

Developer Tech Talks

Brown Bag Technical Discussions

Brown Bag Intentions

I've always loved conferences, but with modern IT budgets and schedules, it's often difficult to go to conferences. Fortunately, we don't have to wait around for a conference to learn new things.

We can just have our own conference.

Sit Back, Relax



This is a safe, open forum
to exchange technical ideas
and have meaningful
discussions.

Hopefully for about 45
minutes.

All code is at Github:

<https://github.com/tgoeke/meetspring>

tim.goeke@gmail.com

Spring Talk Outline

- Meet Spring 3.2
- Crash Course in Spring
- Conclusions

Meet Spring 3.1

3.2 current release

<http://www.springsource.org/download/community>

(or use your dependency manager)

Meet Spring 3.1

What is Spring and why do I care?

Pat Answer: Dependency Injection?

Spring Patterns

Spring Ecosystem

Spring vs X

Meet Spring 3.1

What is Spring and why do I care?

Better Answer: Spring helps me write quality software faster.

Spring Patterns

Spring Ecosystem

Spring vs X

Meet Spring 3.1

Spring Patterns

Core Pattern: Dependency Injection

Eric Evan's Style DDD

MVC

*... and a bunch of
other stuff ...*

We'll cover a selection of this material during the crash course section.

Meet Spring 3.1

Spring Ecosystem *Lot's of stuff beyond the core.*

Web Flow .NET AMQP Grails
Roo Groovy Web Services
Python Social Data
RabbitMQ Virgo RT
Eclipse STS LDAP Android BlazeDS
Security Flex EIP Integration GemFire

We'll only focus on a small section of all this during the crash course section.

Meet Spring 3.1

Spring Ecosystem *Lot's of stuff beyond the core.*

Wicket .NET AMOR Grails

Assertion: Once you've
learned one part of Spring, it's
easy to learn the other parts.

Security

GemFire

We'll only focus on a small section of all this during the
crash course section.

Meet Spring 3.1

Spring vs X

EJB3

Google Guice

<https://code.google.com/p/google-guice/>

Roll Your Own

What's new in Spring 3.2?

We'll focus on two areas:

XML Free Spring Basics
Annotation Based Autowiring

... with some other things sprinkled in along the way...

Crash Course in Spring

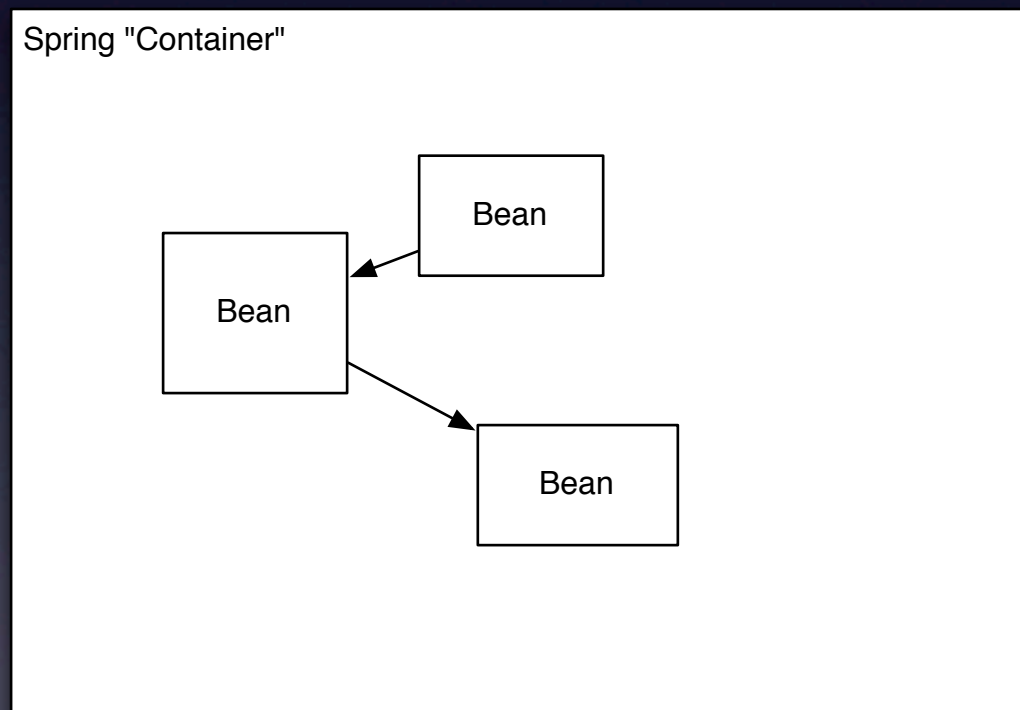
Injection

Traditional
Java Configuration

What is a Spring “container”?

