

Spring-boot-starter-parent Example



Last Updated: August 30,
2020



By: Lokesh
Gupta



Spring Boot, Spring Boot
2



Maven
Basics

In this spring boot tutorial, we will learn about **spring-boot-starter-parent** dependency which is used internally by all spring boot dependencies. We will also learn what all configurations this dependency provides, and how to override them.

What is spring-boot-starter-parent dependency?

The **spring-boot-starter-parent** dependency is the parent POM providing dependency and plugin management for Spring Boot-based applications. It contains the default versions of Java to use, the default versions of dependencies that Spring Boot uses, and the default configuration of the Maven plugins.

Few important configurations provided by this file are as below. Please refer to this [link](#) to read the complete configuration.

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/20
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xs
<modelVersion>4.0.0</modelVersion>
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-dependencies</artifactId>
  <version>${revision}</version>
  <relativePath>../../spring-boot-dependencies</relativePath>
</parent>
<artifactId>spring-boot-starter-parent</artifactId>
```

```

<packaging>pom</packaging>
<name>Spring Boot Starter Parent</name>
<description>Parent pom providing dependency and plugin management for applicati
  built with Maven</description>
<properties>
  <java.version>1.8</java.version>
  <resource.delimiter>@</resource.delimiter> <!-- delimiter that doesn't clash w
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
  <maven.compiler.source>${java.version}</maven.compiler.source>
  <maven.compiler.target>${java.version}</maven.compiler.target>
</properties>

...

<resource>
  <directory>${basedir}/src/main/resources</directory>
  <filtering>true</filtering>
  <includes>
    <include>**/application*.yaml</include>
    <include>**/application*.yaml</include>
    <include>**/application*.properties</include>
  </includes>
</resource>

</project>

```

The `spring-boot-starter-parent` dependency further inherits from `spring-boot-dependencies`, which is defined at the top of above POM file at line number : 9.

This file is the actual file which contains the information of default version to use for all libraries. The following code shows the different versions of various dependencies that are configured in `spring-boot-dependencies`:

pom.xml

```

<properties>
  <!-- Dependency versions -->
  <activemq.version>5.15.3</activemq.version>
  <antlr2.version>2.7.7</antlr2.version>
  <appengine-sdk.version>1.9.63</appengine-sdk.version>
  <artemis.version>2.4.0</artemis.version>
  <aspectj.version>1.8.13</aspectj.version>
  <assertj.version>3.9.1</assertj.version>
  <atomikos.version>4.0.6</atomikos.version>

```

```

<bitronix.version>2.1.4</bitronix.version>
<byte-buddy.version>1.7.11</byte-buddy.version>
<caffeine.version>2.6.2</caffeine.version>
<cassandra-driver.version>3.4.0</cassandra-driver.version>
<classmate.version>1.3.4</classmate.version>

...
...
</properties>

```

Above list is very long and you can read complete list in this [link](#).

How to override default dependency version?

As you see, spring boot has default version to use for most of dependencies. You can override the version of your choice or project need, in `properties` tag in your project's `pom.xml` file.

e.g. Spring boot used default version of google GSON library as **2.8.2**.

```

<groovy.version>2.4.14</groovy.version>
<gson.version>2.8.2</gson.version>
<h2.version>1.4.197</h2.version>

```

I want to use **2.7** of gson dependency. So I will give this information in properties tag like this.

```

<properties>
  <java.version>1.8</java.version>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <gson.version>2.7</gson.version>
</properties>

```

Now in your eclipse editor, you can see the message as : The managed version is 2.7
The artifact is managed in org.springframework.boot:spring-boot-dependencies:2.0.0.RELEASE.

GSON resolved dependency

Drop me your questions in comments section.

Happy Learning !!

Was this post helpful?

Let us know if you liked the post. That's the only way we can improve.

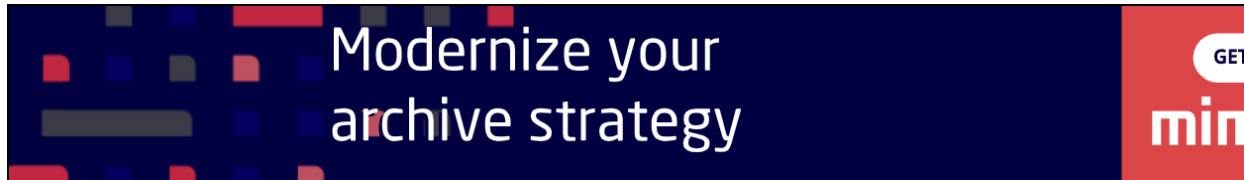
Yes

No

Recommended Reading:

1. [Spring-boot-starter Maven Templates](#)
2. [Maven – Parent and Child POM Example](#)
3. [Spring Boot war Packaging Example](#)
4. [Spring Boot Dev Tools Tutorial](#)
5. [Spring Boot Multi-Module Maven Project](#)
6. [Spring Boot2 @SpringBootApplication Auto Configuration](#)

7. [Spring Boot Remoting – Spring RMI annotation example](#)
8. [Spring retry module example with spring boot](#)
9. [Spring boot RSS feed with rome](#)
0. [Spring boot read file from resources folder](#)



Join 7000+ Awesome Developers

Get the latest updates from industry, awesome resources, blog updates and much more.

Email Address

Subscribe

** We do not spam !!*

5 thoughts on “Spring-boot-starter-parent Example”

Larry

January 7, 2020 at 7:20 am

Saying “...spring-boot-starter-parent dependency which is used internally by all spring boot dependencies” is a bit misleading, as spring-boot libraries themselves don't use spring-boot-starter-parent, they mostly use spring-boot-parent or spring-boot-starters as their parent.

Also, strictly speaking, in Maven terminology, it's not a “dependency”, it's a “parent”. A trivial distinction for those familiar with it, but for newcomers it can be distracting.

[Reply](#)

Lokesh Gupta

January 7, 2020 at 8:51 pm

Thanks for the feedback. I will update the post.

[Reply](#)

jueli lin

November 7, 2019 at 5:08 am

Is there a way to use spring-boot-starter-parent to manage only one module of a project, not the others(as they have been proved all working)?

[Reply](#)

sajeev

November 1, 2019 at 4:37 pm

above explanation your saying "spring-boot-starter-parent" is based and these two .xml file , from this two files we we get spring-boot-starter-parent libraries right

[Reply](#)

kabila

December 27, 2018 at 10:20 am

how to do postmapping in springboot using spring datajpa

[Reply](#)

Leave a Comment

☐ Add me to your newsletter and keep me updated whenever you publish new blog posts

Post Comment





A blog about Java and related technologies, the best practices, algorithms, and interview questions.

Meta Links

- [About Me](#)
- [Contact Us](#)
- [Privacy policy](#)
- [Advertise](#)
- [Guest Posts](#)

Blogs

[REST API Tutorial](#)



Copyright © 2022 · Hosted on [Cloudways](#) · [Sitemap](#)