Servidores de Aplicaciones Arquitectura y Planificación. Análisis de Mercado.

Jaime Cid

Arquitecto de Soluciones WEB

http://blogs.sun.com/jaimecid

Índice

- ESCALABILIDAD. Diseñando arquitecturas masivamente escalables.
- ALTA DISPONIBILIDAD. Diseñando una arquitectura 24x7.
- OPERACION CONTINUA. Estrategias de despliegue de Aplicaciones.
- GESTIONABILIDAD. Integración con gestión de Red. Gestion JMX
- ESTANDARES. J2EE 1.4. Roadmap J2EE 5.0. Novedades.

Índice

- IDENTIDAD y SEGURIDAD. Integración del servidor de Aplicaciones con sistemas LDAP y SSO
- MERCADO Servidores J2EE. Servidores OpenSource vs Servidores comerciales

Contexto

- En la gestión de los proyectos WEB no sólo es importante la arquitectura de desarrollo del proyecto, sino que la arquitectura de sistemas y de operación de la plataforma es fundamental a la hora de proporcionar la calidad de servicio requerida por los usuarios
- Una aplicación funcionalmente perfecta no servira a su proposito si tiene un tiempo de respuesta muy lento o tiene una disponibilidad baja.

Objetivos

- Entender las posibilidades de arquitectura de operación de los servidores de Aplicaciones J2EE
- Planificar adecuadamente el HW para conseguir la máximas prestaciones para un determinado presupuesto económico
- Establecer criterios de decisión para elegir entre productos OpenSource, versiones básicas o versiones Enterprise según las necesidades de cada proyecto

Servidores de Aplicaciones J2EE

Containers

- Concurrency
- Consistency
- Security
- Availability
- Scalability
- Administration
- Integration
- Distribution

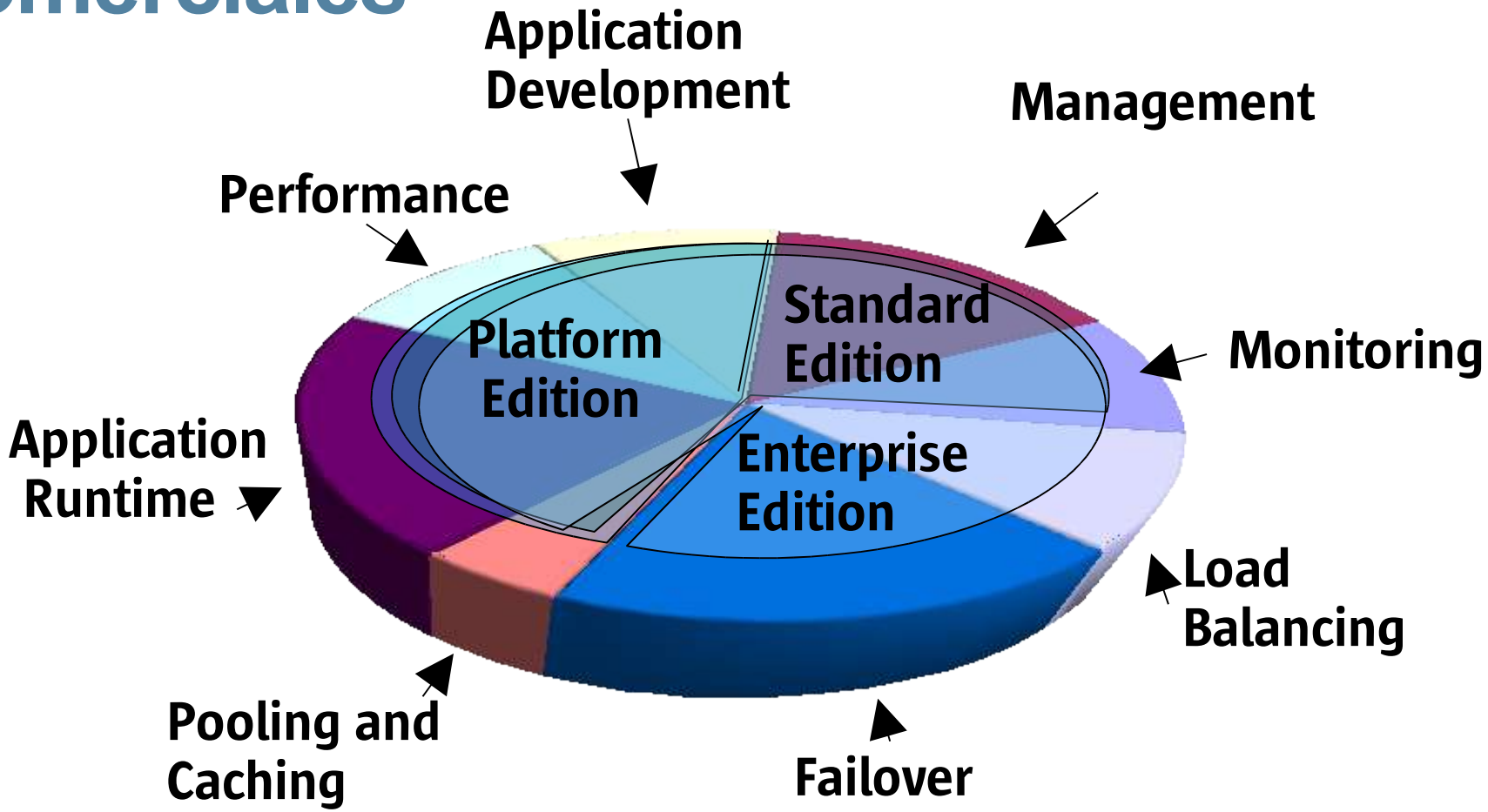
Components

- Presentation
- Business Logic
- Data access (optional)

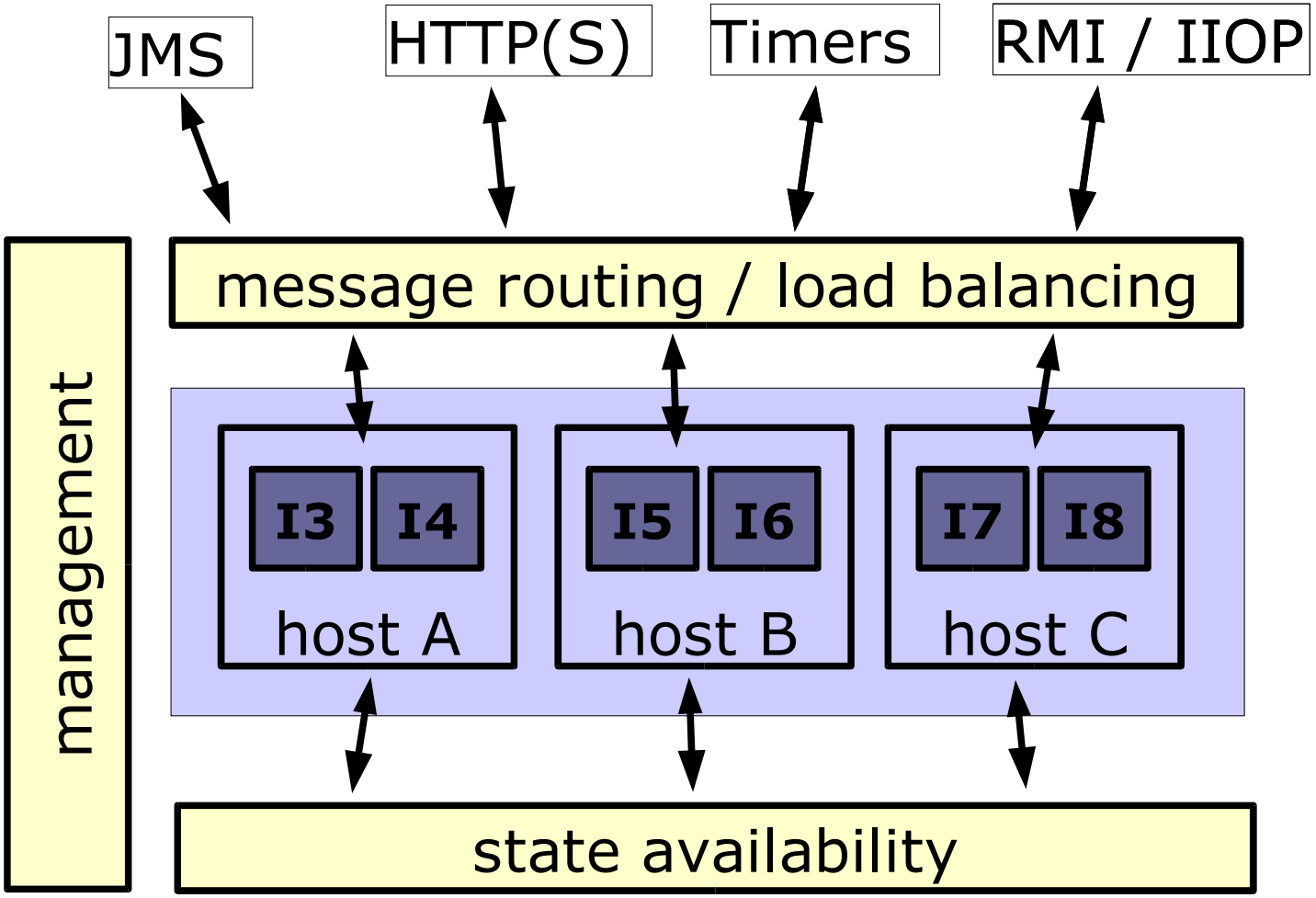
Producto Comercial

Desarrollo aplicación

Servidores de Aplicaciones Comerciales



Funcionalidades extendidas de los Servidores de Aplicaciones Comerciales



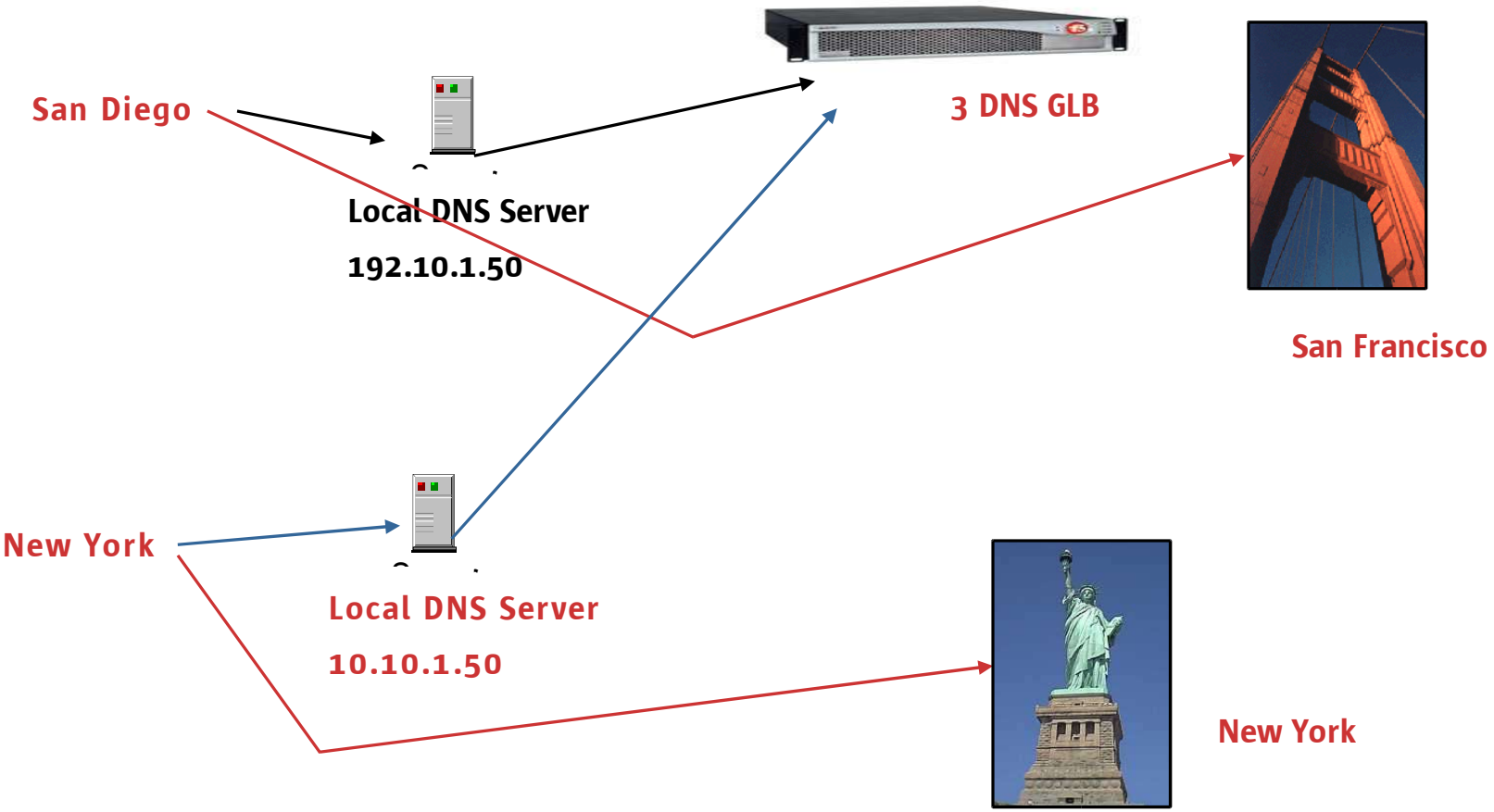
Escalabilidad

¿Qué es escalabilidad?

- *Cómo se comporta una arquitectura cuando únicamente aumenta la carga del sistema y los demás parámetros se mantienen constantes*
- *Idealmente la escalabilidad debe ser lineal, es decir si la arquitectura soporta N usuarios, y se doblan los recursos de la arquitectura, entonces soporta 2xN usuarios.*
- *Escalabilidad vertical: Aumentando el numero de CPUs, y memoria de los mismos servidores*
- *Escalabilidad horizontal: Aumentando el número de servidores de las mismas características*

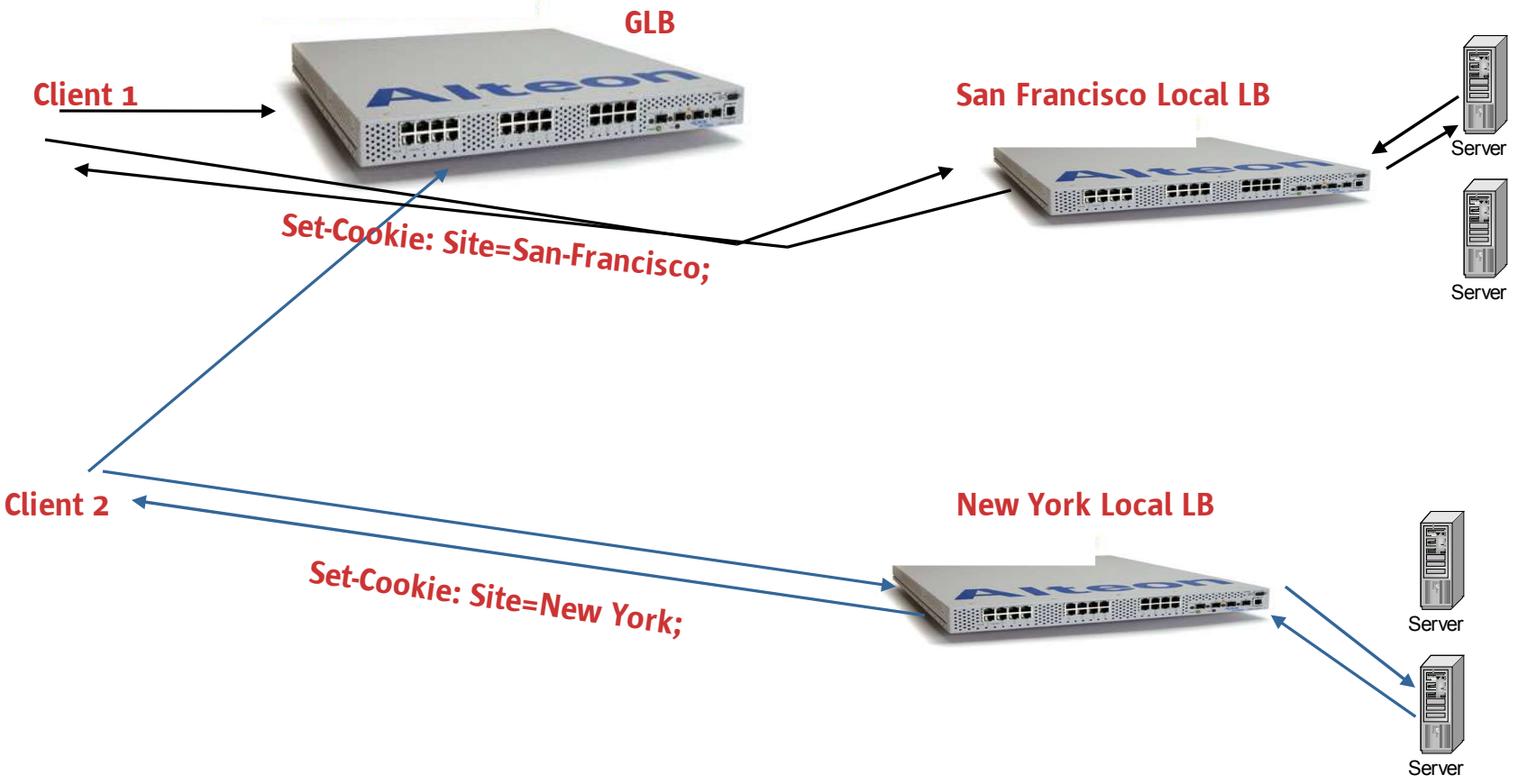
Balanceo de carga

Balanceo de carga basado en DNS

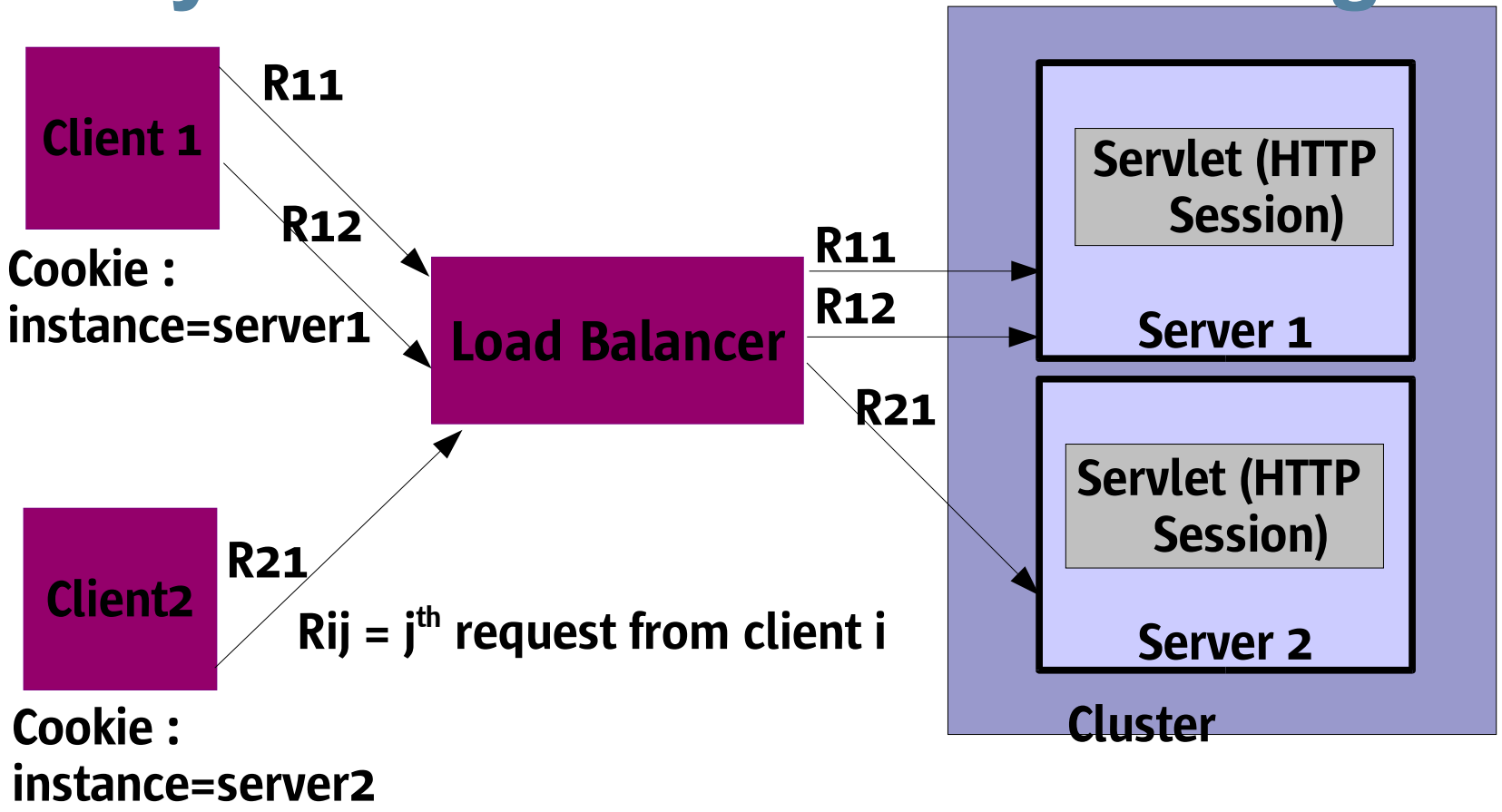


Balanceo de Carga

Balanceo de carga basado en Cookies



Sticky round-robin load balancing



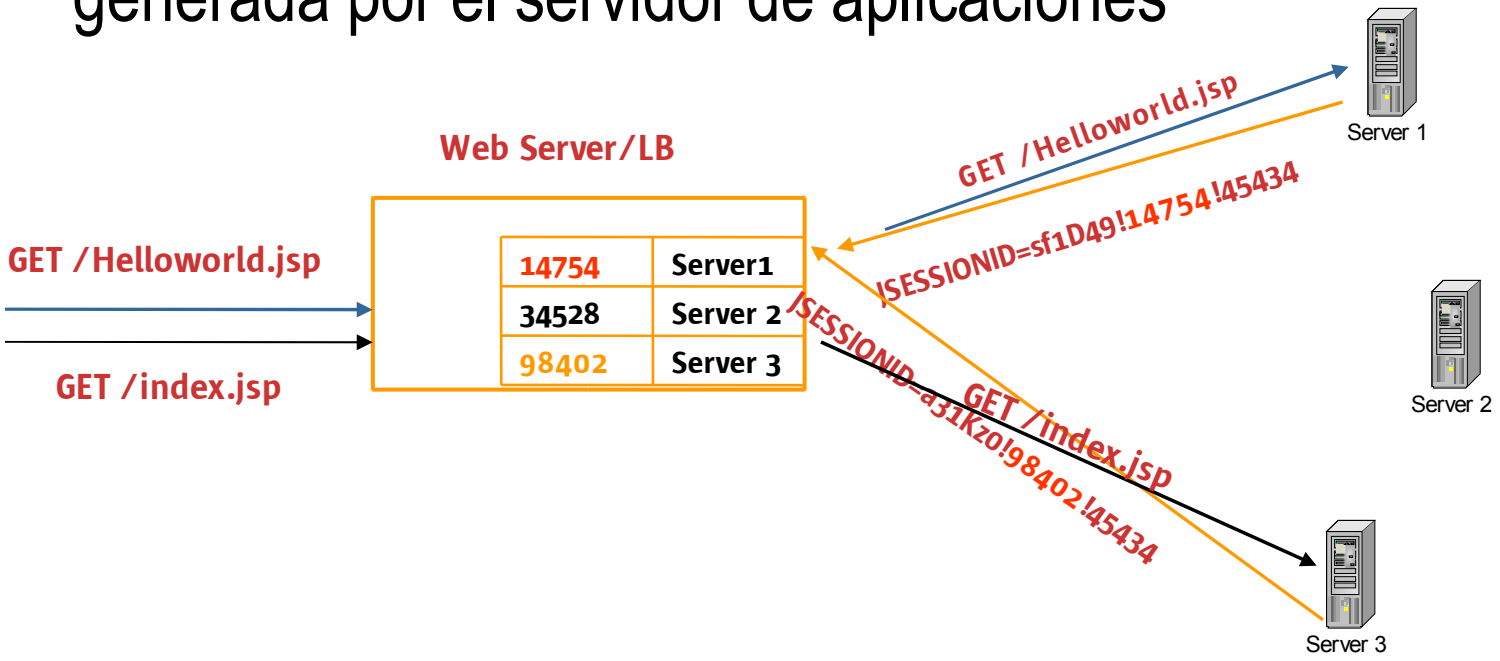
- Permite que dirigir cada usuario a un mismo servidor durante el ciclo de vida de la sesión en la información de sesión
- Implementación mediante cookies de sesión de navegador

Diseño de los plugins de servidor Web

Persistencia del servidor basada en Cookies

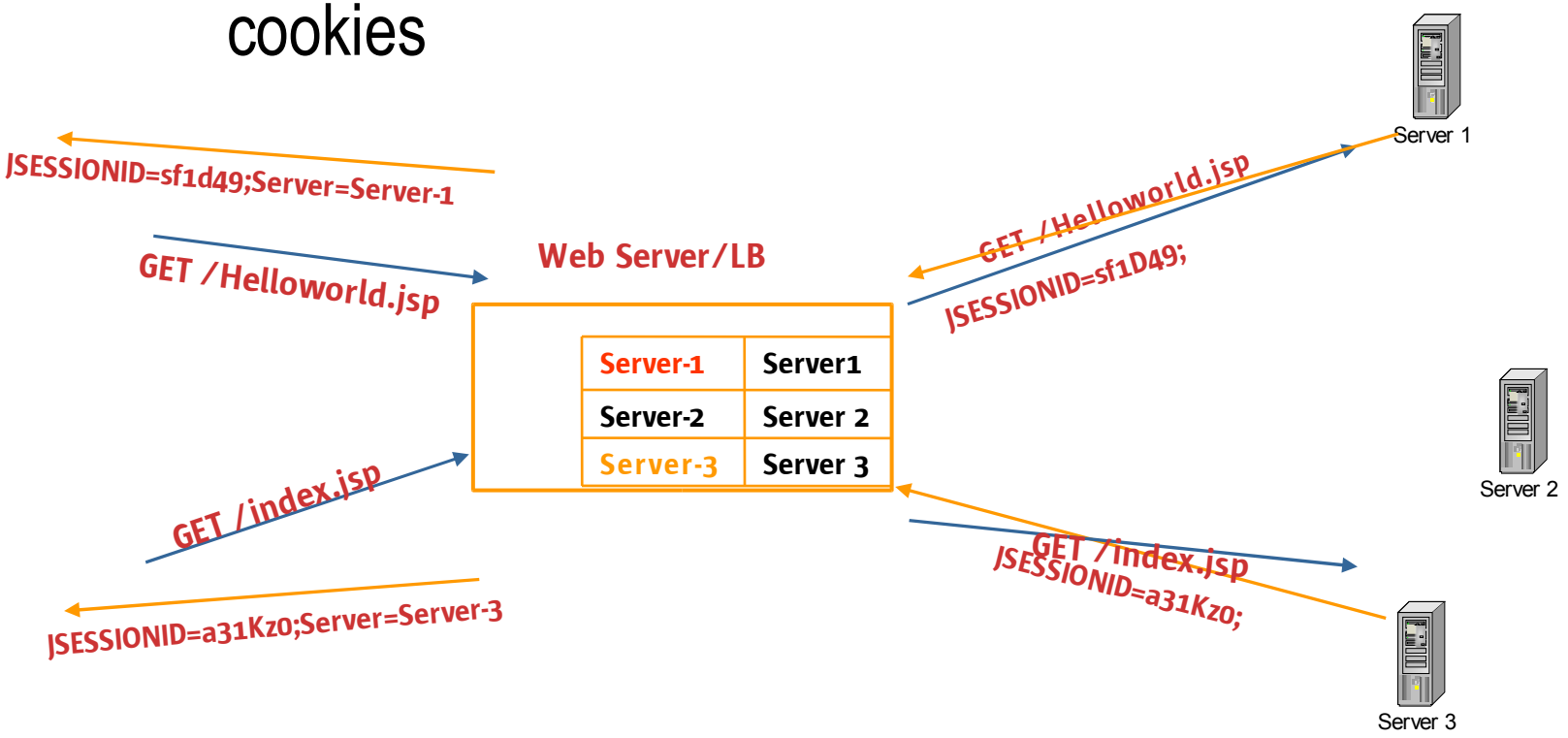
Cookie Pasiva:

- Mantenimiento de la persistencia basado en la cookie generada por el servidor de aplicaciones



Diseño de los plugins de servidor Web

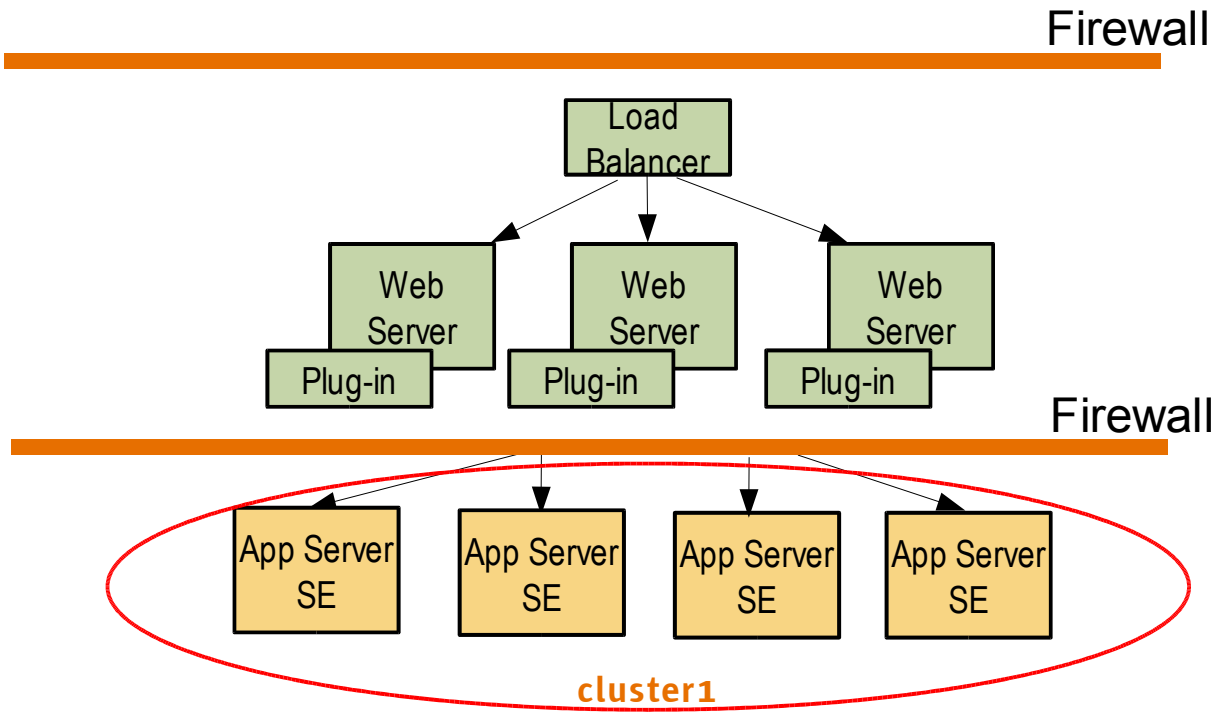
- Persistencia del servidor basada en Cookies
- Cookie Activa
  - El servidor Web o balanceador genera sus propias cookies



Diseño de los plugins de servidor Web

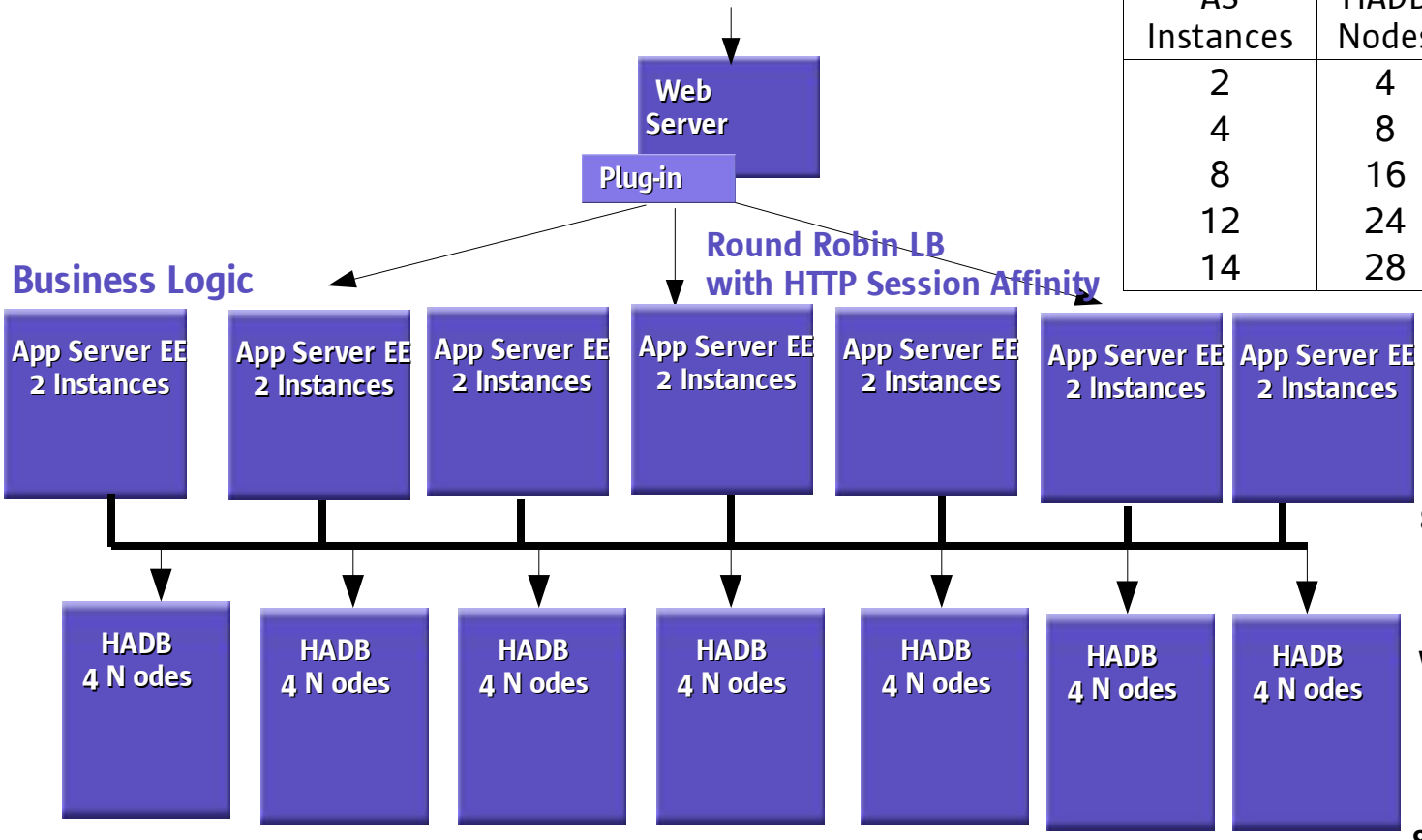
- Protocolos de gestión de servidores de aplicaciones en los balanceadores
  - Registro Estático
  - Registro Dinámico
  - Protocolo IP Multicast
  - Notificaciones en arranque / parada
    - F5 proporciona un API SOAP para conseguirlo

Arquitectura con Balanceador HW y Servidores WEB



Escalabilidad en una topología de 112 CPUs con alta disponibilidad a nivel de sesión

AS Instances	HADB Nodes	Scalability	CPU's in Cluster
2	4	100%	16
4	8	95%	32
8	16	87%	64
12	24	75%	96
14	28	70%	112



V880 (1G RAM per CPU, 1.2GHz Ultra Sparc 3)

14 App. Server Instances

8 CPU's Per Server x 7 =56CPU's

V880 (1G RAM per CPU, 1.2GHz Ultra Sparc 3)

28 HADB Nodes

8 CPU's Per Server x 7 =56CPU's

Session Replication: HTTP Session and Stateful Session Beans

Factores de Escalabilidad

¿Que influye en la escalabilidad?

