

tesi di laurea

Servizi di verifica runtime di Ambienti Intelligenti

Anno Accademico 2009/2010

relatore

Ch.mo prof. Porfirio Tramontana

correlatore

Ch.mo prof. Antonio Coronato

candidato

Alessio Priore

Matr. 534/002061

Obiettivi del lavoro di tesi

- ✓ Integrazione ed implementazione di servizi OSGi per la verifica runtime di applicazioni Ambient Intelligent.
- ✓ Il lavoro svolto si colloca nell'ambito del progetto "Uranus" attualmente in fase di sviluppo presso l'ICAR-CNR (Napoli) per l'assistenza sanitaria.



Caratteristiche fondamentali di un Aml:

- semplicità
- affidabilità
- sicurezza
- interoperabilità

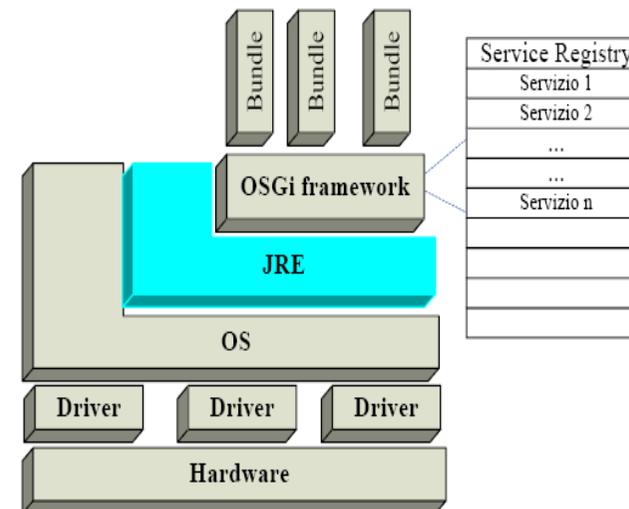


Caratteristiche:

- Indipendenza dalla piattaforma;
- Indipendenza dall'applicazione
- sicurezza
- servizi multipli
- molteplicità di tecnologie legate alle reti
- molteplicità di tecnologie legate alla progettazione dei dispositivi;
- coesistenza di vari standard

Architettura OSGi:

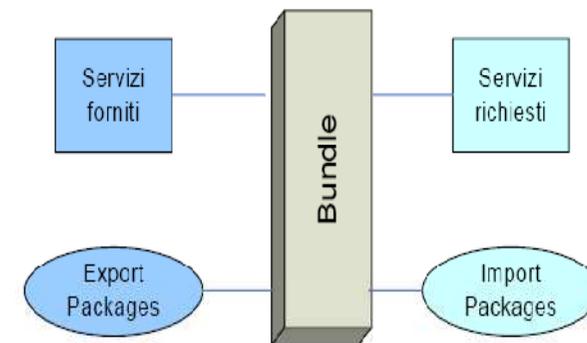
- OSGi Framework
- Servizi
- Bundle



Bundle:

Un bundle è un file java archive (jar) nel quale:

- Sono contenute le risorse per implementare uno o più servizi.
- C'è un file manifest.
- È implementata la classe activator.
- Può esserci la directory opzionale OSGI-OPT



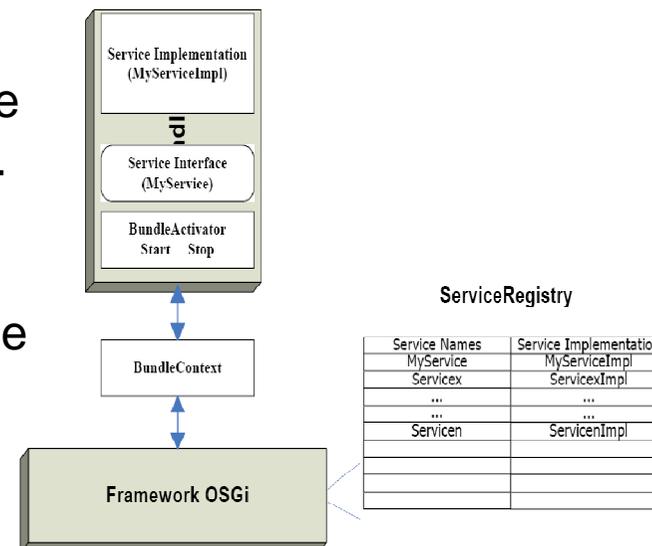
File Manifest:

