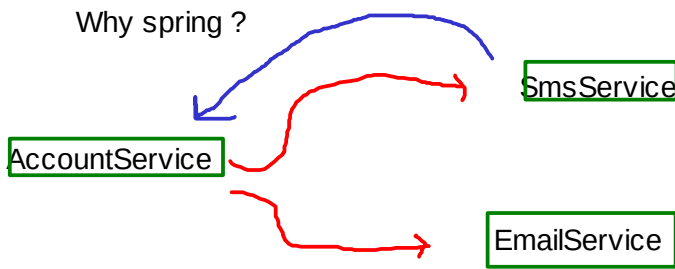




Why spring ?



looking for low level module
ie as is resp for its own dependencies :(

What spring can do?

Part dont need to worry about there dependecny that can be manage by some framework(DI)

any part can be easily swapped out

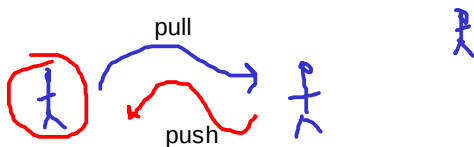
Push model

Pull model

```
MessagingService messagingService=new SmsService();
```

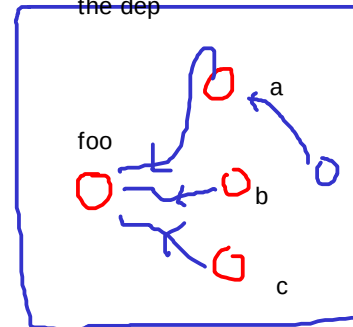
```
AccountService accountService=new AccountService();
accountService.setMessingService(messagingService);
```

```
accountService.transfer(1, 2, 10);
```



```
class Foo{
    private A a;
    private B b;
    private C c;
    ....
}
```

Spring can do this job of managing the dep



Spring act as a container

that manage the life cycle of beans(as, es)

bean wiring

What is spring ? why spring?

let consider a fund transfer app
AccountService need SmsService or EmailService...
i can say that smsservice is a dep of AccountService

if i directly create the object it will create a bad code

then i go for interface(design become good)

but still i need to maintain the dp

then spring come

spring say hey programmer dont manage the dependency
just tell me who depend on what(xml vs anno vs java config)

and i will mange it for you

pull vs push

Spring

2003	Java 1.4	<u>xml based approach</u>	Spring 1.x
2005	java 1.5	<u>annoation</u>	Spring 2.5
2010		<u>java based config</u>	Spring 3.x
2014	----- spring boot	-----wrapper on spring framework	
2023			

they are all gangadhar and shaktiman

ie they are same

```
public class AccountService {  
    private MessagingService messagingService;  
  
    public void setMessagingService(MessagingService messagingService){  
        this.messagingService = messagingService;  
    }  
  
    ...  
  
    public void transfer(int fromAcc, int toAcc, double amount) {  
        System.out.println("fund is tranfered from acc " + fromAcc + " to " + toAcc + " and amount is " + amount);  
        messagingService.sendMessage();  
    }  
}
```

```
public class EmailService implements MessagingService {  
    public void sendMessage() {  
        System.out.println("email is send...");  
    }  
}
```

