nuxeo | Professional Open Source ECM

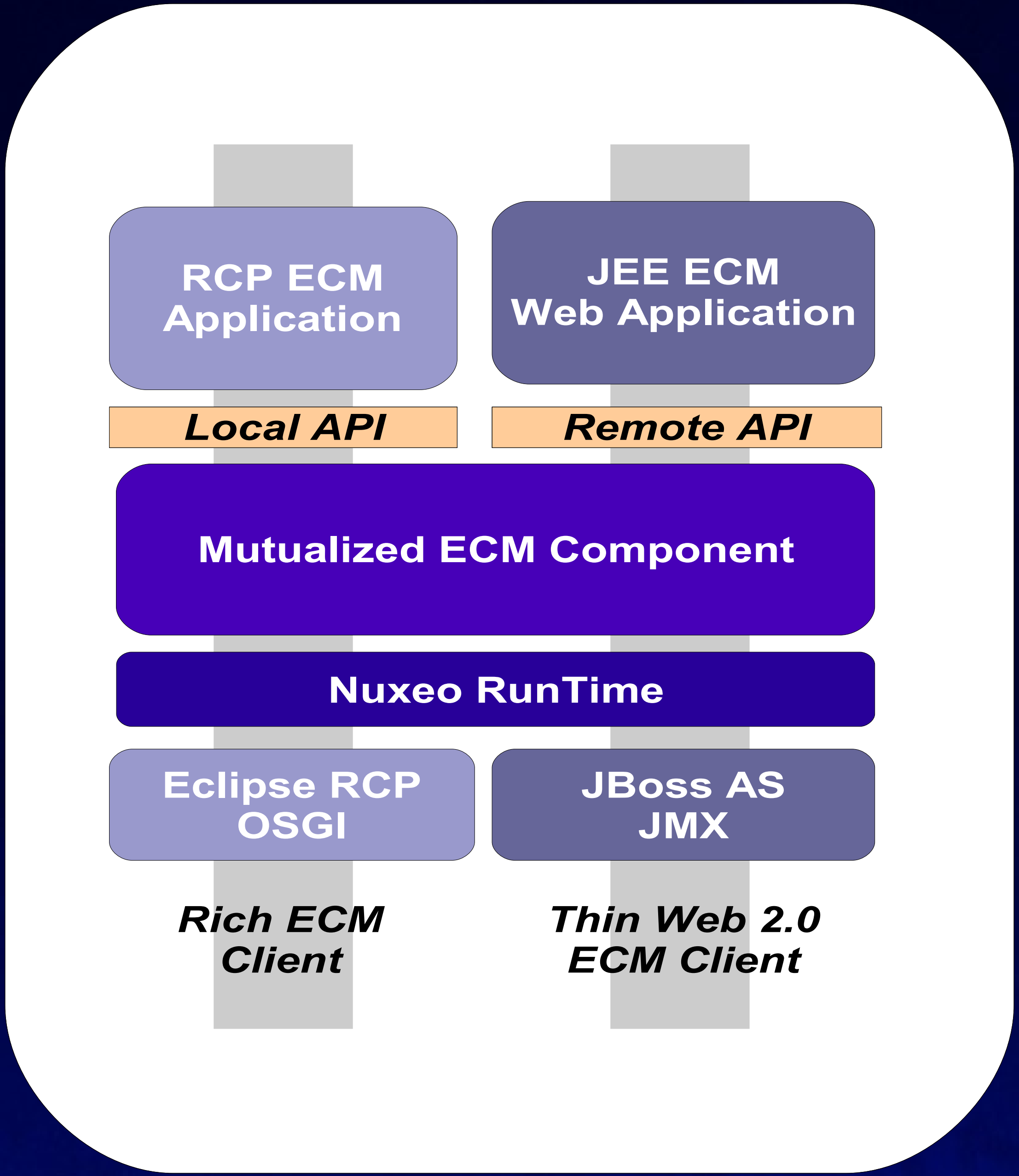
Nuxeo Runtime

Inconvénients de J2EE

- Manque de flexibilité au niveau du déploiement
  - Les descripteurs des modules (XML) ne permettent pas la fragmentation des modules.
  - Références en dur vers les noms des packages dans les descripteurs XML des modules ou dans le code
  - Impossible de rajouter une fonctionnalité sans devoir reconstruire et redéployer entièrement l'application
- Pas de modèle d'extensions (plugins)
  - Impossible de rajouter dynamiquement des extensions sans modifier les modules déjà déployés
  - Pas de notion de dépendance entre modules
  - Pas de support pour le versioning des modules
- Modules difficilement réutilisables sur autres platformes non

