Title: What is Bill-Of-Materials in Maven dependency management?

Post Body:

What is the significant use of Bill-Of-Materials features available in the new Spring release, basically the name sounds good but searched for significant but I fail to find it?

Accepted Answer:

The **Bill Of Materials** (aka *BOM*) is more a *Maven* concept than a *Spring* one. It has emerged with the *2.0.x* release of Maven when the `import` scope has been introduced.

This scope let you declare a *pom* dependency inside your `<dependencyManagement>` section which will result in a sort of merge of the dependency and its dependant project *dependencyManagement* section.

The *BOM* concept came to put that scope in practice, which is no more than a project with a *pom.xml* having as packaging nature **pom** and that has only the `<dependencyManagement>` sectin inside in which we declare all the project dependencies (in your case it should be *Spring* ones) that are aligned together and should work out of the box.

You can think of a *BOM* project as a dependency directory that declares all the related projects **versions** thus save your version resolution efforts.

One your *BOM* imported into your project descriptor, you no more need to declare the version when declaring a dependecy because it will be inherited from that imported pom and it should the be something like the following:

```
<dependency>   <groupId>org.springframework</groupId>   <artifactId>spring-core</artifactId> <!-- or any other spring related
```

Highest Rated Answer: None