

# Extending Spring Boot for the Enterprise

Billy Korando

Developer Advocate - IBM

@BillyKorando 

[william.korando@ibm.com](mailto:william.korando@ibm.com) 

# Subscribe to the Java Newsletter

<https://developer.ibm.com/newsletters/java/>

# For your Spring & JakartaEE needs

<https://cloud.ibm.com/docs/java>



think 2019

# Developer Night

Think. Code. Share.

An evening full of code,  
live demos and tech talks

June 13, Amsterdam

Register now:  
[ibm.biz/DevNight2019](http://ibm.biz/DevNight2019)

IBM



[DESIRE TO KNOW MORE INTENSIFIES]

<https://billykorando.com/>  
<https://developer.ibm.com/series/living-on-the-cloud/>

@BillyKorando

# Agenda

1. The Spring Boot philosophy
2. Improving project management
3. Improving project initialization

# About me

Over 10 years experience

Worked in various industries: insurance regulation, electronic health records, shipping, finance, and government, before switching to developer advocacy

@BillyKorando

Software  
development  
itself is often  
pleasant

Much around  
software  
development is  
painful





**Holly Cummins** @holly\_cummins · Apr 4

“Engineers don’t hate process. They hate process that can’t defend itself.”  
- Michael Lepp

The talk three before mine by @RianVDM at @LandingFest is a great complement to my talk about fun in the workplace.  
Recommended.



**"How to build successful products by prioritizing te...**

Most product management training and advice focuses outward. We are taught to care about the market above all else, and we learn the ins and outs of product di...

slideshare.net



1



2



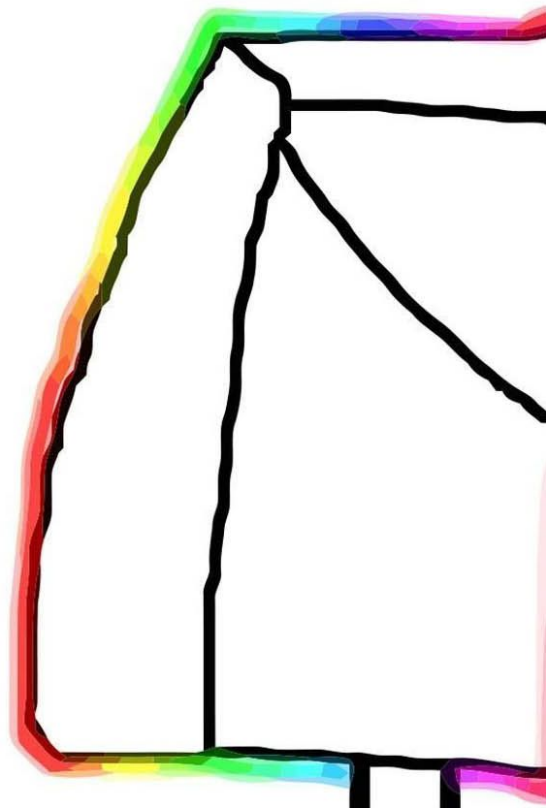
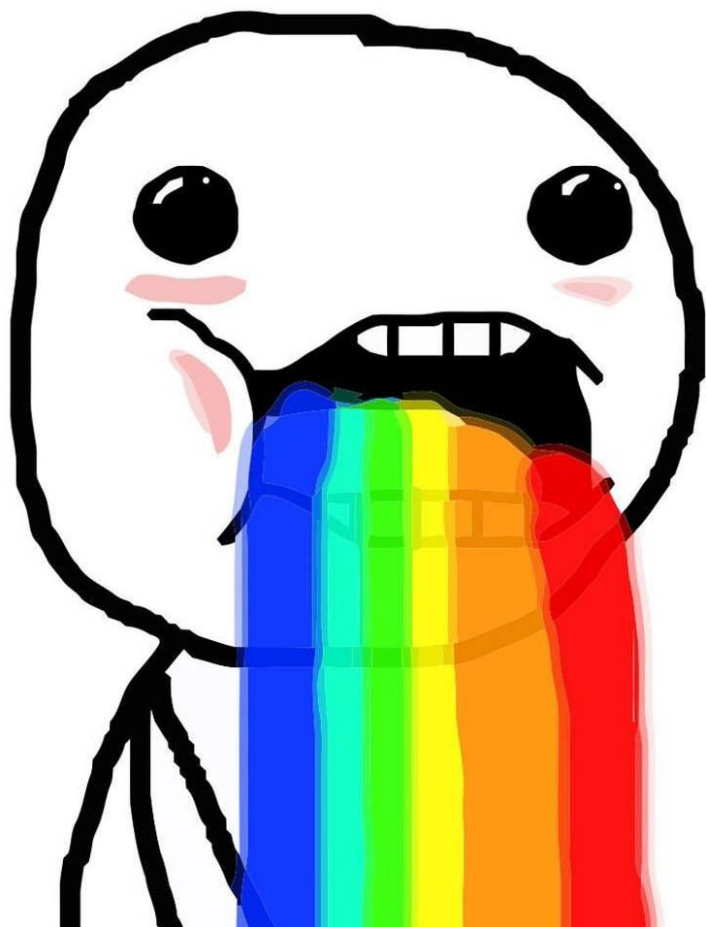
8



