Title: Replicating Maven 'dependencyManagement' tag from inside Gradle build
Post Body:

I am trying to follow [this Spring Boot/Vaadin guide](#) **however** I'm using Gradle, *not* Maven.

At the very top of that guide they say to use the following Maven XML:

```
<dependencyManagement>    <dependencies>        <dependency>            <groupId>com.vaadin</groupId>        <artifact
```

However I don't see a `dependencyManagement` task available via Gradle. So I ask: how do I replicate the same behavior as the `<dependencyManagement/>` XML element above over in 'Gradle land'?

Update: current attempt

```
dependencyManagement {      imports {            mavenBom 'com.vaadin:vaadin-bom:10.0.11'        } }
```

Only problem with this is that when I add it to my `build.gradle` and then run `./gradlew clean` I get the following Gradle error:

   '*Could not find method dependencyManagement() for arguments...*'

Accepted Answer:

This should give you a working build:

```
plugins {      // the Gradle plugin which provides the "dependencyManagement" block      id 'io.spring.dependency-management' ve
```

Run `./gradlew dependencies --configuration compileClasspath` to see that all dependencies are available on your Java compile classpath now.

Edited to reply to the question in the comments: indeed, the import of the BOM leads to a slightly different set of dependencies than would be used without it. You can see the dependencies difference like so:

1. `./gradlew dependencies --configuration compileClasspath > with-BOM.txt`
2. Remove the `dependencyManagement` block and add a version to the single dependency: `compile 'com.vaadin:vaadin-spring-boot-starter:10.0.11'`
3. `./gradlew dependencies --configuration compileClasspath > without-BOM.txt`
4. `diff -u with-BOM.txt without-BOM.txt`

You can see slight differences like `org.webjars.bowergithub.webcomponents:webcomponentsjs:1.2.6` being used with the BOM and version `1.2.2` without it. The reason for that can be found in [the BOM](#) where version `1.2.6` is defined and where the authors also mention the reason for that: "Transitive webjar dependencies, defined here for repeatable builds"

Highest Rated Answer:

Generally maven `dependencyManagement` tag is used to import bom or control transitive versions. Gradle does that with `platform` component, like below example.

```
dependencies {    implementation platform('com.vaadin:vaadin-bom:10.0.11')    implementation ('com.vaadin:vaadin-core') }
```

This does not force you to depend on spring plugin for importing bom.