Motivations for NXRuntime

- Deploy Nuxeo Components on multiple targets
  - OSGi and JBoss as a first step



Motivations for NXRuntime

- Overcome JEE deployment limitations
  - monolithic descriptors (like web.xml)
    - can't have multiple components contributing
  - one archive for one web application
    - can't have a modular webapp without rebuilding each time
  - no declarative dependency between modules
- Overcome JEE component model limitations
  - unable to deploy a new component without rebuilding the while application
  - no support for module versionning

Motivations for NXRuntime

- Provide a base component model for Nuxeo
  - that is platform agnostic
  - that supports hot deployment
  - that supports dependencies management
  - that includes a plugin model
    - define extension point
    - contribute plugins

NXRuntime

- NXRuntime
  - Component model
  - Extension points
- OSGiAdapter
  - OSGi subset implementation
- NXJBossRuntime
  - NXRuntime component adapter on top JBoss JMX (MBean)
- NXJBossExtensions
  - Custom deployer for modular deployment (ear deployer)
  - Custom deployer ordering
  - Deployment listener (ex: OSGiAdapter activation)

NXRuntime – component model

- OSGi = Open Services Gateway initiative
  - http://www.osgi.org/
- OSGi features
  - modular deployment for
    - *bundles* (archives .JAR)
    - *fragments* (subparts of a master bundle)
  - an OSGi *bundle*) can define one or more services
  - a service is identified by an id (ex: org.nuxeo.ecm.core)
  - a bundle can declare it's dependecies
- NXRuntime implements a subset of OSGi

OSGi - Bundle Manifest

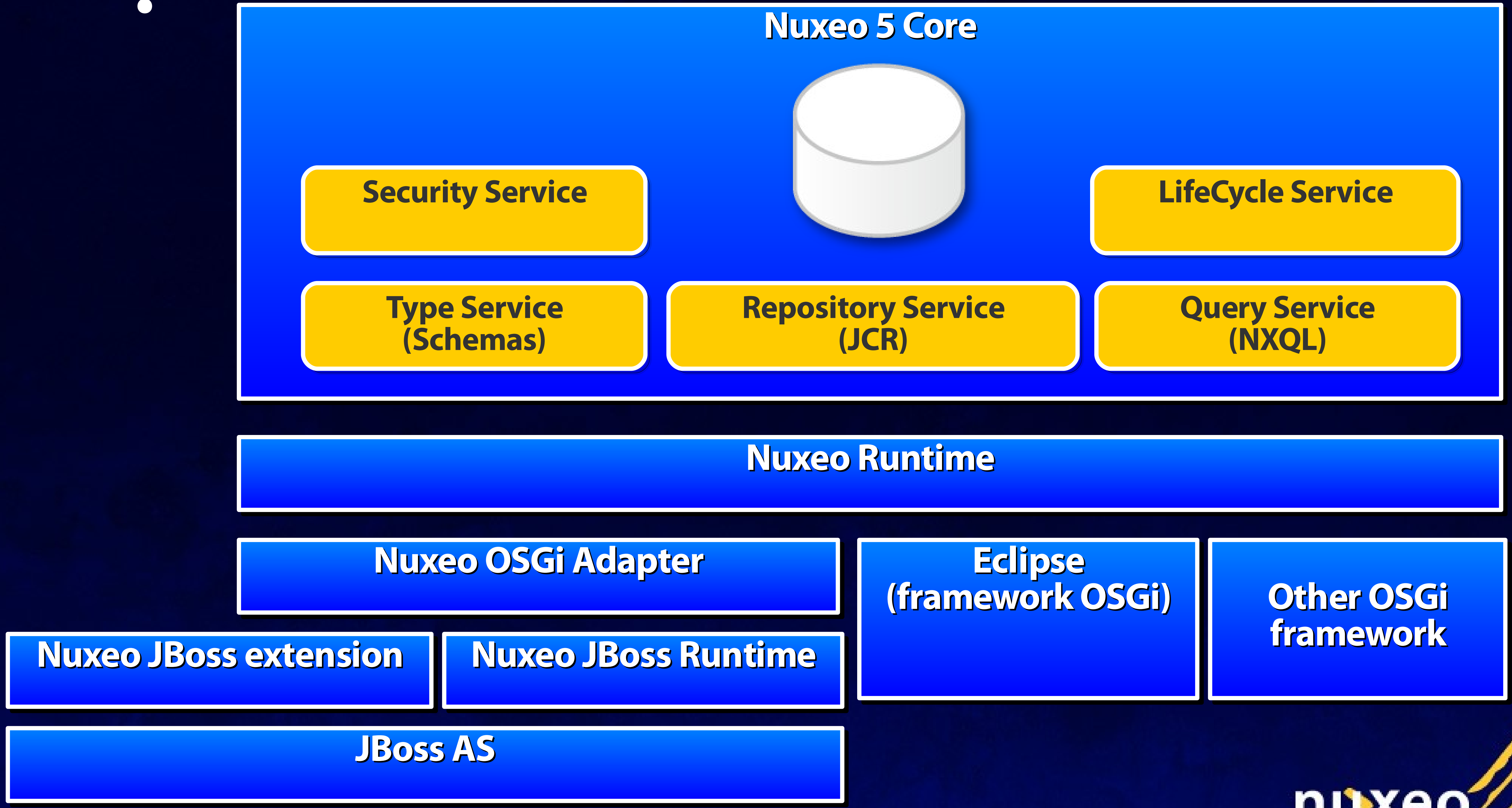
```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: Nuxeo ECM Core
Bundle-SymbolicName: org.nuxeo.ecm.core;singleton:=true
Bundle-Version: 1.0.0
Bundle-Vendor: Nuxeo
Bundle-Localization: bundle
Bundle-Activator: org.nuxeo.ecm.core.NXCoreActivator
Bundle-ClassPath: ., lib/xsom.jar,
 lib/connector-api.jar,
 lib/java-cup-v11a.jar
Export-Package: org.nuxeo.ecm.core,
 org.nuxeo.ecm.core.api,
 org.nuxeo.ecm.core.api.local,
 org.nuxeo.ecm.core.jca,
 org.nuxeo.ecm.core.lifecycle,
 org.nuxeo.ecm.core.model
Require-Bundle: org.nuxeo.ecm.core.api,
 org.nuxeo.runtime
Nuxeo-Component: OSGI-INF/CoreService.xml,
 OSGI-INF/TypeService.xml,
 OSGI-INF/RepositoryService.xml,
 OSGI-INF/CoreExtensions.xml,
 OSGI-INF/SecurityService.xml
```

