

Versions des principales API de JEE / Jakarta EE

de JEE 1.0 à JEE 1.4 (avant 2005) : pas d'annotation , config. XML de dingues !!!

JavaEE 5 et JavaEE 6 : avec annotations et "SUN"

JavaEE 7 : "ORACLE" et "<http://xmlns.jcp.org>"

JavaEE 8 (2018) : "ORACLE"

Jakarta EE 8 (2019) : "Eclipse Foundation"

Jakarta EE 9 (2020) - *béta*: *javax.* to jakarta.* namespace change*.

En règle générale , un serveur d'une version N supporte du code standard (et des configurations "standards") en versions N-1 et N-2 .

Par contre un ancien code complémentaire (non standard) de l'époque N-1 est rarement parfaitement supporté par la version N (ex : extension richfaces3 pour JSF1 / JEE5 qui n'est plus supporté par JSF2 de JEE6) .

Variantes "Full-Profile" et "Web-Profile" :

Les versions Java EE 6 , 7 et 8 existent en au moins 2 déclinaisons :

variantes (profiles)	Contenu	Déploiement	Serveurs
Full-Profile	Toutes les api de Java-EE	.ear	Serveurs complets avec conteneur d'EJB et conteneur WEB
Web-Profile	Api "web" + CDI + JPA + EJB en version "Lite" et JAX-RS depuis JEE7	.war	Serveurs complets et également serveurs simplifiés avec conteneurs "web" seulement (ex : TomEE / tomcat EE)

Principales restrictions de la version "Lite" des EJB 3.1 et 3.2 :

- pas d'accès remote (@Local seulement)
- pas de MDB / JMS (EJB asynchrone avec files d'attentes)

Evolutions des principaux besoins :

Epoque	Application	Profile nécessaire suffisant
2000-2013 (n-tiers / SOA)	Remote EJB (RMI) , JMS 1.1 , web-services "SOAP" et JSF	Full-Profile
2014-2018 (WOA)	JSF et/ou JAX-RS + javascript	Web-Profile
2019-202x (micro-services)	JAX-RS + javascript (JSF)	Web-Profile et <i>Micro-Profile (jakarta EE)</i>

Attention : le "micro-profile" de JakartaEE est encore très récent (pas encore de recul , pas encore considéré comme "mature") !!!!

Tableau des principales versions

<i>API</i>	<i>Fonctionnalités</i>	<i>JEE5</i>	<i>JEE6</i>	<i>JEE7</i>	<i>JEE8 Jakarta</i>	<i>JEE9 Jakarta</i>
Servlet	Cœur web/http	2.5	3.0	3.1	4.0	5.0
JSP	Pages (coté serveur)	2.1	2.2	2.2	2.3	4.0
JSF	Framework Web sophistiqué (MVC, ...)	1.2	2.0	2.2	2.3	3.0
EJB	Objet/service métier , transaction , ...	3.0	3.1 3.1 Lite	3.2 3.2 Lite	3.2 3.2 Lite	4.0 4.0 Lite
JPA	Persistence / ORM java	1.0	2.0	2.1	2.2	3.0
JMS	Envoi et réception de messages (Queue)	1.1	1.1	2.0 et 1.1	2.0 et 1.1	3.0
JTA	Transaction distribuée	1.1	1.1	1.2	1.2	2.0
JSTL	Standard TagLib	1.2	1.2	1.2	1.2	2.0
JavaMail	Envoi et réception de mail	1.4	1.4	1.4	1.4 ou 1.5	1.5 ou 1.6
JCA	Connecteur JEE (syst. propriétaire , ...)	1.5	1.6	1.6	>= 1.6	>= 1.6
JAX-WS	web-services Soap	2.0	2.2	2.2	2.3	>= 2.3
JAX-RS	web-services REST		1.1	2.0	2.1	3.0
CDI	Injection de dépendance (inter-container)		1.0	1.1	2.0	3.0
Validation	Validation via annotations		1.0	1.1	2.0	3.0

