



**CENTRE D'INNOVATION
EN TELECOMMUNICATION
& INTEGRATION DE SERVICES**

OSGi

Part III Architecture Interne

1 Gestion des Classes

Stéphane Frénot

Architectures de Réseaux de Services



ARÈS



1



Plan

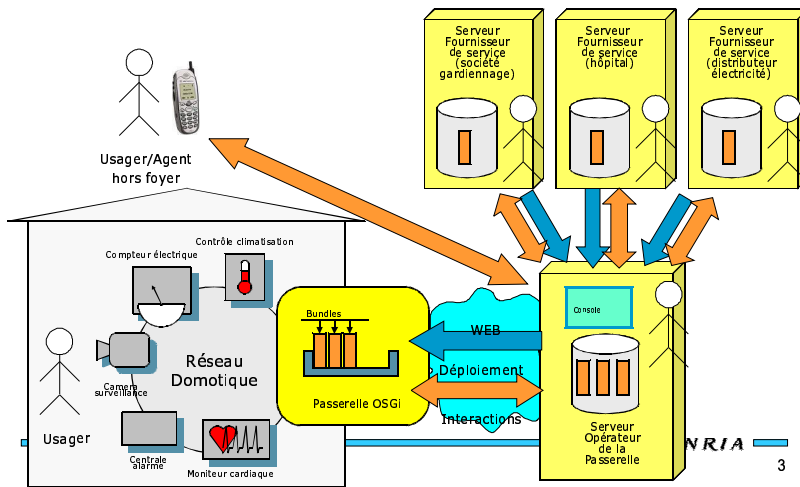
- Architecture générale
- Cycle de vie
- Premier Bundle
- Manifest
- Gestion des packages



2



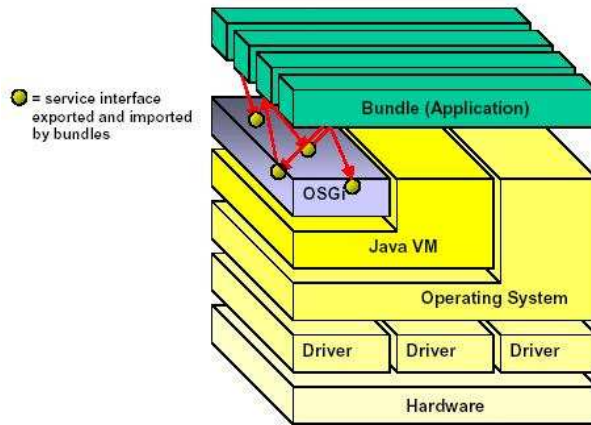
Architecture générale



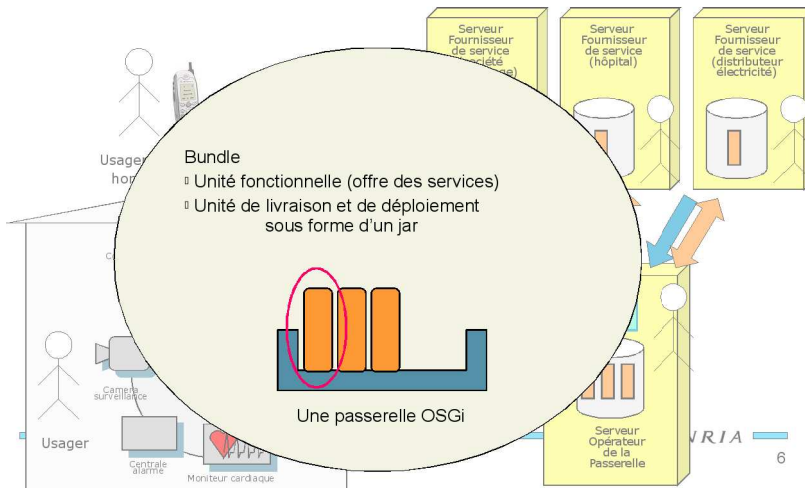
Motivations

- Chargement/Déchargement de code dynamique
 - Langage Java
- Déploiement dynamique d'applications sans interruption de la passerelle
 - Installation, Lancement, Mise à jour, Arrêt, Retrait
- Résolution des dépendances versionnées de code
- Programmation orientée service
- Cible également des systèmes à mémoire restreinte