- El diseño de la aplicación
- Requisitos de Alta disponibilidad y nodos del Cluster
- Coste de la comunicación entre servidores y nodes
- Mecanismos de comunicación con el backend donde se almacenan los datos (Base de Datos, Host, Sistemas ERP, Sistemas CRM, etc)

Factores de Escalabilidad

Requisitos de Alta disponibilidad y nodos del Cluster

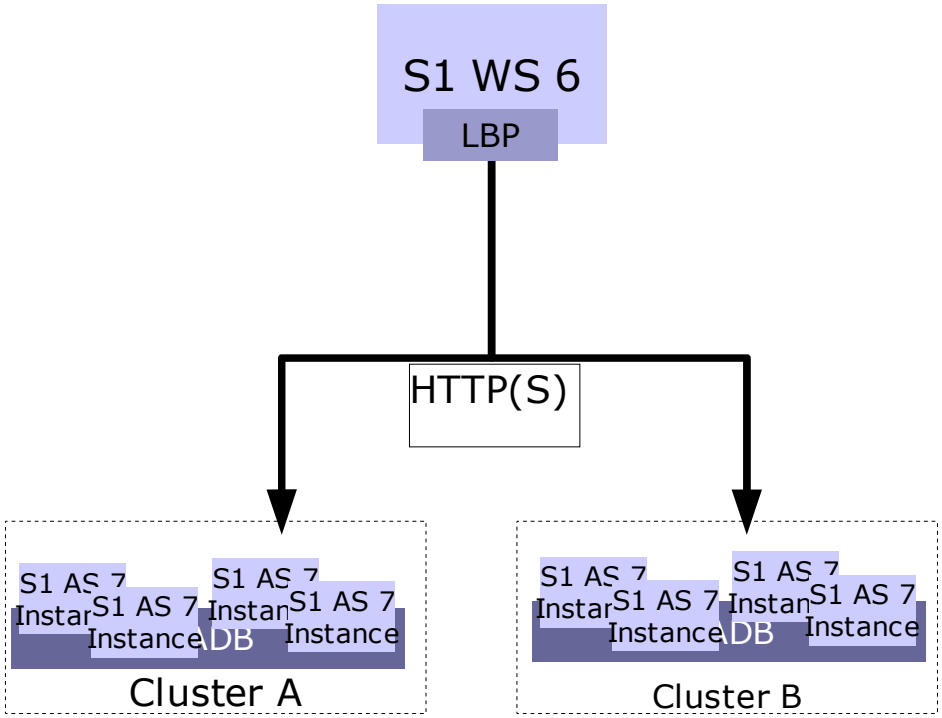
- Número de nodos de un cluster
  - > Cuanto más nodos, mayor coste de comunicación
  - > La transferencia de datos puede inundar el sistema
- La eficiencia de los balanceadores
  - > Detección de caidas de servidores de aplicaciones
- Los costes de replicación de datos se incrementan con el número de nodos
  - > Se puede mitigar con estrategias de replicación
- Nivel de tolerancia a fallos
  - > En caso de caida de un nodo, ¿que puedo perder?

Factores de Escalabilidad

Consejos sobre nodos del Cluster

- Tamaño del cluster
  - A menor número de nodos la escalabilidad es mayor
- Comunicación entre los nodos del cluster
  - “Sesión replicada en todos los nodos”—funciona bien para un número de nodos pequeño (2,3,4)
  - “Principal/ Secundario”—cada nodo tiene una réplica con la sesión almacenada de manera pasiva
  - “Nodos totalmente independientes que comunican con un almacén de sesiones común” - Útil para un elevado número de nodos
- Considerar una estrategia multi-cluster

Multi-Cluster



Factores de Escalabilidad

Consejos de Alta Disponibilidad

- Determinar el nivel requerido de tolerancia a fallos
  - > Cuanto menos datos tenga que replicar mejor
  - > Cuanto mayor sea la frecuencia de replicación mejor
- Planificar el crecimiento en escalabilidad
  - > Elegir estrategias que permitan mayor escalabilidad como almacén de sesiones independiente en caso de previsión de crecimiento
  - > Usar comunicaciones de alta velocidad entre nodos de replicación de sesiones

Matriz con tipo de persistencia, frecuencia, y ámbito

Persistence-type	*Persistencefrequency*	*Persistencescope*	*Supported in*
memory	stopserv*		7.0PE
file	time-based*		7.0PE
ha	time-based	session	7.0EE
ha	time-based	modified-session	7.0EE
ha	time-based	modified-attribute	7.0EE
ha	web-event	session	7.0EE
ha	web-event	modified-session	7.0EE
ha	web-event	modified-attribute	7.0EE

* :- These are not the supported values for this property

Escalabilidad y Alta Disponibilidad en J2EE™

El problema

- Soporte de clustering de Servidores
  - > Balanceo de carga entre instancias
  - > Gestión del failover entre instancias
- Garantizar alta disponibilidad de datos
  - > Información de sesión de cliente (con autenticación)
  - > Datos de negocio almacenados en base de datos
  - > Mensajes pendientes de procesamiento
- Garantizar alta disponibilidad de servicios J2EE
  - > Servicio de Nombres Servicio de Transacciones
  - > Servicio de Mensajes Servicio de EJB Timer
  - > Conectividad a Bases de Datos

Escalabilidad y Alta Disponibilidad en J2EE™

La solución

- Los fabricantes proporcionan guías de configuración y arquitecturas de alta disponibilidad en las versiones Enterprise
- Ajustar la arquitectura de sistemas a los requisitos de Escalabilidad y Alta Disponibilidad
- Guías de diseño y codificación
  - > Mantener la sesión de usuario lo más pequeña posible
  - > Pensar inicialamente que la aplicación debe escalar
- Algunos fabricantes proporcionan APIs propietarias, no incluidas en J2EE

Balanceo de carga de peticiones HTTP

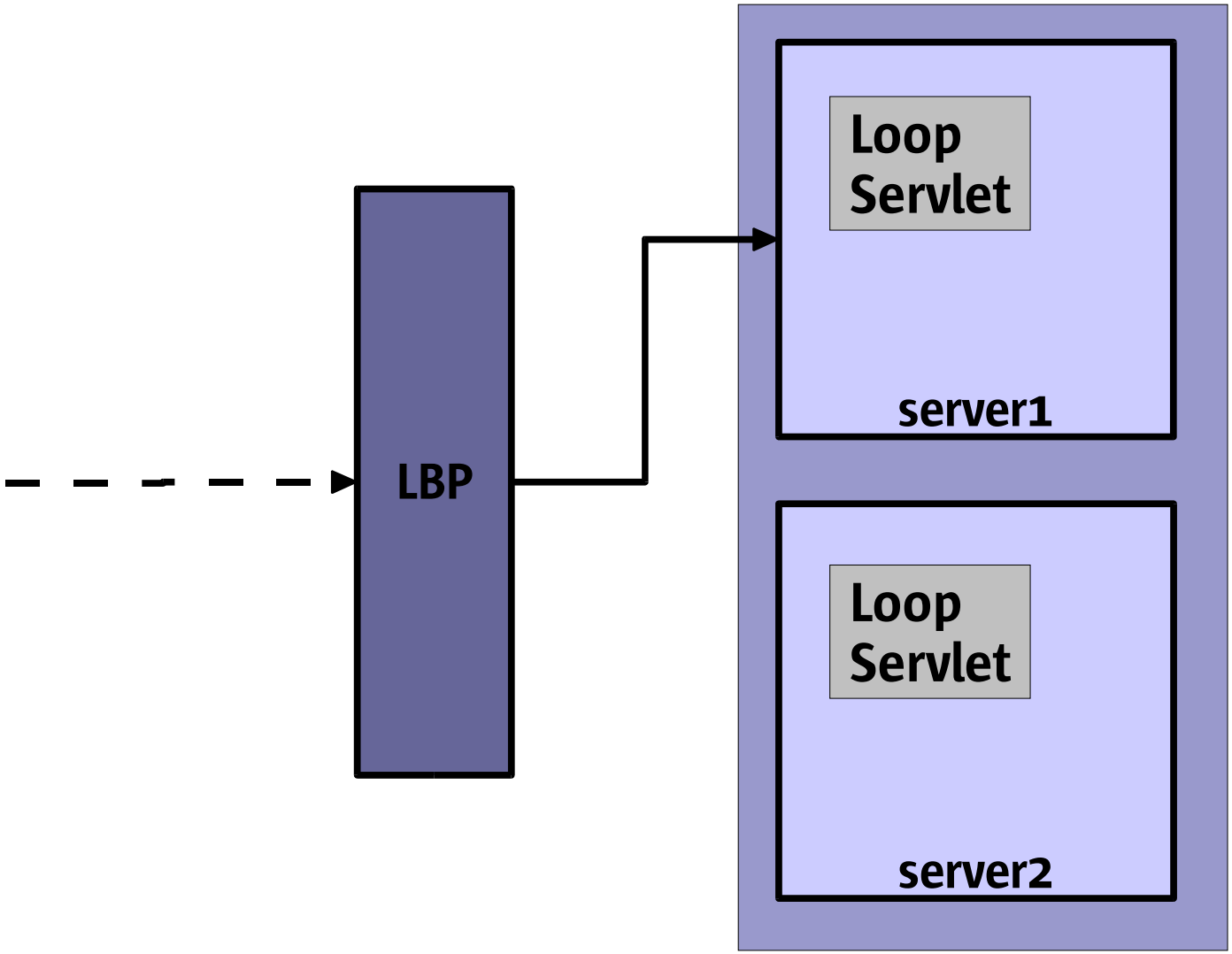
Características

- En general implementado en los plugin de los servidores WEB
- Algoritmos de balanceo
  - Round-robin
  - Round-robin con pesos
  - Basados en el tiempo de respuesta
- Asignación de clientes a instancias (stickiness)
- Monitorización de disponibilidad de instancias
- (Opcional) Reintento de peticiones idempotentes
- (Optional) Soporte de balanceadores HW

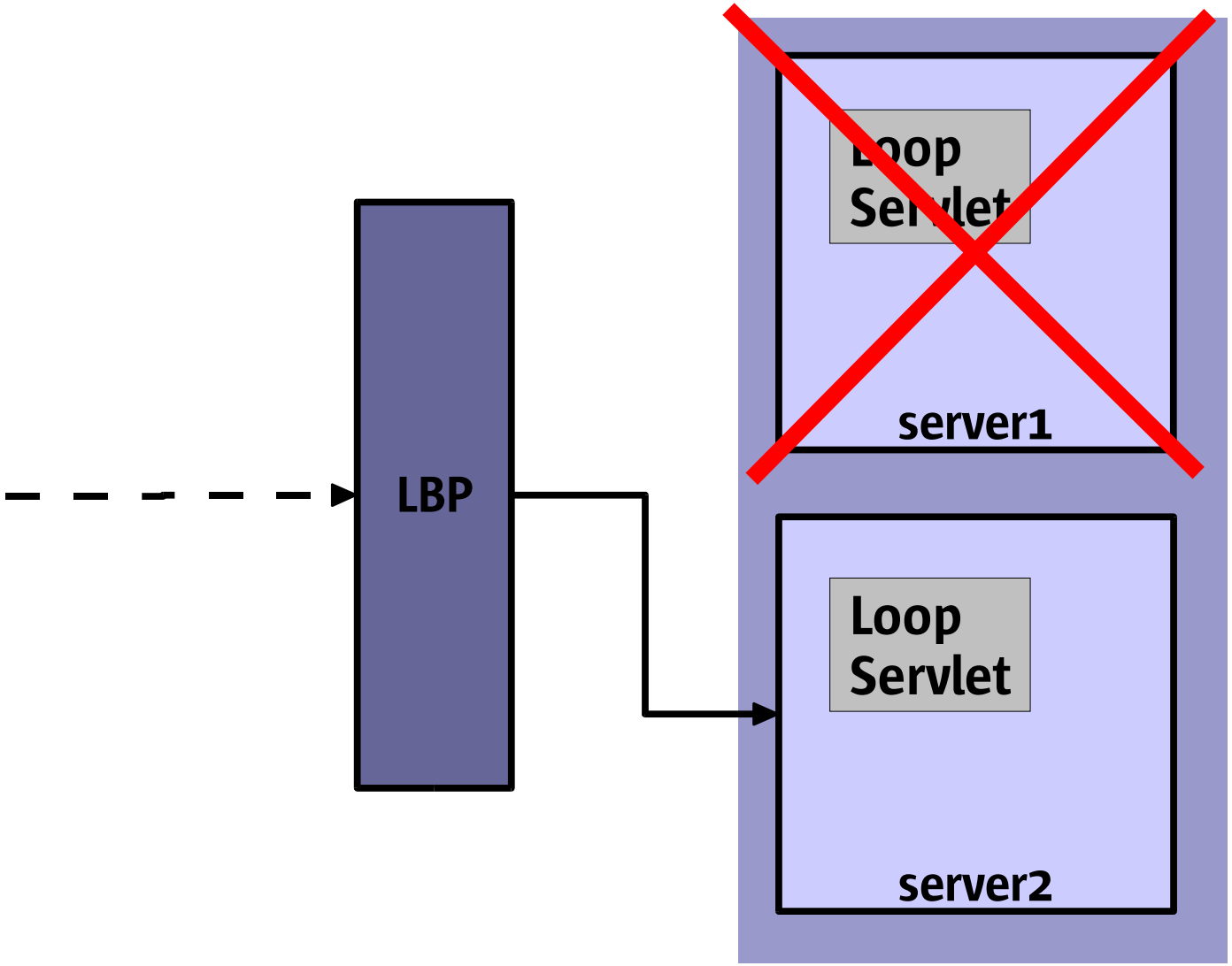
loadbalancer.xml

```
<loadbalancer name="loadbalancer1">
    <cluster name="cluster1">
      <instance name="instance1" enabled="true" listeners="
        http://instance1.domain.com"> </instance>
      <instance name="instance1" enabled="true" listeners="
        http://instance2.domain.com"> </instance>
      <web-module context-root="/webapps-simple" enabled="true" /
  >
      <health-checker url="/" interval-in-seconds="10" />
    </cluster>
    <property name="reload-poll-interval-in-seconds" value="5" />
    <property name="response-timeout-in-seconds" value="30" />
    <property name= "https-routing" value="true" />
</loadbalancer>
```

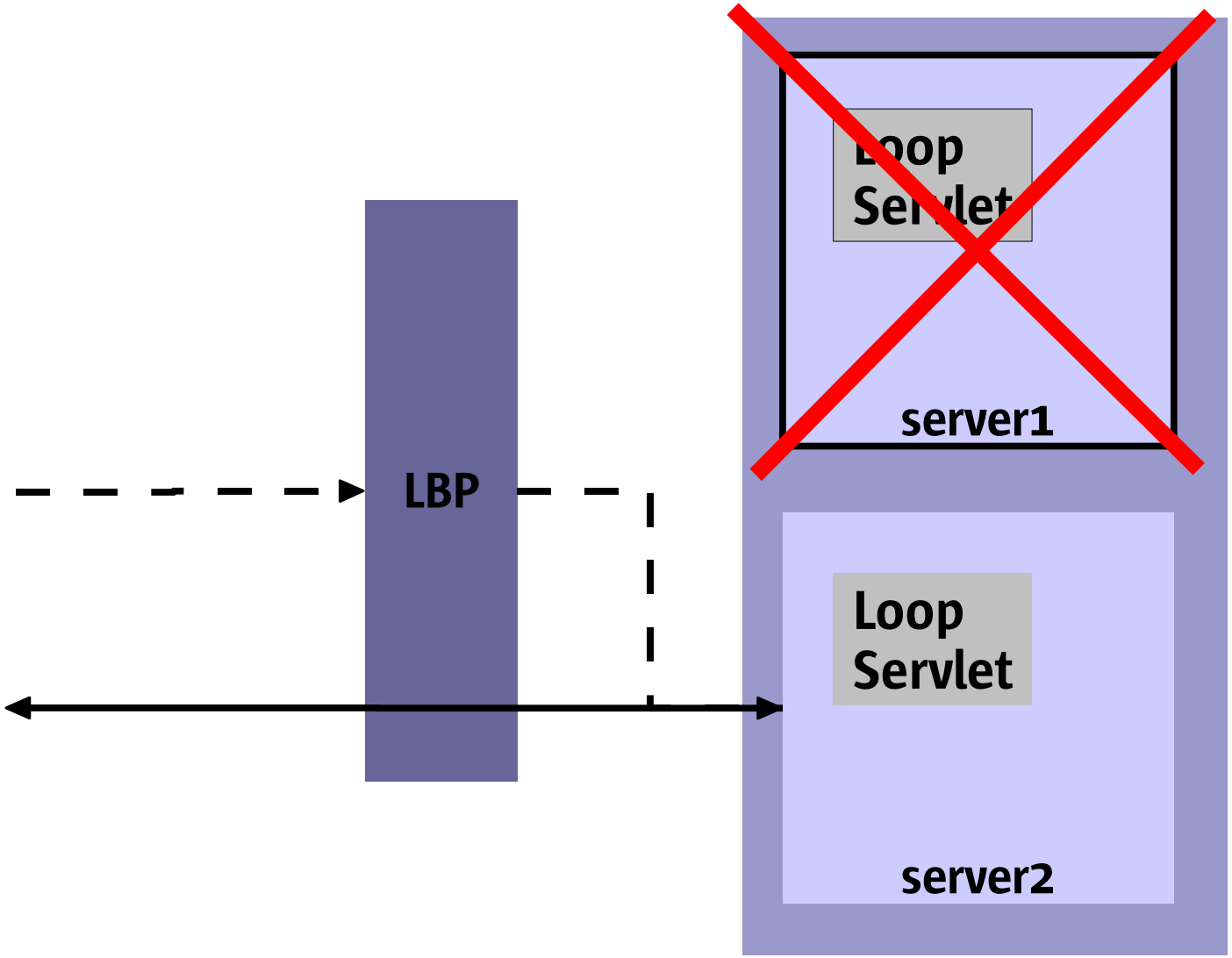
URLs idempotentes



URLs idempotentes



URLs idempotentes



URLs idempotentes

loadbalancer.xml

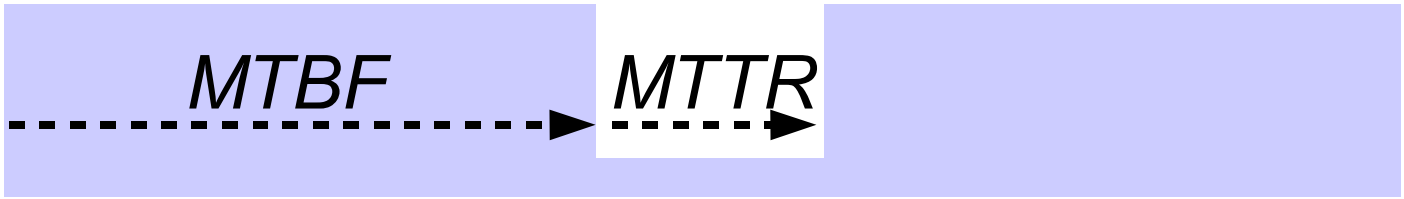
```
<cluster name="cluster1">
      <instance  name="instance1" enabled="true"
        disable-timeout-in-minutes="60" listeners="http://localhost:8080"/>
      <instance  name="instance2" enabled="true"
        disable-timeout-in-minutes="60" listeners="http://localhost:8081"/>
      <web-module context-root="infinite"
         enabled="true" disable-timeout-in-minutes="60" >
            <idempotent-url-pattern url-pattern="/*" no-of-retries="2" />
       </web-module>
```

Alta Disponibilidad

¿Que es Alta Disponibilidad?

El porcentaje del tiempo que el sistema entrega una respuesta correcta dentro de un determinado intervalo de tiempo.

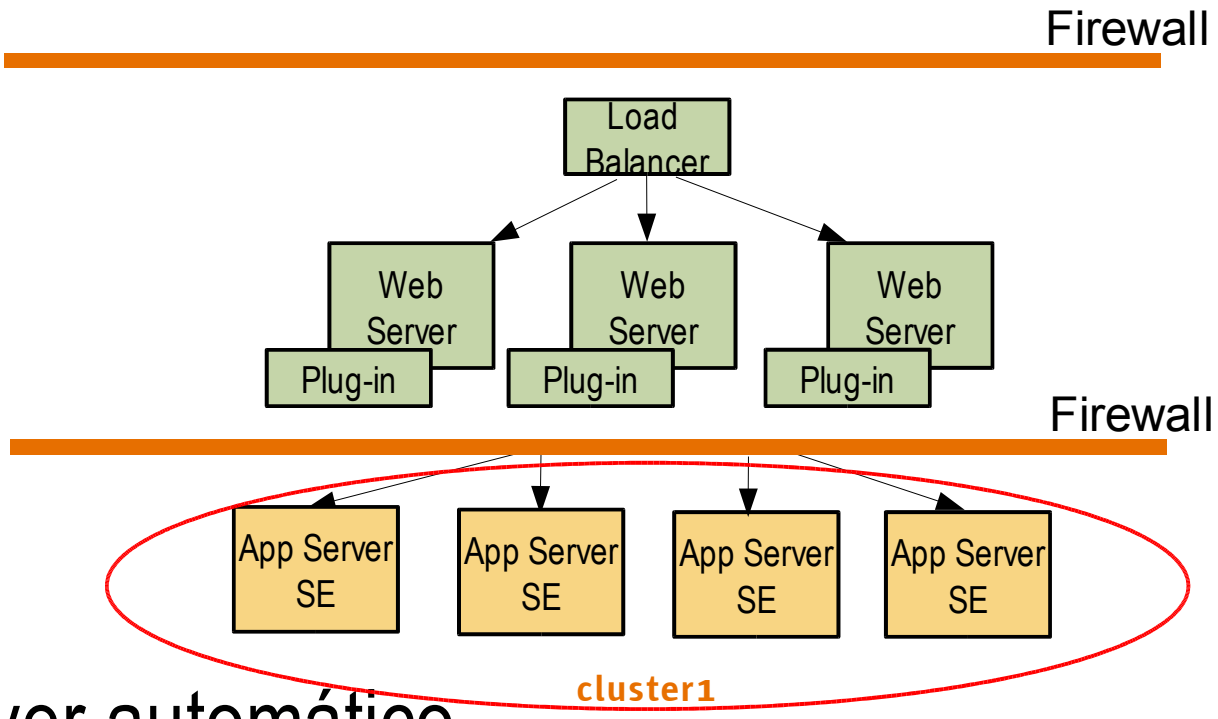
$$A = \frac{MTBF}{MTBF + MTTR}$$



MTBF == Mean Time Between Failure
MTTR == Maximum Time to Repair

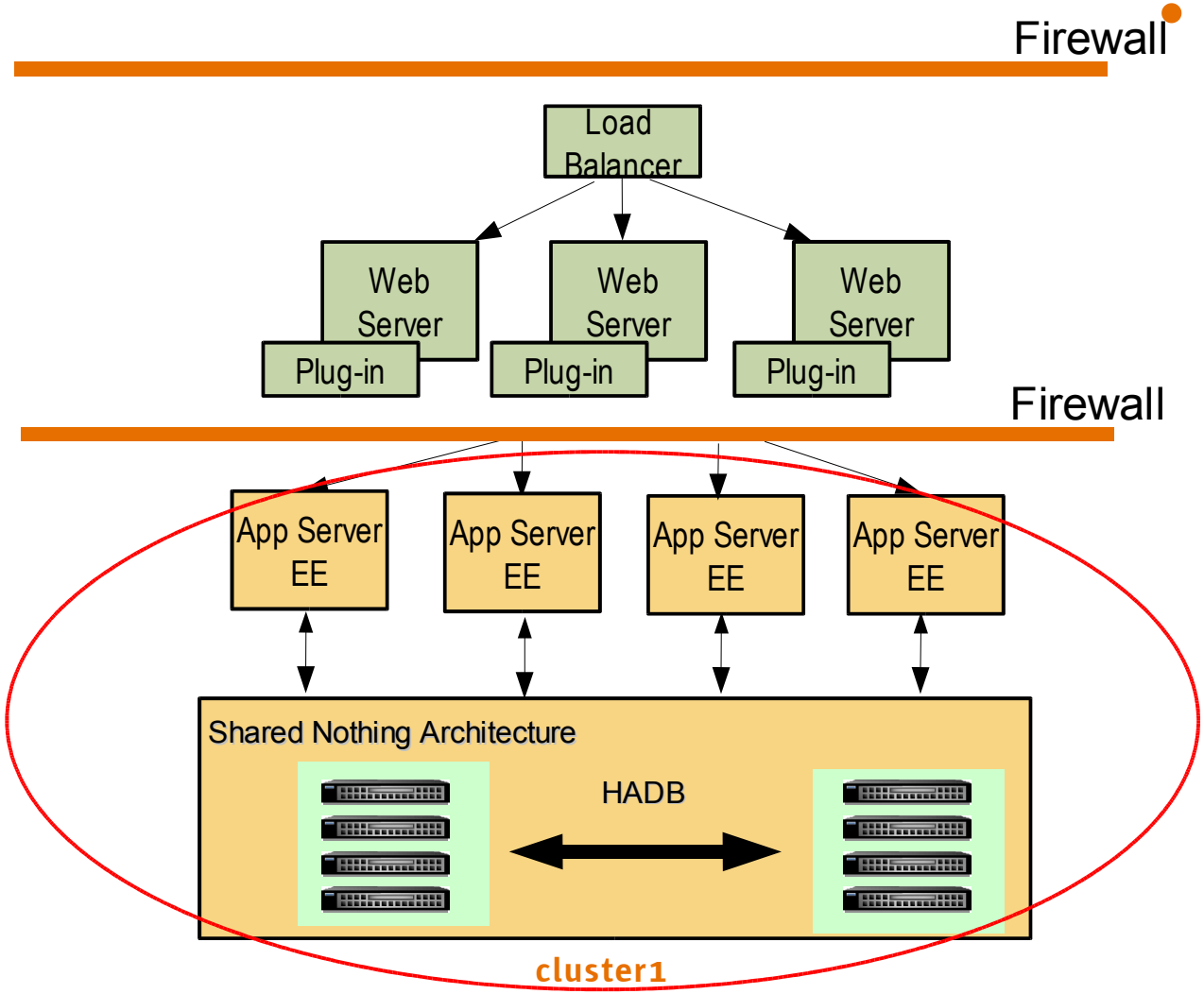
- "5–Nueves" 99.999% ~5 Min. Caida / Año
  - > Incluyendo paradas planificadas y no planificadas
  - > Implica un alto de grado de tolerancia a fallos

Alta dispnobilidad de servicio



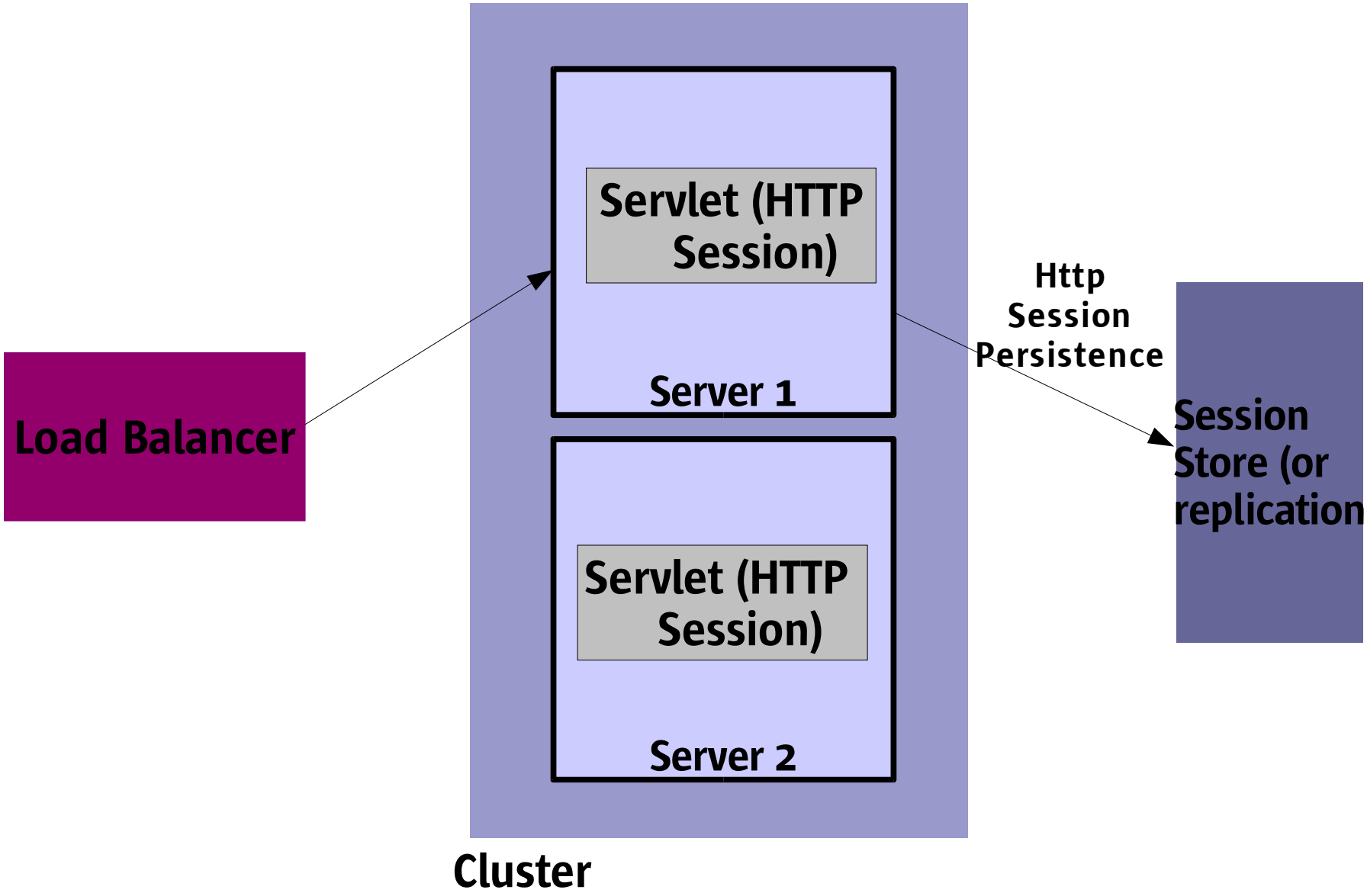
- Failover automático
  - > Detección de caidas de servicio
  - > Asignación de nueva instancia
  - > Comprobación de instancia viva
  - > Muy eficiente en coste
  - > Para aplicaciones sin sesión o cuya pérdida no sea grave para el servicio

Alta Disponibilidad de Sesión

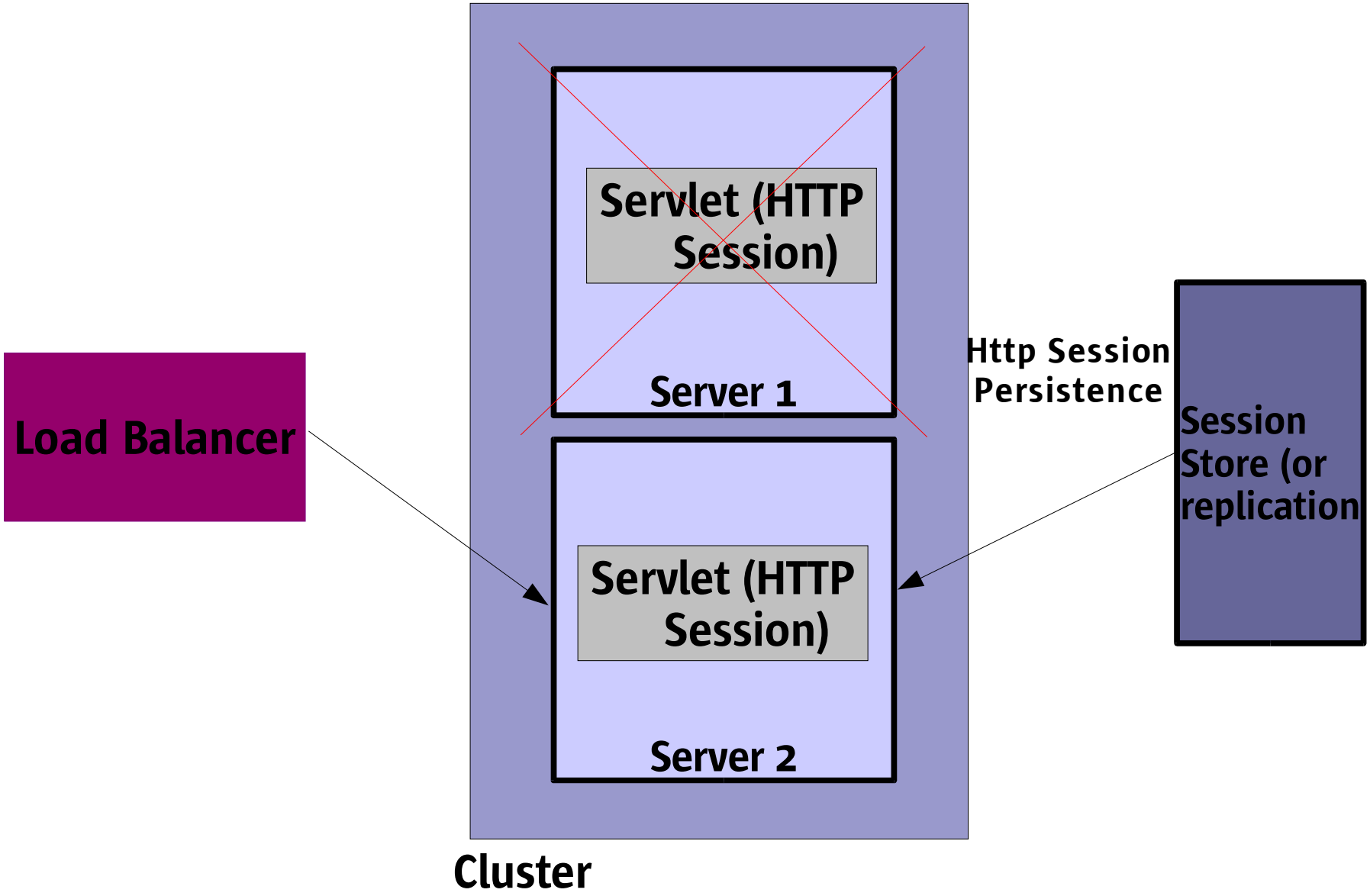


- Aplicaciones de Misión Crítica
  - Servicio 24x7
  - Sesión y estado siempre recuperable
  - Fallos transparentes al usuario