- I parametri del file manifest sono dichiarati da una lista di coppie *nome_header: lista_valori*.

```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: OSGI Logservice over Slf4j Plug-in
Bundle-SymbolicName: org.ekkehard.osgi.over.slf4j;singleton:=true
Bundle-Version: 1.0.0
Bundle-Vendor: org.ekkehard
Bundle-RequiredExecutionEnvironment: J2SE-1.5
Import-Package: org.eclipse.equinox.log,
org.ekkehard.core,
org.osgi.framework,
org.osgi.service.log,
org.osgi.util.tracker,
org.slf4j;version="1.5.3"
Bundle-Activator: org.ekkehard.osgi.over.slf4j.Activator
```

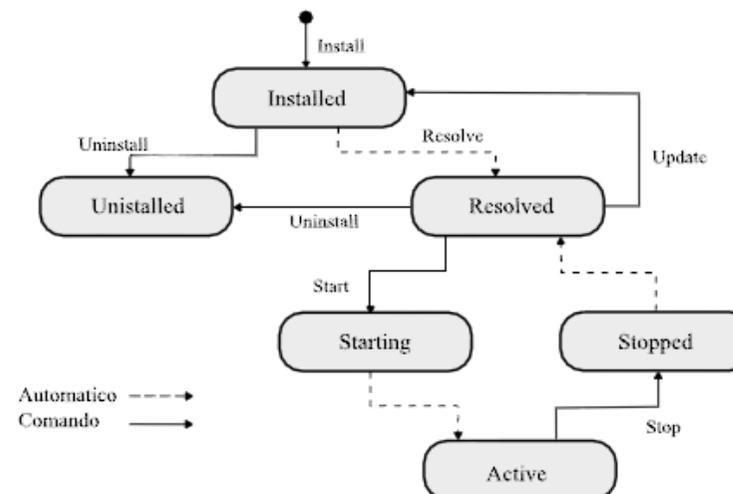
Bundle Context:

- L'interfaccia BundleContext rappresenta l'ambiente di esecuzione del bundle all'interno del framework.
- È creato dal framework quando un bundle viene attivato.
- Il BundleContext è usato dall'activator di un bundle per interagire con il framework:
 - registrazione dei servizi
 - ricerca dei servizi
 - gestione degli eventi
 - gestione della persistenza



Life Cycle:

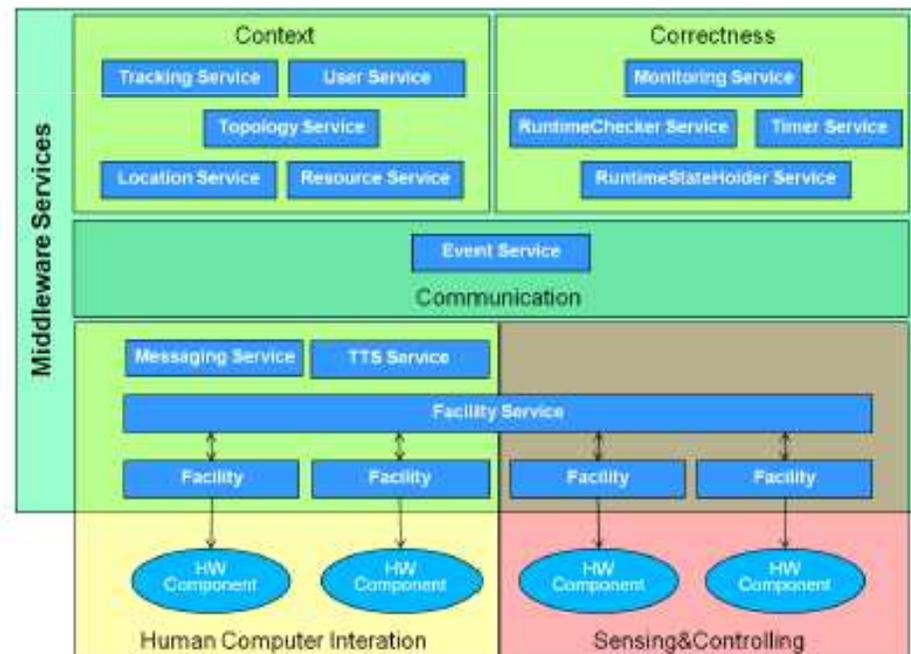
- Installazione
- Registrazione
- Attivazione
- Disattivazione



