

## TERASOLUNA 2.0.5.0 移行ガイド(Web 版)

## 変更履歴

バージョン	日付	改訂箇所	改訂内容
2.0.5.0	2012/12/26	-	新規作成

## ■ 概要

本ドキュメントは、TERASOLUNA Server Framework for Java (Web 版) 2.0.4.0 で作成したアプリケーションを、2.0.5.0 へ移行する際の手順を示すドキュメントである。

### ◆ 2.0.5.0 の変更点概要

- Spring フレームワークのバージョンアップ (2.5.6 > 3.1.3)
- 依存ライブラリのバージョンアップ
- build.xml のクラスパス設定変更

## ■ 移行手順

2.0.4.0 から 2.0.5.0 への移行手順を説明する。

### 前提条件

TERASOLUNA Server Framework for Java (Web 版) 2.0.4.0 を使用したアプリケーションが正常に動作していること。

以後、本書ではこれをアプリケーションと呼称する。

#### ① blank プロジェクトのダウンロード

TERASOLUNA Server Framework for Java (Web 版) 2.0.5.0 の blank プロジェクト (terasoluna-server4jweb-blank\_2.0.5.0.zip) を以下の URL よりダウンロードし、任意のフォルダに展開する。

ダウンロードサイトの URL :

➤ SourceForge.jp <http://sourceforge.jp/projects/terasoluna/releases/>

## ② 依存ライブラリの変更

TERASOLUNA フレームワークや Spring の依存ライブラリの差し替えを行う。  
差し替えるライブラリは以下の通りである。

2.0.4.0のライブラリ	2.0.5.0のライブラリ	種別
<del>spring-2.5.6.SEC01</del>		削除
<del>spring-webmvc-struts-2.5.6.SEC01</del>		削除
commons-lang-2.3	commons-lang-2.5	差し替え
teralib-exception-1.0.1	teralib-exception-1.0.2	差し替え
teralib-filter-1.0.1	teralib-filter-1.0.2	差し替え
teralib-log-1.0.1	teralib-log-1.0.2	差し替え
terasoluna-commons-2.0.4.0	terasoluna-commons-2.0.5.0	差し替え
terasoluna-commonscreenflow-1.0.1	terasoluna-commonscreenflow-1.0.2	差し替え
terasoluna-dao-2.0.4.0	terasoluna-dao-2.0.5.0	差し替え
terasoluna-ibatis-2.0.4.0	terasoluna-ibatis-2.0.5.0	差し替え
terasoluna-taglibex-1.1.0	terasoluna-taglibex-1.1.1	差し替え
terasoluna-thin-2.0.4.0	terasoluna-thin-2.0.5.0	差し替え
terasoluna-web-cr-1.1.0	terasoluna-web-cr-1.1.1	差し替え
	aopalliance-1.0	追加
	spring-aop-3.1.3.RELEASE	追加
	spring-asm-3.1.3.RELEASE	追加
	spring-beans-3.1.3.RELEASE	追加
	spring-context-3.1.3.RELEASE	追加
	spring-core-3.1.3.RELEASE	追加
	spring-expression-3.1.3.RELEASE	追加
	spring-jdbc-3.1.3.RELEASE	追加
	spring-orm-3.1.3.RELEASE	追加
	spring-struts-3.1.3.RELEASE	追加
	spring-tx-3.1.3.RELEASE	追加
	spring-web-3.1.3.RELEASE	追加

「種別」に従って下記の修正を行う。

※「2.0.5.0 のライブラリ」は terasoluna-server4jweb-blank\_2.0.5.0.zip を展開したフォルダの lib フォルダ直下に格納されている。

**削除：**

アプリケーションの lib フォルダ直下から「2.0.4.0 のライブラリ」に記載した jar ファイルを削除する。

**差し替え：**

アプリケーションの lib フォルダ直下から「2.0.4.0 のライブラリ」に記載した jar ファイルを削除し、「2.0.5.0 のライブラリ」に記載した jar ファイルを追加する。

**追加：**

「2.0.5.0 のライブラリ」に記載した jar ファイルをアプリケーションの lib フォルダ直下へ追加する。

### ③ build.xml の修正

②依存ライブラリの変更に従って、build.xml のクラスパスの設定を修正する。  
アプリケーション直下の/ant/build.xml を②依存ライブラリの変更で使用した依存ライブラリ表を参照し、「種別」に従って下記の修正を行う。

#### 削除：

「2.0.4.0 のライブラリ」に記載した jar ファイルのクラスパス設定を削除する。

#### 差し替え：

「2.0.4.0 のライブラリ」に記載した jar ファイルから「2.0.5.0 のライブラリ」に記載した jar ファイルにクラスパス設定の jar ファイル名を変更する。

#### 追加：

「2.0.5.0 のライブラリ」に記載した jar ファイルのクラスパス設定を追加する。

### ●修正例●

#### 【2.0.4.0 の build.xml】

```
<!-- クラスパスの設定 -->
<property name="classpath.lib" value="
    ${lib.dir}/spring-2.5.6.SEC01.jar;
    ${lib.dir}/commons-jxpath-1.3.jar;
    ${lib.dir}/commons-lang-2.3.jar;
    ${lib.dir}/commons-logging-1.1.1.jar;
    * * * (中略) * * *
"/>
```

#### 【2.0.5.0 の build.xml】

```
<!-- クラスパスの設定 -->
<property name="classpath.lib" value="
    ${lib.dir}/commons-jxpath-1.3.jar;
    ${lib.dir}/commons-lang-2.5.jar;
    ${lib.dir}/commons-logging-1.1.1.jar;
    * * * (中略) * * *
"/>
```

### ④ 動作確認

アプリケーションを起動させて、問題なく動作することを確認する。  
フレームワークを經由せずに Spring を利用している処理が存在する場合は、Spring のバージョンアップによる影響が無いことを回帰試験等により確認すること。