Failover de la sesión HTTP

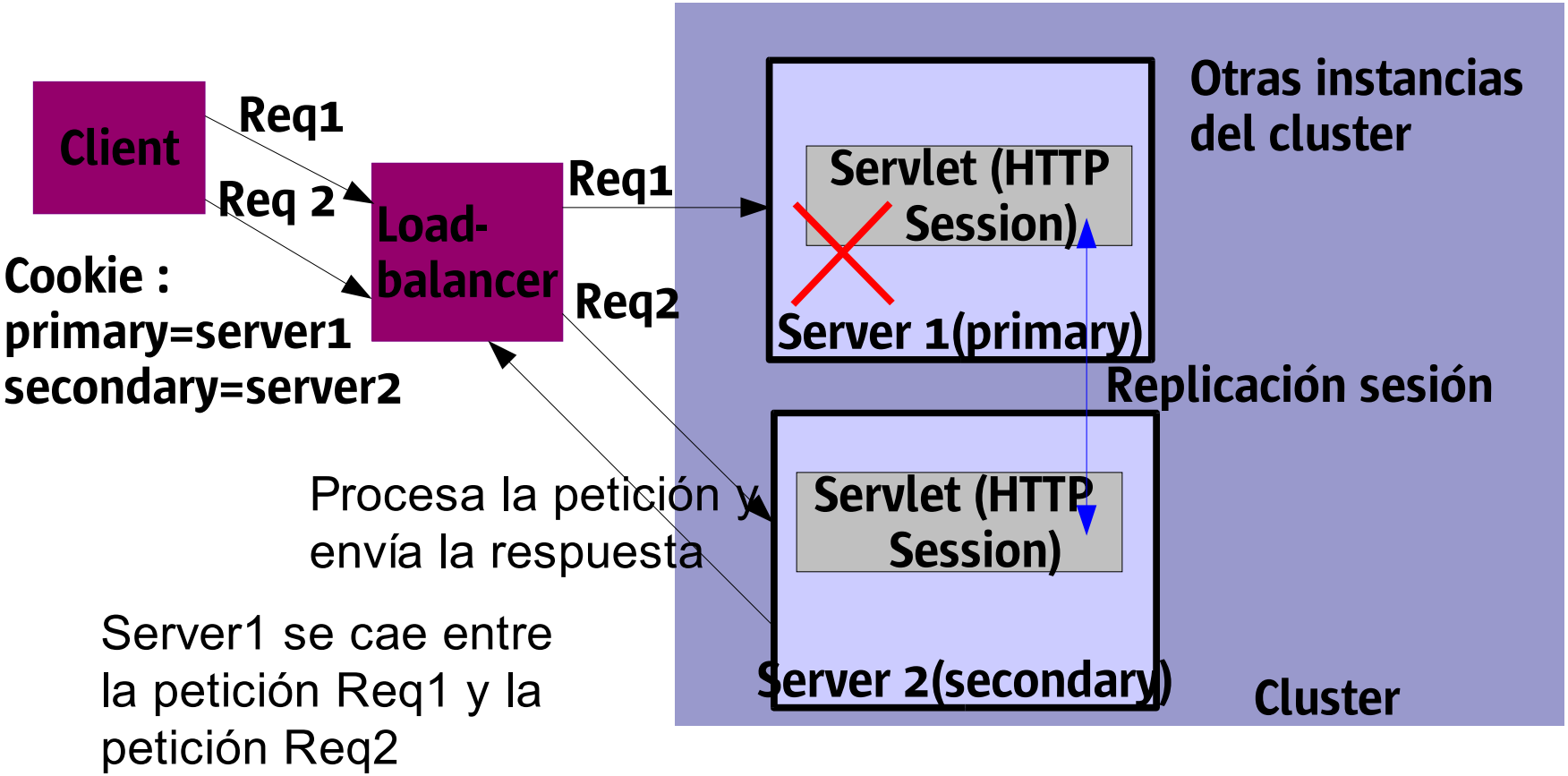


Failover de la sesión HTTP



Failover de la sesión HTTP

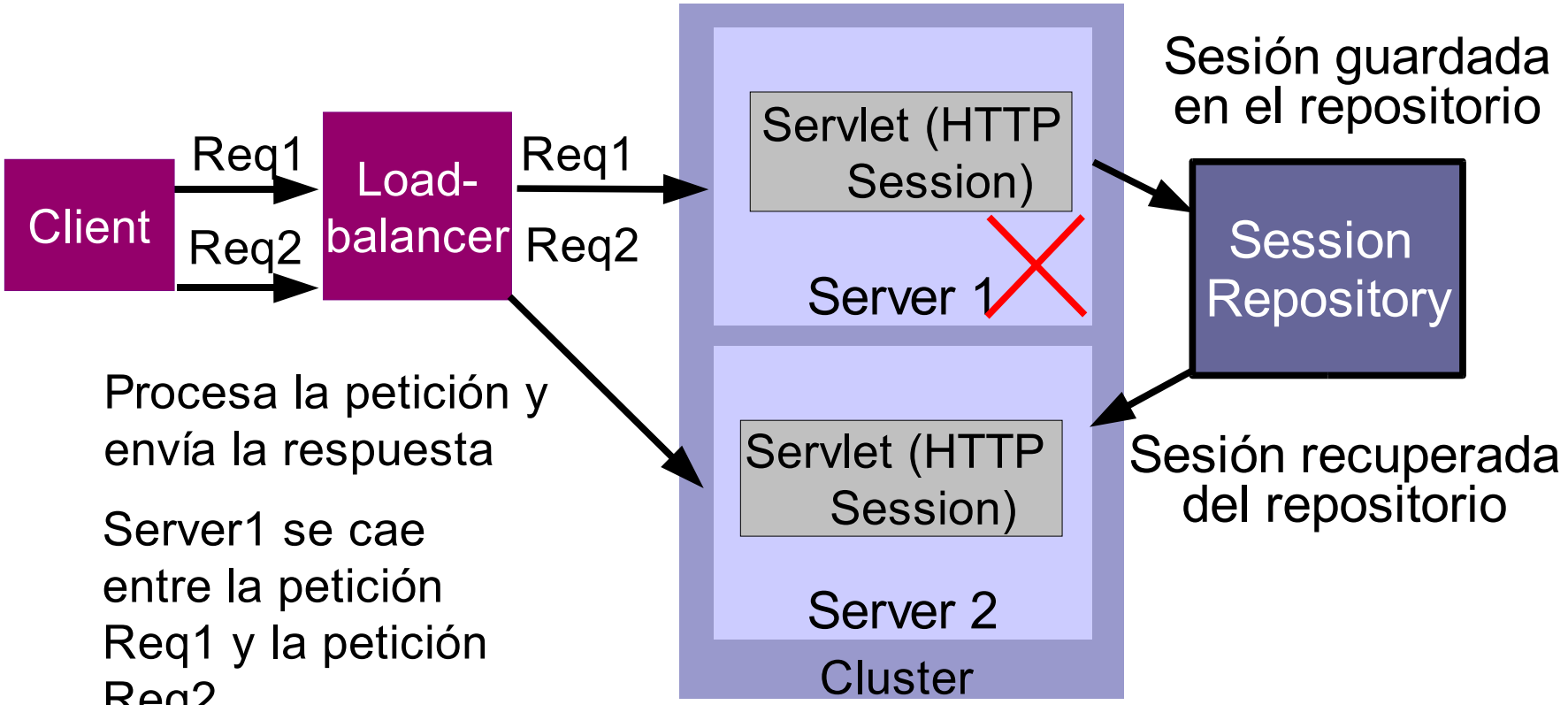
Replicación en memoria



- **Después del failover, el servidor2 pasa a ser el primario**
- **Se crea un tercer servidor secundario, si el servidor 1 queda inactivo**

Failover de la sesión HTTP

Usando un repositorio de sesiones



Estrategias para el Failover de la sesión HTTP

Frecuencia de la persistencia

- Configuración específica de cada fabricante
- Se puede configurar para persistir las sesiones al final de la petición http, o bien en background, cada cierto tiempo
- La estrategia basada en tiempo da una mejor latencia pero con el riesgo potencial de pérdida de la información del último minuto
- Algunos fabricantes proporcionan APIs para establecer estrategias propias de la aplicación perdiendo portabilidad y certificación J2EE

Estrategias para el Failover de la sesión HTTP

Ámbito de la persistencia

- Configuración específica de cada fabricante
- Sirve para reducir el volumen de datos a persistir
- Por defecto siempre se escribe toda la sesión en el almacen de persistencia
- Se pueden configurar otros ámbitos:
  - Sesión modificada
  - Atributo Modificado

Balanceo de carga y Failover invocaciones a EJB™

Técnicas

- InitialContextFactory específico del vendedor
  - Se distribuyen las siguientes llamadas a las diferentes instancias del cluster con una política round-robin:
    - New InitialContext()
    - InitialContext.lookup()
- IORs con información de cluster
  - Los IOR incluyen información del servidor donde se generan, y el cliente reintenta las peticiones con otro nodo en caso de que uno de los servidores del cluster no esté respondiendo.
- Stubs inteligentes (con información de cluster)
  - Incluyen failover y Balanceo de Carga

Balanceo de carga y Failover invocaciones a EJB™

Optimizaciones para evitar congestion en la red

- Afinidad de Servidor
  - Las llamadas a los EJB por parte del mismo cliente siempre se hacen al mismo servidor, minimizando el número de conexiones abiertas simultaneamente
- Colocalización: Útil para Servlet llamando EJB
  - Los servlets llaman al EJB local, dentro de la misma JVM, en vez de balancear las peticiones.
- Afinidad de Transacciones
  - Uso de un mismo servidor para las llamadas que forman parte de una transacción

Failover de Stateless Session Beans

Usar cualquier otra instancia del cluster

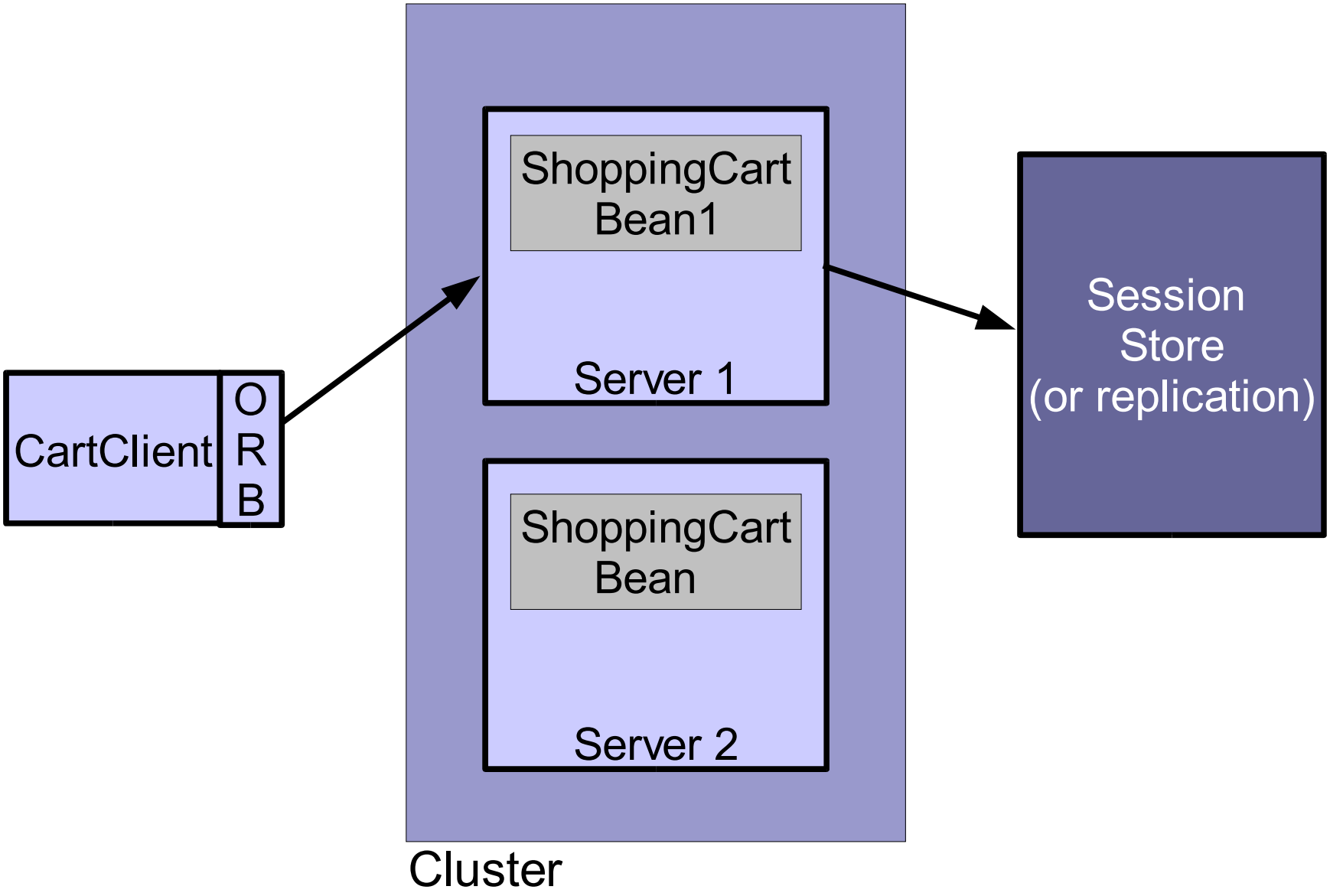
- Secuencia de eventos
  - El Cliente detecta un fallo en la invocación
  - El Cliente identifica un endpoint IIOP alternativo
  - El Cliente reinvoca la petición
  - Cualquier instancia del SLSB sirve para gestionar la invocación ya que no existe información de estado asociada.

Failover de Stateful Session Beans

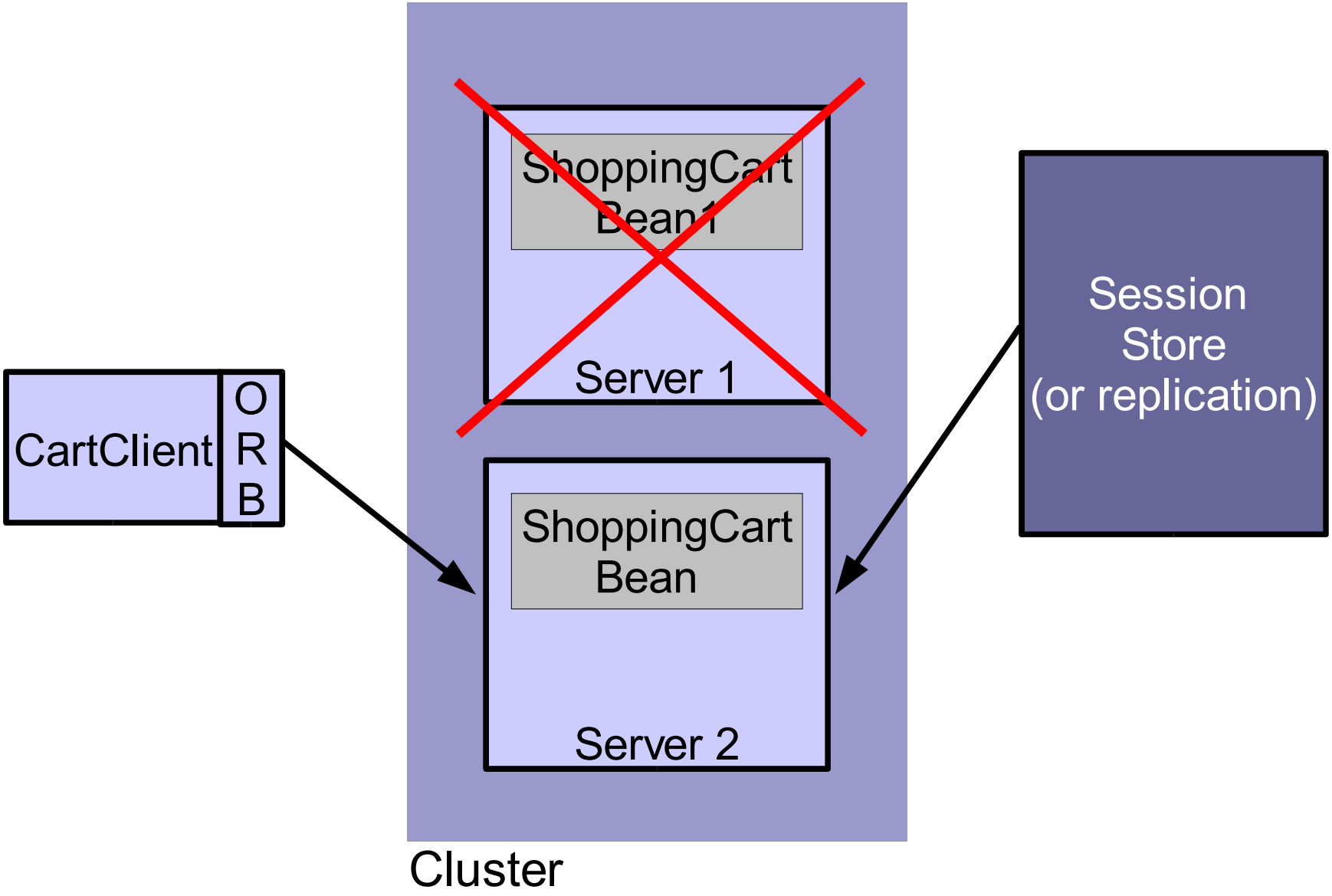
Se requiere Checkpointing

- Las actividades del Client son las mismas que en el caso de SLSB excepto:
  - La instancia asignada para ejecutar la llamada debe reconstruir la información de estado, o bien de repositorio o bien de otra instancia replicada
- Problemas:
  - Las invocaciones de SFSB pueden ser transaccionales — el checkpointing se puede hacer sólo al final de la transacción o del método si no la hubiera
  - Demasiado checkpointing puede afectar al rendimiento

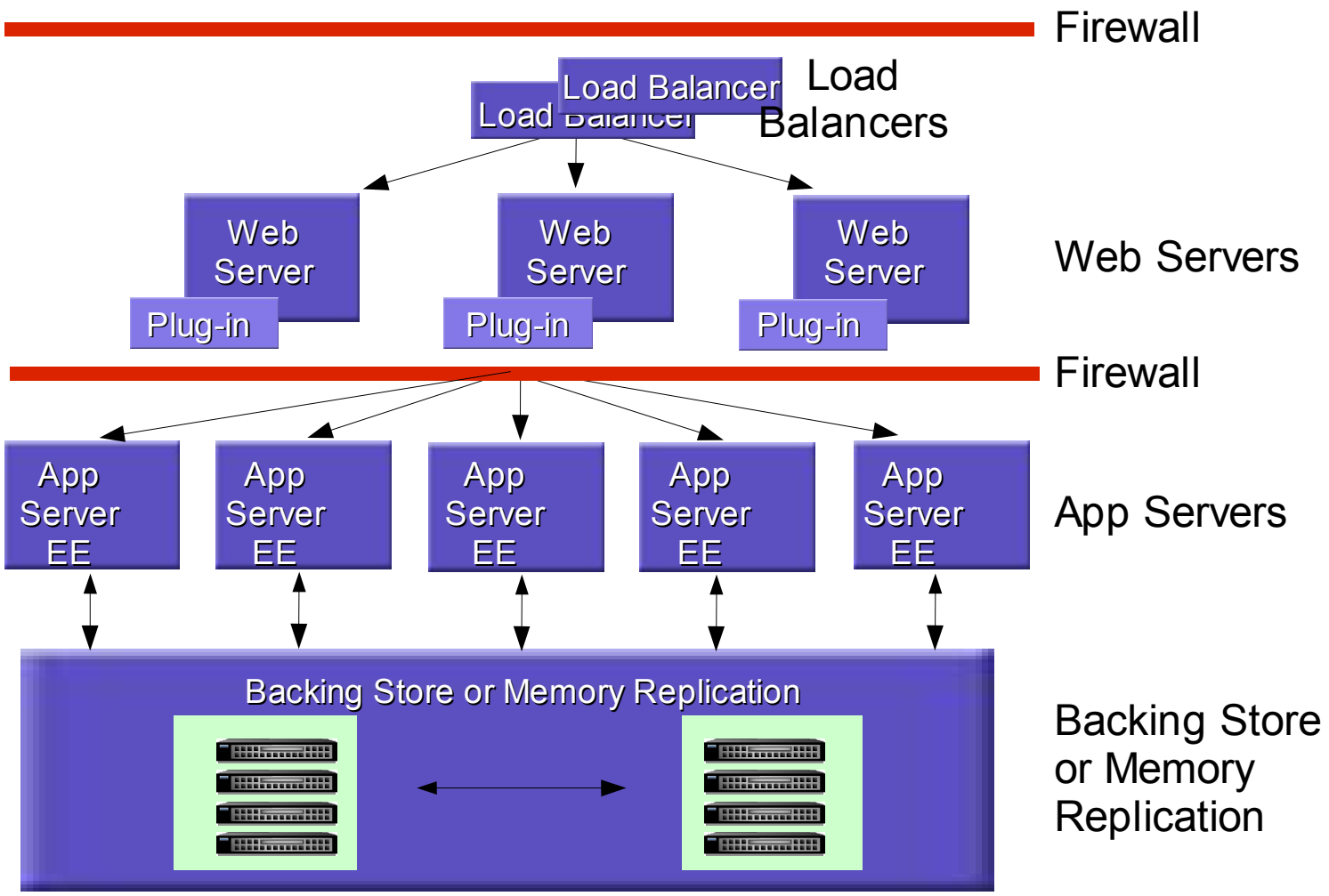
EJB™ Stateful Session Bean Failover



EJB™ Stateful Session Bean Failover



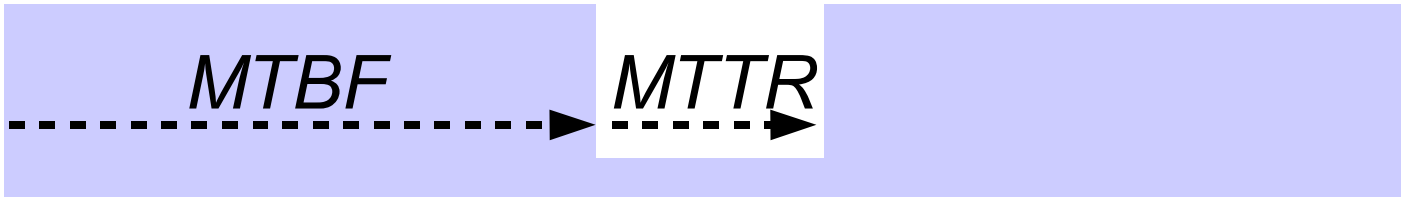
Arquitectura Altamente Disponible



¿Que es Alta Disponibilidad?

El porcentaje del tiempo que el sistema entrega una respuesta correcta dentro de un determinado intervalo de tiempo.

$$A = \frac{MTBF}{MTBF + MTTR}$$



MTBF == Mean Time Between Failure
MTTR == Maximum Time to Repair

- "5–Nueves" 99.999% ~5 Min. Caida / Año
  - > Incluyendo paradas planificadas y no planificadas
  - > Implica un alto de grado de tolerancia a fallos

Fiabilidad y Alta Disponibilidad

Conceptos

- MTTF—Número /Frecuencia de Fallos
- MTTR—Duración de los fallos (Time to Repair)
- Alta fiabilidad no implica Alta Disponibilidad
  - > 1 fallo —1hora para recuperarlo—Alta fiabilidad pero Baja Disponibilidad
  - > 60 fallos —1 segundo para recuperarse de cada fallo. Baja fiabilidad pero Alta Disponibilidad
- Incluso con un sistema poco fiable (máquinas baratas) puedo tener Alta Disponibilidad si la recuperación ante fallos es muy rápida

¿Garantizan los servidores de aplicaciones la alta disponibilidad?

- Los fabricantes se esfuerzan en lograr que esto sea verdad y en aislar a los desarrolladores y operadores de complejidad innecesaria.
- El objetivo de los servidores de aplicaciones es que un cluster de servidores se comporte como un único servidor escalable, un servidor tolerante a fallos virtualizado
- La disponibilidad no es gratuita
  - > Se penaliza el consumo de recursos
  - > Muchas veces penaliza el rendimiento

¿Tiene algun coste activar la alta disponibilidad?

- La alta disponibilidad de sesión implica una replicación de esta información en diferentes nodos. Por tanto tiene los siguientes costes:
  - Memoria para guardar las sesiones replicadas
  - Tráfico de red mientras se produce la replicación
  - Ciclos de CPU para gestionar la replicación

Error 1: Las aplicaciones no se tienen que ocupar de la Alta Disponibilidad sino los productos de Servidor

- Las sesiones HTTP y los EJB de estado debe ser serializables
  - Consejo: las aplicaciones deben hacer el menor uso posible de los objetos de sesión
- Previsión de posibles excepciones:
  - Los fallos que causen expeciones que no deban ser motivo de failover deben ser tratados específicamente para prevenir el posible failover

Error 2. Todas las peticiones son idempotentes

- *"La idempotencia es la cualidad de algo que tiene el mismo efecto si se usa múltiples veces que si sólo se usa una"*
- Ejemplos de operaciones idempotentes
  - "Fijar la temperatura de la habitación a 23º"
  - Cualquier operación de lectura
- Muchas (la mayoría) no son idempotentes!!!
  - "Incrementa la cuenta en 20"
  - "Pasa al estado siguiente"

Error 2. Todas las peticiones son idempotentes

- Las peticiones no idempotentes pueden afectar a la integridad de los datos si se repiten
- Failover implica reintentos en operaciones de negocio
  - Muchas (la mayoría) de las operaciones de negocio no son idempotentes
- Cuidado con funcionalidad del tipo failover transparente

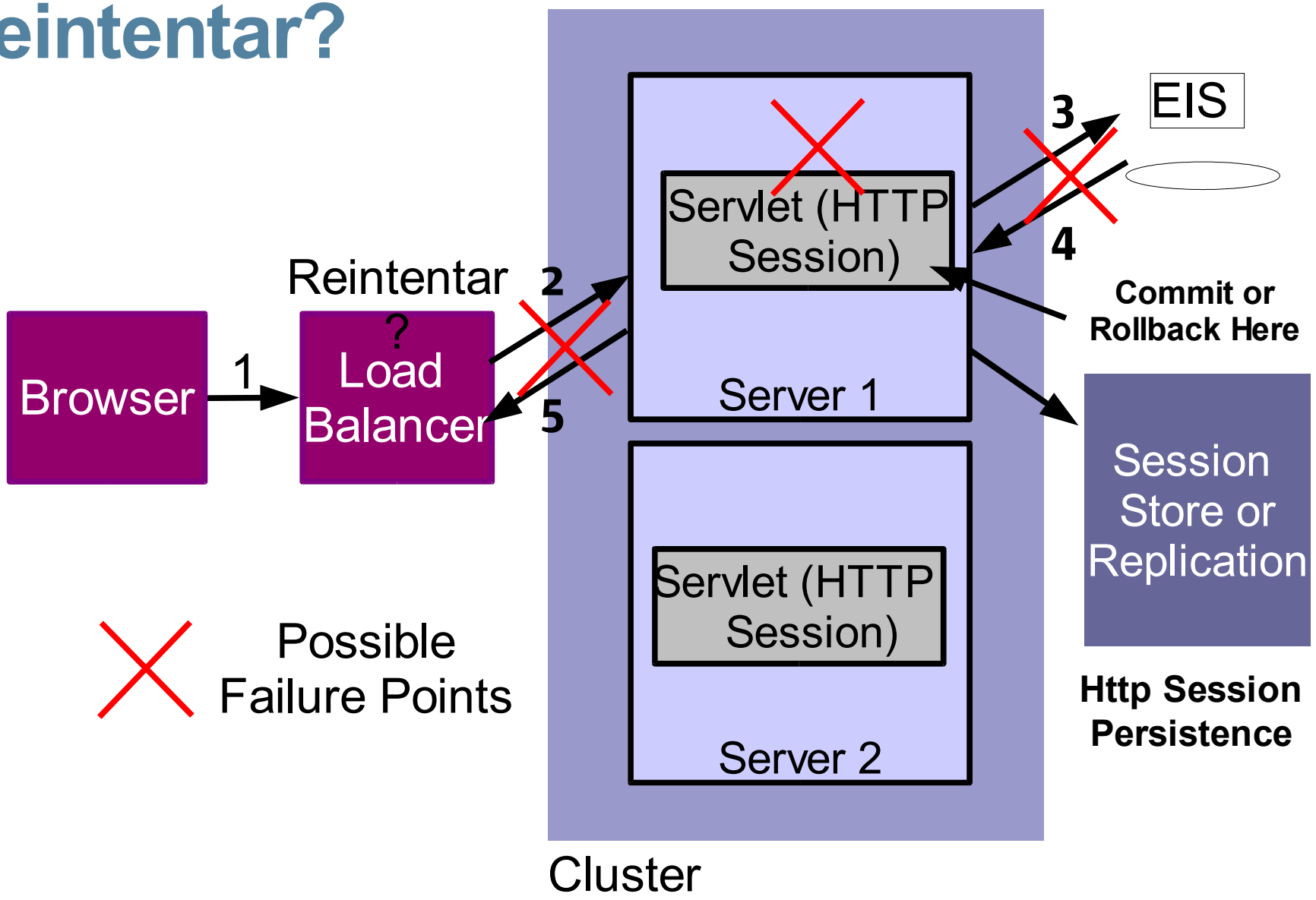
Error 2. Todas las peticiones son idempotentes

- Peticiones HTTP
  - Algunos balanceadores soportan failover transparente sobre peticiones idempotentes
    - Usar con precaución. Hacer un análisis previo.
    - No poner en riesgo la integridad de los datos
- Peticiones RMI/IIOP
  - Los buenos orb distinguen los estados de “no completada”, “si completada” y “en duda” y el failover solo se debería hacer en el primer caso

Error 2. Todas las peticiones son idempotentes

- Mensajes JMS(entrada de MDB's)
  - Usar la transacción manejada por el contenedor para onMessage()
  - El envio de duplicados es posible, la aplicación lo debería tener en cuenta (también en el origen)
  - Ayudas para determinar el reenvio de mensajes:
    - getJMSRedelivered()
      - *"Es probable aunque no se garantiza que el mensaje fue entregado anteriormente pero su recepción no fue confirmada entonces*
    - getJMSCorrelationID()
      - Útil en situacioens donde los mensajes son correlados respecto a los mensajes enviados

Idempotencia—Es seguro reintentar?



Error 3. Usar las sesiones HTTP o EJBs como una Base de Datos

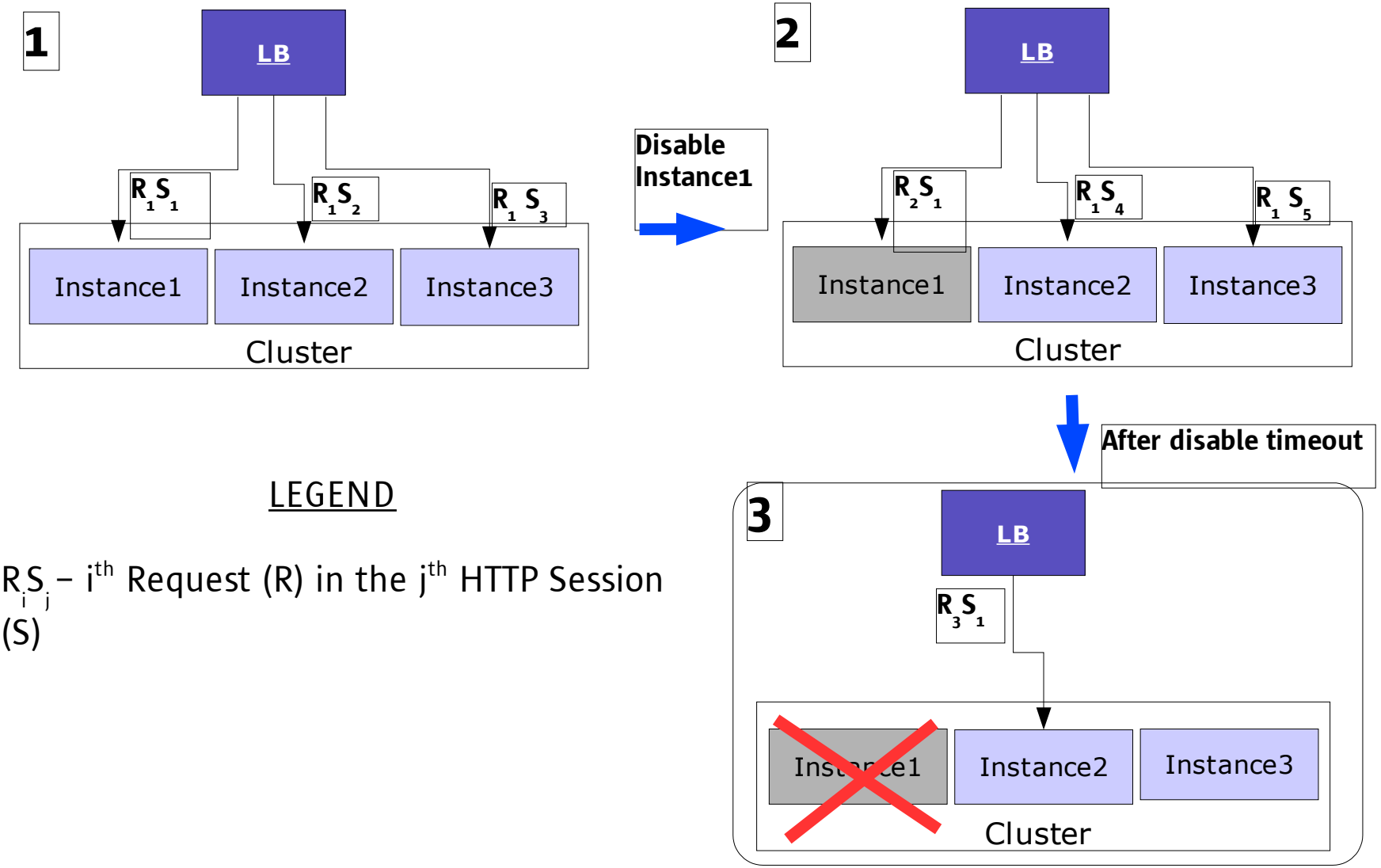
- Guardar demasiados datos en las sesiones
  - En una sola instancia—"sólo" un problema de memoria
  - En clusters, un impacto directo al rendimiento, escalabilidad y alta disponibilidad
- No tienen transaccionalidad
  - No se garantiza sincronización con el back-end
  - En caso de failover se puede romper la sincronización

Error 4. Los upgrades y parches no afectan a la Alta Disponibilidad

Consejo: Probar con Rolling Upgrades y Quiescence

- Que es quiescence
  - Algunos balanceadores lo soportan a dos niveles:
    - Instance-level quiescence
    - Application-level quiescence
- En entornos de producción probar primero en entornos de certificación o pruebas