OSGi framework

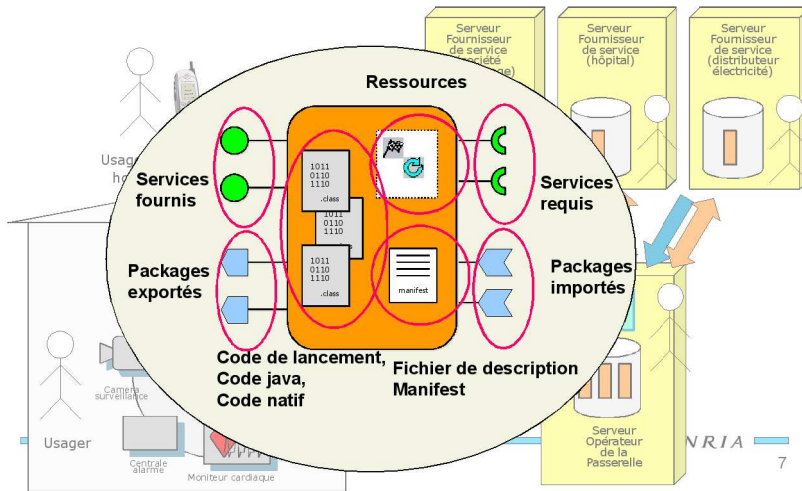


Bundle et Service

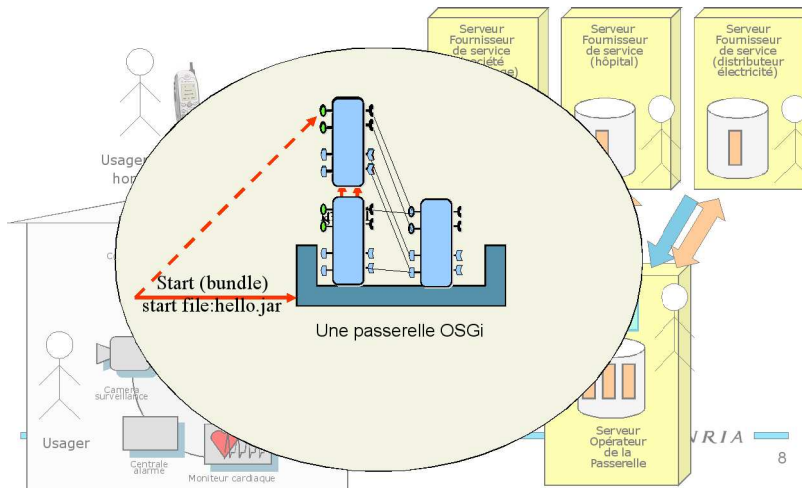




Structure d'un bundle



Bundle et Service



Le bundle hello

```

package hello;
import org.osgi.framework.BundleActivator;
import org.osgi.framework.BundleContext;

public class HelloWorld implements BundleActivator {
    public void start(BundleContext bc){
        System.out.println("Bonjour");
    }

    public void stop(BundleContext bc){
        System.out.println("Au revoir");
    }
}

```



```

Bundle-Name: Hello World
Bundle-Description: The simple bundle
Bundle-Activator: hello.HelloWorld

