**Spring Qualifier with Autowiring**

Often we faced a situation wherein a Spring bean we have multiple properties with same data type. Now if we want to wire it through Autowiring annotation, Spring container confused which bean to wires with which property and ended up with throwing an Exception.

Let try to understand it through an Example,

As IPhone has frontCamera and rearCamera and both are children of ICamera interface, Now if we use Autowire annotation on frontCamera and rearCamera properties. Spring Container won’t understand which bean to wire with which property and throws an Exception.

Example:

Java Code:

**package** com.example.autowire;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**public** **class** IPhone {

String version;

**int** ram;

String iosVersion;

@Autowired

ICamera frontCamera;

@Autowired

ICamera rearCamera;

**public** String getVersion() {

**return** version;

}

**public** **void** setVersion(String version) {

**this**.version = version;

}

**public** **int** getRam() {

**return** ram;

}

**public** **void** setRam(**int** ram) {

**this**.ram = ram;

}

**public** String getIosVersion() {

**return** iosVersion;

}

**public** **void** setIosVersion(String iosVersion) {

**this**.iosVersion = iosVersion;

}

**public** ICamera getFrontCamera() {

**return** frontCamera;

}

**public** **void** setFrontCamera(ICamera frontCamera) {

**this**.frontCamera = frontCamera;

}

**public** ICamera getRearCamera() {

**return** rearCamera;

}

**public** **void** setRearCamera(ICamera rearCamera) {

**this**.rearCamera = rearCamera;

}

@Override

**public** String toString() {

**return** "IPhone [version=" + version + ", ram=" + ram + ", iosVersion="

+ iosVersion + ", frontCamera=" + frontCamera + ", rearCamera="

+ rearCamera + "]";

}

}

**package** com.example.autowire;

**public** **interface** ICamera {

String getDescription();

String getMegaPixel();

}

**package** com.example.autowire;

**public** **class** RearCamera **implements** ICamera{

**private** String description;

**private** String megaPixel;

**public** **void** setDescription(String description) {

**this**.description = description;

}

**public** **void** setMegaPixel(String megaPixel) {

**this**.megaPixel = megaPixel;

}

@Override

**public** String getDescription() {

// **TODO** Auto-generated method stub

**return** description;

}

@Override

**public** String getMegaPixel() {

// **TODO** Auto-generated method stub

**return** megaPixel;

}

@Override

**public** String toString() {

**return** "RearCamera [description=" + description + ", megaPixel="

+ megaPixel + "]";

}

}

**package** com.example.autowire;

**public** **class** FrontCamera **implements** ICamera{

**private** String description;

**private** String megaPixel;

**public** **void** setDescription(String description) {

**this**.description = description;

}

**public** **void** setMegaPixel(String megaPixel) {

**this**.megaPixel = megaPixel;

}

@Override

**public** String getDescription() {

// **TODO** Auto-generated method stub

**return** description;

}

@Override

**public** String getMegaPixel() {

// **TODO** Auto-generated method stub

**return** megaPixel;

}

@Override

**public** String toString() {

**return** "FrontCamera [description=" + description + ", megaPixel="

+ megaPixel + "]";

}

}

**package** com.example.autowire;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**public** **class** Manager {

**public** **static** **void** main(String[] args) {

ApplicationContext context = **new** ClassPathXmlApplicationContext("configFiles/IPhone.xml");

IPhone iphone = (IPhone) context.getBean("iphone");

System.***out***.println(iphone);

}

}

IPhone.xml:

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"*>

<context:annotation-config/>

<bean id=*"iphone"* class=*"com.example.autowire.IPhone"*>

<property name=*"version"* value=*"IPhone6"*/>

<property name=*"ram"* value=*"2"*/>

<property name=*"iosVersion"* value=*"IOS6"*/>

</bean>

<bean id=*"rcamera"* class=*"com.example.autowire.RearCamera"*>

<property name=*"description"* value=*"18-210 lens with Apperature f1.2"*/>

<property name=*"megaPixel"* value=*"1024px"*/>

</bean>

<bean id=*"fcamera"* class=*"com.example.autowire.FrontCamera"*>

<property name=*"description"* value=*"VGA camera"*/>

<property name=*"megaPixel"* value=*"480px"*/>

</bean>

</beans>

Output :

Could not autowire field: com.example.autowire.ICamera com.example.autowire.IPhone.frontCamera; nested exception is org.springframework.beans.factory.NoUniqueBeanDefinitionException: No qualifying bean of type [com.example.autowire.ICamera] is defined: expected single matching bean but found 2: rcamera,fcamera

To overcome this problem, we can use @Qulifier annotation, by this annotation we can explicitly say to Spring container which bean we need to wire with which property,

So here we will modify the IPhone.java little bit,

@Autowired

@Qualifier("fcamera")

ICamera frontCamera;

@Autowired

@Qualifier("rcamera")

ICamera rearCamera;

We introduce @Qulifier(“fcamera”), @Qulifier(“rcamera”) just beneath of @Autowired annotation , by doing this we instruct Spring container to wire Iphone’s fronCamera property with the bean which has id fcamera and wire reracamera with rcamera bean.