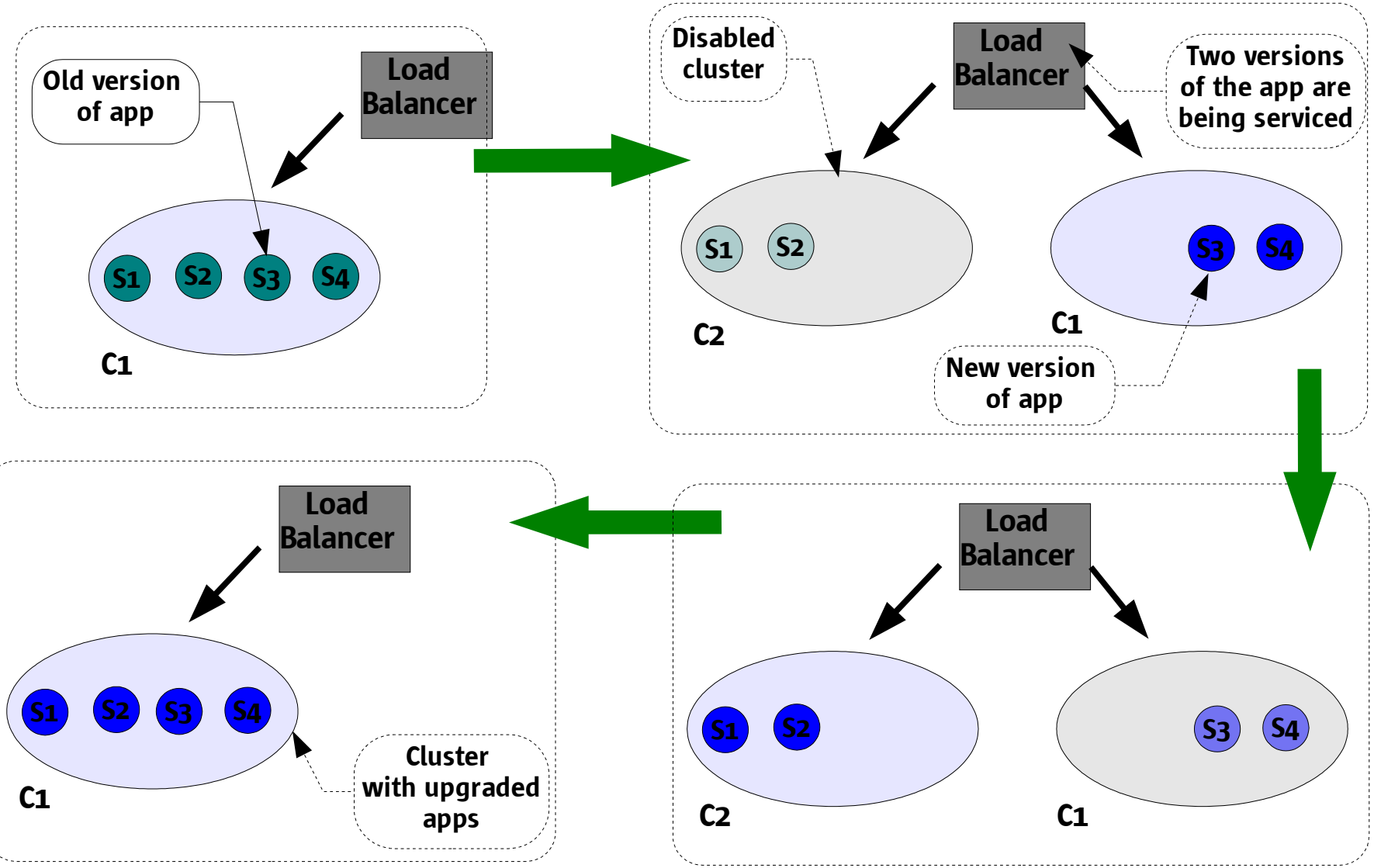
Instance Quiescing



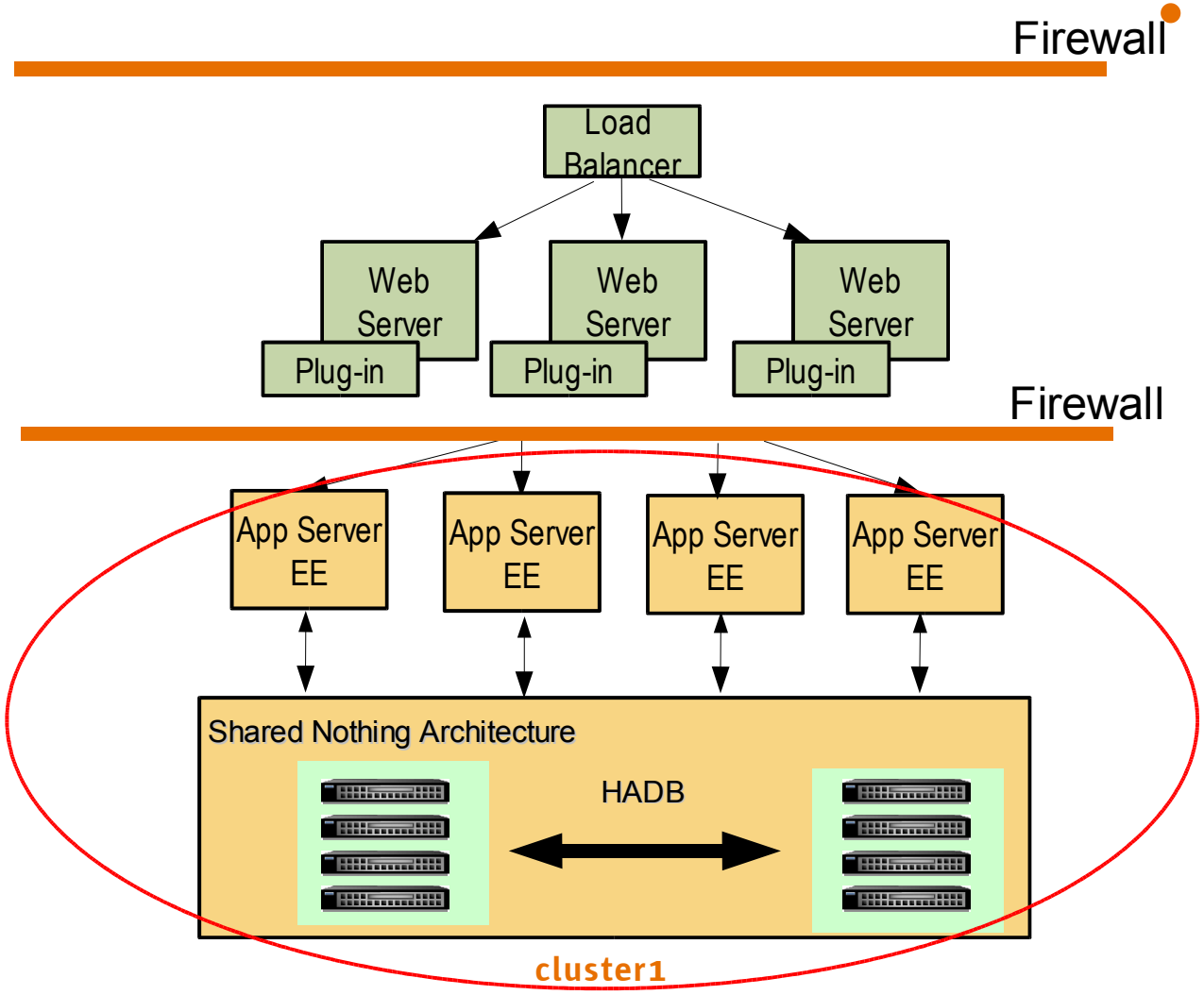
LEGEND

R_iS_j – i^{th} Request (R) in the j^{th} HTTP Session (S)

Application Quiescing

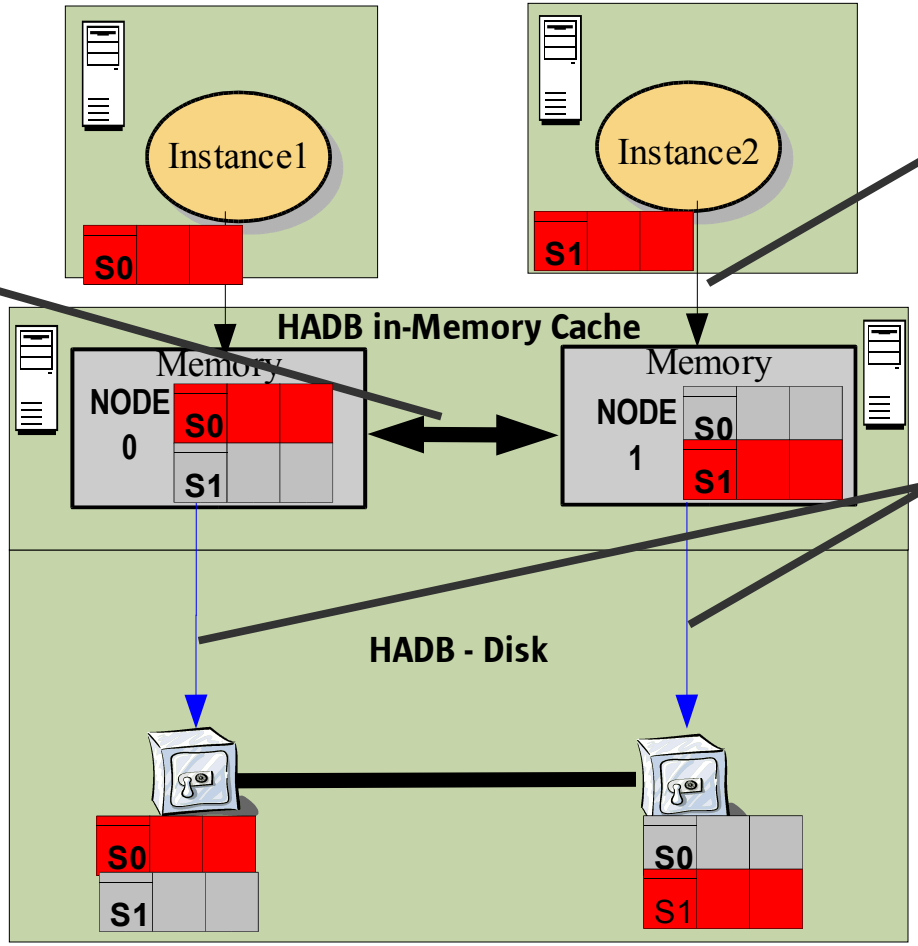


Alta Disponibilidad de Sesión



- Aplicaciones de Misión Crítica
  - Servicio 24x7
  - Sesión y estado siempre recuperable
  - Fallos transparentes al usuario

Persistencia a velocidades de memoria con alta disponibilidad de 5 nueves



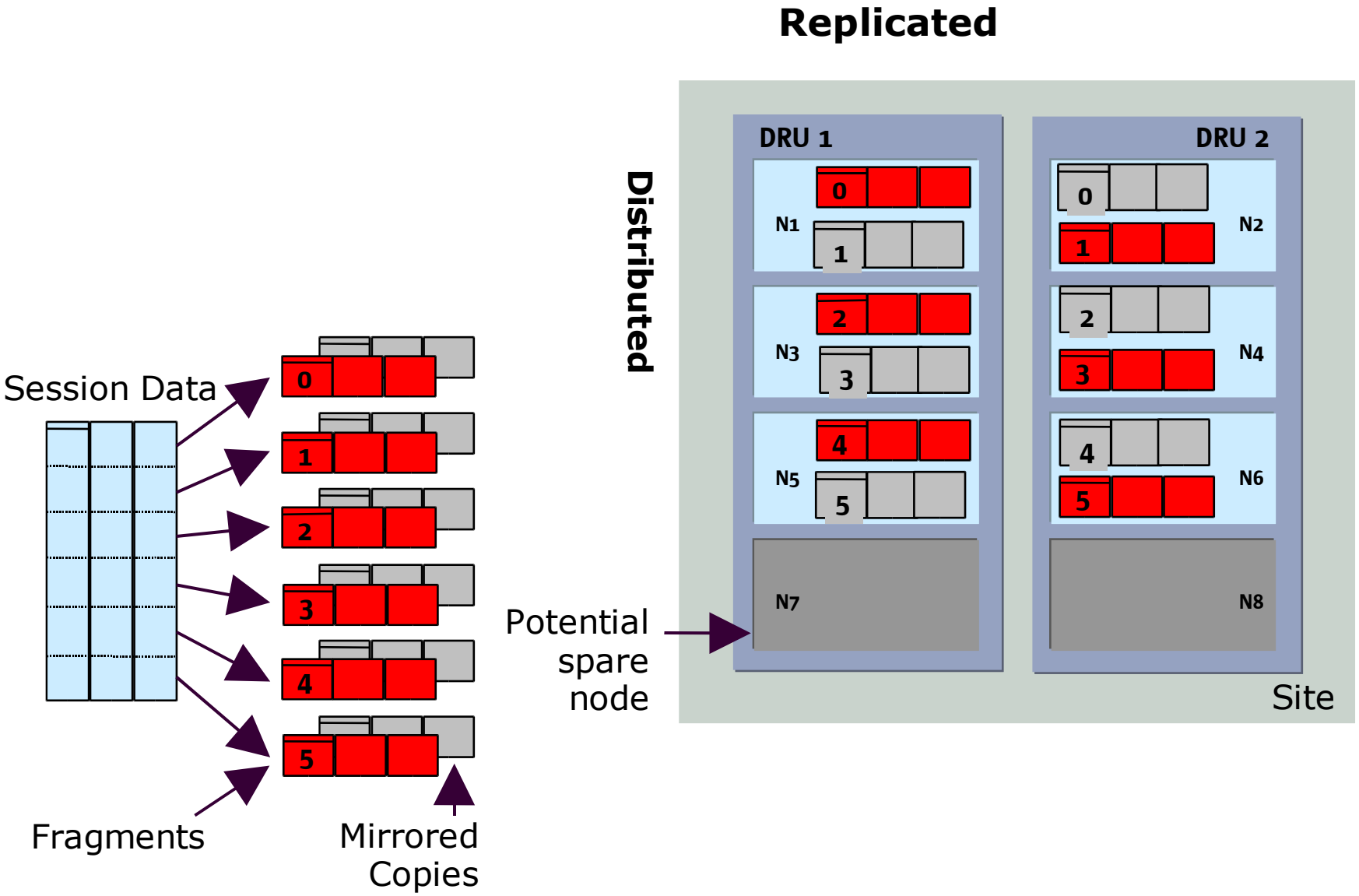
2. Fast synchronous memory to memory replication across nodes, JVM freed quickly

1. Real time in-memory writes from JVM heap to HADB Memory Cache

3. Asynchronous writes to HADB

- **Always-On - Automatic**
- **Self Repairing**
- **No intervention**
- **Massively Scalable**

HADB – Almacén de sesiones

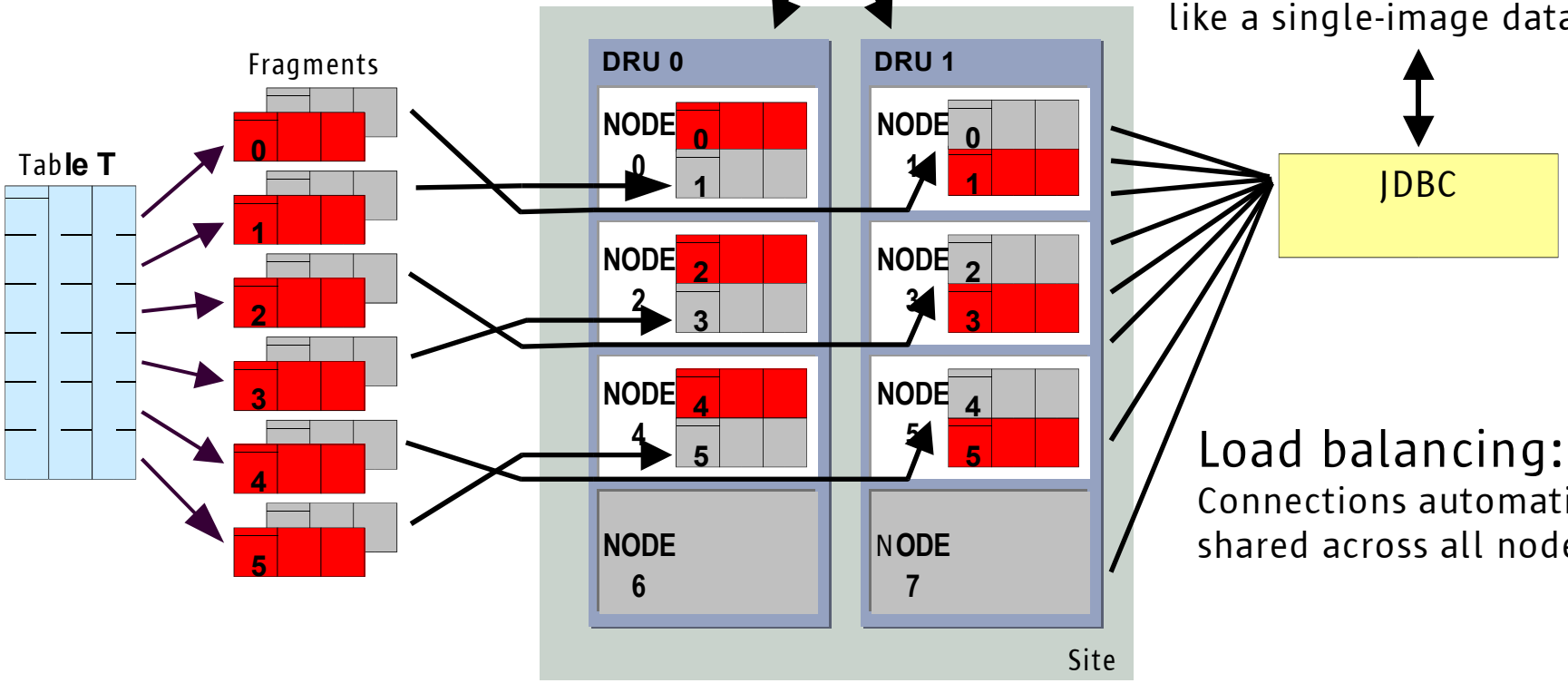


Arquitectura de datos y acceso

Synchronous Replication:
Updates are synchronized
across both copies of the data

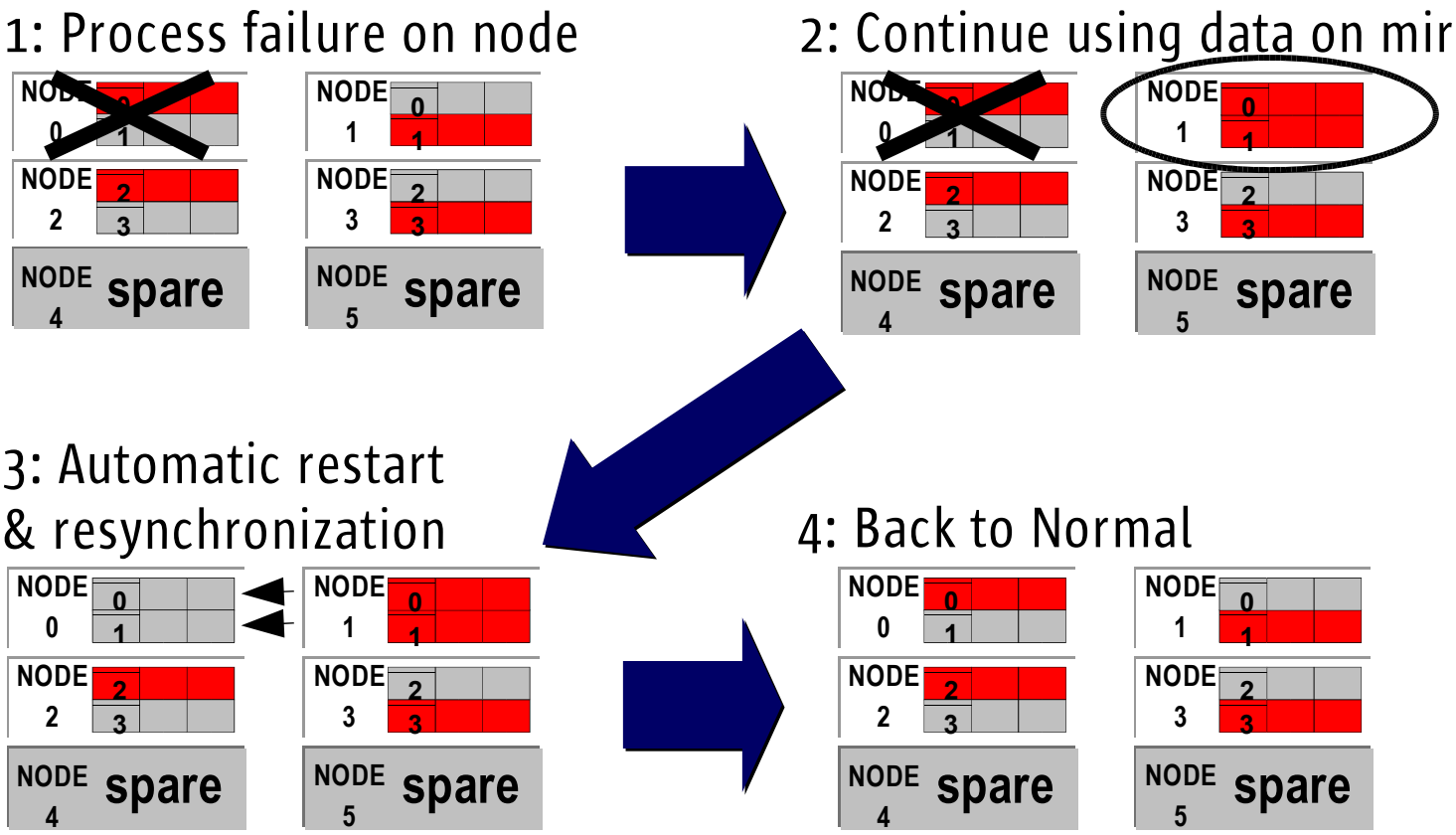
Data Distribution:
Tables automatically distributed
across active nodes

Transparency:
From the outside, the cluster
Is invisible, and Clustra looks
like a single-image database



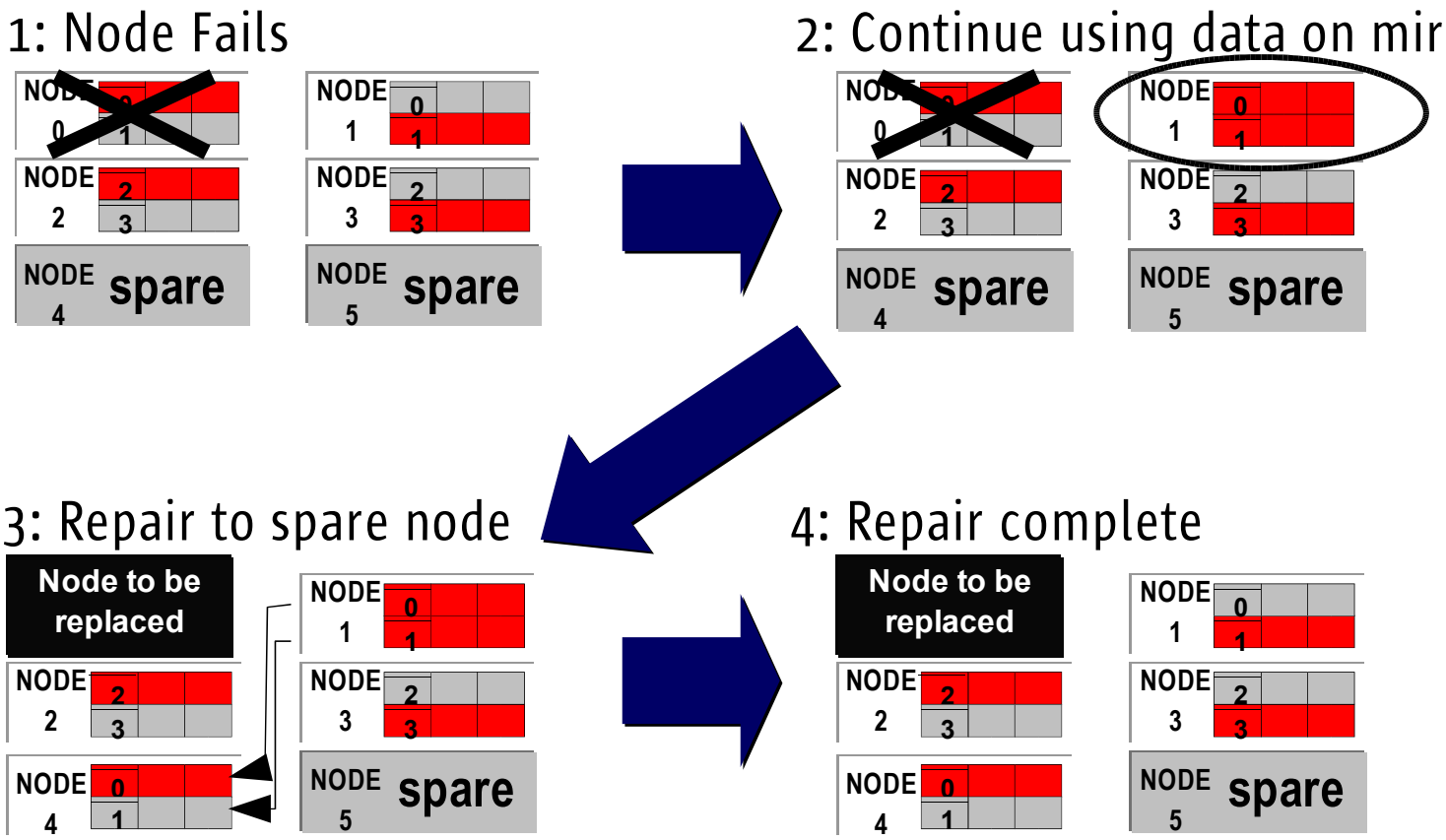
Load balancing:
Connections automatically
shared across all nodes

Fallos SW - Autoreparación



Self-Repair – No Human Intervention

Fallos HW- Autoreparación



Self-Repair – No Human Intervention

Operación continua

Operación continua

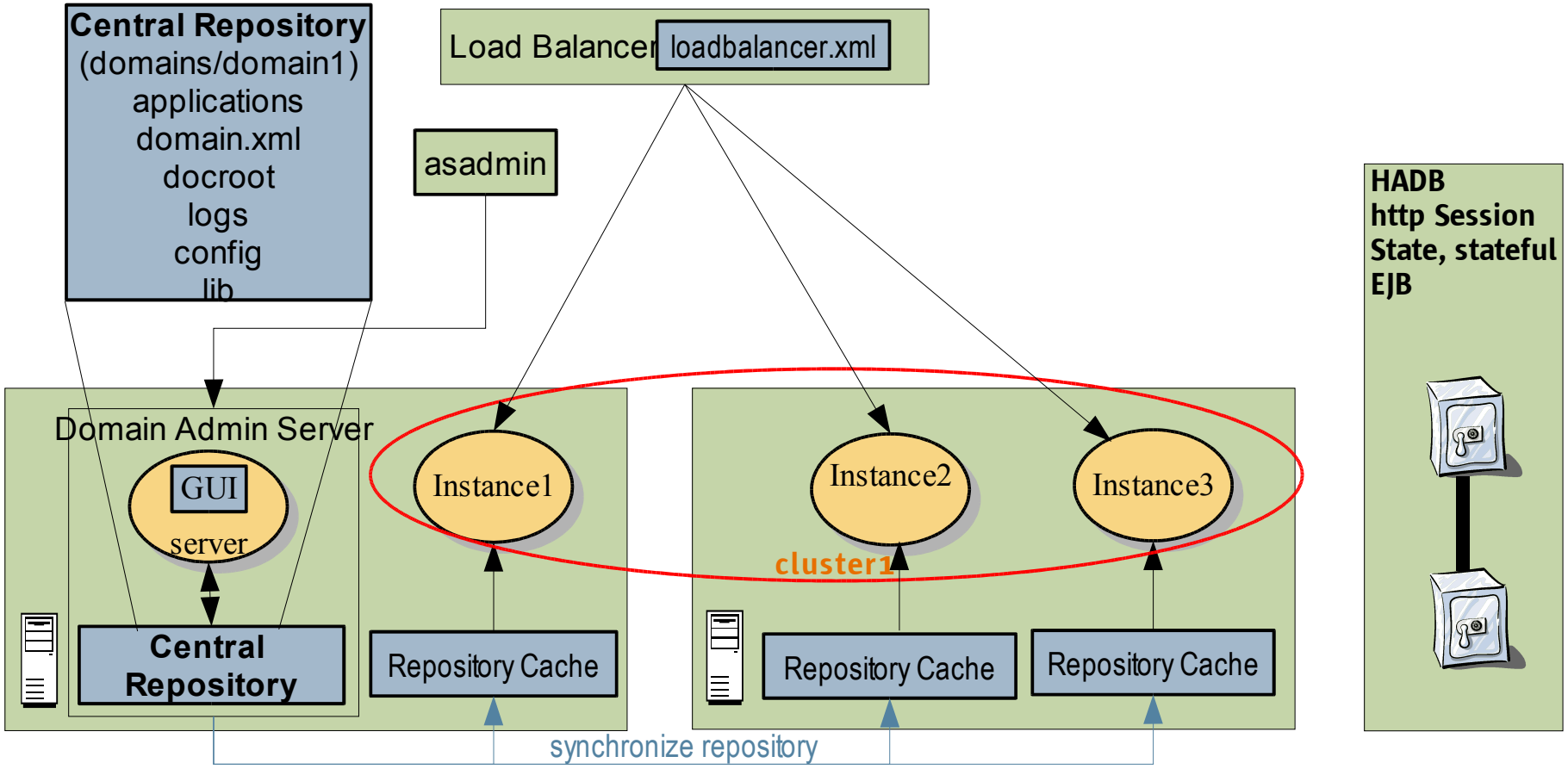
- **Mantenimiento HW**

- **Mantenimiento SW**

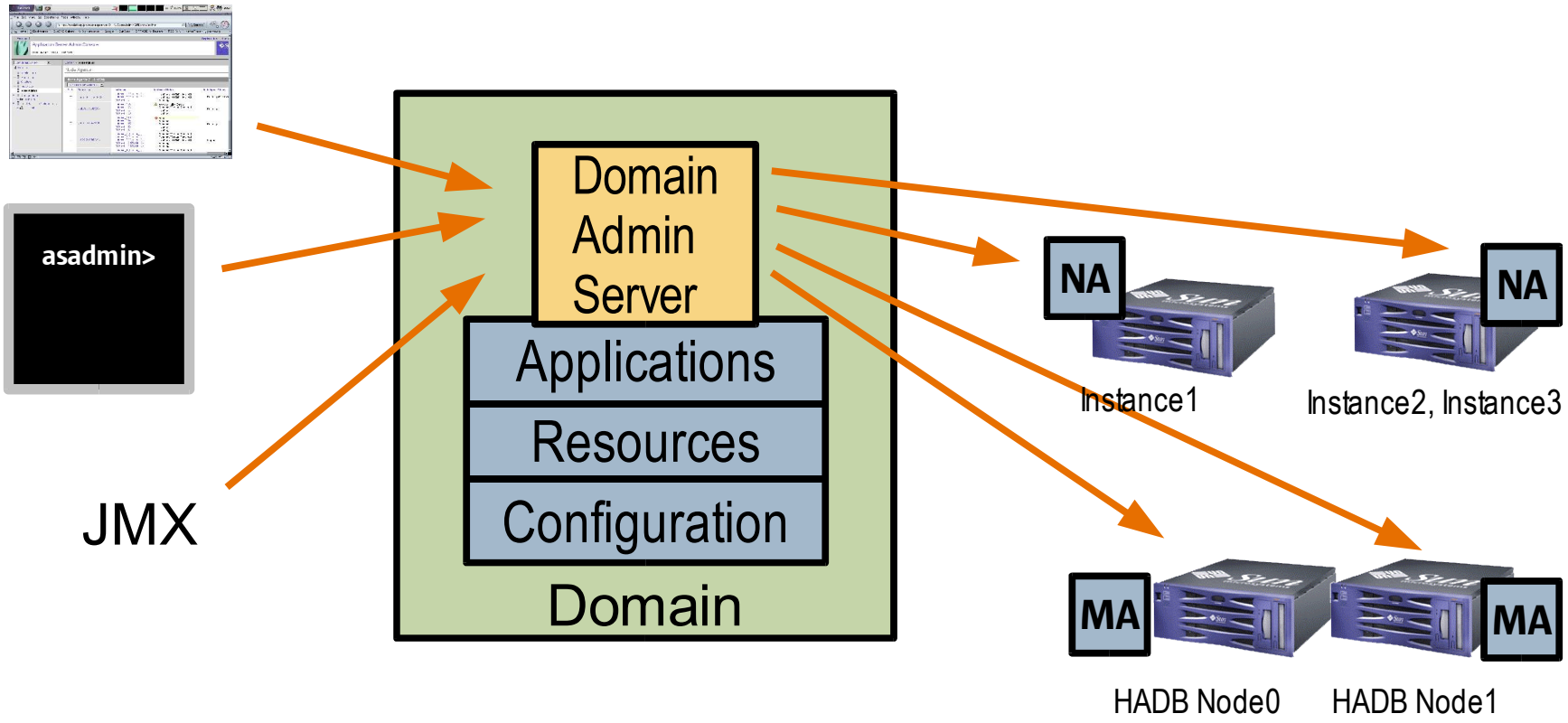
- **Escalabiliad en caliente**

Gestionabilidad

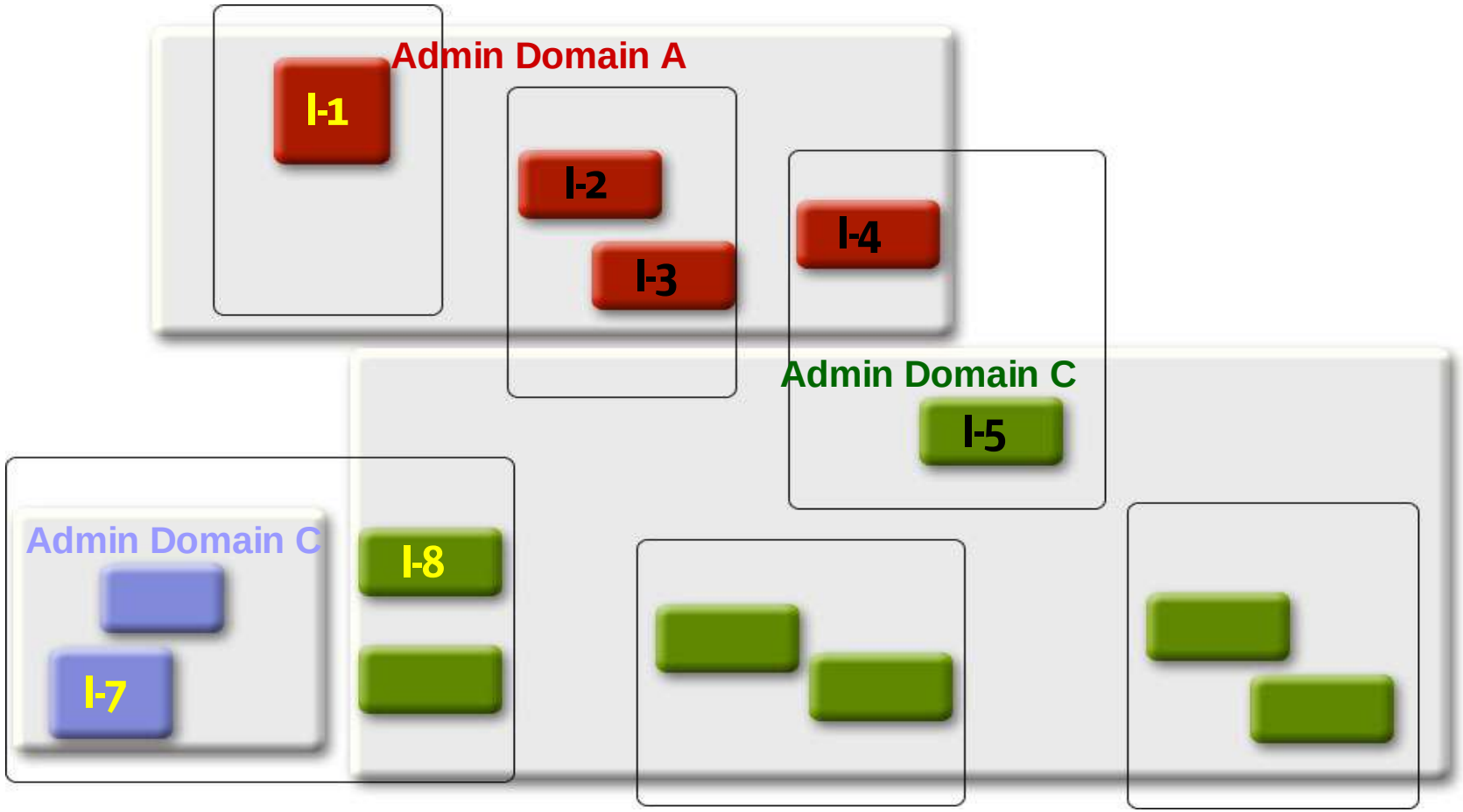
Arquitectura AppServer 8.1 SE/EE



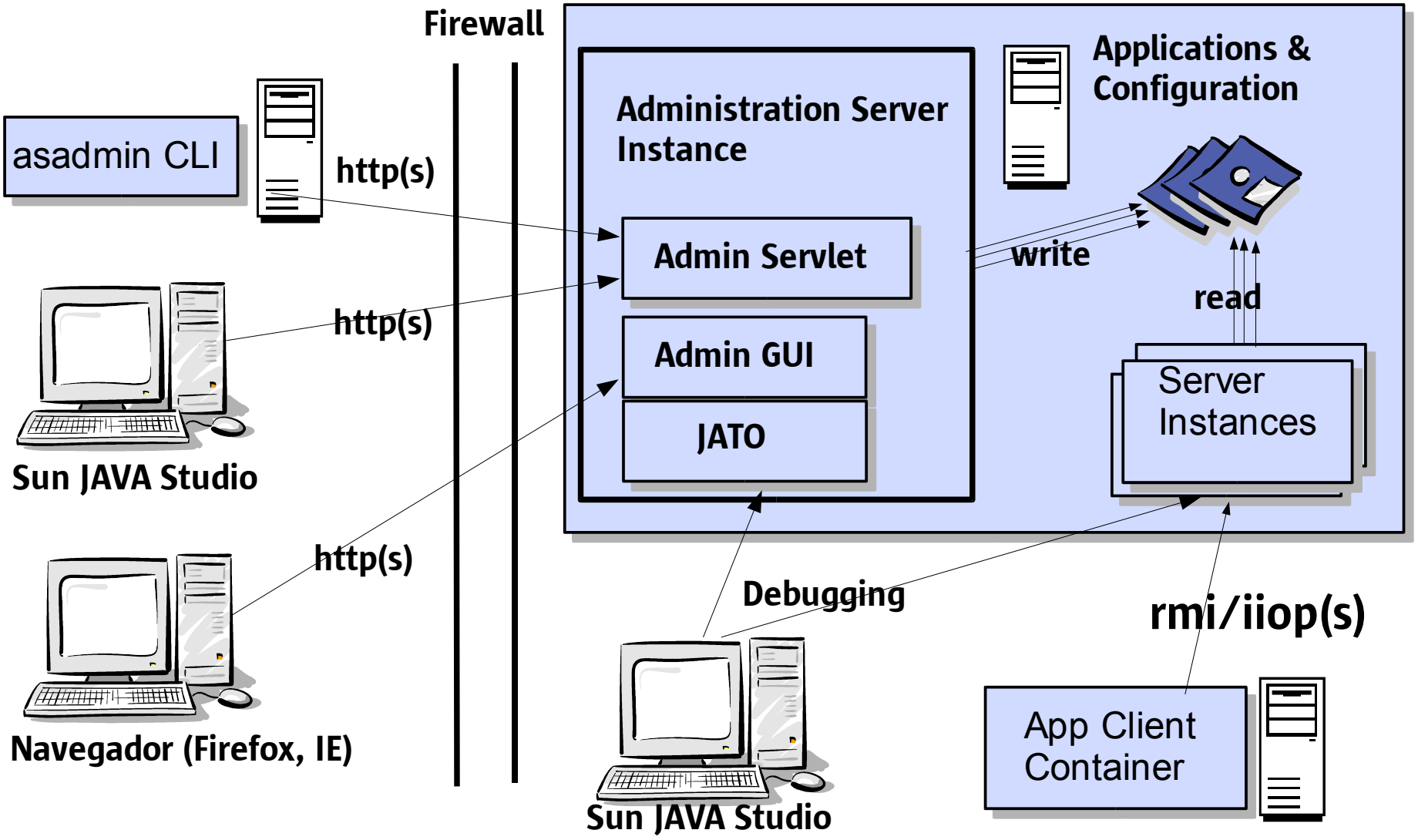
Arquitectura de Administración



App Server 8.1 EE
Dominios Administrativos



Servidor de Administración



Application Server Admin Console

User: admin Server: ilsdev8

Domain
Applications
Resources
Clusters
myCluster
ilsdev1_instance1
ilsdev2_instance2
myCluster2
Instances
Node Agents
Configurations
server-config (Admin C
default-config
ilsdev2_instance1-cor
myCluster-config
JVM Settings
Logger Settings
Deployment Setting
J2EE Containers
Java Message Serv
Security

Domain > Clusters > myCluster > ilsdev1_instance1

General | Properties | Monitor

Monitor | Applications | Resources | Transactions

Monitoring

Refresh

Use this page to observe the runtime state of Application Server components that have monitoring enabled. To enable monitoring for a component, select Configure Monitoring and set the monitoring level of the component to LOW or HIGH. To view monitoring data for a component for which monitoring has been enabled, select it from the View list. The monitoring information for the component will display below the View list. Select Refresh to update the information on the display. Monitoring information will vary depending on the component being monitored. To view monitoring information for Applications, Resources, or to freeze and rollback Transactions, select the appropriate page.