Crash Course in Spring

What is a Spring “container”?



Crash Course in Spring

Dependency Injection

Misko Hevery - Don't Look For Things

<http://www.youtube.com/watch?v=RIfLCWKxHJ0>

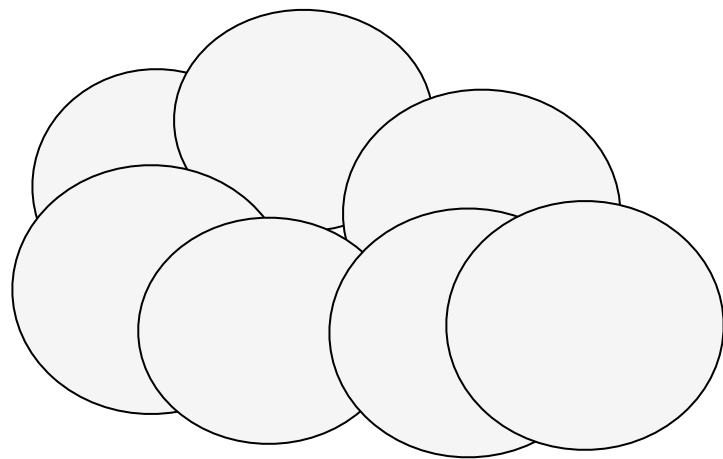
Two piles

Pile of Objects

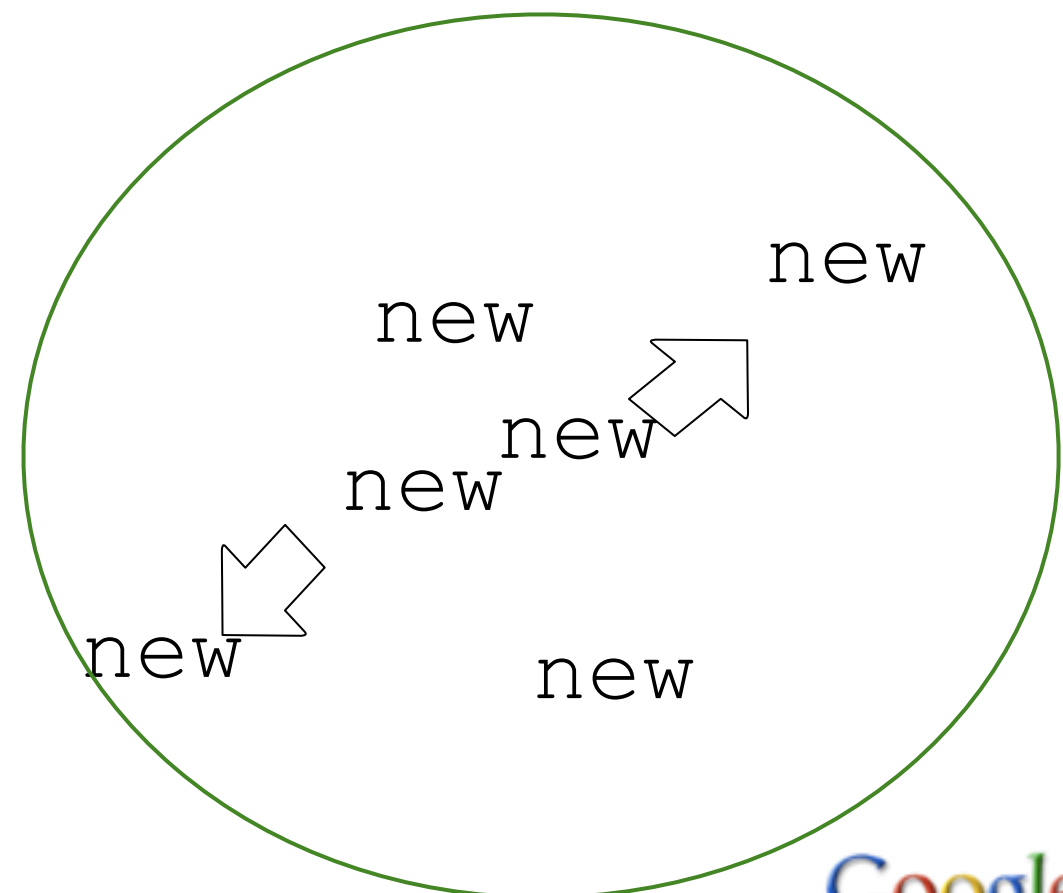
- Gets the collaborators needed

Pile of New Keywords