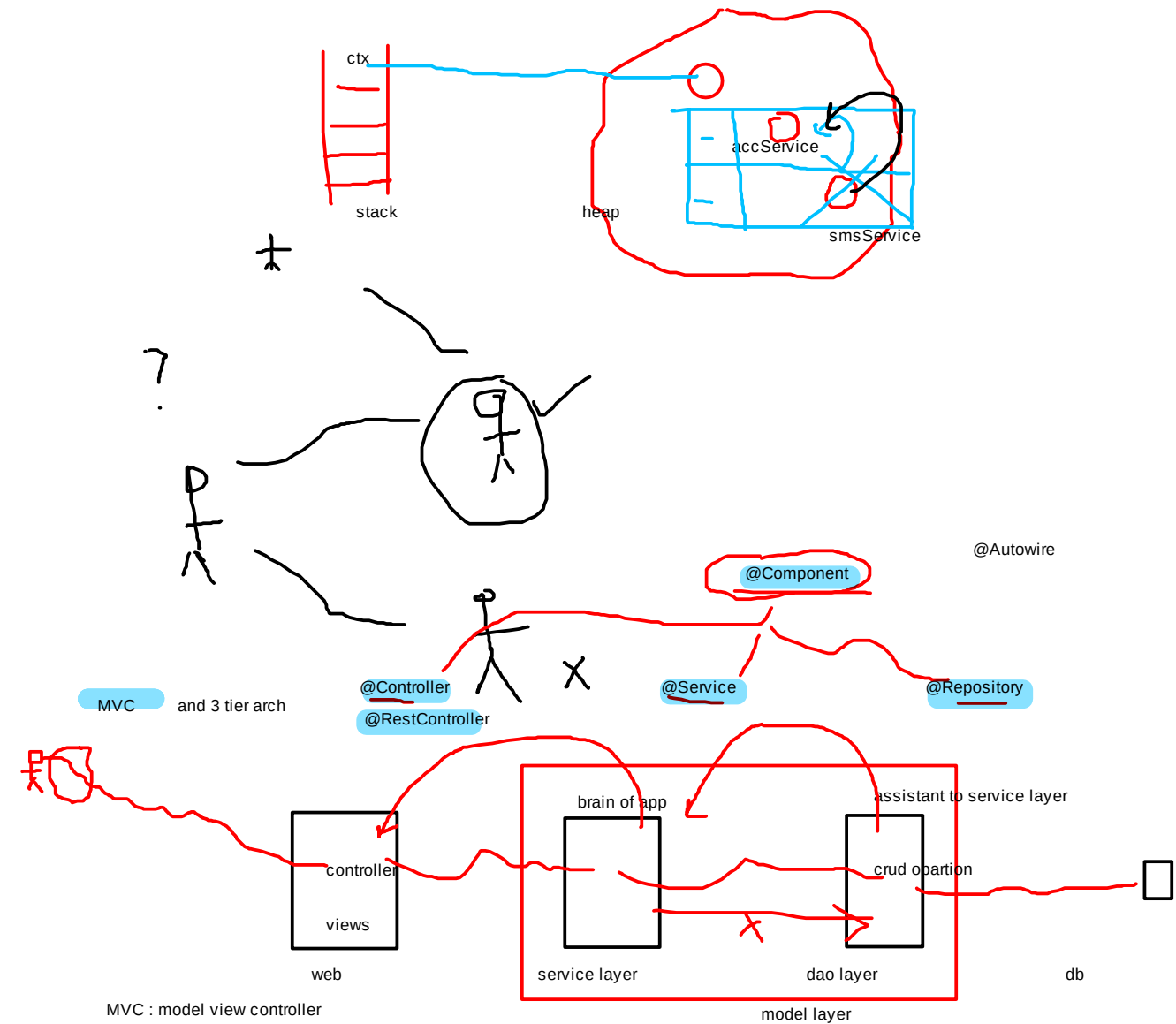
```
public interface MessagingService {  
    public void sendMessage();  
}
```

```
public class SmsService implements MessagingService {  
    public void sendMessage() {  
        System.out.println("sms is send...");  
    }  
}
```

```
<beans xmlns="http://www.springframework.org/schema/beans"  
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
       xsi:schemaLocation="http://www.springframework.org/schema/beans  
       http://www.springframework.org/schema/beans/spring-beans.xsd">  
  
    <bean id="accService" class="com.bankapp.service.AccountService" autowire="byType">  
    </bean>  
  
    <bean id="ms1" class="com.bankapp.service.SmsService"/>  
</beans>
```

```
container  
ApplicationContext ctx=new ClassPathXmlApplicationContext("beans.xml");  
AccountService accountService=(AccountService) ctx.getBean("accService");  
accountService.transfer(1, 2, 10);
```

setter injection

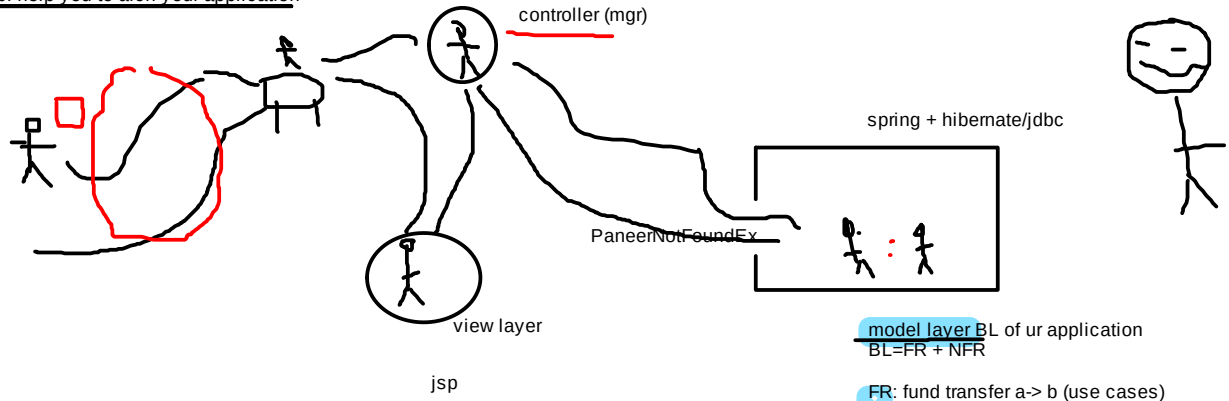


MVC : model view controller

model view controller

when i should go
try catch vs throws

Arch dp: help you to arch your application



model layer BL of ur application
BL=FR + NFR

FR: fund transfer a-> b (use cases)

NFR: XXX ilities : sec, tx logging caching...
(AOP)