[https://twitter.com/holly\\_cummins/status/1113836034276634624](https://twitter.com/holly_cummins/status/1113836034276634624)

@BillyKorando

# Springtime for software development



# Opinionated

@BillyKorando



Phil Webb



@phillip\_webb

Following



For those on [reddit.com/r/java/](https://reddit.com/r/java/) saying it's Spring Boot is "the framework for a framework" here's a diagram:



7:54 PM - 8 Sep 2015

@BillyKorando

You're going to get cake (Spring)

But you have a say in the cake you want



Opinionated, but  
not oppressive

# Guide Signs



"signage on post during daytime" by Adi Goldstein on Unsplash

@BillyKorando



# Not Guide Rails



"four brown wooden ladders on beach" by Antonio Lainez on Unsplash

@BillyKorando

# Preserving mental capacity

I have opinions,  
they are always  
right

- Convention over configuration
  - Don't require developers to make more decisions than they have too
- Be flexible
  - Provide developers an easy path to “change” opinions when necessary
- Preserve mental capacity

# Agenda

- ✓ The Spring Boot philosophy
- 2. Improving project management
- 3. Improving project initialization

# Improving project management

- Using parent poms effectively
- Composing starters/shared libraries
- Leveraging autoconfiguration

# Goals

- Keep project build files simple
- Keep projects easy to maintain for teams and enterprise/architects
- Have build files be descriptive of a project's capabilities/responsibilities

# A Typical Build File

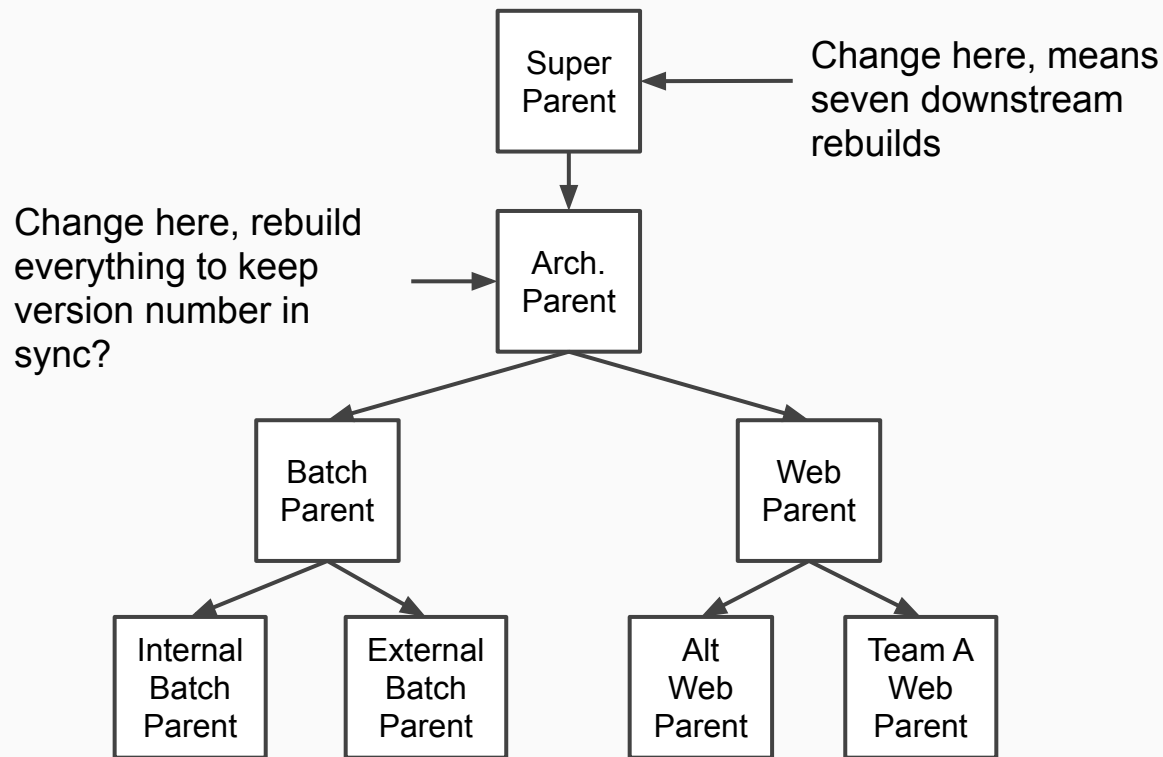
```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <dependencies>
    <groupId>com.ibm.cloud</groupId>
    <artifactId>ibm-cloud-sdk</artifactId>
    <version>3.0.0</version>
    <packaging>jar</packaging>
  </dependencies>
  <groupId>org.springframework.cloud</groupId>
  <name>book-service</name> <artifactId>spring-cloud-dependencies</artifactId>
  <description>Example of a typical Spring Boot project version, version setting that doesn't follow Spring Boot principles</description>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>RELEASE</version>
    <relativePath /> <!-- lookup parent from repository -->
  </parent>
  <build>
    <plugins>
      <plugin>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-maven-plugin</artifactId>
      </plugin>
    </plugins>
  </build>
</project>
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
</dependencies>
```



