

Version de Servlet dans WebSphere Liberty Profile

Compréhension de l'erreur

Le message d'erreur de `console.log` indique une `DDParser ParseException` liée au fichier `web-fragment.xml` dans le `spring-web-6.2.3.jar`. Plus précisément, il indique que le descripteur de déploiement spécifie la version "50"(qui correspond à Servlet 5.0), mais le serveur WebSphere Liberty Profile (WLP) est actuellement provisionné avec la version "40"(Servlet 4.0). Cette incompatibilité se produit parce que :

- **Servlet 5.0** fait partie de Jakarta EE 9 et est requis par les nouvelles versions de Spring Boot (par exemple, Spring Boot 3.x, qui inclut `spring-web-6.2.3.jar`).
- **Servlet 4.0** fait partie de Java EE 8, et WLP est probablement configuré avec la fonctionnalité `javaee-8.0`, qui ne prend pas en charge Servlet 5.0.

Pour corriger cela, vous devez aligner la version de Servlet prise en charge par WLP avec celle requise par votre application. La solution recommandée consiste à mettre à jour WLP pour prendre en charge Servlet 5.0 en activant la fonctionnalité `jakartaee-9.1`.

Solution : Mettre à jour WLP pour prendre en charge Servlet 5.0

Voici comment corriger le problème en mettant à jour WLP pour utiliser la fonctionnalité `jakartaee-9.1`, qui inclut la prise en charge de Servlet 5.0 :

1. Localiser le fichier `server.xml`

- Le fichier de configuration `server.xml` définit les fonctionnalités activées dans WLP.
- Il est généralement situé dans le répertoire du serveur, par exemple `/opt/ibm/wlp/usr/servers/myServer/server.xml`, où `myServer` est le nom de votre serveur.

2. Éditer le fichier `server.xml`

- Ouvrez le fichier `server.xml` dans un éditeur de texte.
- Localisez la section `<featureManager>`, qui liste les fonctionnalités activées pour le serveur. Elle pourrait ressembler à ceci :

```
<featureManager>
  <feature>javaee-8.0</feature>
</featureManager>
```

- Remplacez la fonctionnalité `javaee-8.0` par `jakartaee-9.1` pour activer la prise en charge de Servlet 5.0

:

```
<featureManager>
  <feature>jakartaee-9.1</feature>
</featureManager>
```

- Enregistrez le fichier.

3. Appliquer les modifications en mode développement WLP (si applicable)

- Si vous exécutez WLP en **mode développement** (par exemple, en utilisant la commande `wlp/bin/server run` ou une intégration IDE), vous pouvez appliquer les modifications comme suit :

- **Redémarrage manuel :**

- * Arrêtez le serveur :

```
/opt/ibm/wlp/bin/server stop myServer
```

- * Redémarrez le serveur :

```
/opt/ibm/wlp/bin/server start myServer
```

- **Rechargement à chaud en mode développement :**

- * Si WLP est en cours d'exécution en mode développement (par exemple, via `server run` ou un IDE), il peut détecter automatiquement les modifications apportées à `server.xml`. Cependant, pour vous assurer que la nouvelle fonctionnalité est chargée, un redémarrage est recommandé.

4. Vérifier la correction

- Après avoir redémarré le serveur, redéployez votre application (par exemple, en copiant le fichier WAR dans le répertoire `dropins` ou en utilisant votre méthode de déploiement).
- Vérifiez les journaux du serveur pour confirmer le déploiement réussi. Recherchez des messages comme :

```
[AUDIT    ] CWWKT0016I: Application web disponible (default_host) : http://localhost:9080/myapp/
```

```
CWWKZ0001I: Application myapp démarrée en X.XXX secondes.
```

- Accédez à votre application (par exemple, `http://localhost:9080/myapp/`) pour confirmer qu'elle fonctionne sans erreurs.

Solution alternative : Dégradation de Spring Boot

Si la mise à jour de WLP vers jakartaee-9.1 n'est pas une option (par exemple, en raison de contraintes de compatibilité avec d'autres applications), vous pouvez dégrader votre application Spring Boot vers une version compatible avec Servlet 4.0 :

1. Passer à Spring Boot 2.7.x

- Spring Boot 2.7.x prend en charge Servlet 4.0 (et les versions antérieures comme Servlet 3.1).
- Mettez à jour votre `pom.xml` pour utiliser Spring Boot 2.7.x :

```
<parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.7.18</version> <!-- Utilisez la dernière version 2.7.x -->
    <relativePath/>
</parent>
```

2. Reconstruire et redéployer

- Reconstruisez votre application :

```
mvn clean package
```

- Redéployez le fichier WAR généré vers WLP et vérifiez qu'il démarre sans erreurs.

Résumé

- **Cause de l'erreur** : Votre application (`spring-web-6.2.3.jar`) nécessite Servlet 5.0, mais WLP est configuré pour Servlet 4.0.
- **Correction préférée** : Mettez à jour WLP pour utiliser la fonctionnalité `jakartaee-9.1` en modifiant `server.xml` et en redémarrant le serveur.
- **Correction alternative** : Dégradez vers Spring Boot 2.7.x pour correspondre à la prise en charge de Servlet 4.0 de WLP.

La solution préférée (mise à jour de WLP) garantit que votre application peut utiliser les dernières fonctionnalités de Spring Boot 3.x et Jakarta EE 9. Suivez les étapes ci-dessus pour résoudre le problème et déployer votre application avec succès.