- Dependency Injection managed



Guice

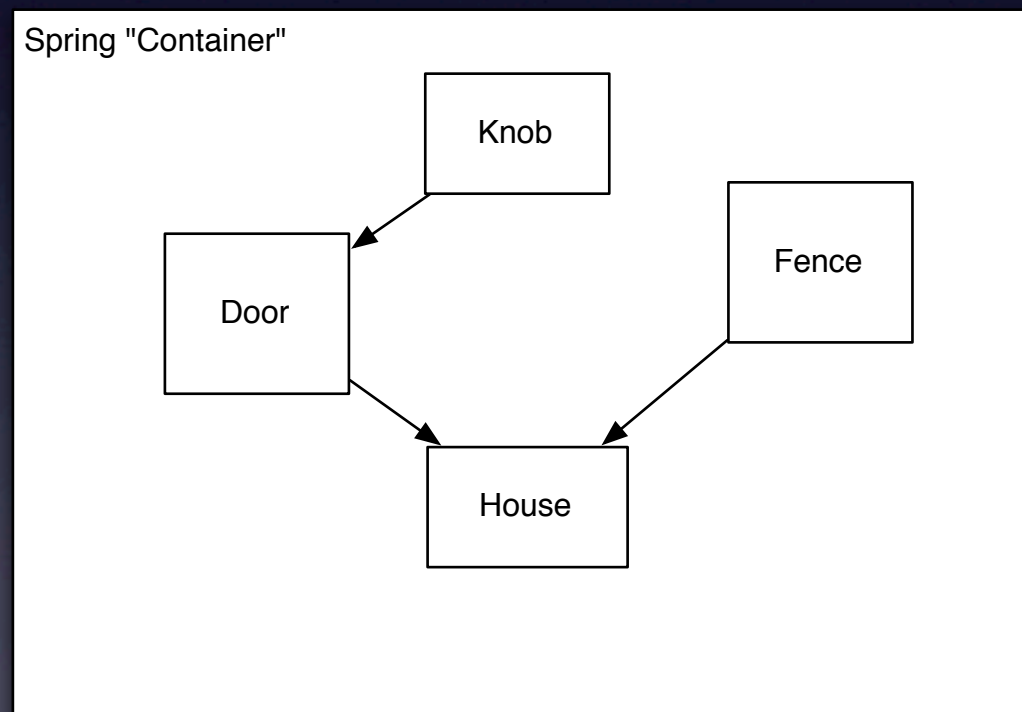


Crash Course in Spring

Demo and Code

Crash Course in Spring

What is a Spring “container”?



Crash Course in Spring

What about the patterns?

Eric Evan's Style DDD

MVC

Wait a bit...

Crash Course in Spring

Autowiring

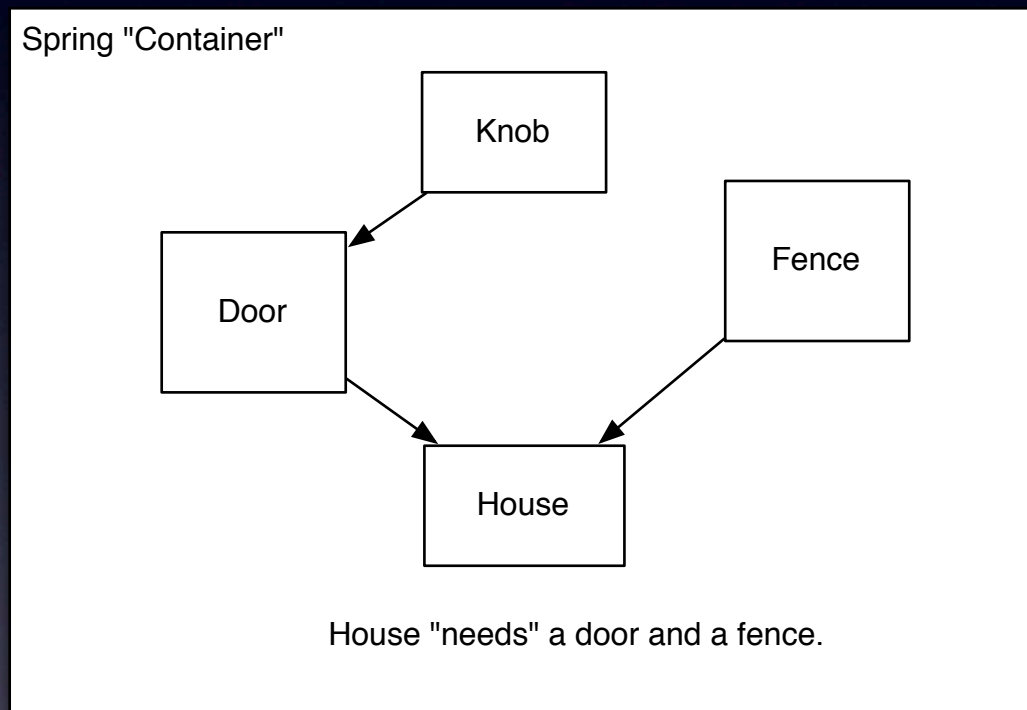
How does it work?

