



**PE INTERNATIONAL**  
SUSTAINABILITY PERFORMANCE

# OSGi goes Enterprise

Eclipse Demo Camp 2013

# Standards

OSGi Enterprise

and

Java EE



# Eclipse Gemini



## Java Persistence API

## Blueprint

DB Access

Naming

Web

Management

# DB Access

## Example

```
public DataSource createDataSource() throws SQLException {  
    DataSourceFactory dataSourceFactory = getDataSourceFactory();  
    Properties properties = new Properties();  
    properties.put(JDBC_URL, "jdbc:postgresql://localhost:5432/gemini");  
    properties.put(JDBC_USER, "admin");  
    properties.put(JDBC_PASSWORD, "secret");  
    DataSource dataSource = dataSourceFactory.createDataSource(properties);  
    return dataSource;  
}
```

# DB Access

## Services

{org.osgi.service.jdbc.DataSourceFactory}

={osgi.jdbc.driver.class=org.h2.Driver,  
osgi.jdbc.driver.name=H2}

Registered by bundle: org.eclipse.gemini.dbaccess.h2\_1.1.0 [164]

Bundles using service:

org.eclipse.gemini.jpa\_1.0.0.RELEASE [38]

{org.osgi.service.jdbc.DataSourceFactory}

={osgi.jdbc.driver.class=org.apache.derby.jdbc.EmbeddedDriver,  
osgi.jdbc.driver.version=3.0,  
osgi.jdbc.driver.name=Derby}

Registered by bundle: org.eclipse.gemini.dbaccess.derby\_1.0.0 [83]

No bundles using service

# JPA

## Example

Manifest-Version: 1.0

Bundle-ManifestVersion: 2

Meta-Persistence: META-INF/persistence.xml

Bundle-Name: PE Jpa Model

Bundle-SymbolicName: com.pe-international.sample.model.jpa

Bundle-Version: 1.0.0.qualifier

Bundle-Vendor: PE-INTERNATIONAL

Import-Package: javax.persistence;jpa="2.0";version="1.1.0",  
javax.persistence.criteria;jpa="2.0";version="1.1.0",  
javax.persistence.metamodel;jpa="2.0";version="1.1.0",  
model.account,  
org.osgi.framework;version="1.6.0",  
org.osgi.service.jpa,  
org.osgi.util.tracker

Bundle-RequiredExecutionEnvironment: JavaSE-1.6

Bundle-Activator: sample.Activator

Export-Package: model.account.jpa

# JPA

## Services

{`javax.persistence.EntityManagerFactory`}

={`osgi.managed.bundles=com.pe-international.sample.model.jpa_1.0.0`,  
`osgi.unit.version=1.0.0.qualifier`,  
`osgi.unit.name=Accounts`,  
`osgi.unit.provider=org.eclipse.persistence.jpa.PersistenceProvider`}

Registered by bundle: `com.pe-international.sample.model.jpa_1.0.0` [67]

Bundles using service:

`com.pe-international.sample.model.jpa_1.0.0.qualifier` [67]

{`org.osgi.service.jpa.EntityManagerFactoryBuilder`}

={`osgi.managed.bundles=com.pe-international.sample.model.jpa_1.0.0`,  
`osgi.unit.version=1.0.0.qualifier`,  
`osgi.unit.name=Accounts`,  
`osgi.unit.provider=org.eclipse.persistence.jpa.PersistenceProvider`}

Registered by bundle: `org.eclipse.gemini.jpa_1.0.0` [38]

No bundles using service.

# Dependency Injection

Blueprint

and

Declarative Services



# Blueprint

## History

2003 Service Binder

2005 Declarative Services with OSGi R4

2006 Spring Dynamic Modules

2009 Donated to Eclipse as Blueprint

2011 Blueprint Container with OSGi R4.2

# Blueprint

## Example

```
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <!-- Consume OSGi Service -->
  <reference id="entityManagerFactory"
            interface="javax.persistence.EntityManagerFactory"
            filter="(osgi.unit.name=Accounts)" />

  <!-- Create Service Instance -->
  <bean id="customerRepository"
        class="com.pe.sample.service.CustomerRepositoryImpl"
        <property name="entityManagerFactory" ref="entityManagerFactory" />
  </bean>

  <!-- Publish OSGi Service -->
  <service ref="customerRepository" auto-export="interfaces" />

</blueprint>
```

# Blueprint

## Constructors

```
// new model.account.jpa.CustomerImpl()
<bean class="model.account.jpa.CustomerImpl"/>

// new model.account.jpa.CustomerImpl("Zaphod")
<bean class="model.account.jpa.CustomerImpl"
  <argument value="Zaphod"/>
</bean>

// model.account.CustomerFactory.createCustomer("Zaphod")
<bean class="model.account.CustomerFactory"
  factory-method="createCustomer">
  <argument value="Zaphod"/>
</bean>

// new model.account.CustomerFactory().createCustomer("Zaphod")
<bean id="customerFactory" class="model.account.CustomerFactory"/>
<bean factory-ref="customerFactory"
  factory-method="createCustomer">
  <argument value="Zaphod"/>
</bean>
```

# Blueprint

## Beans

```
<bean class="model.account.jpa.CustomerImpl">
  // Reference to other bean
  <property name="wife" ref="wifeId" />

  // String as value
  <property name="customerId" value="42"/>

  // List as value
  <property name="accountIds">
    <value>
      <list>
        <value>Acc-43</value>
        <value>Acc-87</value>
      </list>
    </value>
  </property>

  // Map as value
  <property name="accountIdsByProduct">
    <value>
      <map>
        <entry key="DEPOT" value="Acc-43"/>
      </map>
    </value>
  </property>
</bean>
```

# Blueprint

## Services

```
// Publish a service with properties
<service ref="tableView" auto-export="interfaces">
  <service-properties>
    <entry key="view.id" name="customerTable">
  </service-properties>
</service>
```

```
// Reference single service
<reference id="customerTableView"
           interface="ui.ViewContribution"
           filter="(view.id=customerTable)" />
```

```
// Reference multiple services
<ref-list id="allViewContributions"
          interface="ui.ViewContribution" />
```

# Thank you!

Jan Stamer

Senior Software Engineer

[j.stamer@pe-international.com](mailto:j.stamer@pe-international.com)



**PE INTERNATIONAL**  
SUSTAINABILITY PERFORMANCE