View: jvm Configure Monitoring

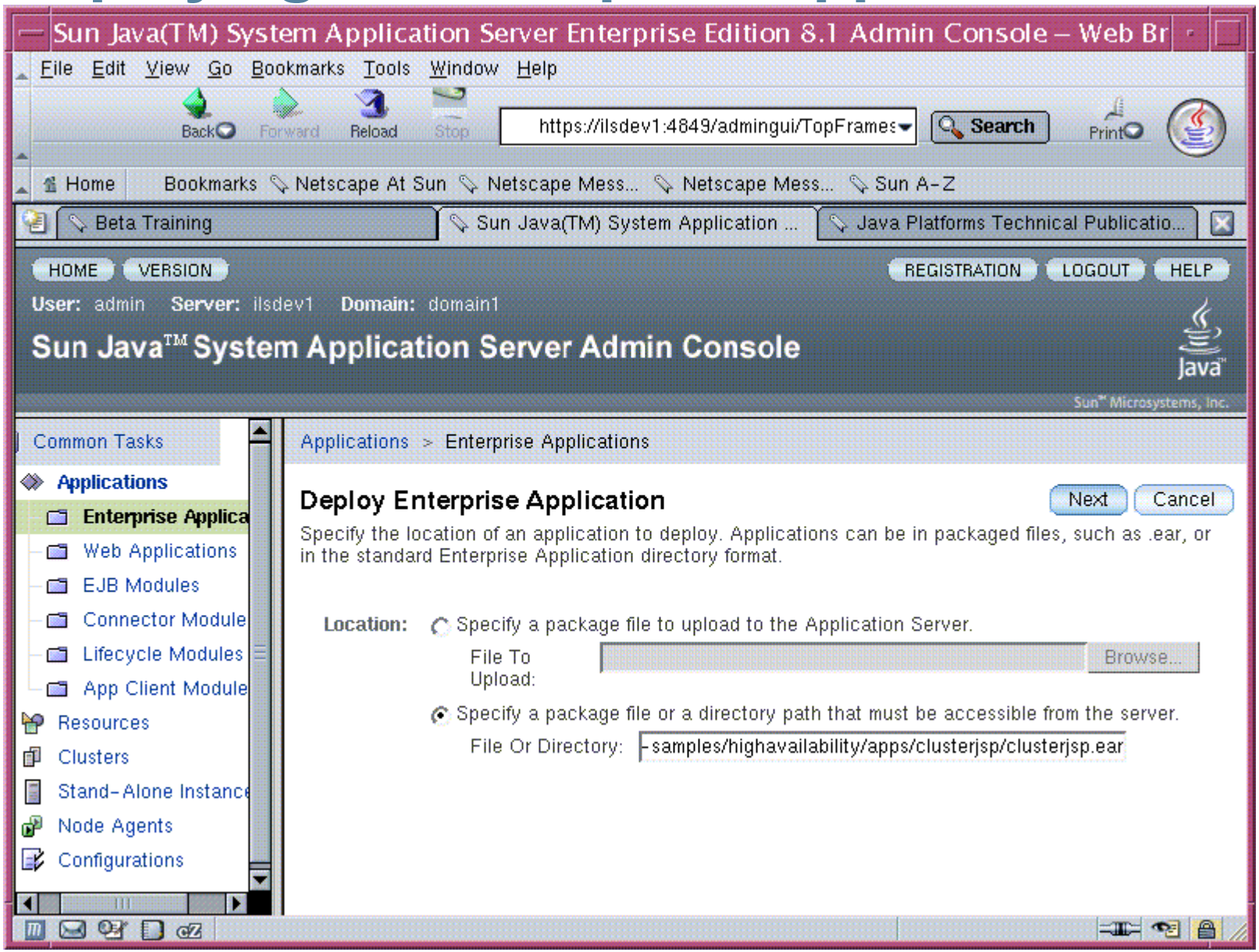
Describes JvmUpTime

Count:	188,834,867 milliseconds (2 Days, 4 Hours, 27 Minutes, 15 Seconds)
Last Sample Time:	Thu Oct 14 18:10:51 PDT 2004
Start Time:	Tue Oct 12 13:43:36 PDT 2004

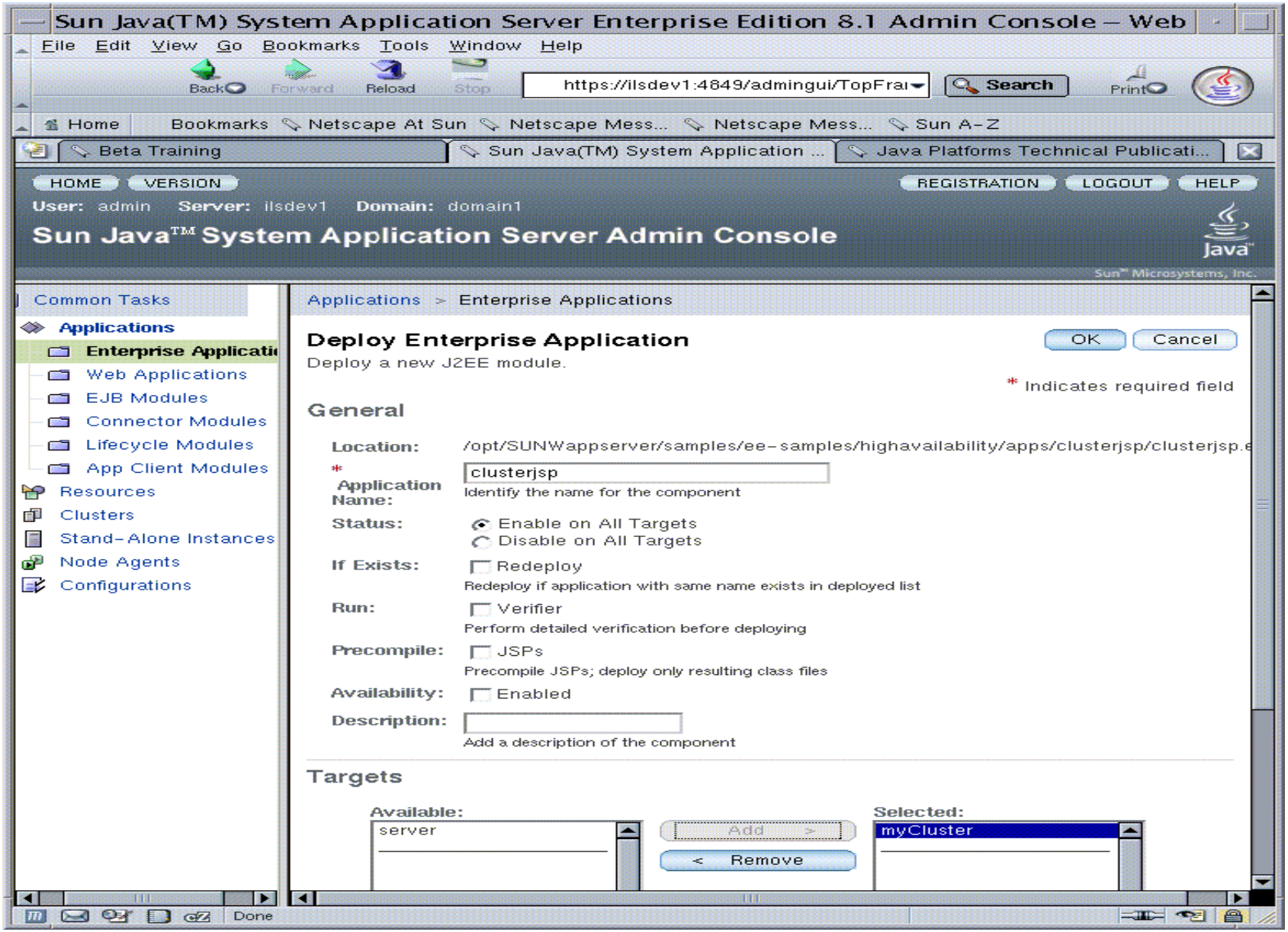
Describes JvmHeapSize

Current: 23,920,640 bytes

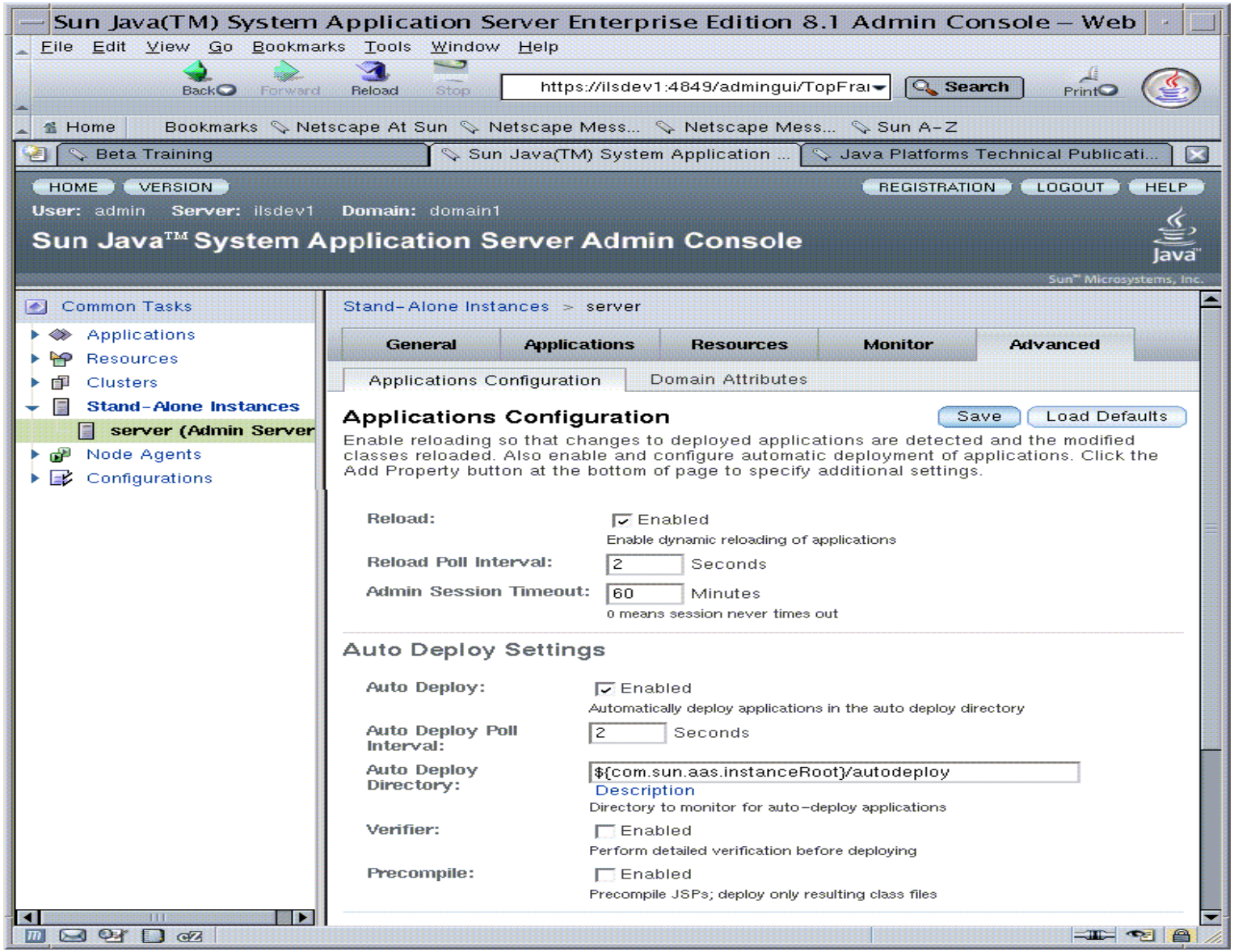
Deploying An Enterprise Application



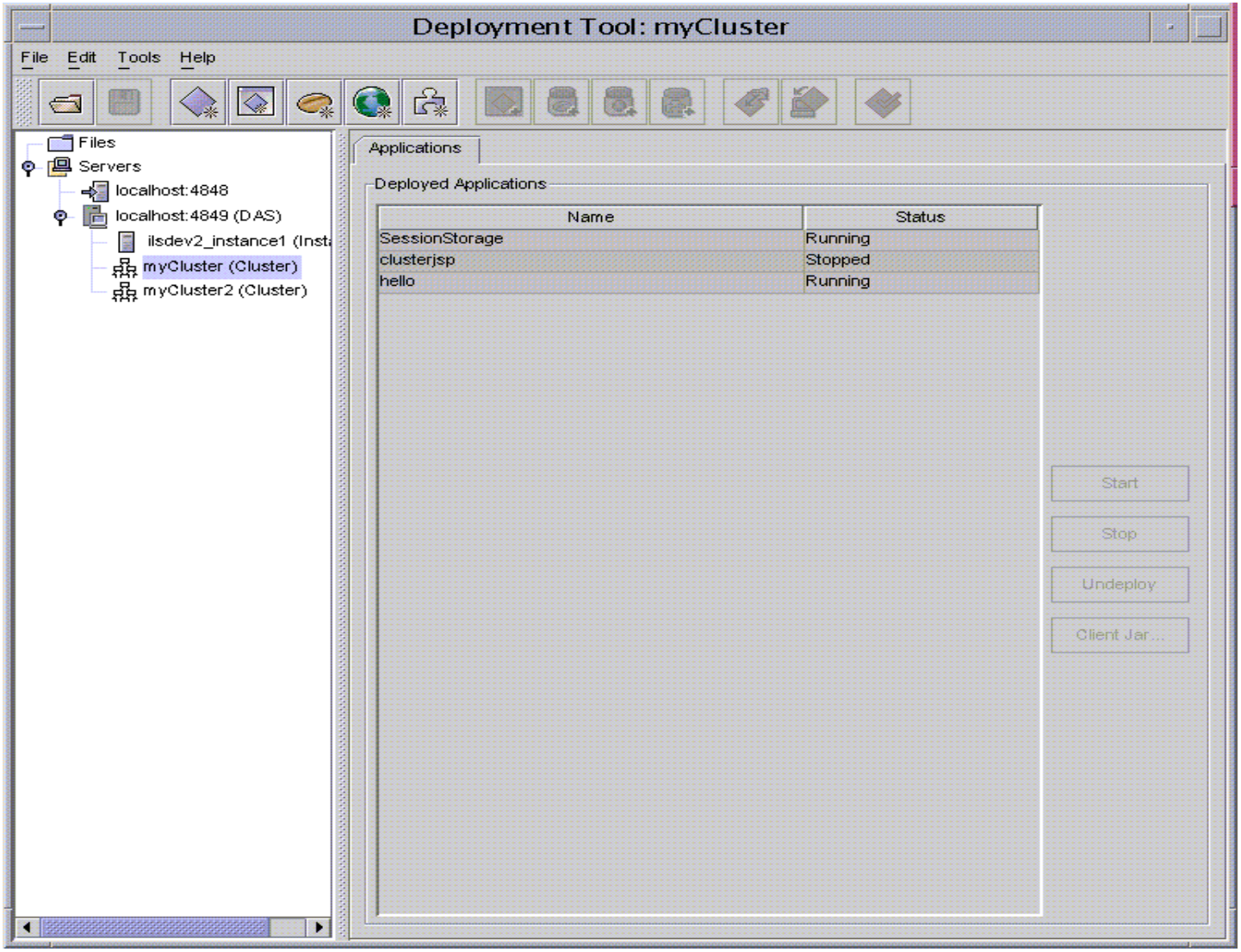
Deploying An Enterprise Application



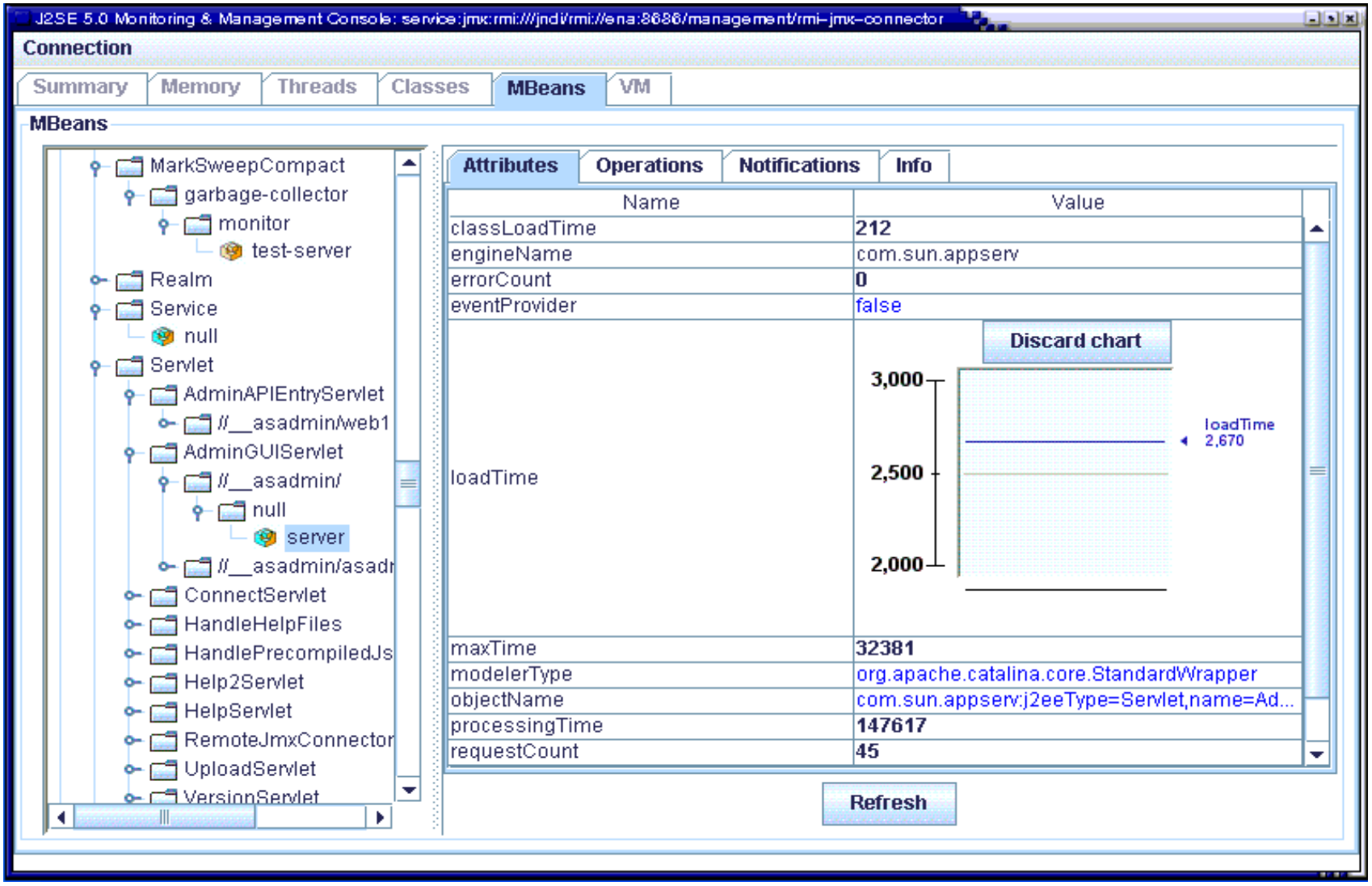
Configuring Auto Deploy



Deploytool – GUI

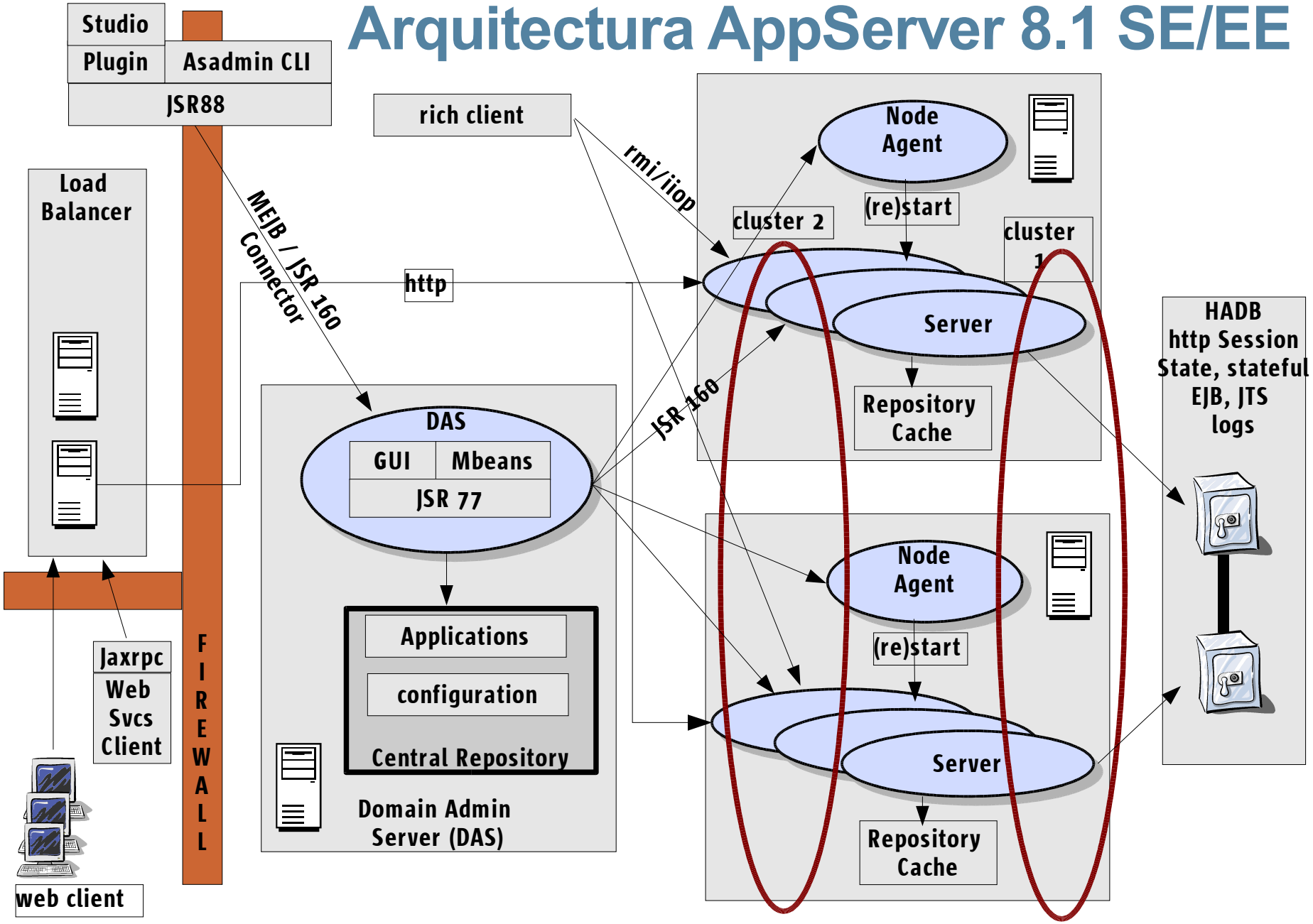


JMX support example



Monitoring Mbeans via J2SE 5.0 JConsole using JSR160 Connector

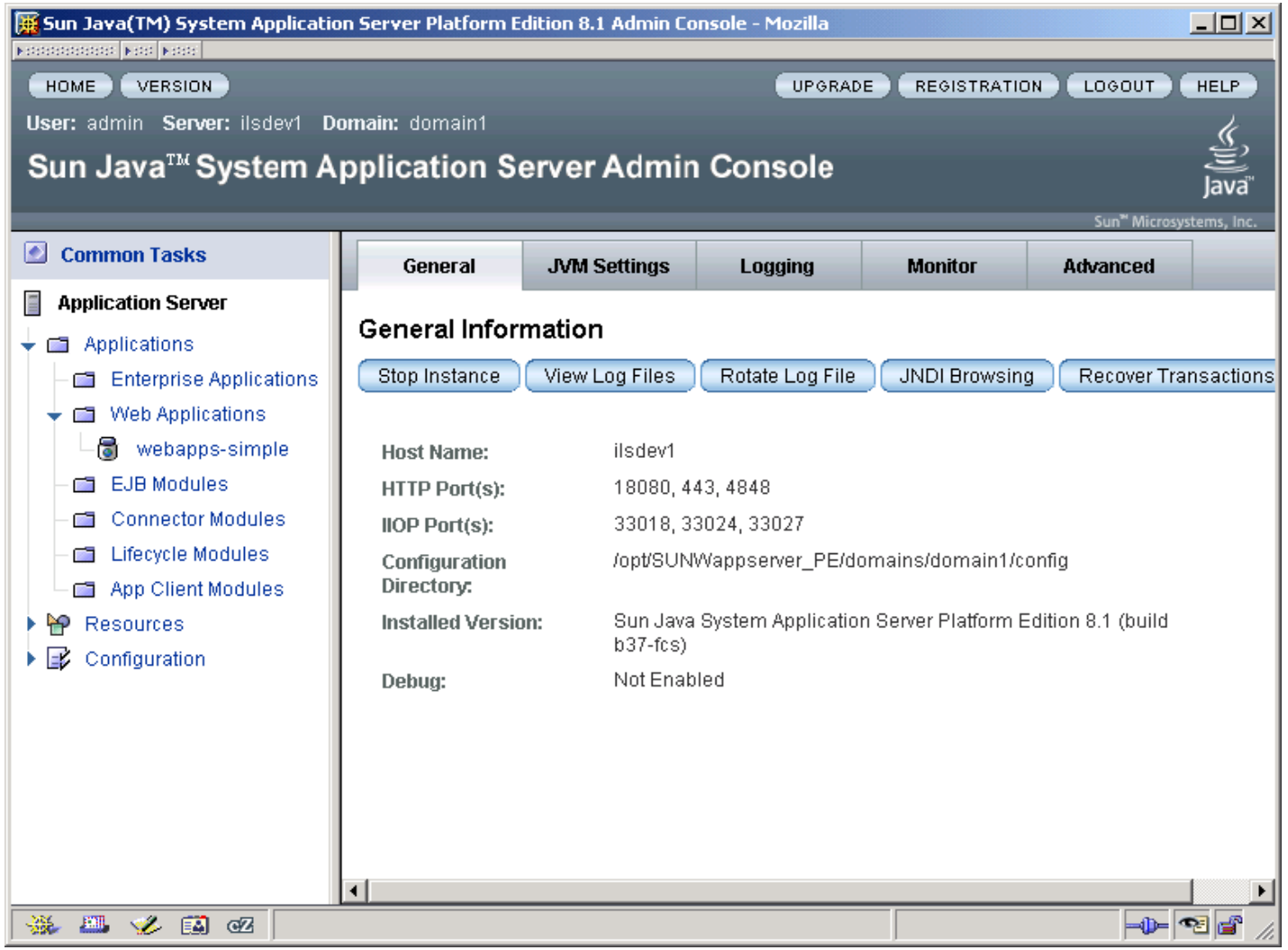
Arquitectura AppServer 8.1 SE/EE



Monitorables and Monitoring Levels

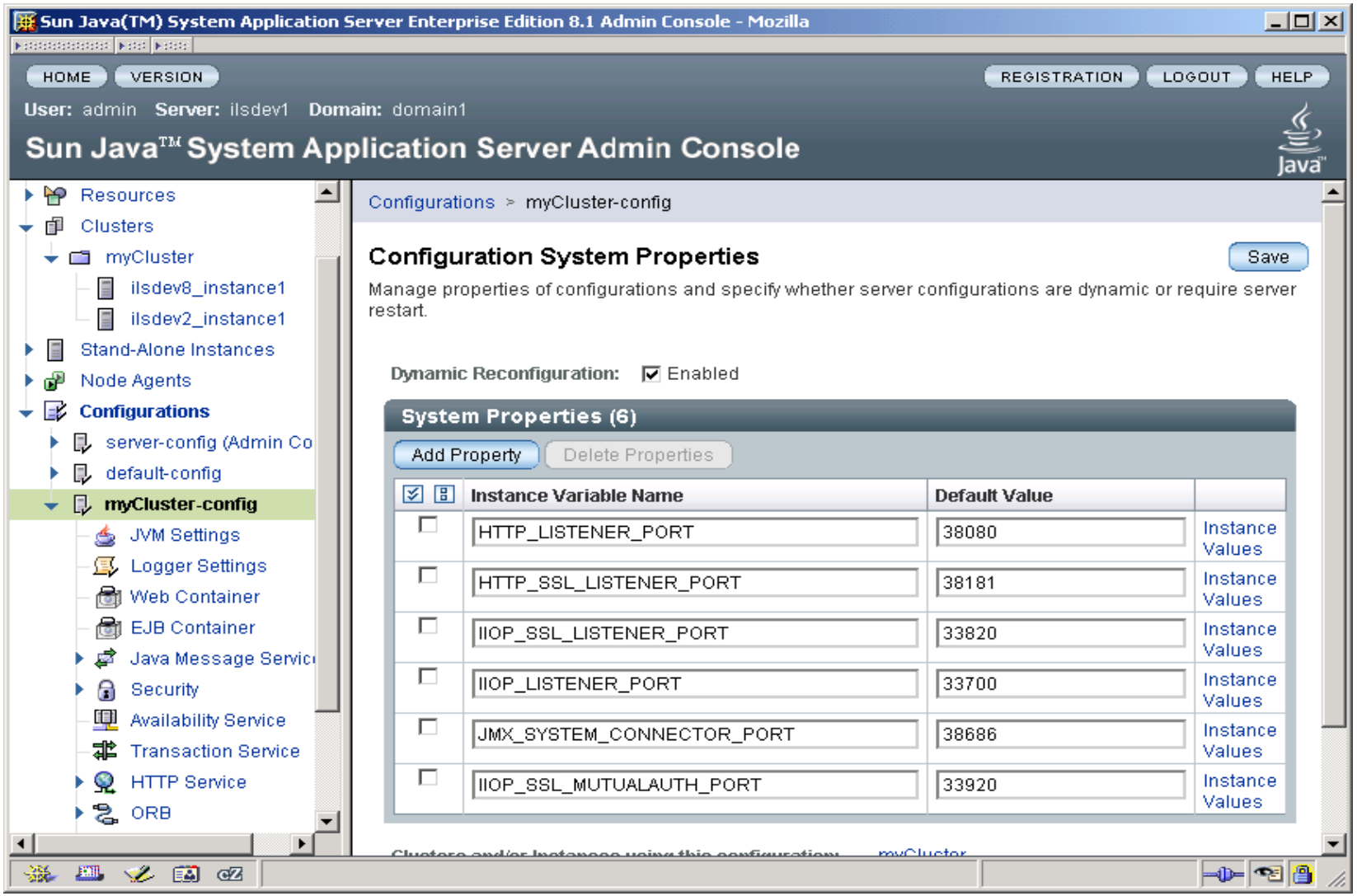
Component/Service	Monitoring Level	Monitorables
JVM	OFF/LOW/HIGH	JVM
HTTP Service	LOW/HIGH	All HTTP listeners and virtual servers
Transaction Service	LOW/HIGH	Any transaction subsystem
JMS/Connector Service	LOW/HIGH	Any JMS/Connector subsystem
ORB	LOW/HIGH	The system ORB used by the Application Server core and its connection managers
Web Container	LOW/HIGH	All deplyed servlets
EJB Container	LOW	EJBs, EJB pools, EJB caches
	HIGH	EJB business methods
JDBC Connection Pool	LOW/HIGH	All JDBC connection pools
Thread Pool	LOW/HIGH	All thread pools