Nouvelles API (en version 1.0) disponibles depuis JEE7 : Java Api for WebSocket , Java Api for JSON , Concurrency Utilities for JEE , Batch Applications for Java Platform

Principales nouveautés de JavaEE 8 :

- **HTTP/2** supporté par **Servlet 4** , PushBuilder dans Servlet 4
- JSON-B (Json-Binding) , un peu comme jackson-databind .
- améliorations de JAX-RS (new reactive client)
- CDI en v2 (utilisable en javaSE)
- **Java EE Security Api** (pour cloud , Paas)
- Bean validation 2.0 (avec alignement java8 (Optional , ...))

Namespaces XML pour les fichiers de configuration

Attention : La marque "SUN" a été reprise/rachetée en **2009** par la marque "Oracle" .

Les API les plus récentes ont maintenant des namespaces qui ont beaucoup évolués ("<http://java.sun.com>" devenu "<http://xmlns.jcp.org>")

Principales entêtes Xml des fichiers de configuration JEE :

web.xml (3.0 / JEE6)	<web-app xmlns:xsi=" http://www.w3.org/2001/XMLSchema-instance " xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd " version="3.0">
web.xml (3.1 / JEE7)	<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd " version="3.1">
web.xml (4.0 / JEE8)	<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd " version="4.0">

faces-config.xml (2.0 / JEE6)	<faces-config xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-facesconfig_2_0.xsd " version="2.0">
faces-config.xml (2.2 / JEE7)	<faces-config xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-facesconfig_2_2.xsd " version="2.2">
faces-config.xml (2.3 / JEE8)	<faces-config xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-facesconfig_2_3.xsd " version="2.3">

beans.xml (1.0 / JEE6)	<beans xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/beans_1_0.xsd " version="1.0">
beans.xml (1.1 / JEE7)	<beans xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/beans_1_1.xsd " version="1.1">
beans.xml (12.0 / JEE8)	<beans xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/beans_2_0.xsd " version="2.0">

persistence.xml (2.0 / JEE6)	<persistence xmlns=" http://java.sun.com/xml/ns/persistence " xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/persistence http://java.sun.com/xml/ns/persistence/persistence_2_0.xsd " version="2.0">
persistence.xml (2.1 / JEE7)	<persistence xmlns=" http://xmlns.jcp.org/xml/ns/persistence " xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd " version="2.1">
persistence.xml (2.2 / JEE8)	<persistence xmlns=" http://xmlns.jcp.org/xml/ns/persistence " xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence_2_2.xsd " version="2.2">

Serveur particulier "TomEE" (Tomcat EE)

(Tomcat + Jakarta EE = **TomEE**) est une **version dérivée de tomcat** qui est **accompagnée des Api du "web-Profile" de Java EE** (récemment renommé Jakarta EE) .

Autrement dit :

- le serveur traditionnel Tomcat ne comporte en lui que les api web fondamentales (servlet et jsp). c'est au développeur de récupérer des api complémentaires (JSF , JAX-RS, ...) via maven (en scope=compile)
- le serveur TomEE comporte déjà en lui les ".jar" pour les api standards de JEE (en version "WebProfile" (EJB-Lite , JSF , CDI , JAX-RS, ...) . Beaucoup de dépendances maven peuvent être exprimées en mode "provided" et certains fichiers de configuration (META-INF/persistence.xml , ...) sont bien interprétés .

Attention : La partie JPA de TomEE est basée sur OpenEJB / OpenJPA plutôt que sur Hibernate .

Variantes de TomEE (versions récentes 7 et 8) :

- **TomEE webProfile** (servlet/jsp , jax-rs / **REST**, jsf , jpa , cdi , ejbLite , ...)
- **TomEE plus** (= TomEE webProfile + JAX-WS / **SOAP** + JMS)
- TomEE MicroProfile (= TomEE webProfile + api récentes "jakarta microProfile")