NXRuntime – component model

- NXRuntime components are packaged as a bundle
  - the bundle and each component can declare it's dependencies
    - a component will wait for it's depencies before beeing started
  - each component is declared by an XML descriptor
    - declares extension points available
    - declares contribution to other extension points

Nuxeo Runtime – Nuxeo 5

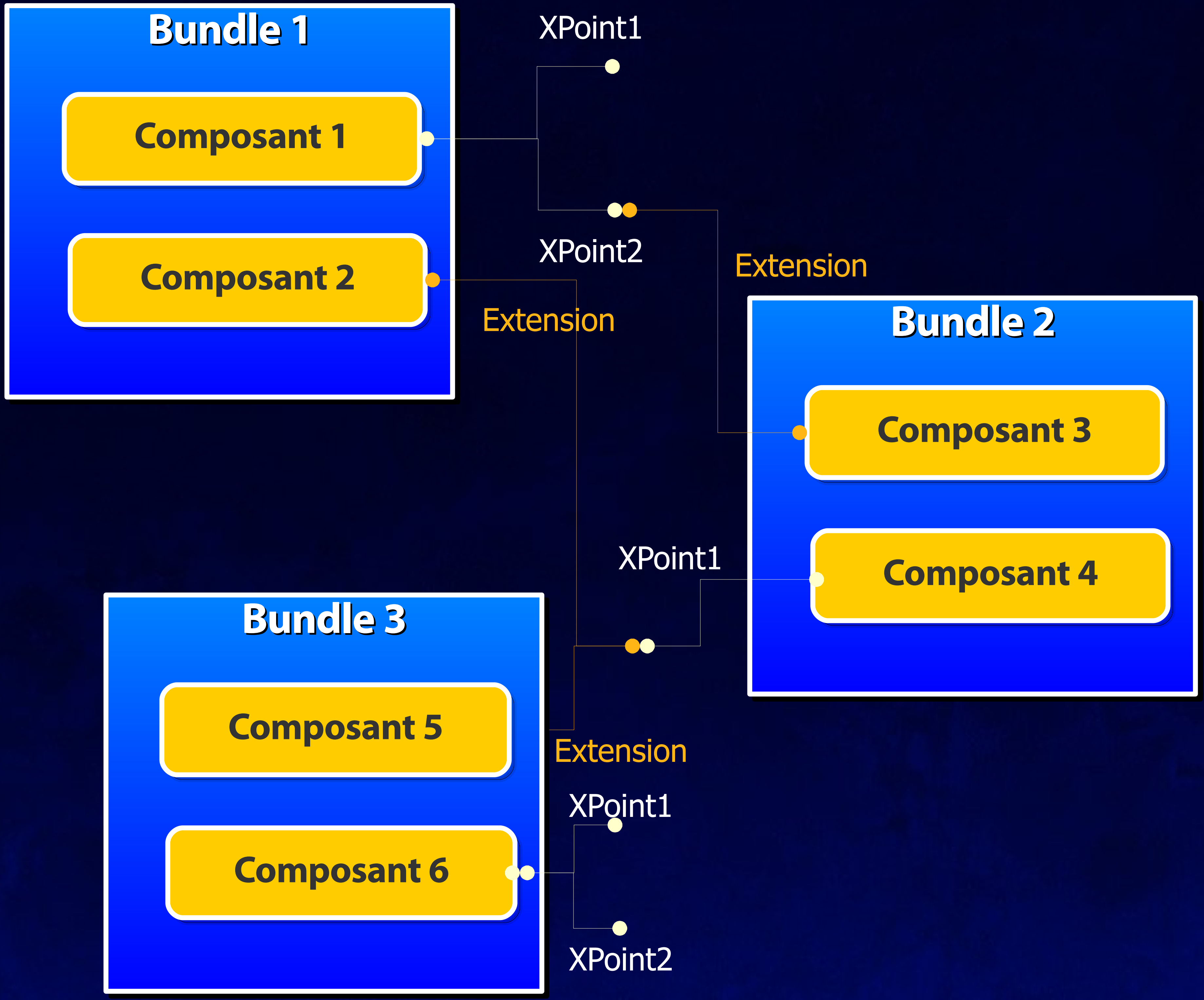
-



NXRuntime Extension Point

- Runtime includes an extension points system
  - Based on the Eclipse model
- You can define new extensions points
  - Let other components extend yours
- You can use extension points
  - Extend or configure via XML descriptors

Extension points



XML Descriptor example

dependencies

EP declaration

Contribution

Contribution

```xml
<?xml version="1.0"?>

<component name="org.nuxeo.ecm.core.listener.CoreEventListenerService">

    <require>org.nuxeo.ecm.core.repository.RepositoryService</require>

    <implementation class="org.nuxeo.ecm.core.listener.impl.CoreEventListenerServiceImpl"/>

    <extension-point name="listener">
        <object class="org.nuxeo.ecm.core.listener.extensions.CoreEventListenerDescriptor"/>
    </extension-point>

    <extension target="org.nuxeo.ecm.core.listener.CoreEventListenerService" point="listener">
        <listener    name="nxruntimelistener"
                     class="org.nuxeo.ecm.core.listener.impl.NXRuntimeEventListener"
        />
    </extension>

    <extension target="org.nuxeo.ecm.core.listener.CoreEventListenerService" point="listener">
        <listener    name="lifecyclelistener"
                     class="org.nuxeo.ecm.core.lifecycle.impl.LifeCycleListener"
        />
    </extension>

</component>
```

