

# Capítulo 3



## Java Servlet

# Servlet



- Que es un servlet?
  - “Server-side entity”
- Primero entender las responsabilidades de un servidor
  - Manejar las conexiones de red
    - Programación a nivel de socket
    - Extraer la información del “request message”
    - Implementar protocolos cliente-servidor (HTTP)
  - Crear las respuestas que enviará de vuelta
    - Puede crear respuestas dinámicamente
    - Obtener datos desde una base de datos
    - Aplicar reglas de lógica de negocio
    - Representar los datos de salida en diferentes formatos

# Servlet



- Diseño de un servidor web
  - Se divide en dos partes ejecutables
    - Una maneja todas las funcionalidades de conexiones
    - La otra maneja todas la lógica de la aplicación
  - Las dos partes se comunican entre ellas con una interfaz estándar
  - Una parte puede ser el servidor web principal
    - Módulo de conexiones de red
    - Administración de clientes
  - Otra parte puede ser scripts CGI
    - Módulos de procesamiento de datos
    - Entrega respuestas



- Extensiones en un servidor
  - Cada requerimiento invoca la creación y destrucción de un nuevo proceso que ejecuta un script CGI
    - Muy ineficiente, especialmente si el CGI realiza rutinas de inicialización
  - Un mejor camino es soportar módulos que se cargan en memoria una vez
    - Se inicializan una vez, cuando se inicia el servidor
    - Cada requerimiento es atendido por el módulo que ya está en memoria disponible para trabajar
    - Muchos servidores están soportando estos módulos desde hace un largo tiempo
    - Esto ha hecho que los CGI queden obsoletos
    - Estos módulos de ejecución son conocidos como “server extensions”

# Servlet



- Servlet extensions
  - En Java se escriben utilizando la Servlet API, los “server extension modules” se llaman Servlet
  - Netscape Server provee NSAPI:
    - Netscape Server Application Programming Interface
  - Microsoft's Internet Information Server provee ISAPI
    - Internet Server Application Programming Interface
- Servlet container
  - Módulo especializado del servidor web
    - Se dedica a cargar y ejecutar los servlets