Now Spring container inject Frontcamera bean into frontCamera and ReraCamera bean into rearCamera respectively.

Modified Code:

**package** com.example.autowire;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.beans.factory.annotation.Qualifier;

**public** **class** IPhone {

String version;

**int** ram;

String iosVersion;

@Autowired

@Qualifier("fcamera")

ICamera frontCamera;

@Autowired

@Qualifier("rcamera")

ICamera rearCamera;

**public** String getVersion() {

**return** version;

}

**public** **void** setVersion(String version) {

**this**.version = version;

}

**public** **int** getRam() {

**return** ram;

}

**public** **void** setRam(**int** ram) {

**this**.ram = ram;

}

**public** String getIosVersion() {

**return** iosVersion;

}

**public** **void** setIosVersion(String iosVersion) {

**this**.iosVersion = iosVersion;

}

**public** ICamera getFrontCamera() {

**return** frontCamera;

}

**public** **void** setFrontCamera(ICamera frontCamera) {

**this**.frontCamera = frontCamera;

}

**public** ICamera getRearCamera() {

**return** rearCamera;

}

**public** **void** setRearCamera(ICamera rearCamera) {

**this**.rearCamera = rearCamera;

}

@Override

**public** String toString() {

**return** "IPhone [version=" + version + ", ram=" + ram + ", iosVersion="

+ iosVersion + ", frontCamera=" + frontCamera + ", rearCamera="

+ rearCamera + "]";

}

}

**package** com.example.autowire;

**public** **interface** ICamera {

String getDescription();

String getMegaPixel();

}

**package** com.example.autowire;

**public** **class** FrontCamera **implements** ICamera{

**private** String description;

**private** String megaPixel;

**public** **void** setDescription(String description) {

**this**.description = description;

}

**public** **void** setMegaPixel(String megaPixel) {

**this**.megaPixel = megaPixel;

}

@Override

**public** String getDescription() {

// **TODO** Auto-generated method stub

**return** description;

}

@Override

**public** String getMegaPixel() {

// **TODO** Auto-generated method stub

**return** megaPixel;

}

@Override

**public** String toString() {

**return** "FrontCamera [description=" + description + ", megaPixel="

+ megaPixel + "]";

}

}

**package** com.example.autowire;

**public** **class** RearCamera **implements** ICamera{

**private** String description;

**private** String megaPixel;

**public** **void** setDescription(String description) {

**this**.description = description;

}

**public** **void** setMegaPixel(String megaPixel) {

**this**.megaPixel = megaPixel;

}

@Override

**public** String getDescription() {

// **TODO** Auto-generated method stub

**return** description;

}

@Override

**public** String getMegaPixel() {

// **TODO** Auto-generated method stub

**return** megaPixel;

}

@Override

**public** String toString() {

**return** "RearCamera [description=" + description + ", megaPixel="

+ megaPixel + "]";

}

}

**package** com.example.autowire;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**public** **class** Manager {

**public** **static** **void** main(String[] args) {

ApplicationContext context = **new** ClassPathXmlApplicationContext("configFiles/IPhone.xml");

IPhone iphone = (IPhone) context.getBean("iphone");

System.***out***.println(iphone);

}

}

IPhone.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"*>

<context:annotation-config/>

<bean id=*"iphone"* class=*"com.example.autowire.IPhone"*>

<property name=*"version"* value=*"IPhone6"*/>

<property name=*"ram"* value=*"2"*/>

<property name=*"iosVersion"* value=*"IOS6"*/>

</bean>

<bean id=*"rcamera"* class=*"com.example.autowire.RearCamera"*>

<property name=*"description"* value=*"18-210 lens with Apperature f1.2"*/>

<property name=*"megaPixel"* value=*"1024px"*/>

</bean>

<bean id=*"fcamera"* class=*"com.example.autowire.FrontCamera"*>

<property name=*"description"* value=*"VGA camera"*/>

<property name=*"megaPixel"* value=*"480px"*/>

</bean>

</beans>

Output :

IPhone [version=IPhone6, ram=2, iosVersion=IOS6, frontCamera=FrontCamera [description=VGA camera, megaPixel=480px], rearCamera=RearCamera [description=18-210 lens with Apperature f1.2, megaPixel=1024px]]

Please note that @Qulifier annotation only and only use with @Autowire annotation. There is no alternatives property in Spring bean XML configuration like (qualifier property).

As in Spring bean XML always we reference a bean through ref property so this scenario will not occur.