

E2E Committee Meeting

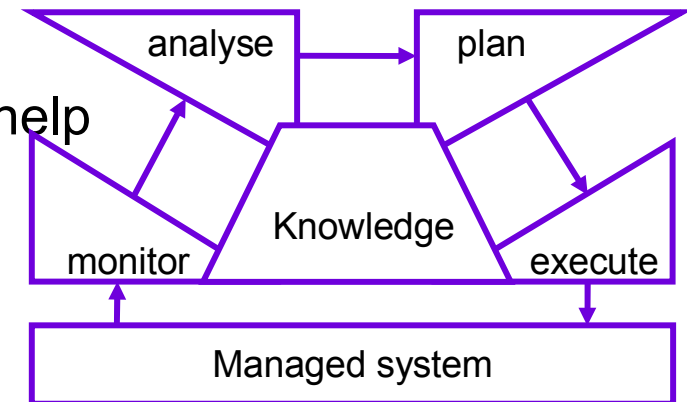
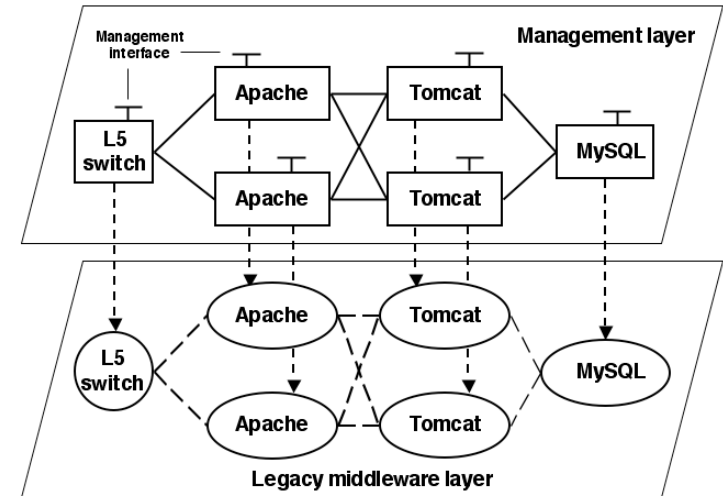
> *Some Use Cases & Requirements*



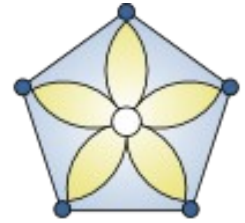
Guillaume.Sauthier@Bull.Net

➤ JADE

- Framework for building autonomous systems
 - Self-management
 - Self-repair
 - Self-sizing
 - Self-protect
- In memory System representation
 - Casually connected
- OSGi first introduced for deployment help
 - Now it's the base of the System



E2E Committee Meeting



➤ JASMINe

- Cluster management software
- Use JADE
- Provides a GUI for deployment
- Generates application configuration (JonAS, Apache, ...)
- Use probes to monitor the system
- Provides a management console
 - Protocol independent (JMX/SOAP)
 - Gateway discovery (DNS-SD)



E2E Committee Meeting

➤ Use Cases

➤ Deployment

- Java EE Application Server deployment
- Java EE applications deployment
- “Offline” gateways deployment
- OBR Management API (write: add resources)

➤ Configuration

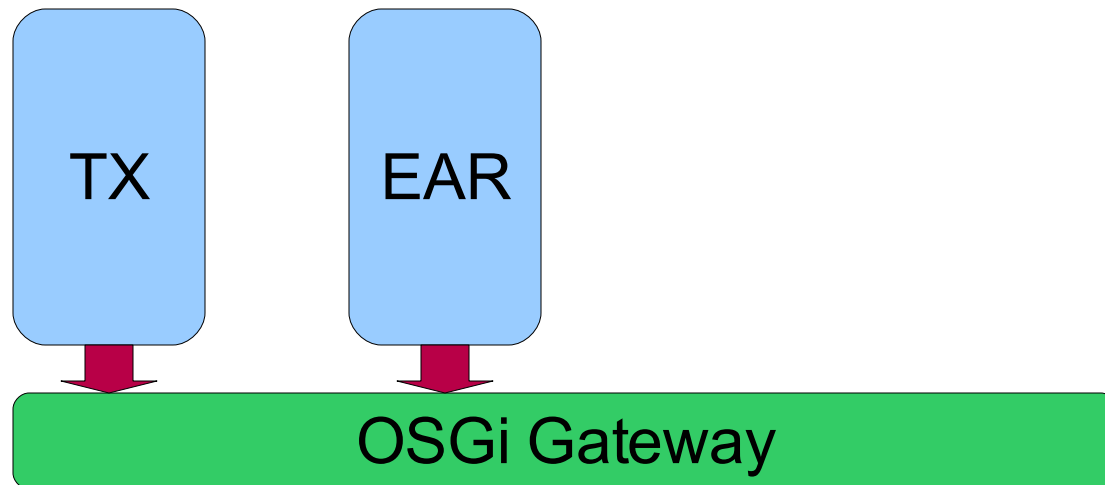
- Remote configuration provisioning (Pull / Push)
- Dynamic reconfiguration (Update)
- On demand services (per application needs)