# Servlet

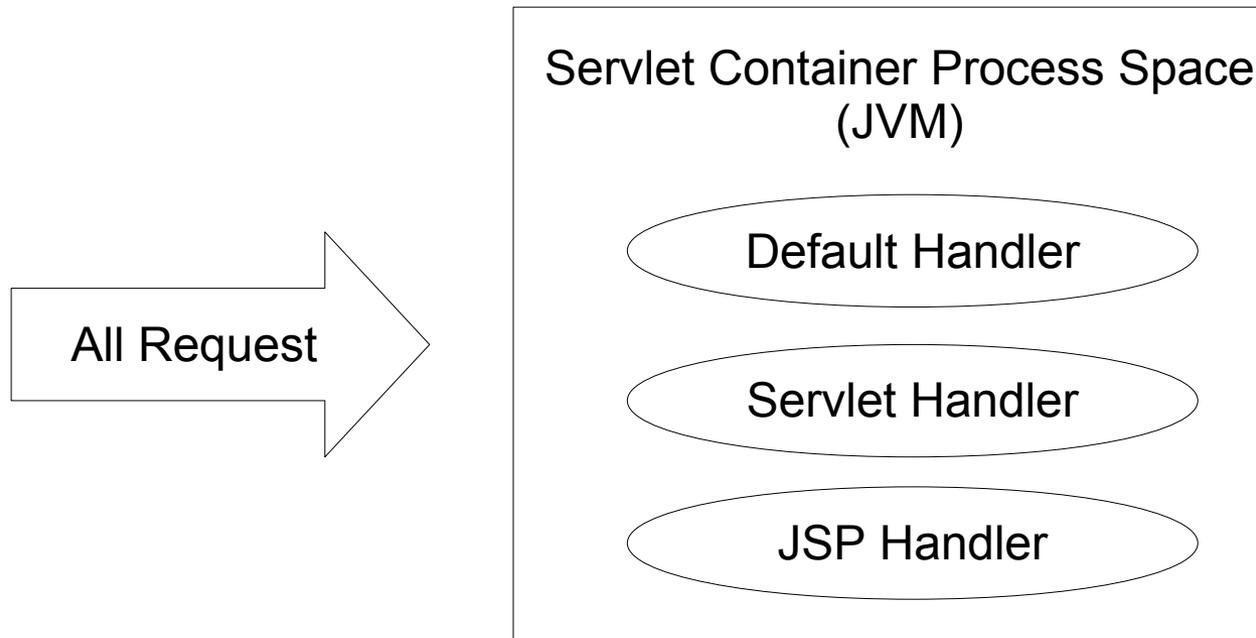


- Servlet container
  - El browser envía un requerimiento al servidor web
    - Si se está pidiendo un archivo HTML, el servidor lo maneja directamente
    - Si se pide la ejecución de un servlet, el servidor delega el trabajo en el “servlet container”
    - El “servlet container” ejecuta el servlet solicitado
    - El servlet se puede conectar con bases de datos o sistemas de archivos para generar una respuesta dinámica
- Un “contenedor de servlet” es parte de un servidor web
  - Pero se ejecuta en un proceso separado

# Servlet



- Servlet containers, se pueden clasificar en tres tipos
  - Standalone: servidores web basados en Java
    - Dos módulos: servidor web y “servlet container”



# Servlet



- Servlet Container

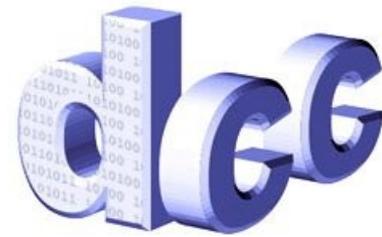
- Standalone:

- Tomcat es un ejemplo, cuando ejecuta todas las partes por su cuenta
    - Se puede ejecutar tomcat como cualquier programa Java dentro de la JVM
    - Contiene “handlers” para contenido estático (HTML) y “handlers” para servlets y JSP