Piattaforma Uranus:

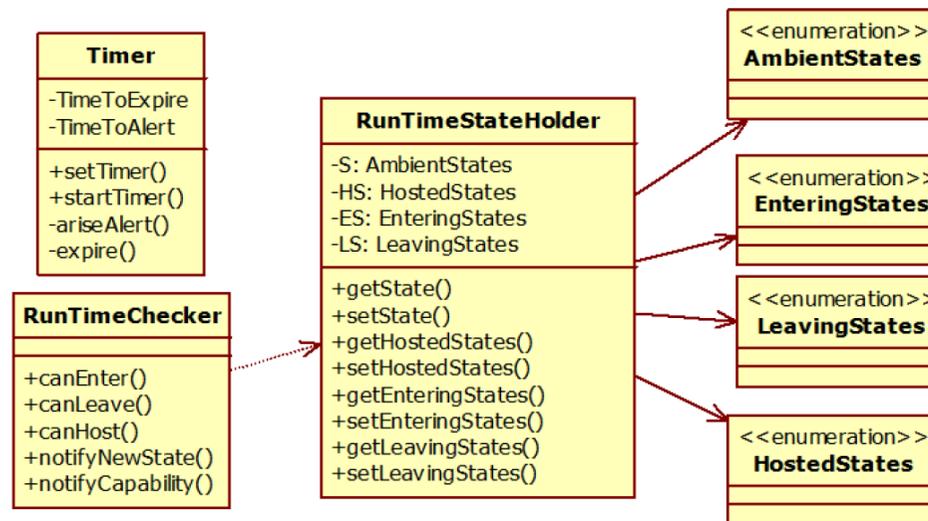
Servizi offerti dalla sezione del context:

- Topology Service
- Tracking Service
- Location Service
- User Service
- Resource Service



Servizi per la verifica della correttezza:

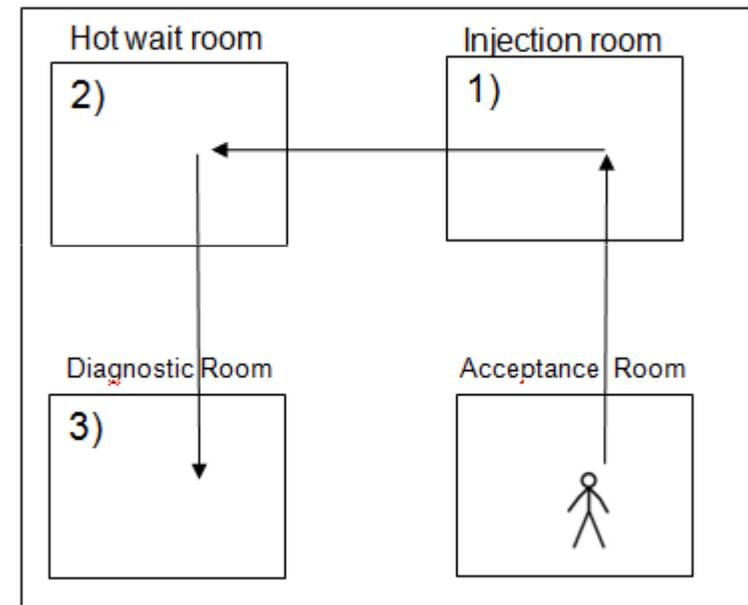
- RunTimeStateHolder Service
- RunTimeChecker Service
- Timer Service
- Monitoring
- LocationService
- MessagingService