Administration Console - JNDI Browsing

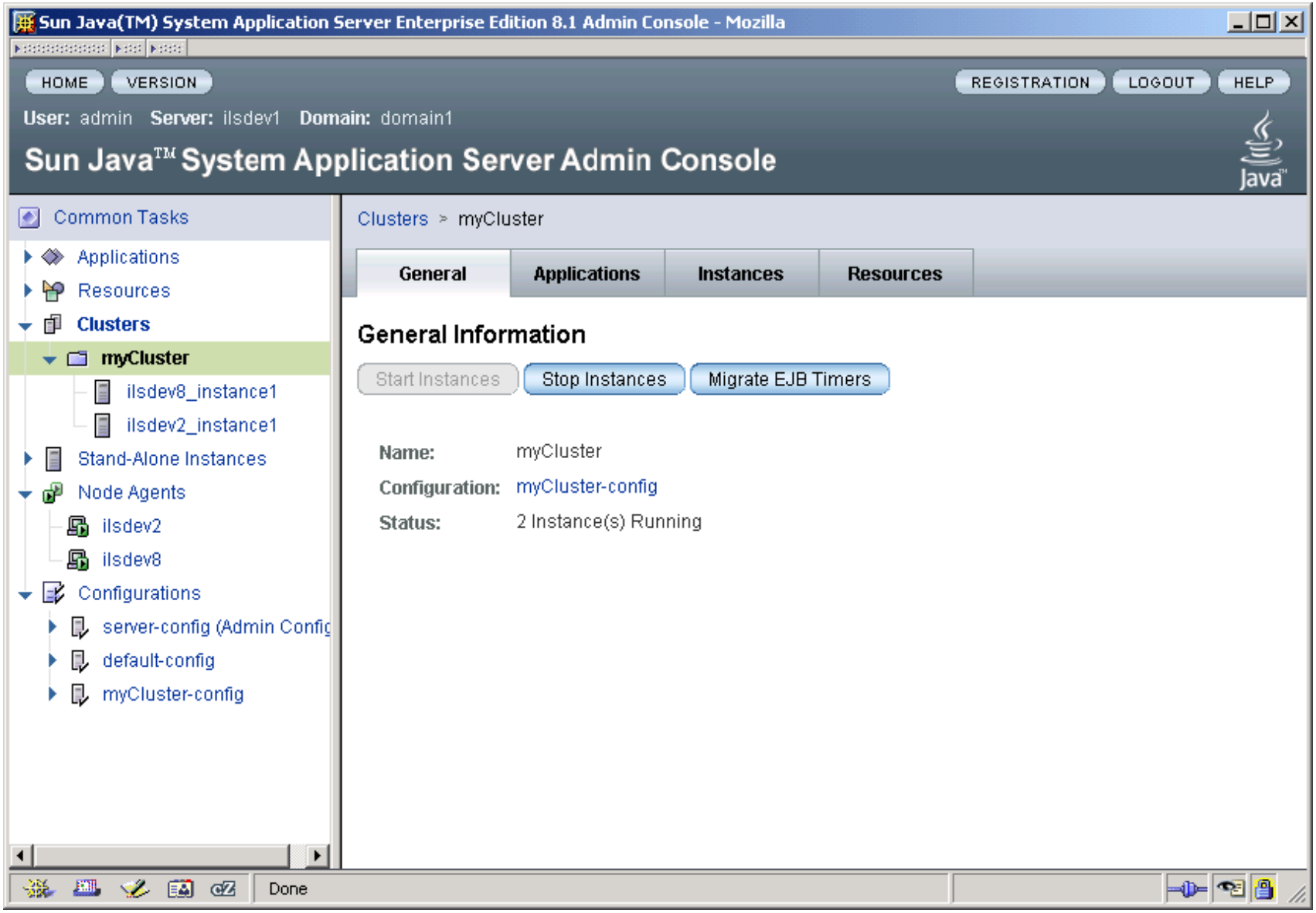


JNDI Tree Browsing

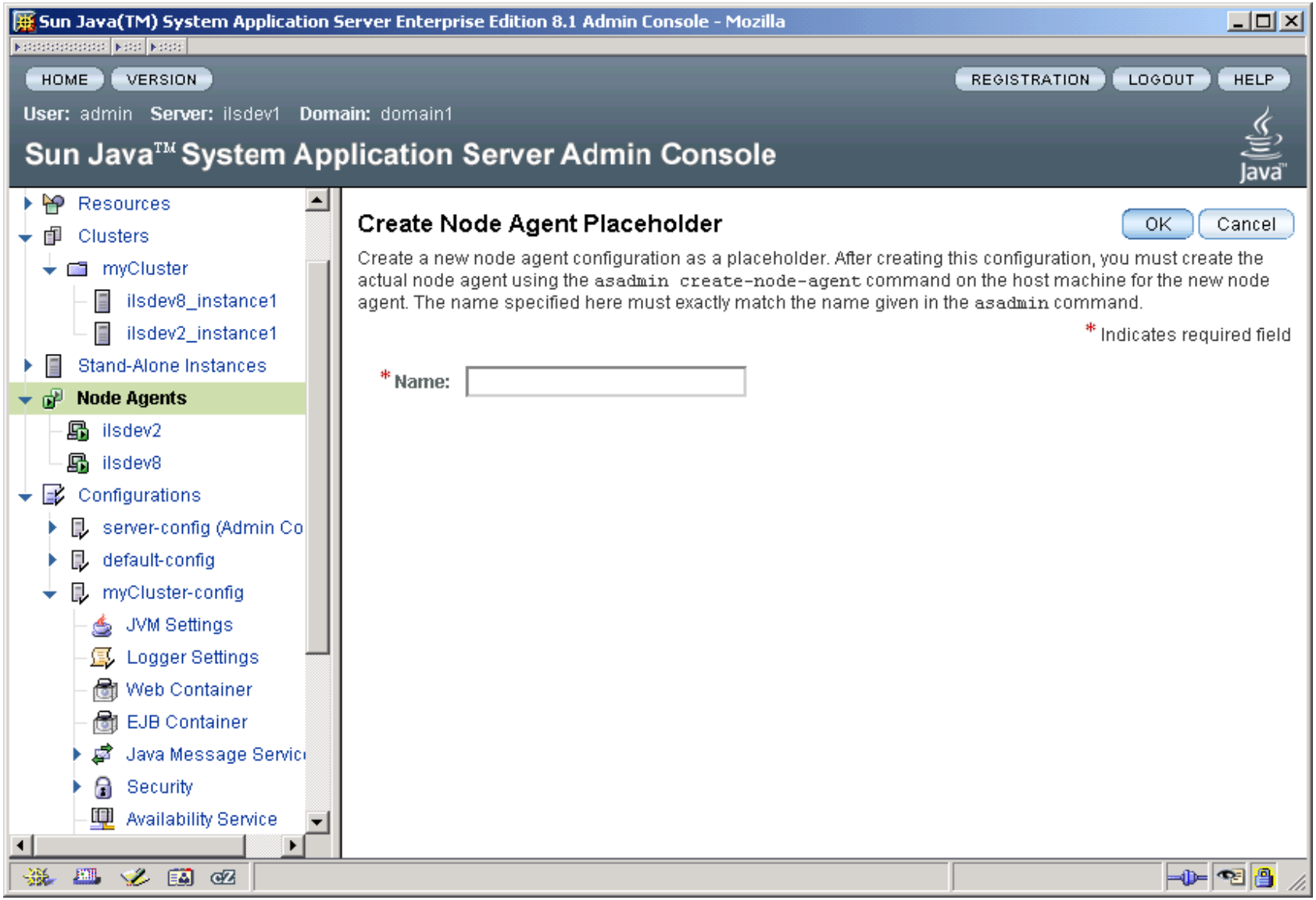
GUI - Configuring Cluster



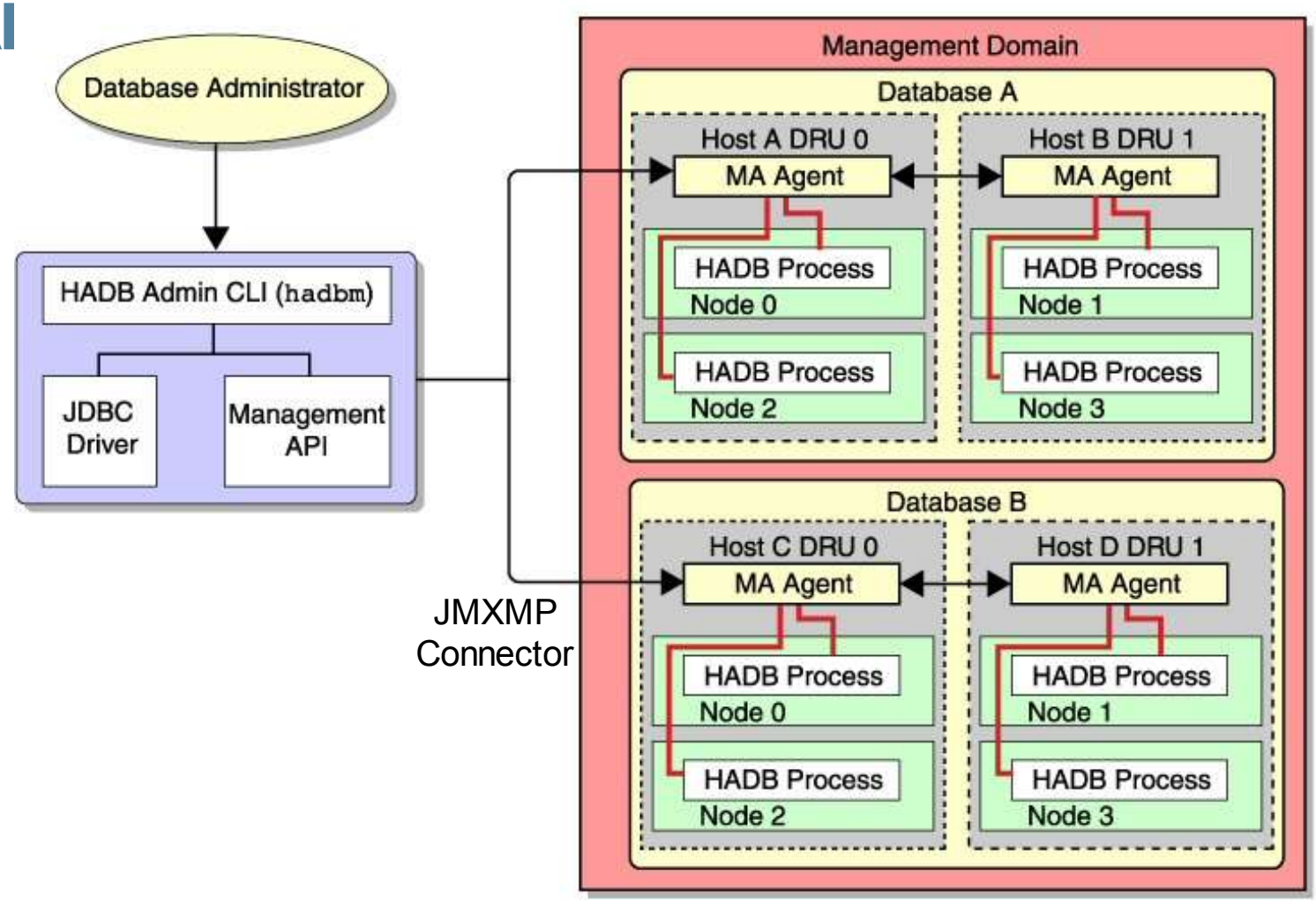
GUI - Administering Cluster



GUI – Creating NA Placeholders



New HADB Management AI



Estandares J2EE

J2EE Platform Evolution

- J2EE Platform v1.2 (Dec 1999)
  - Java Servlet and JavaServer Pages™ (JSP™) specifications-based Web presentation container
  - Enterprise JavaBeans™ (EJB™) architecture-based container
  - Declarative transaction and security support
  - Component-based application development
  - Java Message Service (JMS) asynchronous messaging
  - Both synchronous (RMI/IIOP) and asynchronous (JMS) component integration

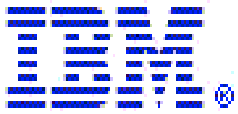
J2EE Platform Evolution

- J2EE Platform v1.3 (Sept 2001) adds:
  - EJB platform container managed persistence
  - Back-end integration via J2EE Connector Architecture
- J2EE Platform v1.4 (Nov 2003) adds:
  - Web services
    - Low-level APIs supporting web services protocols
      - Java Architecture for XML Binding (JAXB)
      - Java API for XML-based RPC (JAX-RPC)
    - Higher level support of EJB specification-based beans as web services endpoints
    - First industry support of full WS-I Basic Profile
  - Bi-directional connectors

J2EE Platform in the Market

- 28 compatible products (v1.3 or 1.4)
- Adoption by Open Source Community
  - Open source community recognizes the value of standards and compatibility
  - Market benefits from dynamism of open source developers
- Last 6 new vendors/projects
  - Apache Software Foundation
  - Caucho
  - JBoss
  - Kingdee
  - Objectweb Consortium
  - Webmethods

J2EE Platform v1.4 Compatible



IBM WebSphere Application Server Technology for Developers, Version 6.0

Oracle Application Server Containers for J2EE 10g (10.0.3)—Developer Preview

Sun Java System Application Server Platform Edition 8

Tmax Soft JEUS 5.0 Application Server

Trifork T4 Application Server

More Compatible Products Coming

New in J2EE 1.4

- JAX-RPC 1.0
- SAAJ 1.1
- JAXR 1.0
- J2EE Management 1.0
- JMX 1.2
- J2EE Deployment 1.1
- J2EE Auth Contract for Containers 1.0
- Web Services for J2EE 1.1

Updated in J2EE 1.4

- J2SE™ 1.4
- JAXP 1.2
- Servlet 2.4
- JSP™ 2.0
- EJB™ 2.1
- Connector 1.5
- JMS 1.1
- JavaMail™ 1.3

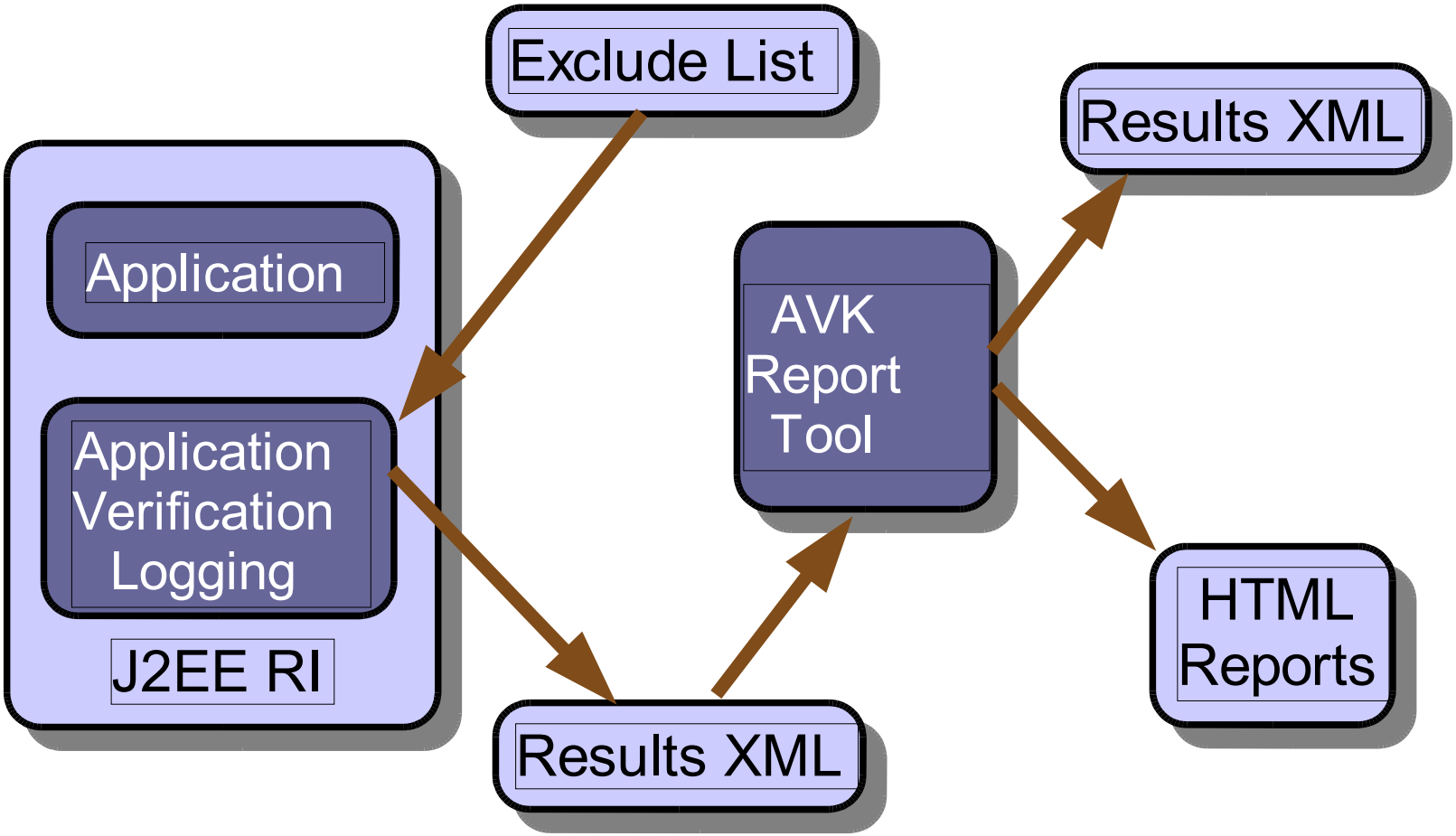
J2EE 1.4 Theme

- Allow J2EE applications to be exposed as SOAP/HTTP based Web Services
- Integrate existing Web Services
- Use JAX-RPC
- Architecture based on JSR-109

Application Verification Kit (AVK)

- Free Java™ Application Verification Kit for the Enterprise
  - > Reduces development risks
  - > Reduces software lifecycle costs
  - > Reduces migration expenses...
- Java Powered for the Enterprise Logo™
  - > Identifies portable enterprise solutions
  - > Reduces vendor lock-in
  - > Free for vendors meeting program goals

J2EE AVK Tools View



Sun JAVA System Application Server 9.0 (J2EE 5.0) Roadmap

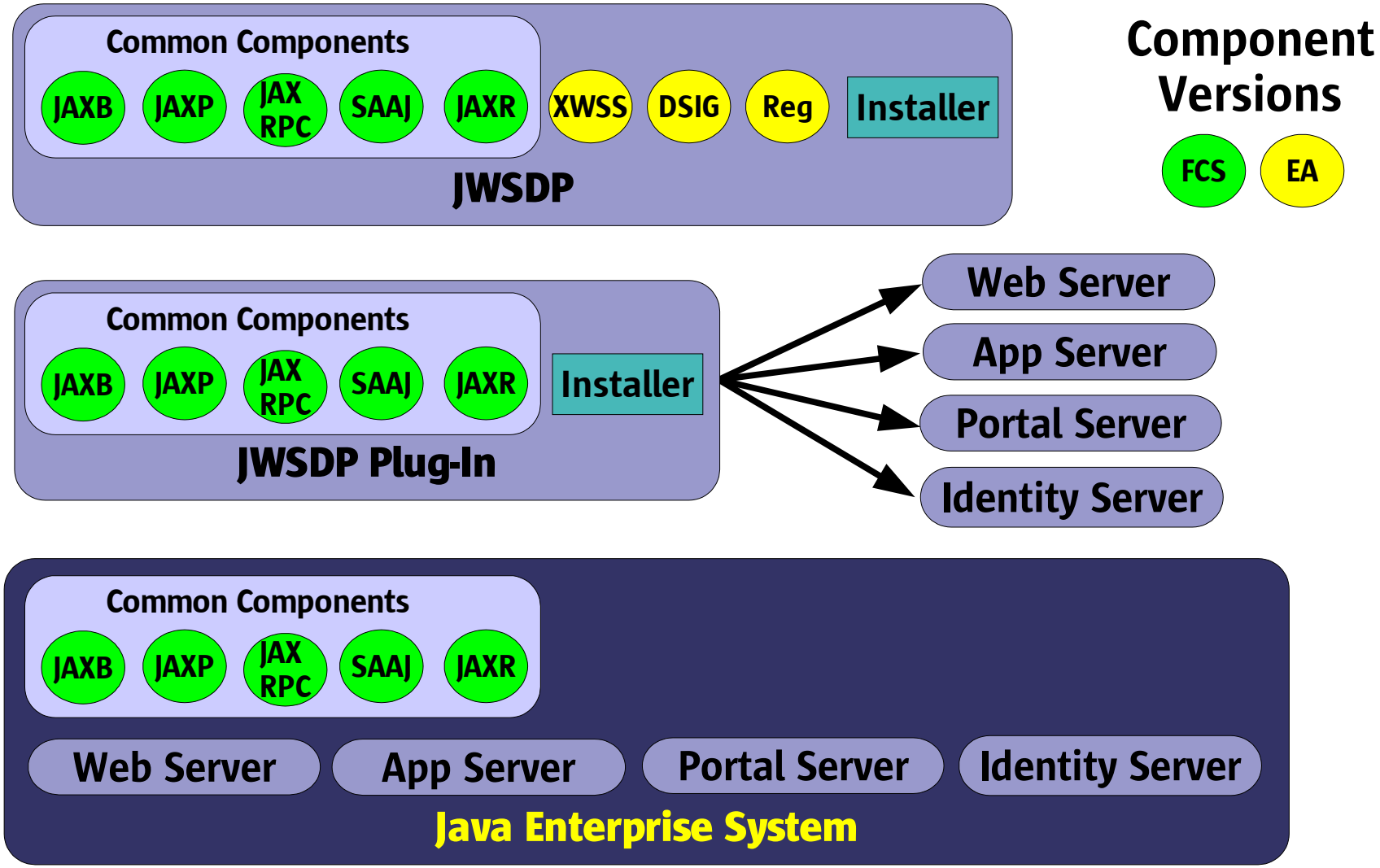
J2EE 5.0 Major Features

- Simplified web services support
- More web service standards support
- Greatly simplified EJB development
- New persistence API, effectively replacing EJB CMP
- Easy web applications with JSF

J2EE 5.0 New Contents

- JSP Standard Tag Library (JSR-52)
- JSP 2.1 (JSR-245)
- StAX (JSR-173)
- Web Services Metadata (JSR-181)
- JAXB (JSR-222)
- JAX-RPC 2.0 (JSR-224)
- Common Annotations (JSR-250)
- JavaServer Faces (JSR-252)
- New persistence API / EJB 3.0 (JSR-220)

JWSDP: Java Web Services Developer Pack Release Vehicles



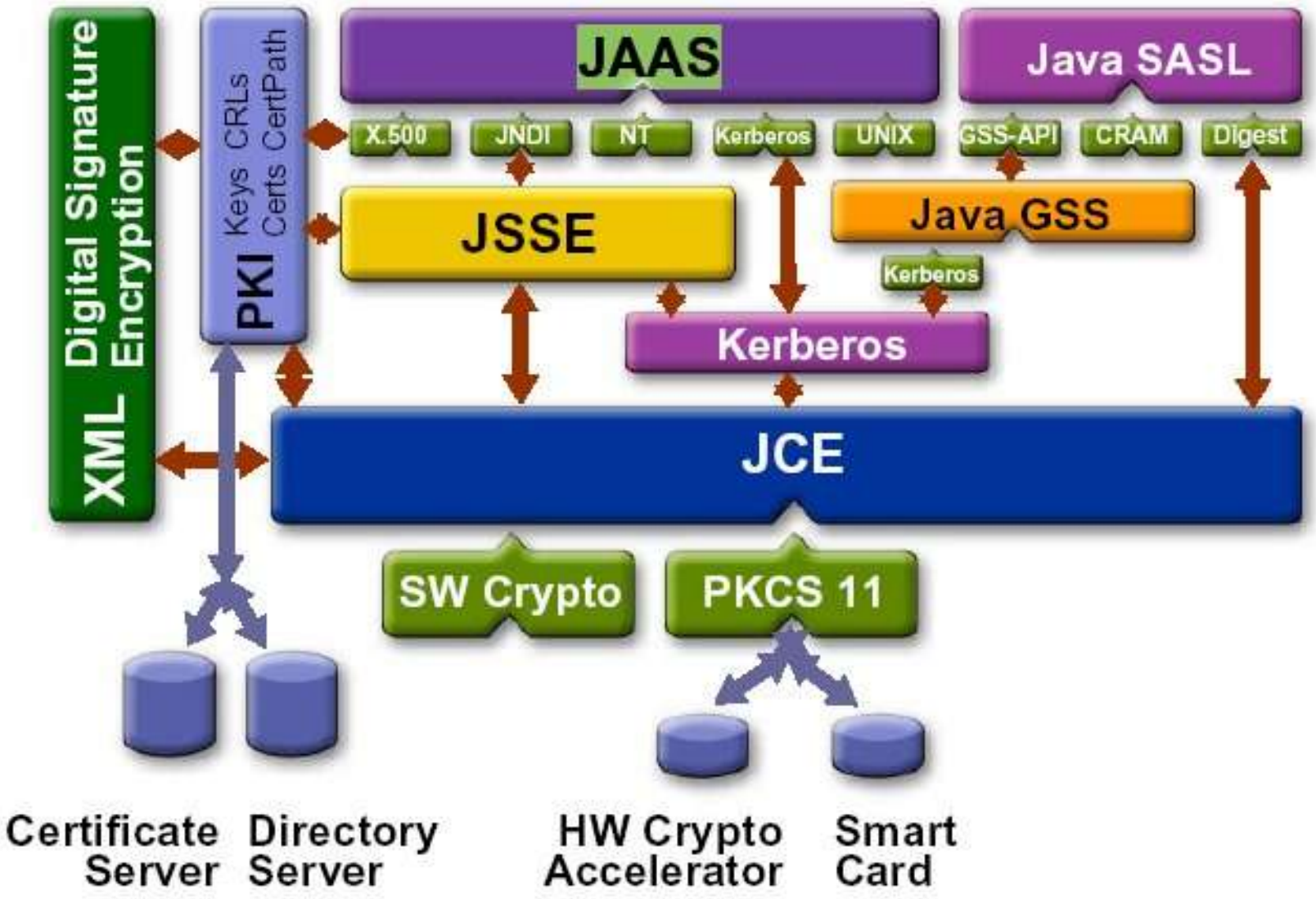
J2EE 6.0 Contents (possible)

- Updates of all core specs
- Additional specs
  - JSR-168 Portlets
  - JSR-196 Authentication SPI for containers
  - JSR-207 Process Definition for Java
  - JSR-208 Java Business Integration
  - JSR-225 Xquery API
  - JSR-227 A Standard Data Binding and Access Facility
  - JSR-235 Service Data Objects
  - JSR-261 Web Services Addressing

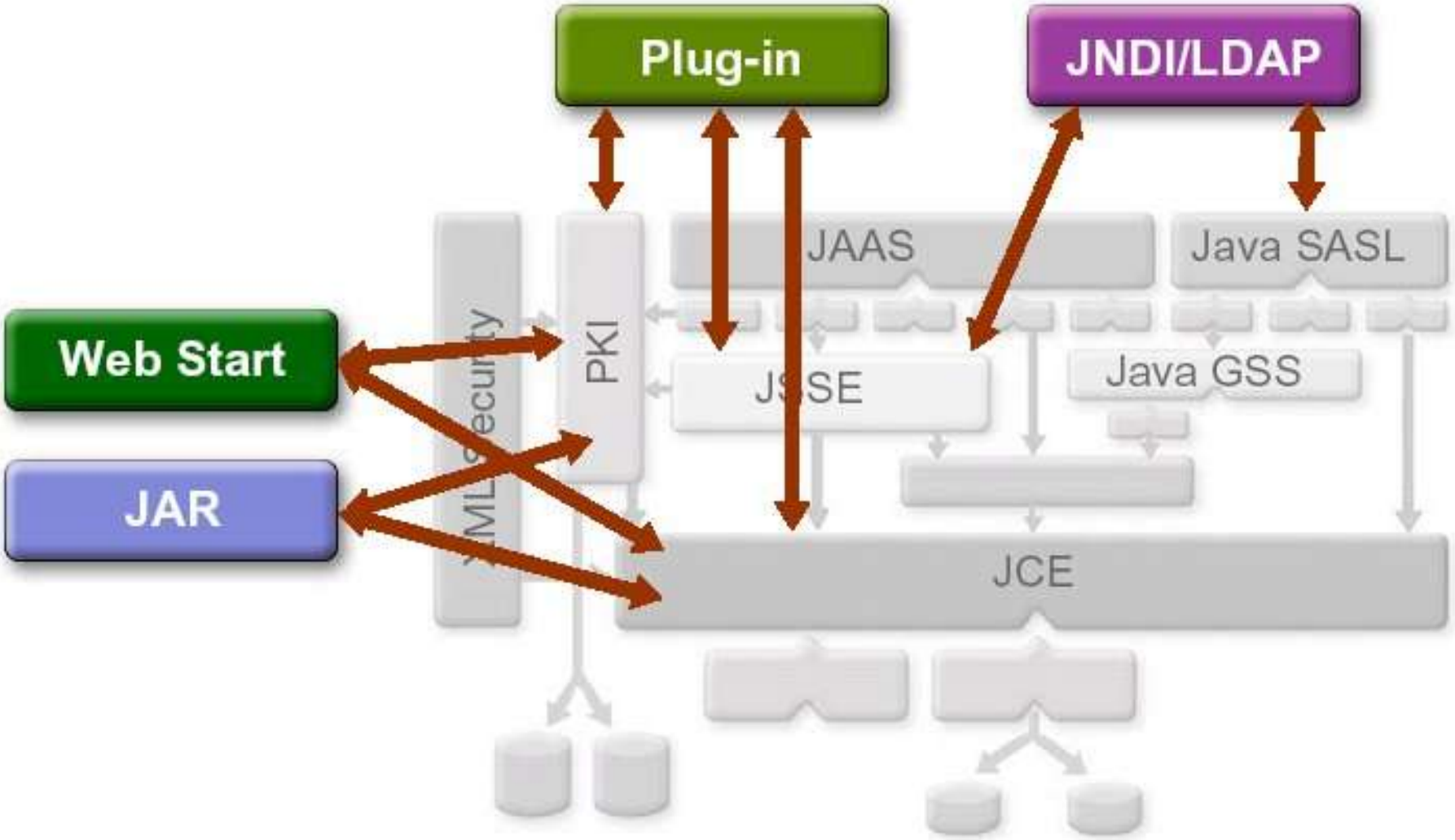

App Srvr 6
• J2EE 1.2 based
• Upgrade from Netscape/Kiva & NetDynamics
CY 02
App Srvr 7.x
• J2EE 1.3
• Developer Friendly
• Web Services & JSP Performance
• Persistent Failover - Leveraging Sophisticated 59's available clustra tech
• Horizontal & Vertical scalability
• Sun Java Enterprise System
• Integrated MQ and broker failover
• Development and deployment tools support, Studio Integration
• Basis for Portal, Integration, Identity
CY 03
App Srvr 8.x
• J2EE 1.4
• WS-I Basic Profile
• WS Security
• Developer Ease of Use (Tools, JSF ..)
• Ease of management and monitoring (JMX, MBeans)
• Excellent price/performance
• Enhanced Availability
• JMS broker failover
• Enhanced Horizontal and Vertical Scalability
• Sun Java Application Platform Suite (Java ES)
• Solaris 10
• J2SE 5
CY 04/05
App Srvr 9
• J2EE 5
• SOA runtime
• Java Business Integration
• Federated Identity
• Composite Applicatiion development
• Composite application monitoring
• First class webservices management
• Policy based administration
• Adaptive Clustering
• Hetrogenous clusters
• Excellent price/performance
• Enhanced Availability
• Full JMS message failover
• Enhanced Horizontal and Vertical Scalability
• TRUSTED Solaris, Zones
CY 06