# Why Does This Happen?

- Enterprises have a lot of demands
  - Security
  - Logging
  - Metrics
  - Database
    - Specific database drivers
- These demands change fairly often
- Lack of guidelines on how to do things right

# A Genealogy of Parent Build Files



So what is the  
solution?

# Manage Dependencies, Don't Just Declare Them

```
<properties>
  <security-starter.version>1.0.0</security-starter.version>
  ...
</properties>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.ibm.cloud</groupId>
      <artifactId>security-starter</artifactId>
      <version>${security-starter.version}</version>
    </dependency>
  </dependencies>
</dependencyManagement>
```

## Someone Set Us Up the BOM

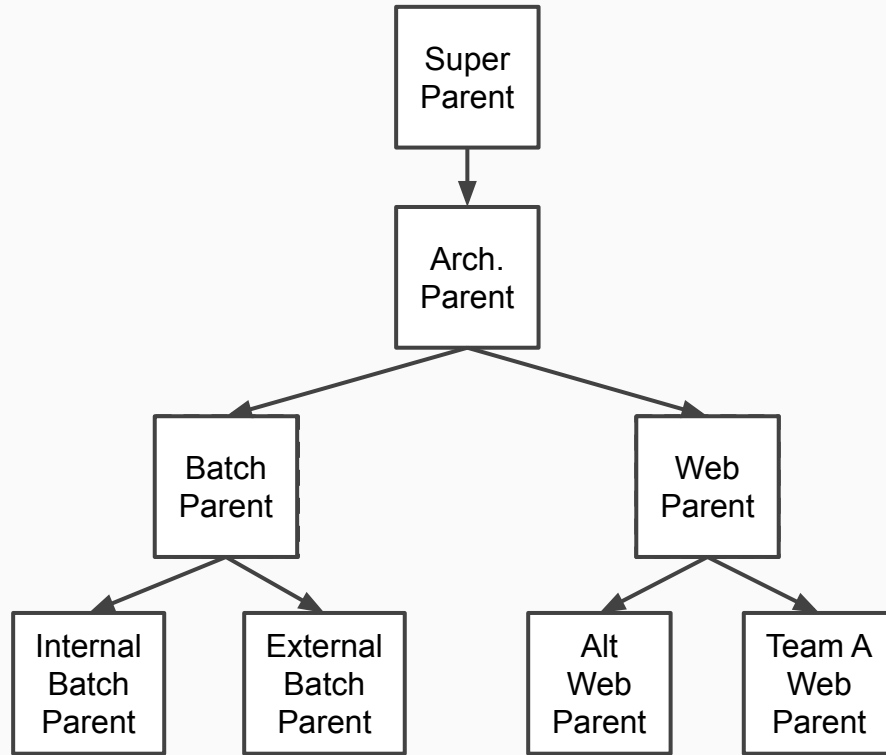
- Don't try to cram every dependency into the parent pom
- Use BOMs to handle dependencies of different domains/project types