- Tomcat

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

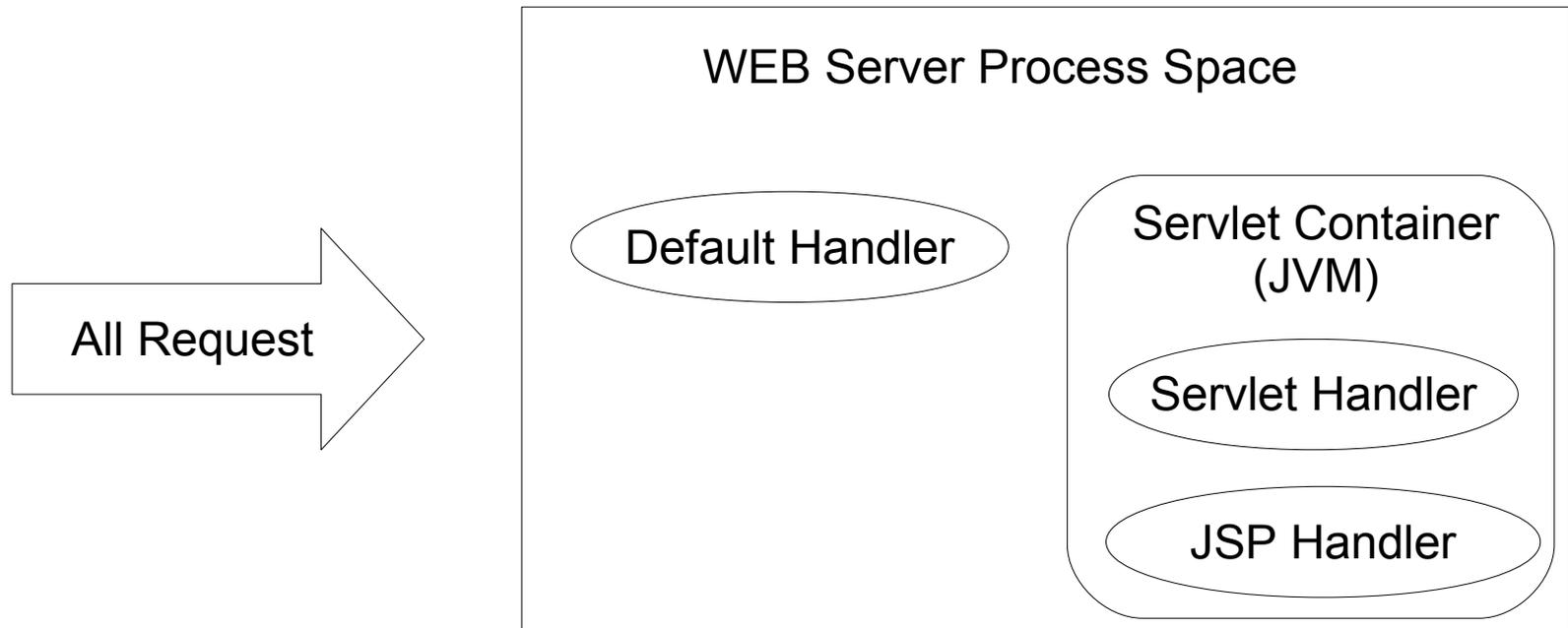
# Servlet



- Servlet container

- In-Process:

- El servidor web y el contenedor de servlet son programas diferentes



# Servlet



- Servlet container: In-process
  - Un ejemplo puede ser tomcat ejecutándose dentro del servidor web Apache
    - Apache carga la JVM que ejecuta Tomcat
    - El servidor web controla el contenido estático
    - Tomcat maneja los servlets y JSP