➤ Management

- Remote gateway management
- Large scale Java EE servers management (Farms)
- Java EE Server Self-Management (Autonomous)

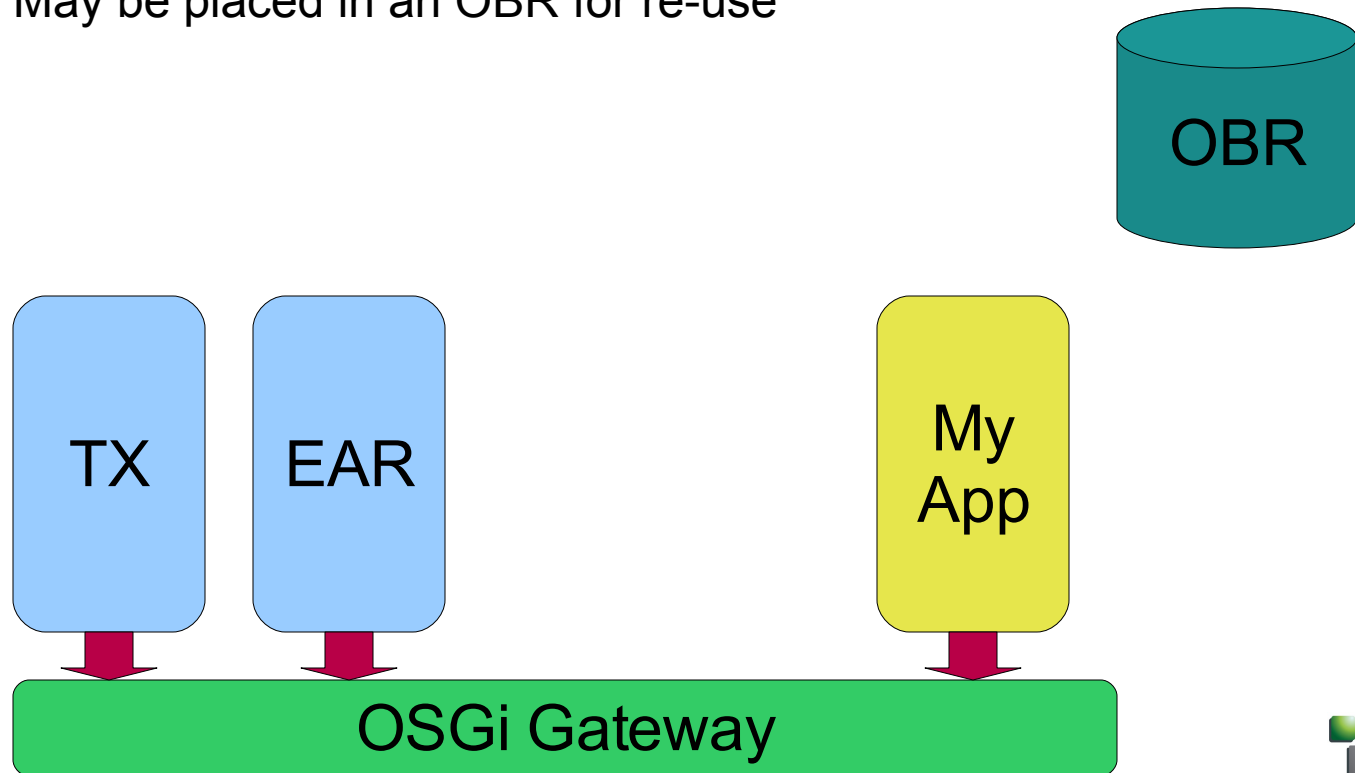
Use Cases / Deployment

- Java EE Application Server deployment
 - JOnAS 5 based on OSGi
 - Provides a set of Services (Deployers, Transaction, ...)
 - JOnAS Services deployable as bundles
 - Running OSGi gateway
 - Install JOnAS blocks on an empty infrastructure



