microsystems

Conclusion: Servlet Futures

- Deployment Descriptors for deploying servlets, their related class files, and content into any server
- Servlet Engine Compatibility tests
- You can help shape servlets by sending your suggestions to servlet-feedback@eng.sun.com

Conclusion: Questions & Answers

- http://java.sun.com/products/servlet
- servlet-interest mailing list
- servlet-feedback@eng.sun.com
- james.davidson@sun.com