# Someone Set Us Up the BOM

```
<project ...>
  <groupId>org.my-org.batch</groupId>
  <artifactId>my-org-batch-dependencies</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>pom</packaging>
  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>org.my-org.batch</groupId>
        <artifactId>batch-security</artifactId>
        <version>1.2</version>
      </dependency>
    </dependencies>
  </dependencyManagement>
</project>
```

# Someone Set Us Up the BOM

```
<project ...>
  ...
  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>org.my-org.batch</groupId>
        <artifactId>my-org-batch-dependencies</artifactId>
        <version>${my-org-batch-dependencies.version}</version>
        <type>pom</type>
        <scope>import</scope>
      </dependency>
    </dependencies>
  </dependencyManagement>
</project>
```





# A Typical Build File con't...

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.ibm.cloud</groupId>
  <artifactId>book-service</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>book-service</name>
  <description>Example of a typical Spring Boot project in an enterprise setting that doesn't follow Spring Boot principles</description>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.0.5.RELEASE</version>
    <relativePath /> <!-- lookup parent from repository -->
  </parent>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
    <java.version>1.8</java.version>
    <spring-cloud.version>Finchley.SR1</spring-cloud.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
```

Opinionated = Bundling  
+ Auto-configuration

## Bundle Libraries Commonly Used Together

Library for database connectivity:

Spring Data + the necessary database driver

Library for caching:

Spring caching + driver for caching service

## Provide Defaults that can be Overridden

@ConditionalOn...

MissingBean

Class

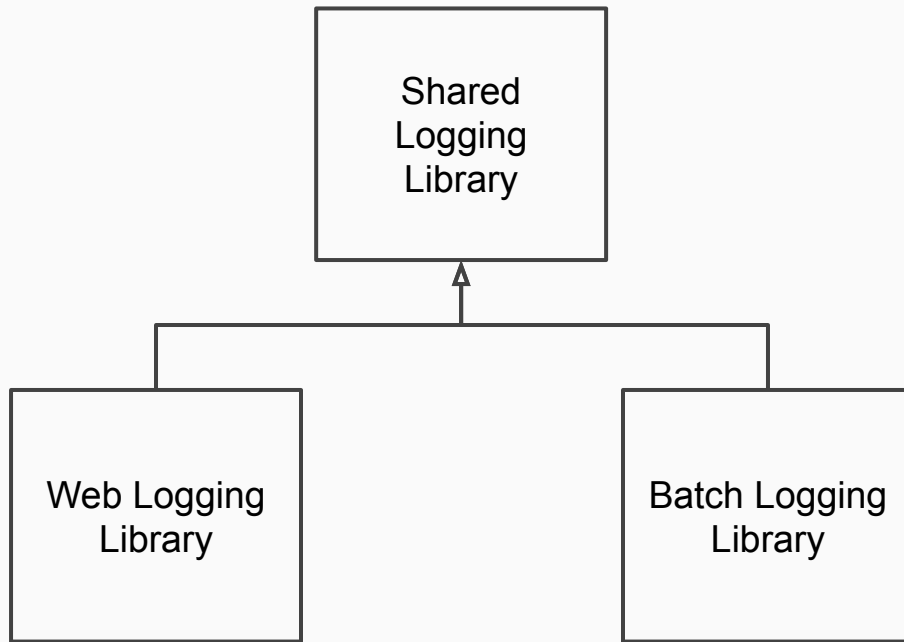
Property

Resource

etc... [link](#)

@AutoConfigureBefore/After

# Auto-configuration Reduce Library Bloat



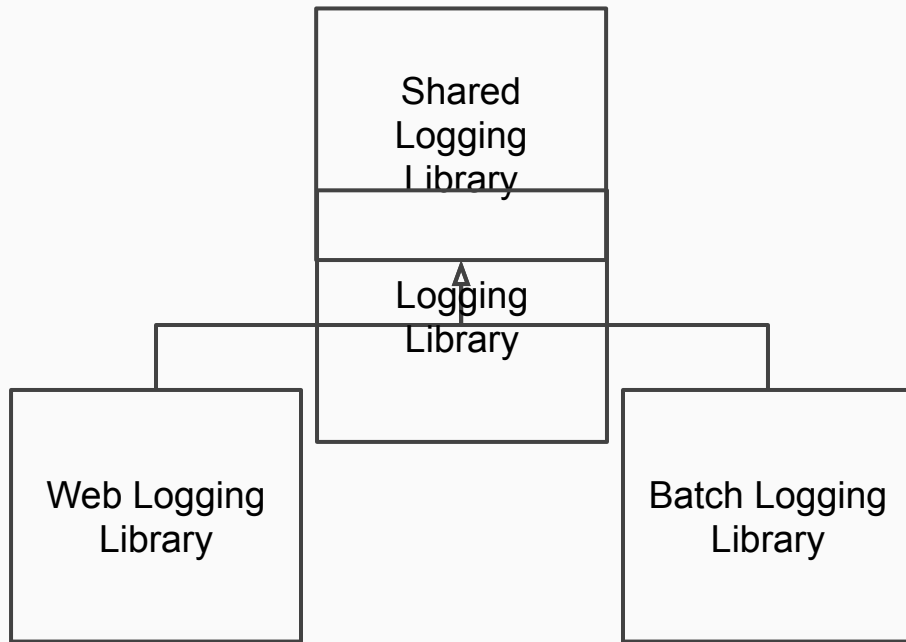
## Auto-configuration Reduce Library Bloat