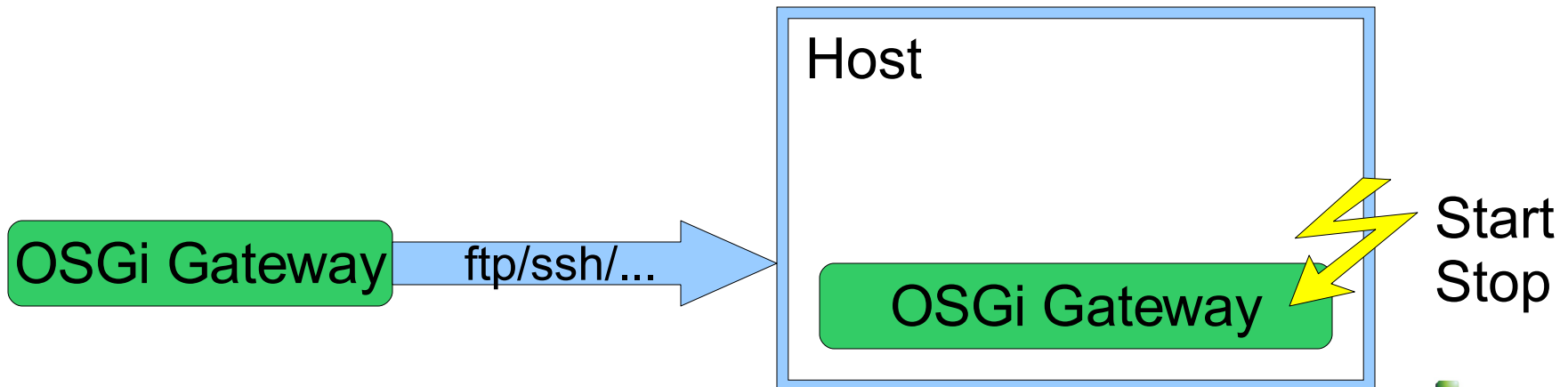
Use Cases / Deployment

- Java EE Application deployment
 - Java EE applications are converted into bundles by deployers
 - On the fly conversion (locally modified only)
 - May be placed in an OBR for re-use



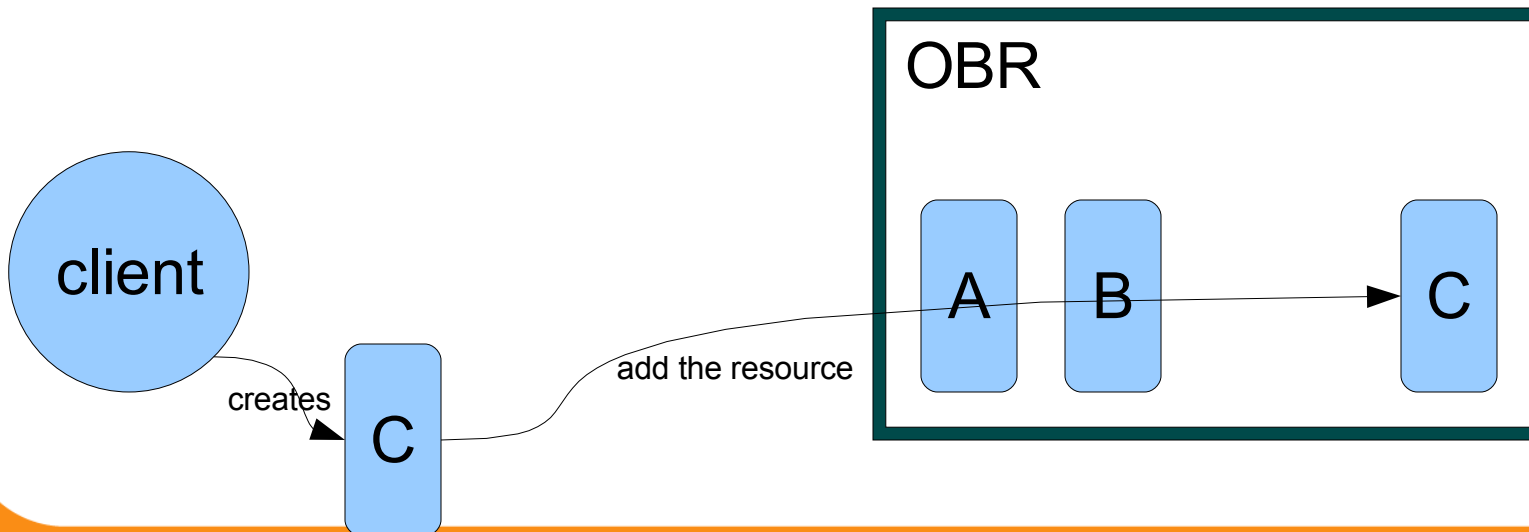
Use Cases / Deployment

- “Offline” gateway deployment
 - Context:
 - Java EE servers farms or grids
 - The minimal execution environment may not be installed
 - Need a way to deploy remotely the gateway
 - Binaries upload
 - Gateway life-cycle support



Use Cases / Deployment

- API for OBR management
 - Current API focus on repository browsing and resource resolution
 - JOnAS may need an API for OBR modifications
 - Add a resource (EjbJars, Libs, ...)