Identidad y Seguridad

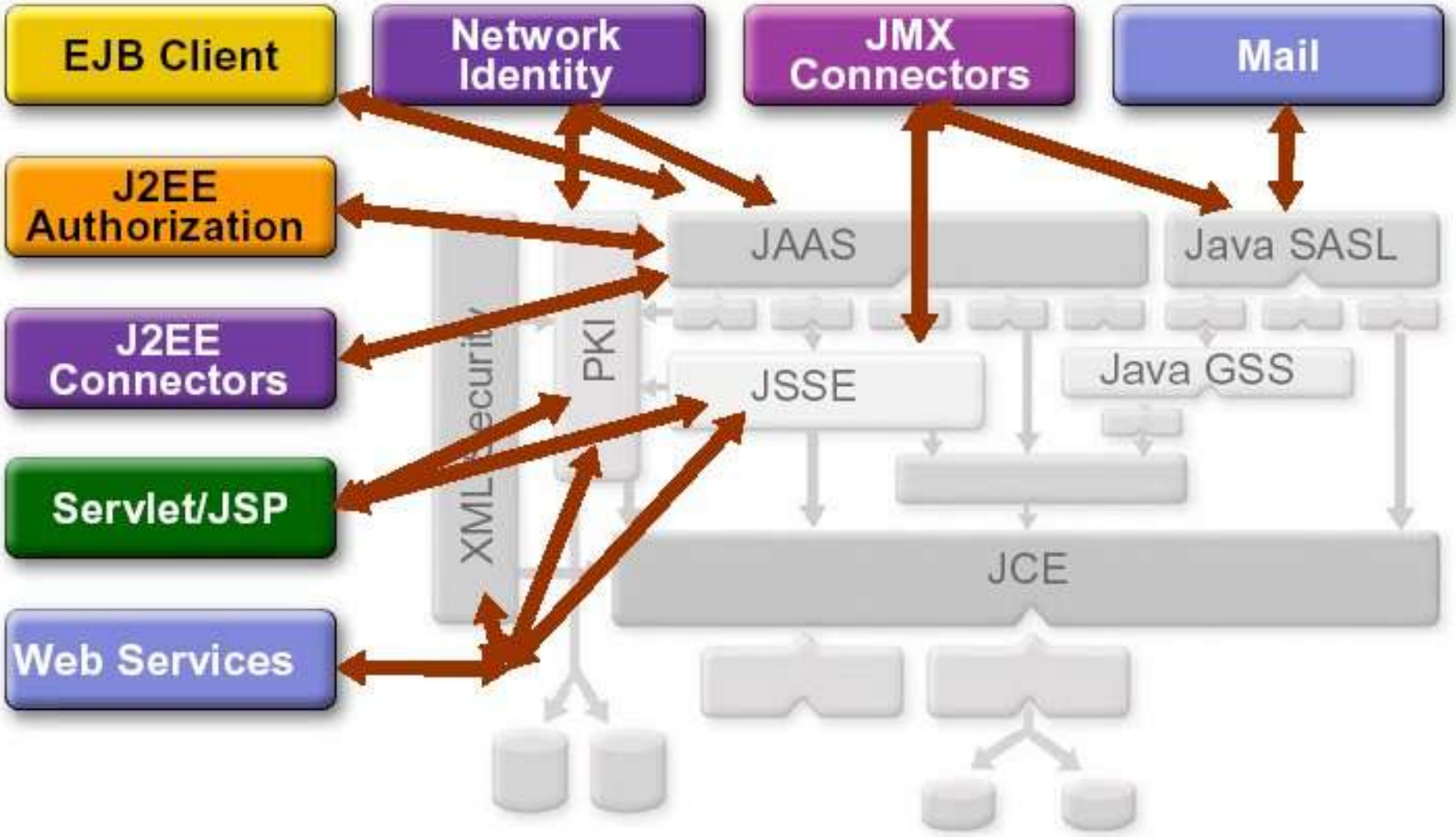
JAVA Security Architecture



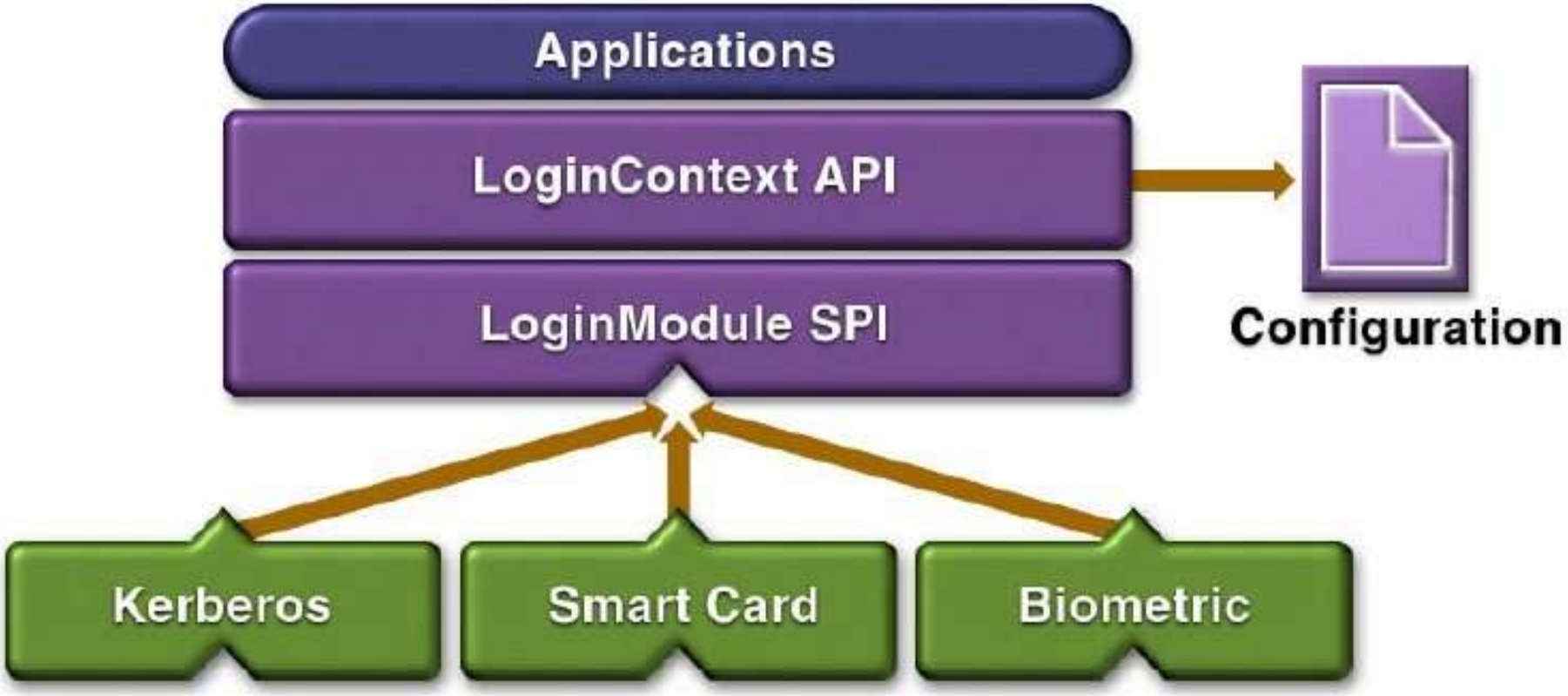
Security in J2RE / J2SE



Security in J2EE



Java Authentication & Authorization Service (JAAS)



SJS App Server 7.x Security

- Full J2EE Security model compliance
- EJB v2.0 Security model compliance
- Java Servlet 2.3 Security model
- Single Sign-On (SSO)
- Java RMI-IIOP Client Security
- Support for file, LDAP, Certificate & solaris Authentication realms
- Declarative security with XML Role Mapping

server.policy

- Contains security manager permisions
- One for AS instance
- Default minimal permissions for Apps
- Disable security manager by removing the following in server.xml
- `<jvm-options> -Djava.security.policy=.../config/server.policy </jvm-options>`

Security information

- User Information: Name, password, and other information
- Realms:
  - File - $INSTANCE/config/keyfile
  - Certificate - User identities in the Sun ONE AS security context
  - LDAP - User information is in LDAP directory
  - Solaris - Underlying Solaris user database
  - Custom – JDBC Realm
- Security roles in Application descriptors

server.xml default realm

```
<security-service default-realm="file" anonymous-role="ANYONE" audit-
  enabled="false">
    <auth-realm name="file"
  classname="com.iplanet.ias.security.auth.realm.file.FileRealm">
      <property name="file"
  value="/var/opt/SUNWappserver7/domains/domain1/instancia1/config/keyf
  ile"/>
      <property name="jaas-context" value="fileRealm"/>
    </auth-realm>
    <auth-realm name="ldap"
  classname="com.iplanet.ias.security.auth.realm.ldap.LDAPRealm">
      <property name="directory" value="ldap://localhost:389"/>
      <property name="base-dn" value="o=isp"/>
      <property name="jaas-context" value="ldapRealm"/>
    </auth-realm>
    <auth-realm name="certificate"
  classname="com.iplanet.ias.security.auth.realm.certificate.Certificat
  eRealm">
</auth-realm>
<auth-realm name="solaris"
  classname="com.iplanet.ias.security.auth.realm.solaris.SolarisReal">
<property name="jaas-context" value="solarisRealm"/>
</auth-realm>
  </security-service>
```

Basic Auth sample : web.xml

```
  <security-constraint>
      <web-resource-collection>
        <web-resource-name>basic secuity test</web-resource-name>
        <url-pattern>/*</url-pattern>
      </web-resource-collection>
      <auth-constraint>
         <role-name>staffmember</role-name>
      </auth-constraint>
  </security-constraint>

  <login-config>
      <auth-method>BASIC</auth-method>
      <realm-name>basic-file</realm-name>
  </login-config>

<!-- replace last section with this section to run CLIENT-CERT
   sample.
  <login-config>
      <auth-method>CLIENT-CERT</auth-method>
      <realm-name>basic-file</realm-name>
  </login-config>
```

Form Auth sample : web.xml

```
<security-constraint>
...

    <auth-constraint>
      <role-name>staffmember</role-name>
    </auth-constraint>
  </security-constraint>

  <login-config>
    <auth-method>FORM</auth-method>
    <realm-name>default</realm-name>
    <form-login-config>
      <form-login-page>/login.jsp</form-login-page>
      <form-error-page>/error.jsp</form-error-page>
    </form-login-config>
  </login-config>

  <security-role>
    <role-name>staff</role-name>
  </security-role>
```

Basic Auth sample : application.xml

```
<application>
  <display-name>SecurityBasicApp</display-name>
  <description>Application description</description>
  <module>
    <web>
      <web-uri>WebBasicApp.war</web-uri>
      <context-root>basic</context-root>
    </web>
  </module>

  <security-role>
    <description>null</description>
    <role-name>staffmember</role-name>
  </security-role>
</application>
```

Basic Auth sample : sun-application.xml

```
<sun-application>

 <security-role-mapping>
    <role-name>staffmember</role-name>
    <principal-name>EMAILADDRESS=j2ee, CN=j2ee, UID=j2ee, OU=app,
  O=Sun, C=US</principal-name>
    <principal-name>j2ee</principal-name>
    <group-name>staff</group-name>
 </security-role-mapping>

</sun-application>
```

Single Sign-On Operation

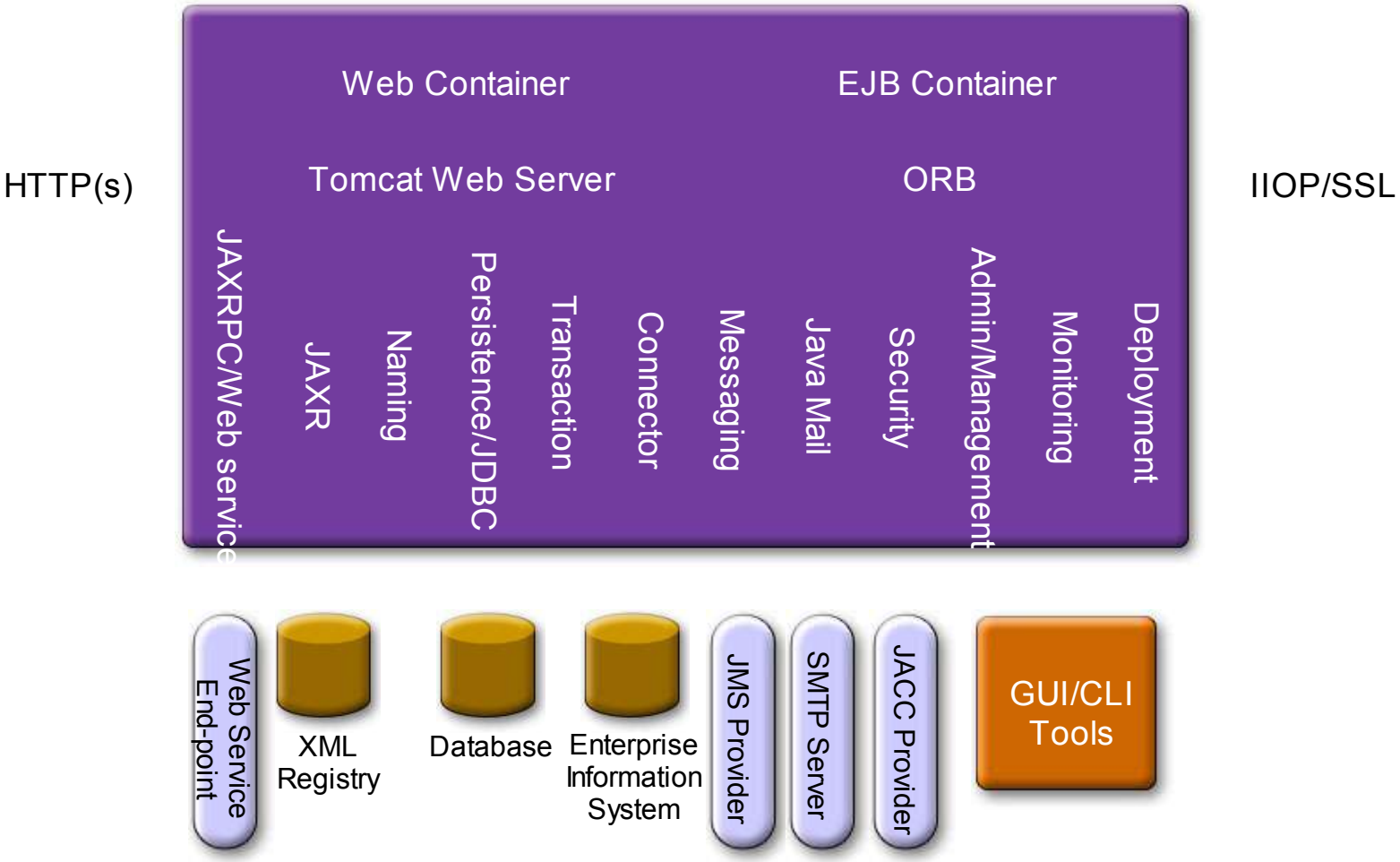
- User authenticates once to access multiple resources
- Policy Agent on Web or Application server intercepts resource requests and enforces access control
- Client is issued SSO token for session validation with session service
- SSO Token
  - > Web based applications use browser session cookies or URL rewriting to store SSO token
  - > Non web applications use the SSO API to obtain the SSO token to validate the user s identity Operation in Principle

Sun Java System Access Manager

- Single Sign-On (SSO)
- JAAS authentication framework (from 6.0)
- Conditional and referral policies (from 6.0)
- SAML and liberty enablement (from 6.0)
- Identity administration (users, policies and services)
- Delegated administration
- Centralized and configurable logging (from 6.0)
- Additional agents

AS8 Architecture Overview

S1AS8 PE Architecture Diagram



S1AS8 PE Security Components

- Authentication – Java Authentication & Authorization Service (JAAS) Based
- Authorization – Java Authorization Contract for Containers (JACC) Based
- Audit
- IIOP Security – Common Secure Interoperability version2 (CSIv2) based

Architectural Differences between S1AS PE 7 and 8

- Removal of NSS API's
- Incorporation of JSSE
- NSS cert DB replaced by JSSE keystore
- Authorization Changed by incorporated JACC
- Web core replaced by Tomcat
- Modifications to the Audit Subsystem
- Introduction of realm-per-app for feature

Architectural Similiarities between S1ASPE 7 & 8

- Authentication almost similar to 7
  - Except addition of realm-per-app feature
- Common Secure Interoperability version 2 (CSIv2) for IIOP Security is unchanged

AS8 Major Subsystems

Authentication Subsystem

- Codebase located under com/sun/enterprise/security/auth/-
- Authentication based on JAAS API's
  - Jaas LoginContext class performs authentication on client and server side
  - $SERVER_INSTALL/domains/domain_name/config/ login.conf configures Jaas LoginModules (useful for adding debug flags). Look at Jaas API's.
  - LoginModules understand Authentication Realms

Authentication Subsystem (Contd)

- Authentication Realms
  - File
    - Keyfile is the username/password database
    - Located under config/keyfile
  - Certificate
    - keystore.jks is the keystore with private keys
    - cacerts.jks is the trustore
    - Located under config/keystore.jks and config/truststore.jks
    - Check these files in case of SSL related issues
  -

Authentication Subsystem (Contd)

- Solaris
  - Manually configured in domain.xml
- LDAP
  - Manually configured in domain.xml
  - LDAP Dynamic Group
- Realms supported via Realm apis present under com/sun/enterprise/security/auth/realm

Authorization Subsystem

- Changed entirely between 7 & 8
- Incorporates JACC API's
- Two distinct components
  - Appserver specific component
  - Provider specific component

Authorization Subsystem (Contd)

- Appserver Specific component
  - Codebase located under
    - com/sun/enterprise/security/acl
    - com/sun/enterprise/security/application
    - com/sun/enterprise/security/authorize
    - com/sun/web/security
  - Responsible for translating the DD security specific information and calling JACC provider subsystem.
  - Bridge between EJB/Web Containers and JACC.

Authorization Subsystem (Contd)

- Provider Specific Component (JACC Provider)
  - Under a separate module called jacc-provider
  - Implements JACC Specification
  - Converts DD security specific information to java policy files.
  - 1 Policy file per application
  - Policy file located under $SERVER_INSTALL/domains/domain_name/generated/policy/
  - Policy for server = Sum of all policy files + server.policy file.

Authorization Subsystem (Contd)

- Loads policy into the server at deployment time
- At runtime delegates policy decision to JDK policy implementation
- Jacc provider specified in domain.xml

Audit Subsytem

- Improvements over previous audit subsystem.
- Hooks for user specified audit classes
- Audit classes specified in domain.xml
- User can extend com/sun/appserv/security/AuditModule to add his own modules
- Default implementation under com/sun/enterprise/security/audit

IIOP Security (CSIv2)

- IIOP Security or CSIv2 – no changes between 7 & 8
- Codebase under com/sun/enterprise/iiop/security
- Implements the CSIv2 protocol – wire level protocol
- Marshals/unmarshals security credentials

Single Sign On

- Default SSO turned on
- Modifications in domain.xml: sso-enabled to turn it off

Configuration for Security

- Admin asadmin commands:
  - Audit module configuration:
    - create-audit-module
    - delete-audit-module
    - list-audit-modules
  - File realm
    - Create-file-user
    - Delete-file-user
    - List-file-users
    - update-file-user

Configuration (Contd)

- Realms configuration
  - create-auth-realm
  - Delete-auth-realm
  - list-auth-realms
- SSL configuration:
  - Create-ssl
  - delete-ssl

- Manual Configurations
  - JACC provider configuration in domain.xml
  - LDAP/Solaris/Certificate realm updation
- Through the web application

Mercado J2EE

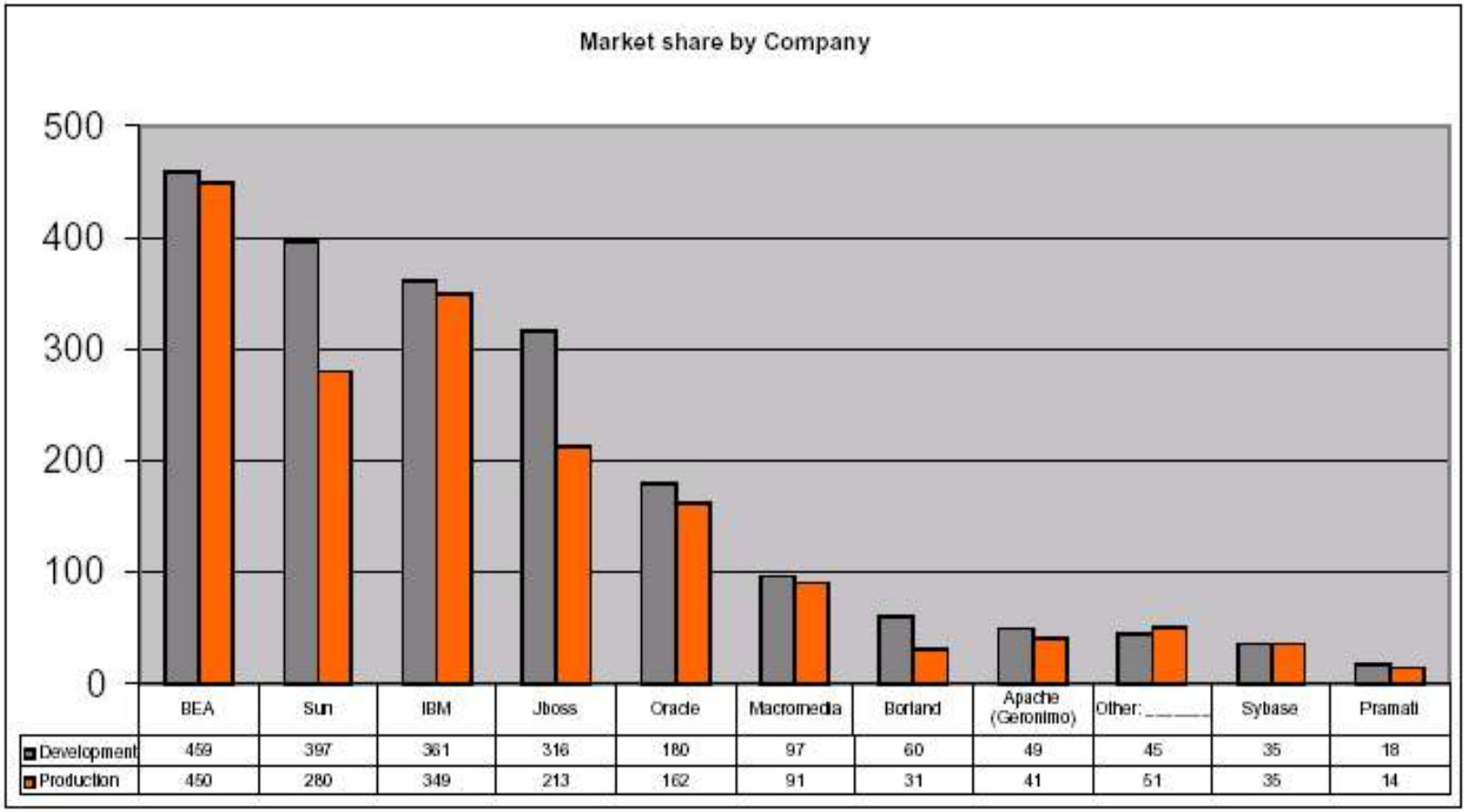
Agenda

- Análisis de Mercado J2EE (BEA, IBM, SUN, ORACLE, JBOSS). Evolución de los modelos de uso, venta y licenciamiento.
- Servidores de aplicaciones OpenSource: Tomcat, JBOSS, ObjectWeb JONAS, Glassfish, Apache Geronimo
- Estrategia de los fabricantes: Sun J2EE 5.0 OpenSource (Glassfish), IBM Websphere Community Edition (Apache Geronimo), RedHat (Jonas), Novell SuSE (JBOSS)
- Herramientas de desarrollo OpenSource: Eclipse & Netbeans
- Roadmap J2EE 5.0 (Java EE 5.0). Estandares de Web Services en JAVA. Estrategia de Sun.
- Roadmap J2SE 6.0 (Java SE 6.0). Evolucion de la maquina virtual.

Análisis de Mercado J2EE (BEA, IBM, SUN, ORACLE). Evolución de los modelos de uso, venta y licenciamiento

The Middleware Company, Aug04

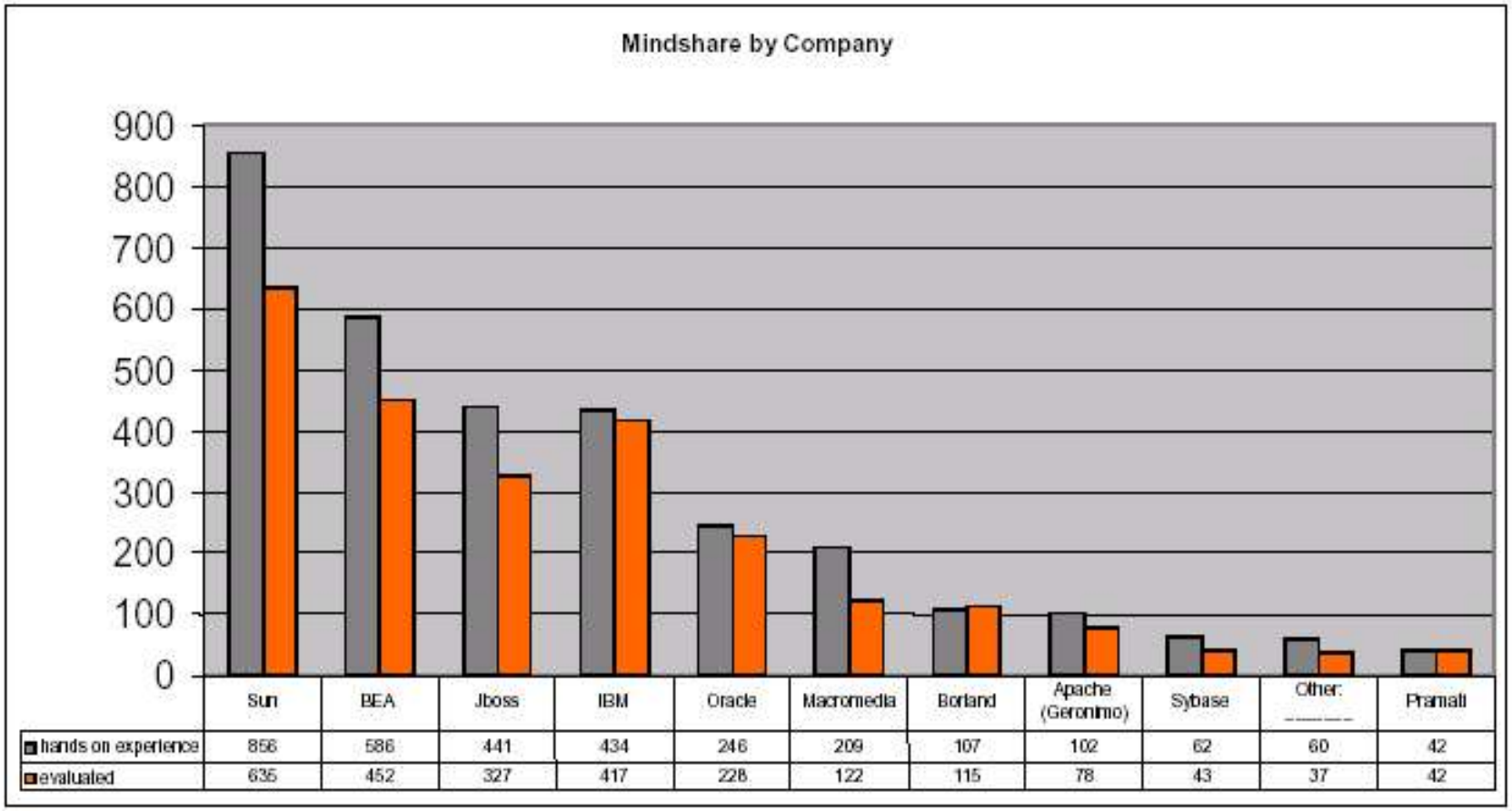
6.3.1.2 Market share by company



	BEA	Sun	IBM	Jboss	Oracle	Macromedia	Borland	Apache (Geronimo)	Other: _____	Sybase	Pramati
Development	459	397	361	316	180	97	60	49	45	35	18
Production	450	280	349	213	162	91	31	41	51	35	14

The Middleware Company, Aug04

6.2.1.2 Mindshare by Company



	Sun	BEA	Jboss	IBM	Oracle	Macromedia	Borland	Apache (Geronimo)	Sybase	Other: _____	Pramati
hands on experience	856	586	441	434	246	209	107	102	62	60	42
evaluated	635	452	327	417	228	122	115	78	43	37	42

Crecimiento de J2EE Open Source

- Los Servidores de Aplicaciones Open Source están ganando terreno a los Servidores Comerciales Tradicionales.
- Factores:
  - Libertad de uso y descarga
  - Ciclo de desarrollo y mantenimiento muy ágil
  - Comunidades muy fuertes
  - Volumen, masa crítica
  - Precios. Descompesacion frente al HW: Moore, Multi-Core, Virtualización.

Microprocessors (Moore)

Growing Transistor Counts (Moore's Law)

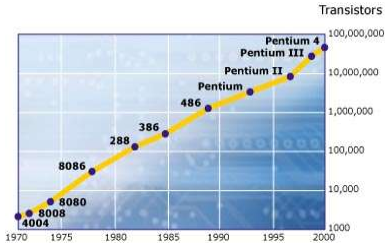
2K 4 8 16 32 64 128 256 512 1M 2 4 8 16 32 64 128 256 512 1B 2

1970 1980 1990 2000 2010

1 1 2 2 3 3 3a

4004 8080 8086 80386 Pentium Pro Opteron Pentium EM64T
4 bits 8 bits 16 bits 32 bits 32 bits 64 bits
640 B pa 64 KB pa 1 MB pa 4 GB va RISC "micro-ops" 16 XB va

physical address (pa) exceeds processor's bit-width – no virtual Address (va)



Sun Processors

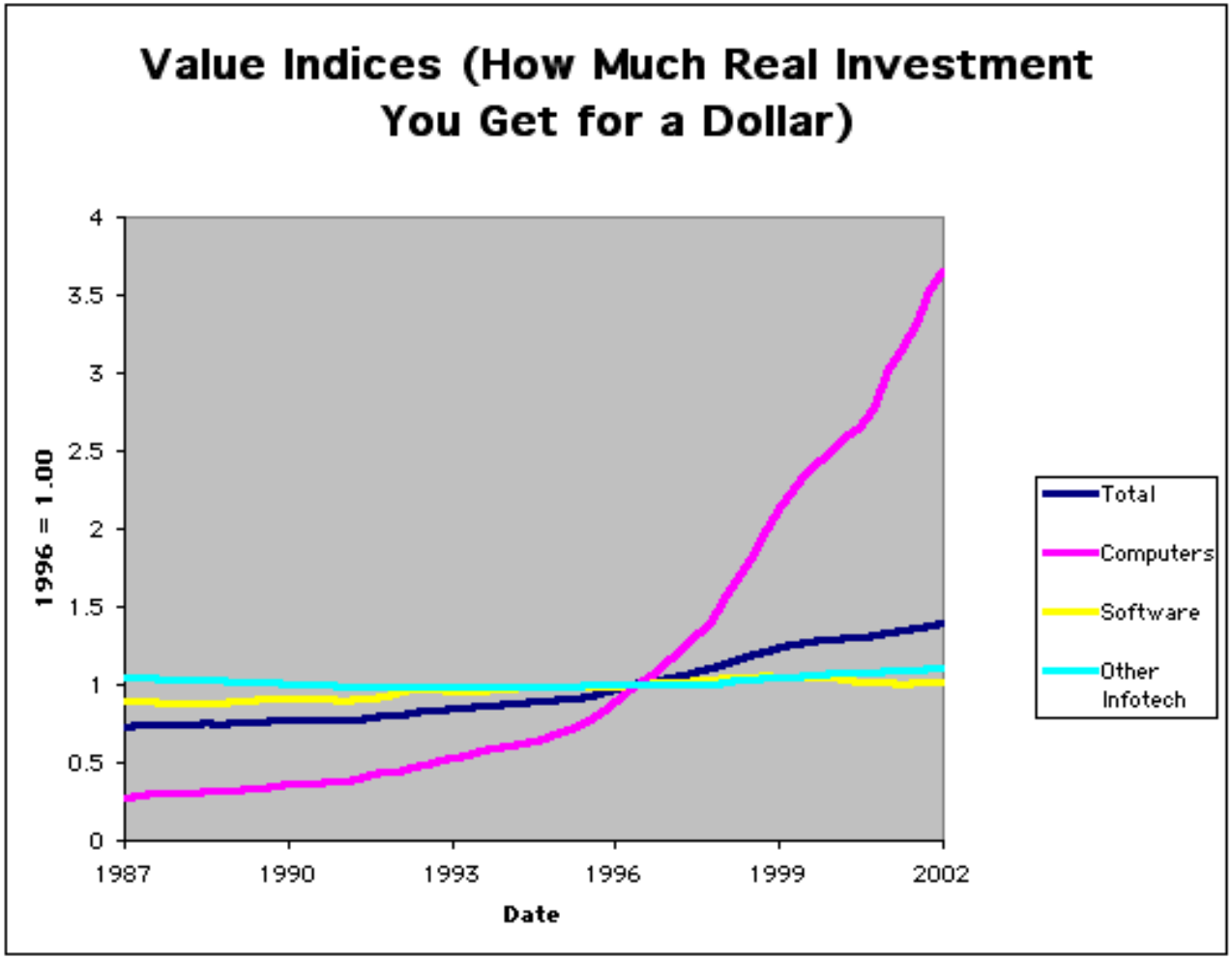
ULTRASPARC DRIVEN

1st 2nd generation

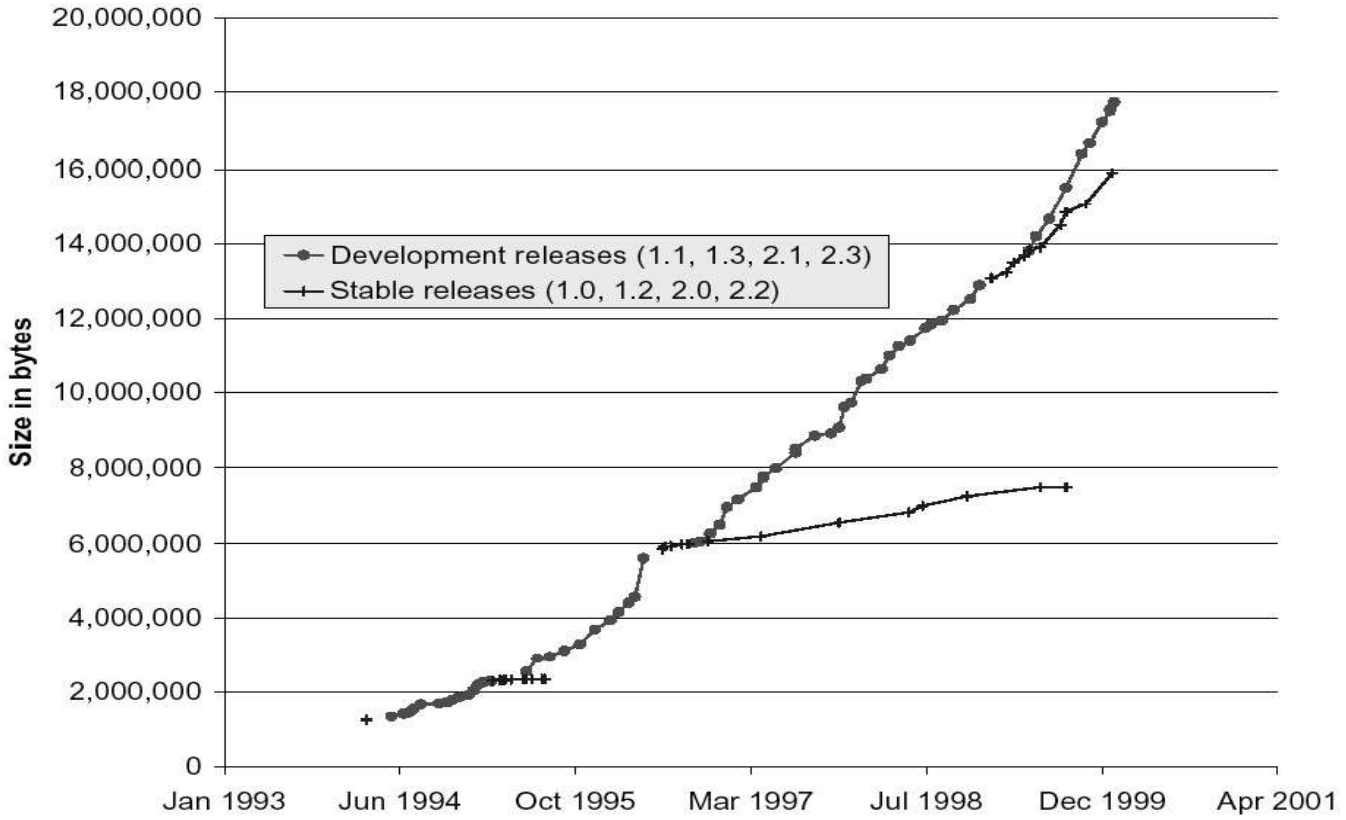
SPARC 32-bit RISC 64-bit RISC CMTRadical CMT
Commodity 68000 80386 Not RISC Not 64 bit AMD Opteron CMT

1 Intel attempts to replace 16-bit x86 architecture with new 32-bit i432 "mini-mainframe" architecture; converts x86 to 32 bits 4 years later
2 Intel attempts to replace 32-bit x86 CISC architecture with new 32-bit i860 RISC architecture; converts x86 to RISC "micro-ops" 6 years later
3 Intel attempts to replace 32-bit x86 RISC architecture with new 64-bit Itanium VLIW architecture; converts x86 to 64 bits 3 years later

Where Is Moore's Law for Software?



Moore´s Law for the linux Kernel (applicable to Open Source develop.)



1. Growth of the compressed `tar` file for the full Linux kernel source release.

Software Licensing Scenarios

IDC: Multi-core chips to disrupt software pricing by 2007

- Core neutral: Sun, Microsoft, VMware, Red Hat
- Extra charging per core: IBM, BEA, Oracle
- Matt Eastwood (vice president of enterprise server research at IDC):
  - In 2006, most of the servers being shipped will be multi-core
  - Processors turn on an 18-month cycle, so by 2007 you'll be on that second wave, when software is optimized for multi-core.
- Joe Clabby (vice president and practice director with Boston-based Summit Strategies):
  - Software shift to a service or per-user based model.