nuxeo

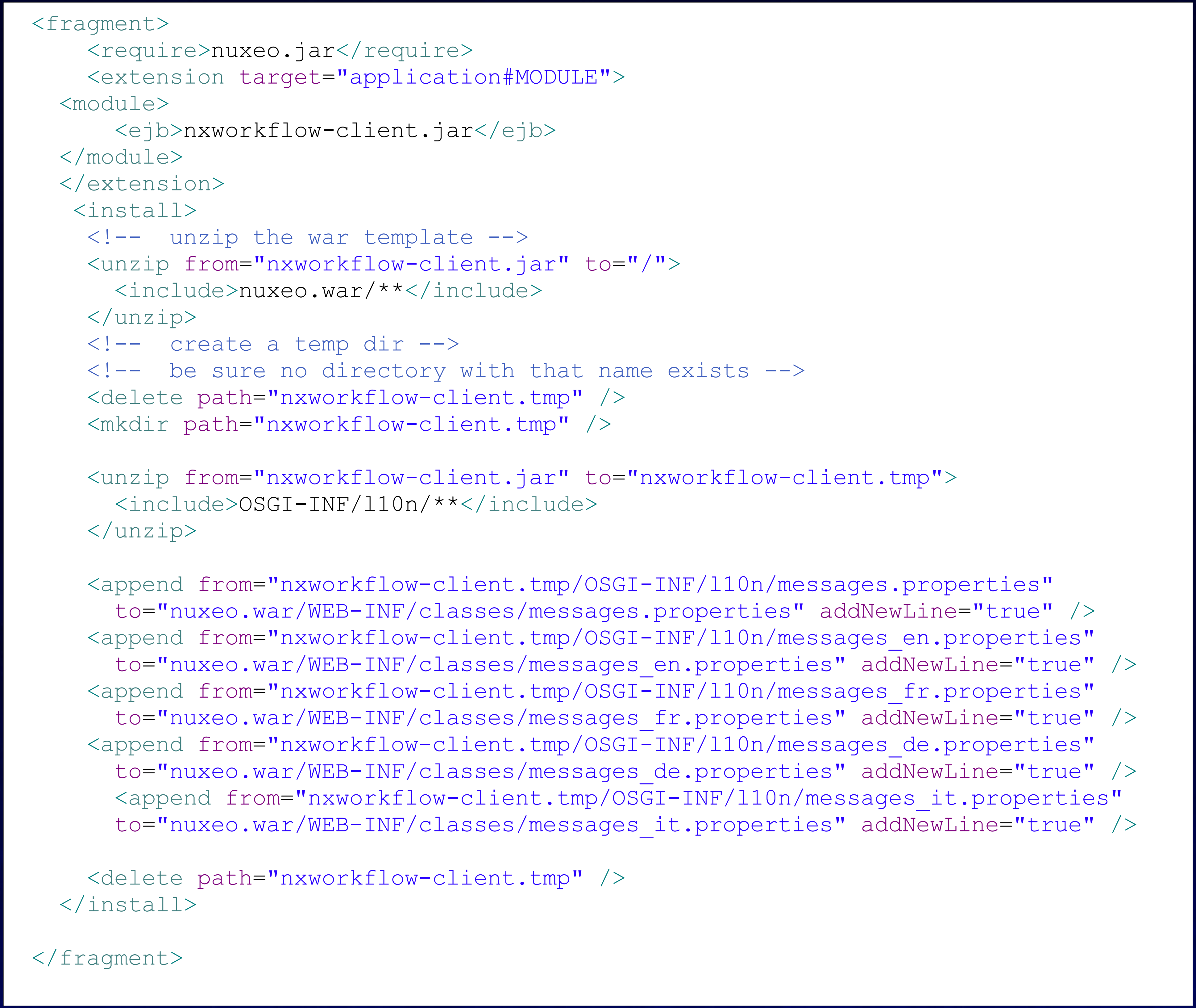
Extension examples

- Pluggable repository
- Schemas and content type registration
- Actions and views registration
- Register event handlers
- Register transformation plugins

Nuxeo Deployment

- Nuxeo Runtime let you deploy contributions to EP
- Nuxeo Runtime let you contribute to the main WebApp
  - contribute views
  - contribute partial configuration
    - web.xml
    - faces-config.xml
    - application.xml
    - ...

Contributing web files



```
<fragment>
    <require>nuxeo.jar</require>
    <extension target="application#MODULE">
  <module>
      <ejb>nxworkflow-client.jar</ejb>
  </module>
  </extension>
   <install>
    <!--  unzip the war template -->
    <unzip from="nxworkflow-client.jar" to="/">
      <include>nuxeo.war/**</include>
    </unzip>
    <!--  create a temp dir -->
    <!--  be sure no directory with that name exists -->
    <delete path="nxworkflow-client.tmp" />
    <mkdir path="nxworkflow-client.tmp" />

    <unzip from="nxworkflow-client.jar" to="nxworkflow-client.tmp">
      <include>OSGI-INF/l10n/**</include>
    </unzip>

    <append from="nxworkflow-client.tmp/OSGI-INF/l10n/messages.properties"
      to="nuxeo.war/WEB-INF/classes/messages.properties" addNewLine="true" />
    <append from="nxworkflow-client.tmp/OSGI-INF/l10n/messages_en.properties"
      to="nuxeo.war/WEB-INF/classes/messages_en.properties" addNewLine="true" />
    <append from="nxworkflow-client.tmp/OSGI-INF/l10n/messages_fr.properties"
      to="nuxeo.war/WEB-INF/classes/messages_fr.properties" addNewLine="true" />
    <append from="nxworkflow-client.tmp/OSGI-INF/l10n/messages_de.properties"
      to="nuxeo.war/WEB-INF/classes/messages_de.properties" addNewLine="true" />
      <append from="nxworkflow-client.tmp/OSGI-INF/l10n/messages_it.properties"
      to="nuxeo.war/WEB-INF/classes/messages_it.properties" addNewLine="true" />

    <delete path="nxworkflow-client.tmp" />
  </install>

</fragment>
```