Crash Course in Spring

Demo and Code

Crash Course in Spring

Autowiring Conceptual Model

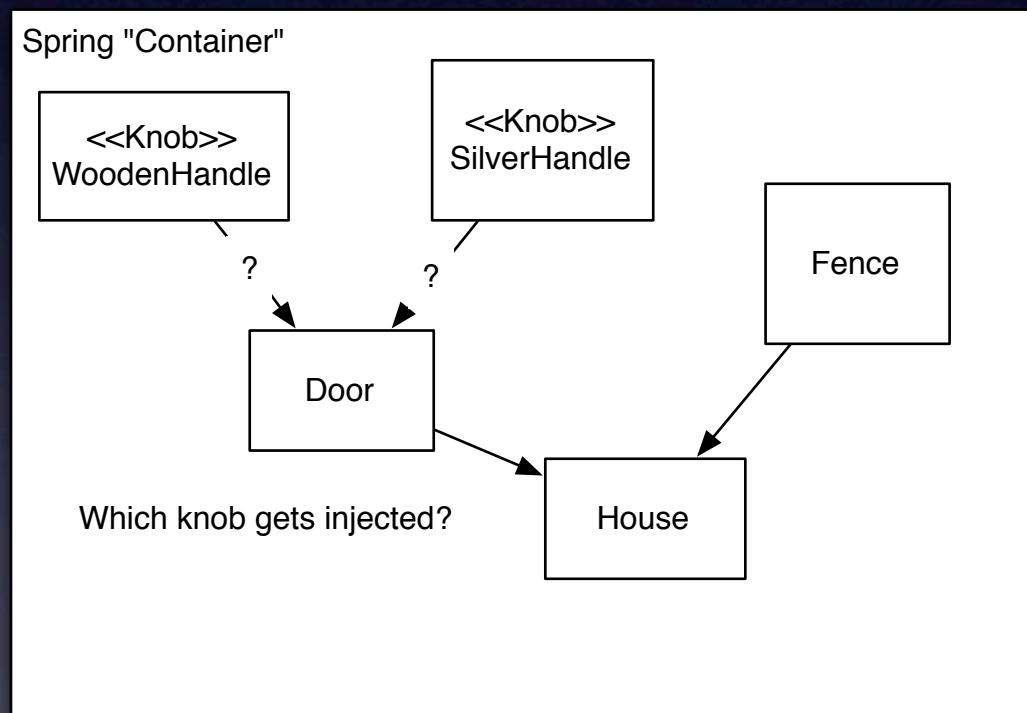


Spring looks for
objects that can go
together.

```
public LittleHouse(Door door, Fence fence) {  
    this.door = door;  
    this.fence = fence;  
}
```