Fourth Quarter 2005 SPECjAppServer2004 Results

Tester	System Title	JOPS	J2EE Server Nodes	J2EE Server CPU Description	J2EE Server Instances
IBM Corporation	WebSphere 6.0.2.3 Application Server on IBM System p5 550 Cluster	2921.48	8	32 cores, 16 chips (SMT on)	8
Sun Microsystems Inc.	BEA WebLogic Server 9.0 on Sun Fire T2000 Cluster	3328.80	6	48 cores, 6 chips	6
Sun Microsystems Inc.	BEA WebLogic Server 9.0 on Sun Fire T2000	615.64	1	8 cores, 1 chip	1
Sun Microsystems Inc.	BEA WebLogic Server 9.0 on Sun Fire X4100 Cluster	1781.47	5	20 cores, 10 chips (2 cores/chip)	5
Sun Microsystems Inc.	Sun Java System Application Server 8.2 Platform Edition on T2000	436.71	1	8 cores, 1 chip, 8 cores/chip (4 threads/core)	3

Last updated: *Wed Dec 7 18:30:44 EST 2005*

Free Version

8.1SE SPECj 2004 Results

- 8.1SE (Standard Edition) delivers enterprise performance at fractions deployment cost of WebSphere 6
  - 90% performance at 1/3 of Deployment Cost
  - Application Tier Costs: HW, OS, App Server License and 3yrs of SW & HW maintenance

	JOPS (Trx/Sec)	Dollar Per JOPS
WebSphere 6	1,343	$505
8.1SE	1,201	$164

Software Licensing Scenarios
Virtualization

- Gordon Haff (Illuminata analyst):
  - > far more disruptive, in my opinion, is virtualization, which divorces the logical or virtual system from the physical server and can do it in a very dynamic way. Virtualization and its cousins [grid, utility computing, etc.] will make pricing that's tied to physical components of any type increasingly an anachronism

Servidores de aplicaciones OpenSource: Tomcat, JBOSS, ObjectWeb JONAS, Glassfish, Apache Geronimo

Open Source Definition Basic Requirements

- Redistribution—Any party can sell or give away the software, with no royalties/ fees required
  - But check licenses for other conditions…
- Source Code—Source code must be included or easily available
- Derived Works—License must allow modifications to be distributed under the same terms as original distribution was received

Open Source Licenses

http://opensource.org/licenses/

- Apache Software License
- New BSD
- CDDL
- Eclipse Public License
- GPL
- LGPL
- Mozilla
- Sun Public License
- PHP License

Apache Software License

- Derived from original BSD Unix license
- Very flexible usage terms: “give us credit”
- For the Java community, this applies to software from jakarta.apache.org and xml.apache.org
- But other packages have adopted it as well
- Lots of commercial products including some Apache Open Source Software (Most of Web Servers / Application Servers)

How Do OS Software Projects Make Decisions?

- Case Study—Jakarta Project at Apache
  - http://jakarta.apache.org/
- Defined roles:
  - User—download and use software
  - Contributor—proposes code or documentation changes
  - Committer—write access to source code repositories
    - Proposed and elected by other committers
- Committers make all the binding decisions!

Will Your Contribution Get Accepted?

- Is the implementation too specific to a particular product requirement?
- Would the proposed change break interfaces and functionality that other users depend on?
- Does any developer on the project care enough about your needs to commit the patches?
  - One time “drive by patches” less likely to be accepted than a series
  - Typically creates an implicit responsibility to maintain the updated code

If You Are Going To Contribute Back

- Most useful practice—aim to have one or more staff members be elected as committers
  - Directly implement changes in the code
  - Advocate to protect interfaces and functionality your product depends on
  - Can vote against proposed changes by others not in line with your needs
- Being a committer comes with a cost
- But the benefits can easily outweigh the cost on software important to you

Tomcat Active Developers

- Amy Roh (amyroh): Catalina, Admin webapp
- Bill Barker (billbarker): Connectors
- Costin Manolache (costin): Catalina, Connectors
- Filip Hanik (fhanik): Clustering
- Glenn Nielsen (glenn): Catalina, Connectors
- Henri Gomez (hgomez): Co[illegible]
- Jan Luehe (luehe): Jasper
- Jean-Francois Arcand (jfarcand): Catalina
- Jean-Frederic Clere (jfclere): Connectors
- Kin-Man Chung (kinman): Jasper
- Mladen Turk (mturk): Connectors
- Remy Maucherat (remm): Catalina, Connectors, Docs, Releases
- Tim Funk (funkman): Catalina, Docs
- Yoav Shapira (yoavs): Docs, JMX

Amy Roh

- Apache Jakarta Tomcat Comitter
- Amyroh at sun.com
- Amy Roh is a committer for Tomcat. She works for the Servlet and JSP reference implementation team at Sun.

Jan Luehe

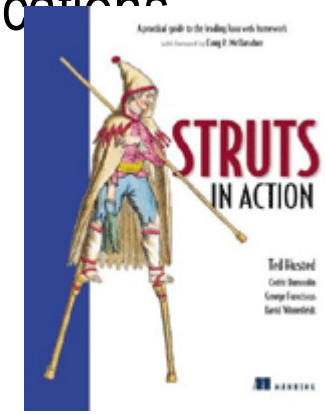
- jan.luehe at sun.com
- Developer of Apache Commons EL (JSP 2.0 Expression Language)

Jean-Francois Arcand

- Jean-Francois Arcand is working for Sun Microsystems since 2000. He currently works on Jakarta Tomcat as well as SUN's Application Server. Before joining Sun, he has worked as a software architect for compagnies such as France Telecom, Microcell Telecom and HMS Software, in both Java and C++. Jean-Francois lives and works from home in Prevost, a very small city in Quebec where life is perfect.

Craig McClanahan

- **Craig McClanahan is a senior Staff Engineer at Sun Microsystems**. He is currently a senior architect on the team that is building Sun Java Studio Creator. Previously, he was co-specification lead for JavaServer Faces 1.0 (JSR-127), and the **original creator of Struts**, an open source framework for building web applications.
- Author of:



Craig R. McClanahan

- Craig McClanhan is a Senior Staff Engineer at Sun Microsystems. His current responsibilities include being the Web Layer Architect for the J2EE Platform, as well as the co-Specification Lead for JavaServer Faces (JSR-127). Craig is also a member of the Apache Software Foundation, and a member of the Jakarta PMC. He has been heavily involved in the development of Tomcat, and is the founding developer for Struts and several of the Jakarta Commons library packages.
- Involved with servlet and JSP technology since around 1998, helped Apache JServ to get from version 0.9 to version 1.0

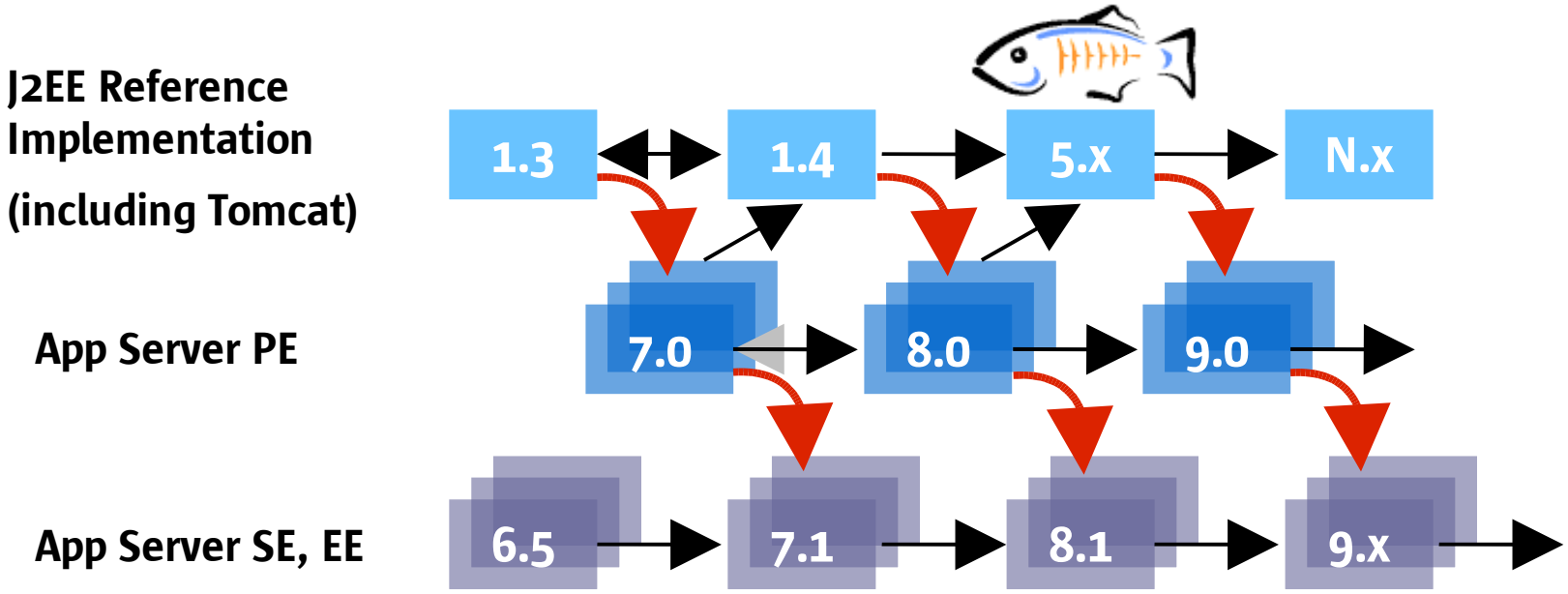
Servidores que usan Tomcat

- JBOSS Application Server
- Apache Geronimo
- Sun JAVA System Application Server
- Red HAT Application Server
- Novell eXtend
- GlassFish (J2EE 1.5 OpenSource)

GlassFish Project: J2EE 5.0 Open Source. SJS Application Server 9 PE

- https://glassfish.dev.java.net/
- The GlassFish Project is a gathering place for developers who wish to participate in the open source community developing of the latest version of Sun's Java System Application Server PE 9.0

PQRI: Production Quality Reference Implementation



From J2EE SDK 1.3.1 Double effort in J2EE RI to be ready to build AS EE on top of it

Time

Estrategia de los fabricantes: Sun J2EE 5.0 OpenSource (Glassfish), IBM Websphere Community Edition (Apache Geronimo), RedHat (Jonas), Novell SuSE (JBOSS)

Redefining Software Infrastructure

Simpler, Easier, Better Value, More Innovation

Java

Web Services

Sun microsystems

Web Server	~~Per CPU~~
App Server	~~Per CPU~~
Directory Server	~~Per Entry~~
Identity Management	~~Per Identity~~
Portal Server	~~Per CPU~~
Messaging Server	~~Per Mailbox~~
Clustering	~~Per Node~~
Database	~~Per CPU Core~~
File System	~~Per Terabyte~~

Subscription model
$/Employee-Yr

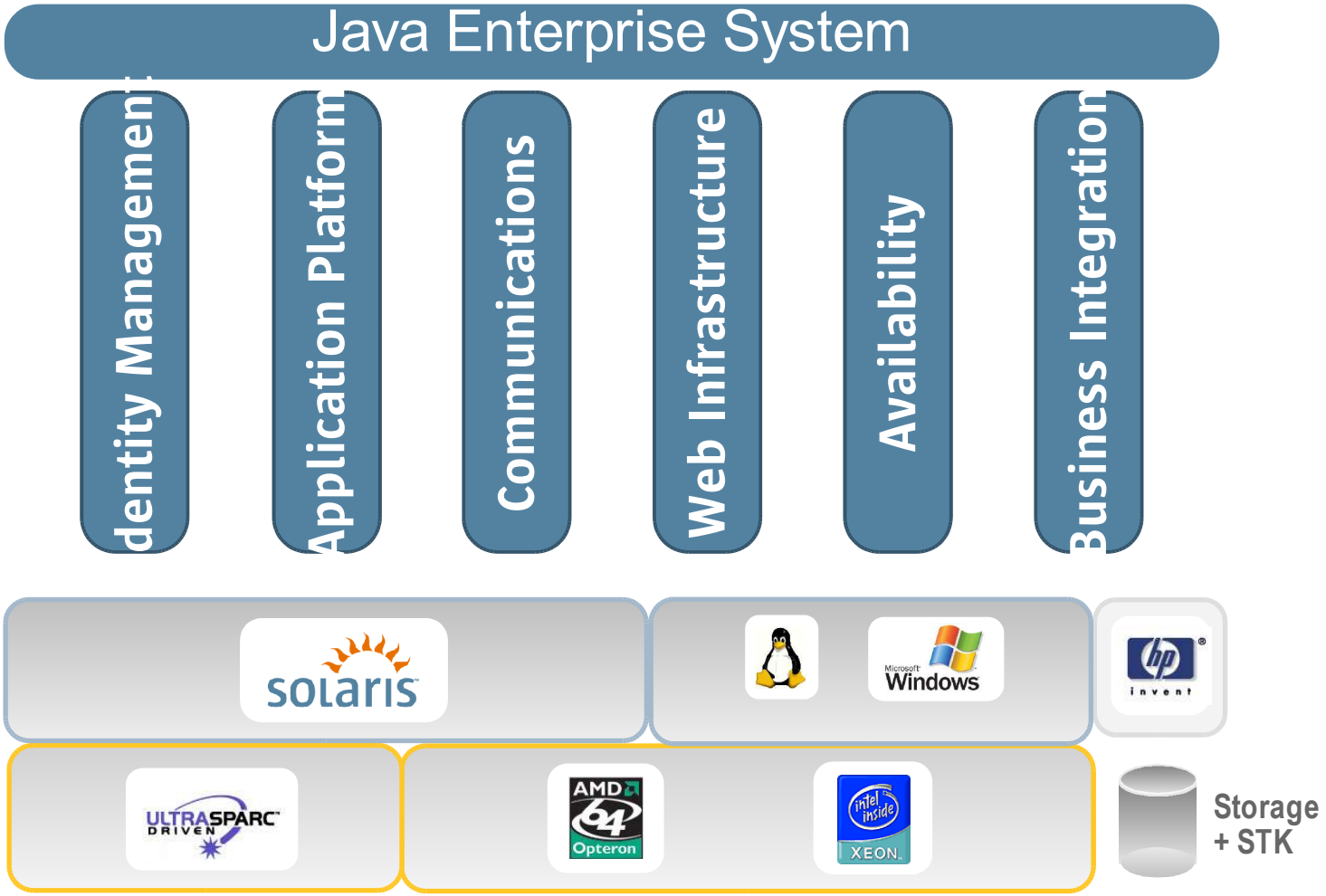
Robust, Flexible & Simple

Choice Requires a "Truly Open Software" Architecture
Open = Easy to Change

Choice Requires Access Without Risk

Download It and Use It Free, for Whatever You Like – Without Risk

Buy It to Get Services: Warranty, Indemnification and Support



The Vision – “Participation Age”

Everyone and Everything Participating on the Network

La comunidad OpenSolaris

Fecha Nacimiento:

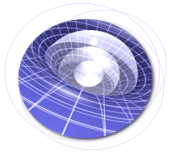
14 de junio, 2005

En 4 Meses:

- **1 Millón Downloads**
- **Más de 9,000 miembros**
- **8,700 aportaciones**

Source: www.opensolaris.org

Compartiendo la innovacion con la comunidad Open Source



GridEngine

Solaris X internationalization technology

Project JXTA

NFSv4

Li18nux.org

Mozilla.org

The WBEMsource Initiative

Free Standard Group

opensolaris

Open Source Licenses

http://opensource.org/licenses/

- Apache Software License
- New BSD
- CDDL
- Eclipse Public License
- GPL
- LGPL
- Mozilla
- Sun Public License
- PHP License

Mozilla License

- http://www.opensource.org/licenses/mozilla1.1.php
- The MPL has a limited amount of 'copyleft' - more copyleft than the BSD family of licenses, which have no copyleft at all, but less than the LGPL or the GPL.
- Allows to build propietary software on top of MPL software if MPLed files are no modified and new files contain no MPLed code
- Example:
  - Google branded browser with Google Talk
  - Sun Branded Browser

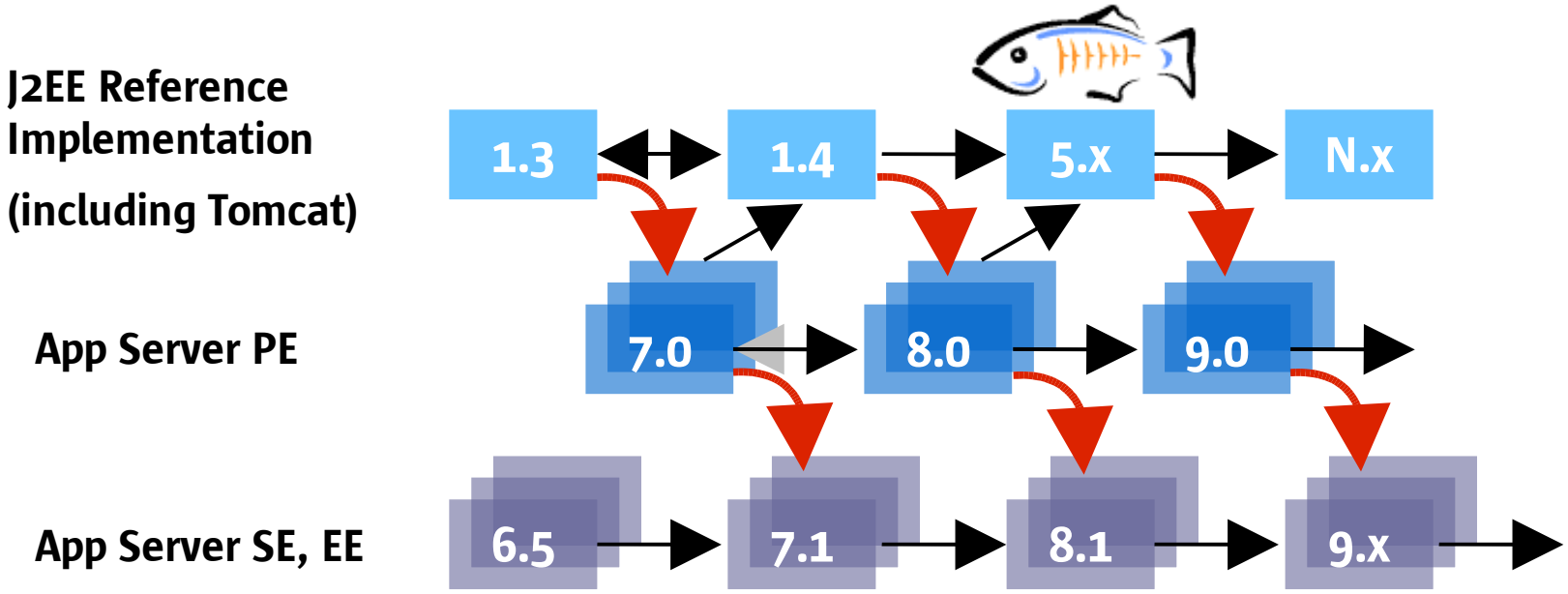
Common Development and Distribution License (CDDL)

- http://www.opensource.org/licenses/cddl1.php
- A variant of the Mozilla Public License (MPL)
- The CDDL allows proprietary products to be built on top of open-source code
- Open Source Initiatives using this license:
  - OpenSolaris
  - GlassFish J2EE 1.5
  - Open-ESB
  - OpenSSO

GlassFish Project: J2EE 5.0 Open Source. SJS Application Server 9 PE

- https://glassfish.dev.java.net/
- The GlassFish Project is a gathering place for developers who wish to participate in the open source community developing of the latest version of Sun's Java System Application Server PE 9.0

PQRI: Production Quality Reference Implementation



From J2EE SDK 1.3.1 Double effort in J2EE RI to be ready to build AS EE on top of it

Time

Open ESB (Enterprise Service Bus)

- https://open-esb.dev.java.net/
- The open-esb project will implement an enterprise class Enterprise Service Bus (ESB) runtime with sample Service Bindings and Service Engines. The core of this software development kit is based on JSR-208, Java Business Integration technology

Open Web Single Sign-On (OpenSSO)

- http://opensso.dev.java.net/
- The Open Web SSO project provides core identity services to facilitate the implementation of transparent single sign on as an infrastructure security component. Targeted towards the web tier, this project provides the foundation for achieving seamless integration of diverse web applications that typically operate against a disparate set of identity repositories and are hosted on a variety of platforms such as web and application servers.

Analistas

- Ovum, Enero de 2006
  - Sun's strategy is bold, innovative, full of potential and, more importantly, Sun is right
  - We agree whole-heartedly with the principles behind Sun's strategy. We are in the midst of a profound change in the business model that underpins the software industry - the balance of revenues is shifting from one-off licences to those from service and support, and whilst it will take at least five years for the transition to complete, Sun's vision is spot on. We also believe that the open source development model is a hugely powerful approach to software innovation, which will over time dominate the entire industry.
  - So there's no doubt that, where it comes to the huge tectonic shift that is underway in the industry, Sun has seen the future.

IBM WebSphere Community Edtion

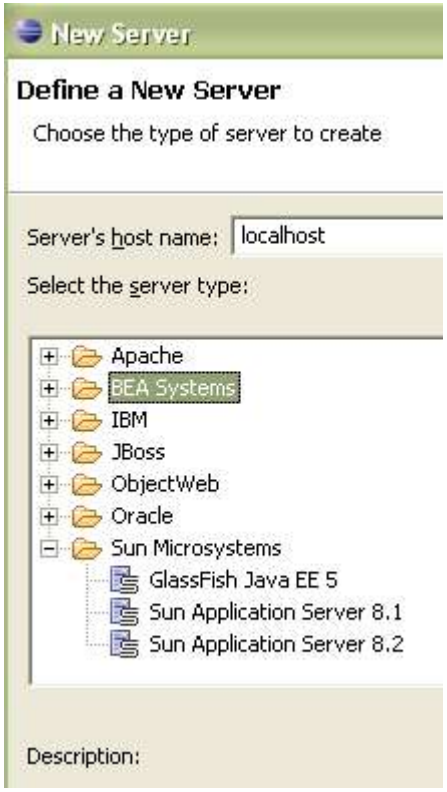
- Distribución soportada de Apache Geronimo
- http://geronimo.apache.org/
- Apache Geronimo incluye:
  - Apache Tomcat and Jetty for Servlets 2.4 and JSP 2.0
  - OpenEJB for EJB 2.1
  - ActiveMQ for JMS 1.1
  - MX4J for JMX 1.2
  - TranQL for JDBC
  - Apache Axis for Web Services
  - HOWL (ObjectWeb) for JTA 1.0

Herramientas de desarrollo OpenSource: Eclipse & Netbeans

Sun Open Source IDEs position

- the truth is that the world needs a single IDE as much as it needs a single operating system or a single software license. Maintaining choice is the lifeblood of open source and I'm delighted to see both NetBeans and Eclipse gaining support for software that had previously been positioned as partisan
- Simon Phipps

Eclipse supports GlassFish



https://glassfishplugins.dev.java.net/download/

NetBeans AppServer Support

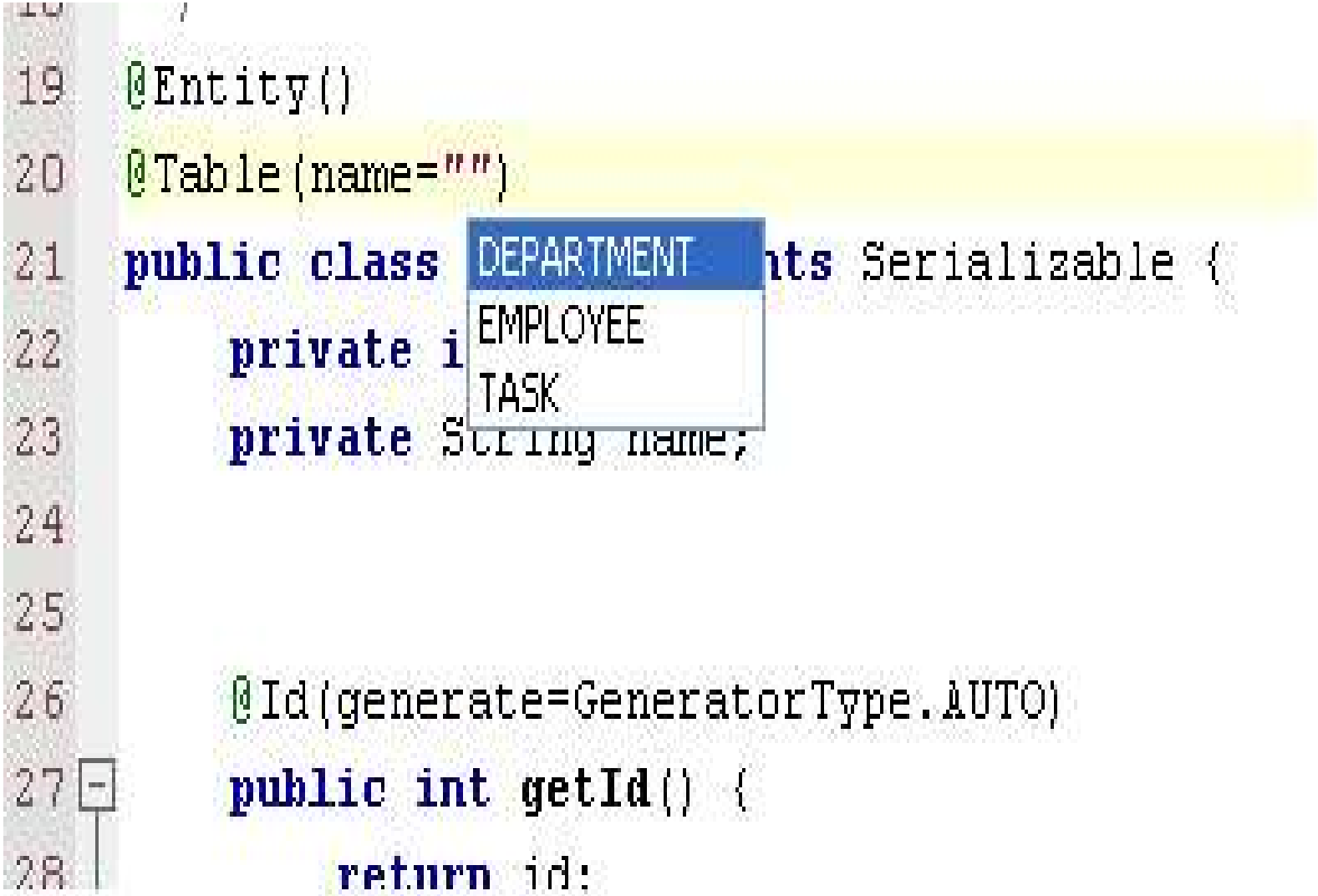
Oracle apoya Netbeans

- http://www.sun.com/2006-0111/feature/index.html
- 11 de Enero de 2006. Oracle Sun Kickoff
- Focus on three main areas:
  - Collaboration on Java technology, with Oracle's continued commitment to Java technology for the next 10 years
  - Oracle's endorsement of the NetBeans IDE, which will fuel innovation through open source software and community-based development
  - Offering the leading real-world performance of Oracle SW on Sun's hot x64 processor-based servers, multicore UltraSPARC processor-based servers, and storage systems

NetBeans & J2EE 5.0

- (J2EE 1.5 == J2EE 5.0 == Java EE 5)
- Under development
- Target GlassFish and others
- EJB 3.0 annotations
- Web Services annotations
- Ease of Development
- JSF and Struts support
- NetBeans will be ready the same day as Java EE 5

NetBeans & J2EE 5.0



Roadmap J2EE 5.0 (Java EE 5.0). Estandares de Web Services en JAVA.

J2EE 5.0 Major Features

- Simplified web services support
- More web service standards support
- Greatly simplified EJB development
- New persistence API, effectively replacing EJB CMP
- Easy web applications with JSF

J2EE 5.0 New Contents

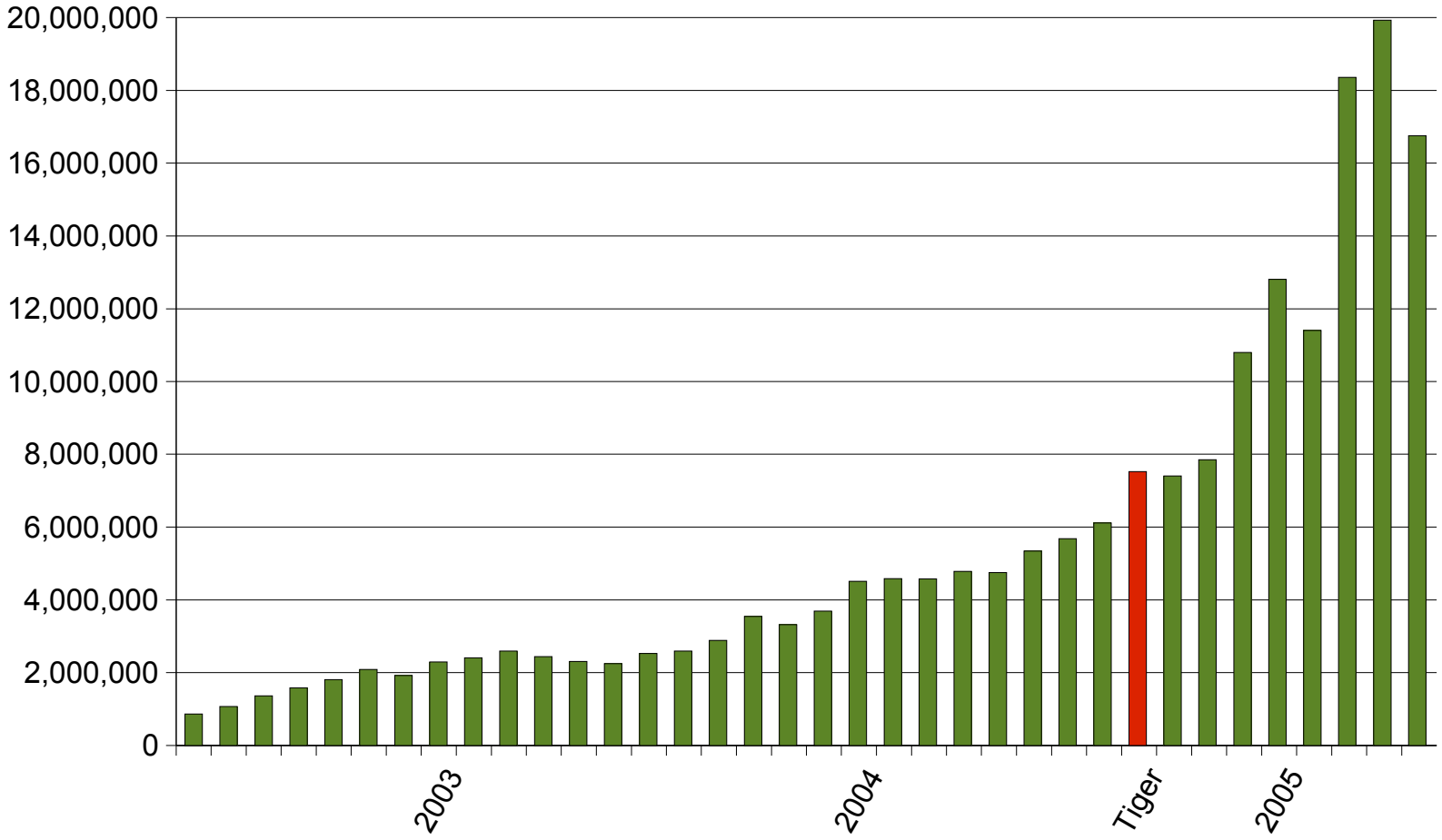
- JSP Standard Tag Library (JSR-52)
- JSP 2.1 (JSR-245)
- StAX (JSR-173)
- Web Services Metadata (JSR-181)
- JAXB (JSR-222)
- JAX-RPC 2.0 (JSR-224)
- Common Annotations (JSR-250)
- JavaServer Faces (JSR-252)
- New persistence API / EJB 3.0 (JSR-220)

J2EE 6.0 Contents (possible)

- Updates of all core specs
- Additional specs
  - JSR-168 Portlets
  - JSR-196 Authentication SPI for containers
  - JSR-207 Process Definition for Java
  - JSR-208 Java Business Integration
  - JSR-225 Xquery API
  - JSR-227 A Standard Data Binding and Access Facility
  - JSR-235 Service Data Objects
  - JSR-261 Web Services Addressing

Roadmap J2SE 6.0 (Java SE 6.0). Evolucion de la maquina virtual

Completed J2SE™ Platform Downloads



1.4.0	**Merlin**	**2002/2/13**
1.4.1	Hopper	2002/10/16
1.4.2	Mantis	2003/5/29
5.0	**Tiger**	**2004/9/30**
6	**Mustang**	**2006/Q3**
7	**Dolphin**	**2008/Q1**

J2SE 6.0: Mustang Themes

- Compatibility, Stability, & Quality!
- Diagnosability, Monitoring, & Management
- XML & Web Services
- Ease of Development
- Enterprise Desktop
- Transparency

JSR 270: Java SE 6 Release Contents

Expert Group Members

Apache
BEA
David Bock
Capgemini
Google
HP
IBM
Ikayzo
Intel
JBoss
Doug Lea
Metasolv
Oracle
Sam Pullara
SAP
SAS Institute
Michael Santos
ThoughtWorks

Mustang Component JSRs

202: Class File Update

199: Compiler API
269: Annotation Processors
260: Javadoc™ Tag Update
221: JDBC™ 4.0
223: Scripting

105: XML Digital Signature
173: Streaming API for XML
222: JAXB 2.0

250: Common Annotations
181: WS Metadata
224: JAX-WS 2.0


JMX
BigDecimal updates
z-ordering
jstat
RMI dynamic proxies
Generics
Gnome Skins
jps
JDBC Rowsets
Autoboxing
extended for loop
faster startup
jconsole
printf
JVM sharing
synth L&F
SAX 2.0
Concurrency utilities
Unicode Surrogates
importing constants
OpenGL
IP reachability
Ocean L&F
SASL
performance
XAWT
unsynchronized StringBuffer
improved cookie support
XML Schema
apt
DOM 3
JVM Monitoring
fatal error handlers
Stack trace API
Remote JMX
improved footprint
varargs
swing printing
AMD64
Enumerated types
XDnD
Packed JARs
metadata
OCSP
New Memory Model
scanning
JVMPI Profiling
performance ergonomics


Compiler API
Longhorn
Look & Feel
MBeans
metadata
JTable upgrades
Splash
screens
Split Verifier
Windows system tray
Unicode Normalizer
Services
Attach on demand
chmod
Parallelize Concurrent
GC
JConsole upgrades
Core JVM
performance
Mustang
Annotation processors
SwingWorker
Parallel old-space GC
Web Services Stack
Password
prompting
JVM DTrace
JavaDoc
LCD fonts
Docs in
Chinese
JDBC 4.0
Faster JNI
Free disk space
Tags
JAXB 2.0
More gfx acceleration
Improved OOM diagnosability
JVM & CLR Co-
Existence
More desktop
integration
FireFox support
Pluggable Locales
Scripting Languages
Native L&F
HTTP cookie manager
XAWT
More GC
Fidelity
JavaScript engine
XML digital signatures
Improved text rendering
Ergonomics

Resolution-Independent Graphics

Client Ergonomics

Filesystem API

Docking

Modules

Tiered Compilation

Dolphin

Friends

invokedynamic bytecode

Tracing

HTML 4.0

BeanShell

Asynchronous I/O

Language-level XML

Performance

Class-data sharing for applications

Class-loader based application isolation

Method References

Web Services for JMX

Servidores de Aplicaciones

Arquitectura. Planificación. Análisis de Mercado.

Jaime Cid

http://blogs.sun.com/jaimecid