E2E Committee Meeting

➤ Use Cases

➤ Deployment

- Java EE Application Server deployment
- Java EE applications deployment
- “Offline” gateways deployment
- OBR Management API (write: add resources)

➤ Configuration

- Remote configuration provisioning (Pull / Push)
- Dynamic reconfiguration (Update)
- On demand services (per application needs)

➤ Management

- Remote gateway management
- Large scale Java EE servers management (Farms)
- Java EE Server Self-Management (Autonomous)

Use Cases / Configuration

- Remote configuration provisioning (Pull / Push)
 - Configuration provisioning is a hard task in a clustered environment
 - JOnAS currently supports PUSH mode through JASMINe
 - Configuration is generated off-line
 - Uploaded to the host
 - Instance is started
 - Not very scalable, no dynamism

Use Cases / Configuration

- Dynamic reconfiguration
 - On code update, configuration properties may changes

Config Admin

Use Cases / Configuration

- On demand services
 - Start with a minimal 'kernel'
 - JOnAS Services are added/activated on demand
 - Applications ask for a particular service
 - Services are then installed/started
 - JOnAS Services should be stopped/removed
 - When no one are using them
 - Reducing the memory footprint

E2E Committee Meeting

➤ Use Cases

➤ Deployment

- Java EE Application Server deployment
- Java EE applications deployment
- “Offline” gateways deployment
- OBR Management API (write: add resources)

➤ Configuration

- Remote configuration provisioning (Pull / Push)
- Dynamic reconfiguration (Update)
- On demand services (per application needs)

➤ Management

- Remote gateway management
- Large scale Java EE servers management (Farms)
- Java EE Server Self-Management (Autonomous)

Use Cases / Management



- Remote gateway management
 - On-line/connected mode
 - Protocol agnostic
 - Protocol may change given the network configuration
 - Firewalls, routers may force us to change the protocol
 - Introspection API
 - Browse Bundles, Services, ...

The screenshot displays the JASMINe OSGi Gateway Administration interface. The main window is titled "JASMINe : OSGi Gateway Administration". It features a "File" menu and a "Preferences" tab. The interface is divided into two main panes:

- Passerelles OSGi:** A tree view showing a list of OSGi bundles. The bundles are numbered and their status is indicated by a colored circle (green for [ACTIVE], red for [RESOLVED]). The list includes:
 - 0: [ACTIVE] System Bundle (0.8.0.SNAPSHOT)
 - 1: [ACTIVE] Apache Felix Shell Service (0.8.0.SNAPSHOT)
 - 2: [ACTIVE] Apache Felix Shell TUI (0.8.0.SNAPSHOT)
 - 3: [ACTIVE] Apache Felix Bundle Repository (0.8.0.incubator)
 - 4: [ACTIVE] AdminMBean (0.4.0)
 - 5: [ACTIVE] JMX Agent (0.1.0)
 - 7: [ACTIVE] ObrMBean (0.1.0)
 - 8: [ACTIVE] JmDNS Registration (0.1.0)
 - 9: [ACTIVE] JmDNS (0.1.0)
 - 18: [RESOLVED] Service Binder (0.8.0.SNAPSHOT)
- Référentiels de Bundles OSGi:** A list of OSGi bundle repositories. The selected repository is "http://oscar-osgi.sf.net/obr2/repository.xml (Untitled)". The list includes:
 - Apache Felix Bundle Repository
 - Apache Felix Shell Service
 - Apache Felix Shell TUI
 - telnetd

At the bottom, there is a "Console" and "Properties" tab. The "Properties" tab shows the following data:

Property	Value
Repository	
Last Modified	Fri Dec 15 00:10:44 CET 2006
Name	Untitled
URL	http://oscar-osgi.sf.net/obr2/repository.xml

Use Cases / Management

- Large scale Java EE servers management (Farms)
 - High numbers of managements targets
 - Deployment plans introduced
 - Strategies on failures/errors (retry, skip)
 - Requirements
 - Reliable (ACID ?)
 - Secured
 - Structured

Use Cases / Management

- Java EE Server Self-Management (Autonomous)
 - Self-sizing
 - Applications can be configured to better support a workload
 - In a cluster, other instances may be started to share the load
 - Related to dynamic reconfiguration



Architect of an Open World™