```



```

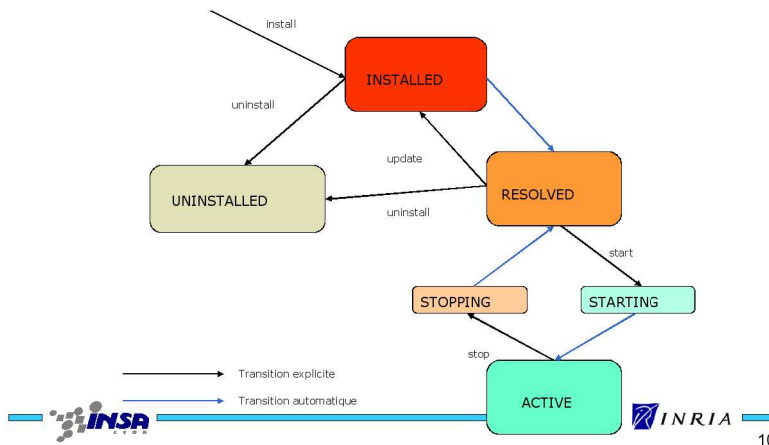
200 Tue Jun 15 16:20:00 CEST 2004 META-INF/MANIFEST.MF
723 Tue Jun 15 16:20:00 CEST 2004 hello/HelloWorld.class