Crash Course in Spring

Autowiring Issues



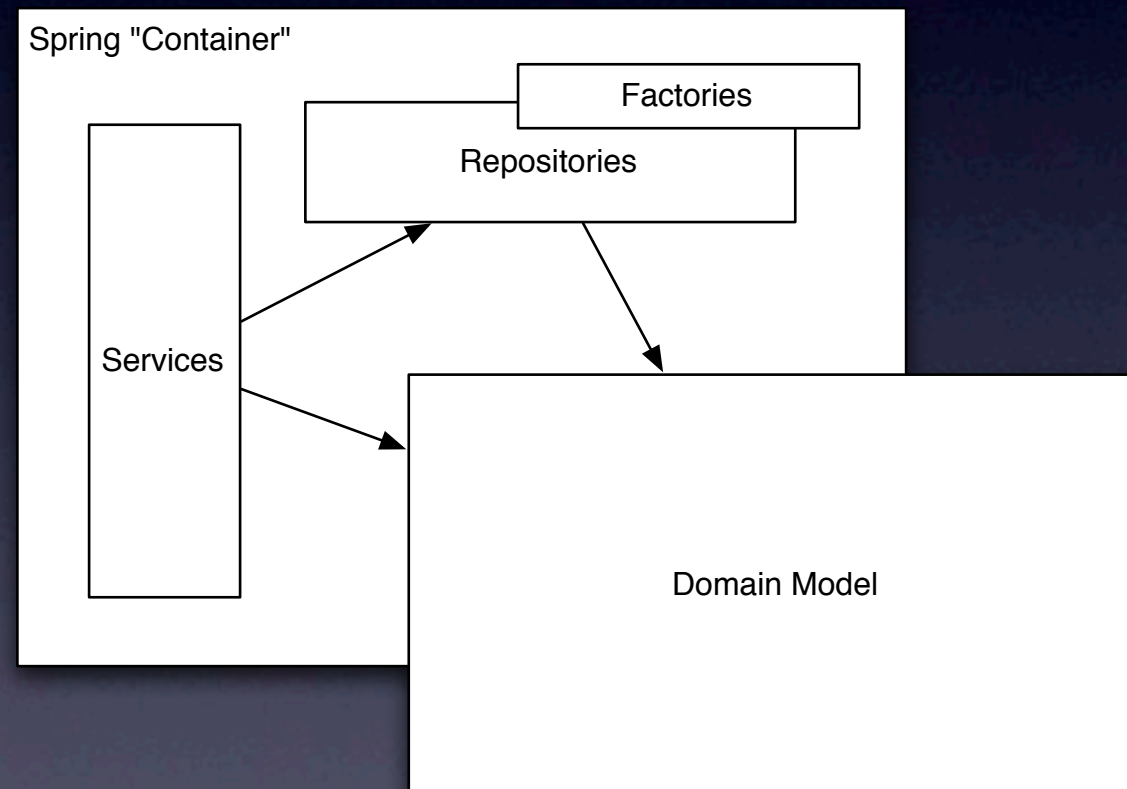
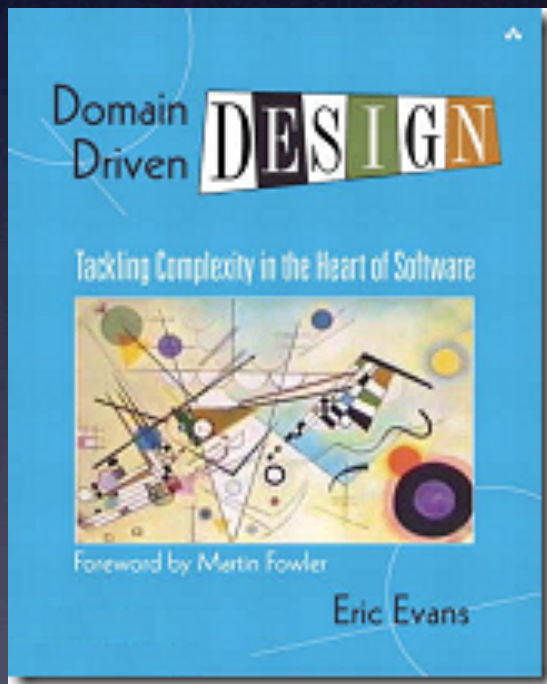
Spring can't resolve this set of objects. But, we can use qualifiers to make it work.

Crash Course in Spring

Demo and Code

Crash Course in Spring

Support for DDD



Super basic version of DDD.

Crash Course in Spring

Demo and Code

Crash Course in Spring

Support for MVC

```
package homebuilder.webapp.controller;

import homebuilder.webapp.services.BuilderSupplyService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;

@Controller
public class IndexController {

    @Autowired
    private BuilderSupplyService builderSupply;

    @RequestMapping( value = "/", method = RequestMethod.GET )
    public String getIndex( Model model ) {

        return "index";
    }

}
```


Crash Course in Spring

Demo and Code

Crash Course in Spring

Review of Misc Items We Used

Conclusions

Did Spring show any promise?
How can it be used in my application?
When does Spring really shine?

When is Spring not a good fit?