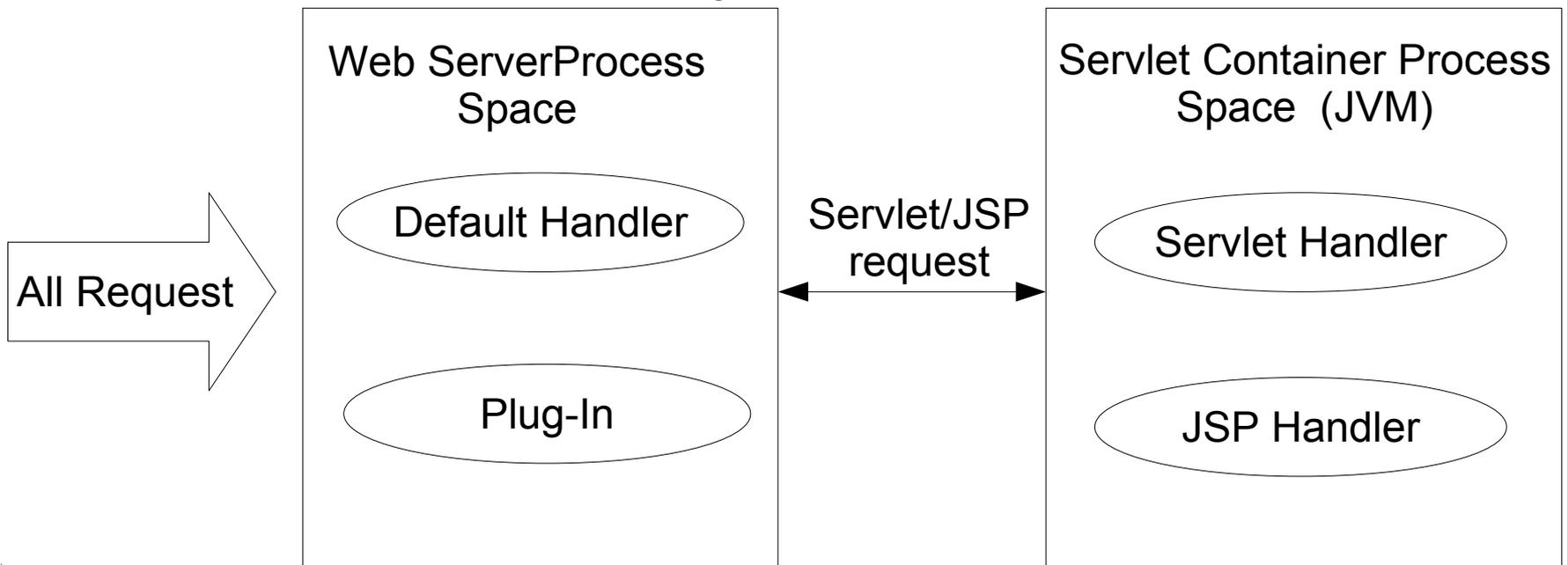
# Servlet



- Servlet Container:

- Out-of-Process:

- El servidor web y el contenedor de servlet son programas independientes
    - Ambos se ejecutan en procesos separados, la comunicación se realiza con un plug-in



# Servlet



- Servlet Container
  - Out-of-process
    - Un ejemplo es Tomcat ejecutándose como un proceso separado
    - Debería estar configurado para recibir requerimientos desde un servidor web Apache
    - Apache carga el plug-in mod\_jdk para comunicarse con Tomcat
- Cada uno de los tipos de “servlet container” tiene
  - Ventajas, limitaciones y aplicabilidad
- Algunos “servlets containers” son mucho más
  - WebLogic (BEA), WebSphere (IBM)

# Servlet



- Relación entre “servlet container” y “servlet API”
  - La especificación de Sun Microsystems provee un framework de comunicación entre ellos
    - Estándar e independiente de la plataforma
  - Este framework está hecho por un conjunto de clases e interfaces Java: servlet API
  - Servlet API está dividida en dos partes
    - javax.servlet
      - Interfaz central de la API
      - Los servlets la implementan de forma directa o indirecta
    - javax.servlet.http
      - Provee funcionalidades básicas para HTTP servlets
      - Las interfaces y clases extienden desde javax.servlet para implementar el protocolo HTTP

# Servlet



- **Ventajas de servlet API**
  - **Flexibilidad**
    - Para agregar una nueva funcionalidad al servidor, se implementa el servlet específico y se agrega al servidor
    - No es necesario modificar el servidor
  - **Separación de responsabilidades**
    - El servidor principal solo se preocupa de conexiones y comunicaciones
    - Los servlets se preocupan de interpretar los requerimientos y generar las respuestas
  - **Es Java**
    - Se pueden usar las características de orientación a objetos
  - **Portabilidad**
    - Se puede desarrollar en un contenedor y llevarlo a otro
    - Menos en las soluciones propietarias



- Desventajas de servlet API
  - Se debe ajustar al conjunto de reglas definidas
    - Es necesario seguir las convenciones para poder implementar servlets estándares
  - Los contenedores de servlet no proveen soporte para otros protocolos
    - En teoría usando la API se podrían construir servlets para casi cualquier protocolo
    - Los contenedores de servlet no proveen soporte para todos

# Servlet



- Que es un servlet?
  - Conceptualmente es una pieza de código que puede:
    - Ser agregada a un servidor para aumentar sus funcionalidades
    - Ser usada para generar respuestas dinámicamente
  - Para un contenedor de servlet, un servlet es:
    - Una clase Java como cualquier otra
    - Una clase que implementa la interfaz `javax.servlet.Servlet`
  - Para un desarrollador web, un HTTP servlet es una clase que:
    - Extiende `javax.servlet.http.HttpServlet`
    - Vive en un contenedor de servlet (como Tomcat)
    - Sirve requerimientos HTTP