```



9

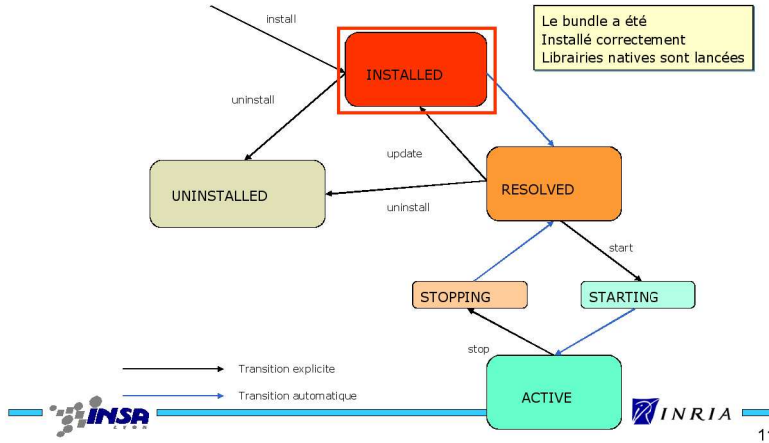
Cycle de vie et état d'un bundle



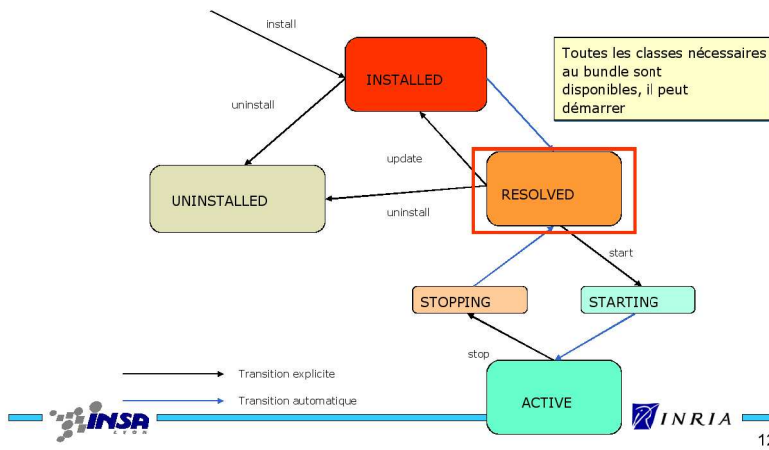
10



Cycle de vie et état d'un bundle

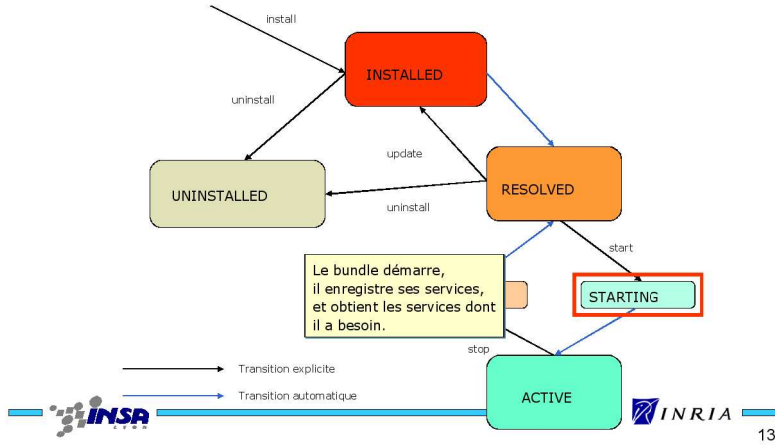


Cycle de vie et état d'un bundle

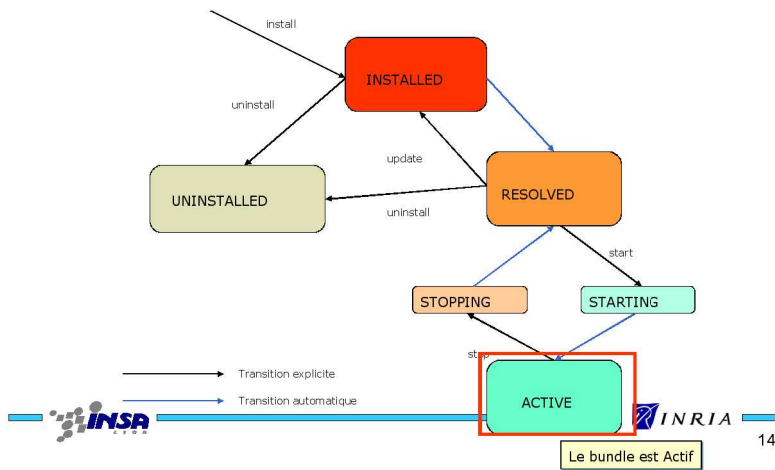




Cycle de vie et état d'un bundle

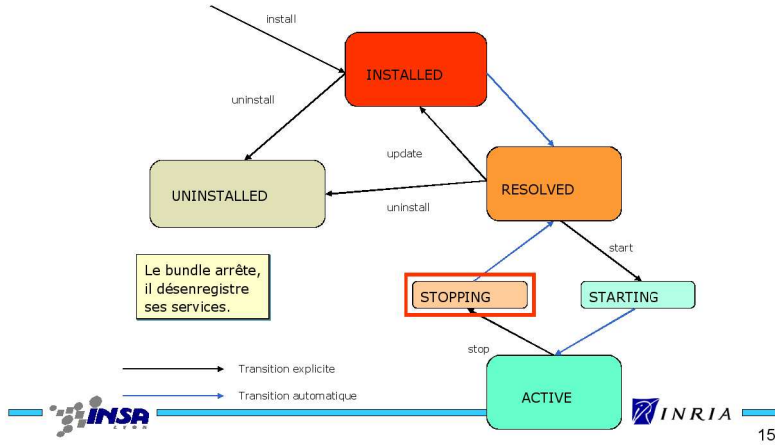


Cycle de vie et état d'un bundle

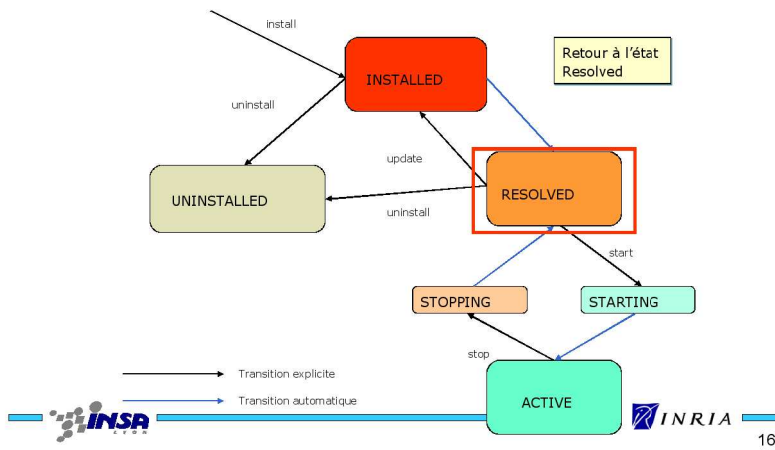




Cycle de vie et état d'un bundle

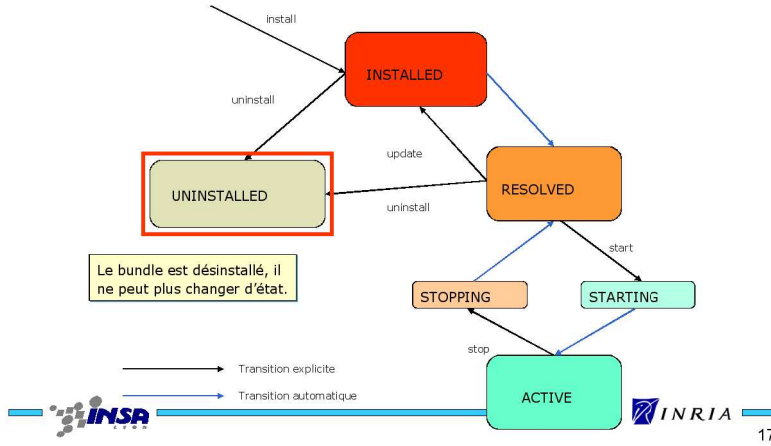


Cycle de vie et état d'un bundle

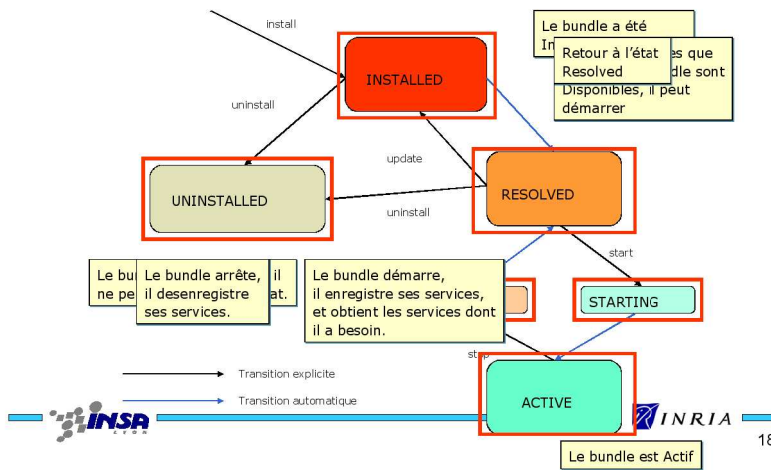




Cycle de vie et état d'un bundle

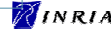
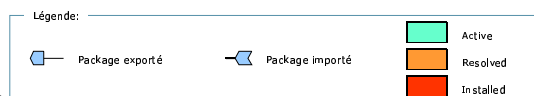
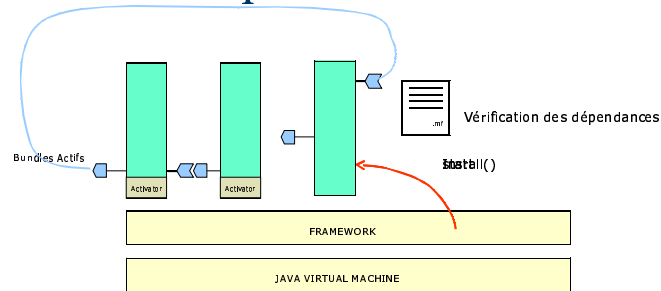


Cycle de vie et état d'un bundle





Scénario: Import de classe



19



Liaison entre packages

■ Déclaration de packages

<i>Bundle-Name</i>		<i>Nom du bundle</i>
<i>Bundle-Description</i>		<i>Chaîne de description du bundle</i>
<i>Bundle-Activator</i>		<i>Nom de la classe Activator</i>
Import-Package		Packages requis (avec/sans la version de spécification)
Export-Package		Packages fournis (avec/sans la version de spécification)

■ Oscar

– > exports

- il peut ne pas y avoir d'Activator dans un bundle





20



Emballage de librairie

- Un bundle peut « emballer » une librairie (un jar) afin de mettre à disposition les classes qu'elle contient (notion de bibliothèque)
- L'archive est stockée telle quelle dans le bundle conteneur