```
@ConditionalOnWebApplication
@Configuration
public class WebLoggingConfiguration {
    //Some special web logging needs
    @Bean
    @ConditionalOnMissingBean(LoggingProvider.class)
    public LoggingProvider eventStreamingLoggingProvider(
        ...
    }
}
```

## Auto-configuration Reduce Library Bloat

```
@ConditionalOnNotWebApplication  
@Configuration  
public class BatchLoggingConfiguration {  
    //Some special batch logging needs  
}
```

# Auto-configuration Reduce Library Bloat





# Agenda

- ✓ The Spring Boot philosophy
- ✓ Improving project management
- 3. Improving project initialization

# Improving project initialization

- Customizing initializr
- Building a wizard app to guide project initialization

# Goals

- Guide developer to making the right choices
- Reduce work and mental burden around project initialization
- Ensure projects are initialized consistently, properly, and fully

start (dot)  
my-org (dot)  
com

# YOUR CORP INITIALIZR

bootstrap your application now

## Project Metadata

Artifact coordinates

Group



Artifact

Name

Description

Package Name

## Core

☐ Security

Standard Security Library

[javadoc](#)

[guide](#)

[reference](#)

☐ JDBC Data

Library for handling standard data work on the database

## Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Selected Dependencies

```

initializr:
  group-id:
    value: com.ibm
  package-name:

  value: com.ibm
description:
  value: Brief description of project here
  description: Brief description of project here
env:
  maven:
    parent:
      groupId: com.ibm.cloud
      artifactId: boot-enterprise-parent
      version: 1.0.0
      include-spring-boot-bom: false

  boms:
    spring-cloud:
      groupId: com.ibm.batch
      artifactId: batch-dependencies
      versionProperty: ibm-batch.version
      order: 50
      mappings:
        - versionRange: "[1.0.0.RELEASE,1.3.0.RELEASE)"
          version: 1.0.0
        - versionRange: "[2.0.0.RELEASE,1.1.x.RELEASE)"
          version: 2.0.0
  repositories:
    my-corp:
      name: My-Corp Snapshots
      url: https://my-corp.org/repositories/snapshots/
      snapshotsEnabled: true
dependencies:
  - name: Core
    content:
  - name: Security
    id: security
    groupId: com.ibm.cloud
    artifactId: security-starter
    scope: runtime
    versionRange: 1.0.0
    starter: true
    description: Standard Security Library
    links:
      - rel: javadoc
        href: https://spring.io/guides/gs/securing-web/
        description: Securing a Web Application
      - rel: guide
        href: https://spring.io/guides/tutorials/spring-boot-auth2/

```

<https://github.com/spring-projects/spring-boot/>

<https://github.com/spring-io/initializr>

<https://youtu.be/xtW79rzi0I4>

# Q & A

@BillyKorando

william.korando@ibm.com

**IBM Cloud sign-up:** <https://ibm.biz/BdzdZL>

**Slides:** <http://ibm.biz/extending-spring-boot>