Caso studio (1/2)

Descrizione dello scenario:

- Topologia del dipartimento di medicina nucleare.
- Il dipartimento si compone delle seguenti camere:
 - Acceptance Room
 - Injection Room
 - Hot Waiting Room
 - Diagnostic Room.
- Sono disponibili servizi e risorse di vario genere



Stati del paziente:

- WaitingToBeinjected (W)
- JustInjected (I)
- Ready (R)

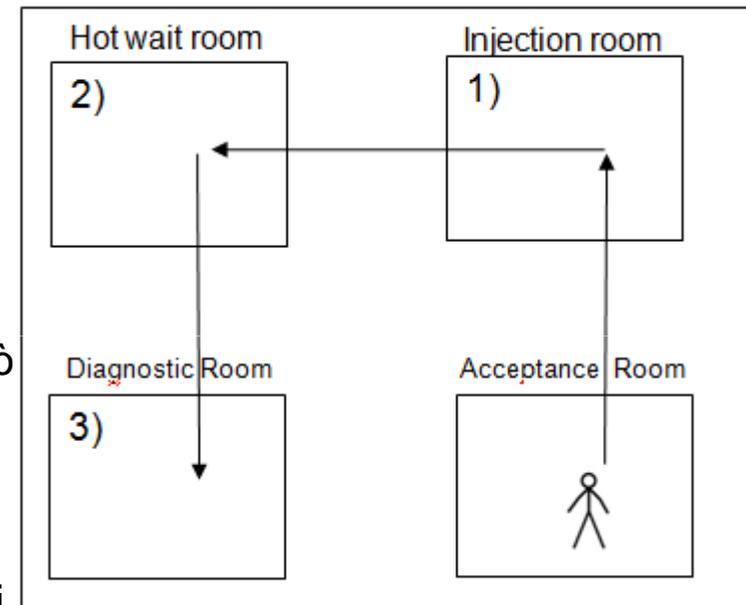
Caso studio (2/2)

Stati del paziente (continua):

- Examined (E)
- Decayed (D)

Descrizione degli eventi:

- 1) RunTimeStateHolder verifica se il paziente può accedere alla I.R.
- 2) Il paziente cambierà il suo stato in ready (R), examined (E) o decayed (D).
- 3) Nella DiagnosticRoom sono eseguite le analisi.



Il paziente può entrare, uscire o rimanere nella H.W.R. solo se si verificano le seguenti condizioni:

- ✓ Patient canEnter(HotWaitingRoom), iff Patient $\in \{I, E\}$
- ✓ Patient canLeave(HotWaitingRoom), iff Patient $\in \{R, D\}$
- ✓ HotWaitingRoom canHost(Patient), iff Patient $\in \{I, R, E, D\}$



Conclusioni

Obiettivi raggiunti:

- ✓ Realizzazione ed implementazione in Java di servizi OSGi per la verifica a runtime.
- ✓ Integrazione di nuove funzionalità per servizi esistenti

Sviluppi futuri:

- ✓ Sviluppo di una base di dati dalla quale caricare tutti i dati relativi alle varie location, risorse e utenti
- ✓ Rendere indipendente l'implementazione dei servizi OSGi dalla una JavaVirtualMachine e di un'implementazione di framework OSGi
- ✓ Un meccanismo di comunicazione con un database che può essere **JDBC**(Java Database Connectivity):un pacchetto applicativo(API) per collegarsi a database relazionali SQL.