<i>Bundle-Name</i>	<i>Nom du bundle</i>
<i>Bundle-Description</i>	<i>Chaîne de description du bundle</i>
<i>Bundle-Activator</i>	<i>Nom de la classe Activator</i>
<i>Import-Package</i> 	<i>Packages requis (avec/sans la version de spécification)</i>
<i>Export-Package</i> 	<i>Packages fournis (avec/sans la version de spécification)</i>
Bundle-ClassPath	Emplacement des classes et ressources du bundle



21



Bundle-Classpath

- Représente (dans le manifeste) les chemins (dans le JAR) de recherche des classes et des ressources
- 3 cas
 - Bundle-Classpath: . ou Pas de Bundle-Classpath
 - Recherche dans le JAR
 - Bundle-Classpath: .,demo/nested.jar,test/nest.jar
 - Recherche dans le JAR puis dans le JAR inclus
 - Bundle-Classpath: demo/nested.jar
 - Recherche dans le JAR inclus
 - Aucune classe ou ressource n'est recherchée dans le JAR
- Exemple :
 - parser xml, moteurs de jsp, annuaire ldap, serveur jmx, encodeur mp3



22



Chargement de classes (i)

- 1 ClassLoader par Bundle
 - Chargement, Mise à Jour, Déchargement
- Principe de la recherche des classes
 - La classe est dans le JRE
 - La classe est dans un package ni importé ni exporté
 - Utilisation de la classe chargée à partir du BUNDLE-CLASSPATH
 - La classe est dans un package importé
 - Utilisation de la classe chargée par le CL d'un autre bundle
 - La classe est dans un package exporté mais déjà exporté par un autre bundle
 - Utilisation de la classe chargée par le CL de l'autre bundle
 - La classe est dans un package exporté mais non exporté par un autre
 - Utilisation de la classe chargée à partir du BUNDLE-CLASSPATH





Emballage de code natif

- Osgi peut contrôler le chargement de bibliothèques de code natif
 - Soit pour que java puisse faire des appels natifs (interface JNI)
 - Soit pour charger des drivers pour le systèmes d'exploitation (encodeur mp3)
- Le framework sélectionne automatiquement la librairie à charger en mémoire.



Gestion du code natif

<i>Bundle-Name</i>	<i>Nom du bundle</i>
<i>Bundle-Description</i>	<i>Chaîne de description du bundle</i>
<i>Bundle-Activator</i>	<i>Nom de la classe Activator</i>
<i>Import-Package</i> 	<i>Packages requis (avec/sans la version de spécification)</i>
<i>Export-Package</i> 	<i>Packages fournis (avec/sans la version de spécification)</i>
<i>Bundle-ClassPath</i>	<i>Emplacement des classes et ressources du bundle</i>
Bundle-NativeCode	Bibliothèques natives à charger fonction du proc, du SE, ...



25



Bibliothèques natives

- Bundle-NativeCode dans le MANIFEST
 - Spécifie l'emplacement des bibliothèques dépendantes du système et du processeur, à charger dynamiquement (par le ClassLoader)

Bundle-NativeCode: `com/mycomp/impl/nativesample/libnat.so;`
`osname=Solaris;`
`processor=sparc;`
`osversion=5.5;`
`com/mycomp/impl/nativesample/libnat.so;`
`osname=SunOS;`
`processor=sparc;`
`osversion=2.5;`
`com/mycomp/impl/nativesample/nat.dll;`
`osname=Windows NT;`
`processor=x86;`
`osversion=4.0`

- Propriétés du framework
`org.osgi.framework.processor`, `org.osgi.framework.language`,
`org.osgi.framework.os.name`, `org.osgi.framework.os.version`



26



Résumé avant la suite

- Activator : « poignée » pour démarrer un bundle
- import-export : technique pour mettre à disposition dynamiquement de nouvelles classes
- Bundle-classpath : permet d'emballer une librairie dans un bundle
- Native-Code : permet d'emballer du code natif