Inject @Scheduled fixedRate value from Spring Boot application.yml file

Asked 5 years, 3 months ago   Active 5 months ago   Viewed 25k times

I know I can inject the value from properties file with the following syntax:

28

```
@Scheduled(fixedRate=${myRate})
public void getSchedule(){
    System.out.println("in scheduled job");
}
```

6

However I can't guess how to accomplish the same if the configuration is in YAML file.

Thanks in advance,

spring   spring-boot

asked Dec 12 '14 at 14:24

Juan Carlos González
724   1   7   16

3 Answers

In my application.properties (YAML) I put this

42

```
console:
    fetchMetrics: 5000
```

Then in my simple Task class I push the definition :

```
        logger.info("What's up ?");
    }
```

Please notice that `fixedRate` expects a `long` and you want to inject a placeholder, you will need `fixedRateString`

 edited Oct 7 '19 at 18:45   answered Jan 7 '15 at 10:46

rajadilipkolli   **Wajax**
2,734   1   17   36   **500**   6   8

---

1   Thanks for your answer. When I try this on a Spring Boot project, I'm getting an IllegalStateException with the message Invalid fixedRateString value "${general.pathMonitorPollingRate}" - cannot parse into integer. I've debugged this error into AbstractBeanFactory from ScheduledAnnotationBeanPostProcessor and I've seen that the value cannot be resolved because the resolvers list is empty. Don't know if this is configuration problem or an issue in Boot. – Juan Carlos González   Jan 8 '15 at 12:10

@JuanCarlosGonzález This is probably won't help you anymore, but I was running into the same "cannot parse into integer" issue. I am able to get around it by adding the Configuration, EnableAutoConfiguration and EnableScheduling annotations to my config. Without auto configuration, it breaks. Can't figure out why – k.schroeder31 Feb 5 '16 at 19:52

Thanks for your help. I think @EnableAutoConfiguration must register the proper StringValueResolver in order to evaluate annotation properly. – Juan Carlos González   Feb 8 '16 at 8:28

2   The "ConsoleConfig" class isn't required here - it doesn't add any value and isn't used at all since the configuration value is taken directly from the YAML and used for the fixedRateString – Tom Feb 20 '19 at 13:06

Note that your property can also be string resolved by Duraction.parse() ie: "PT20.345S" for 20.345 seconds, "P2D" for two days and so on. It doesn't have to be only long with given milliseconds. docs.spring.io/spring-framework/docs/current/javadoc-api/org/... – estn Jan 22 at 12:09 ✎
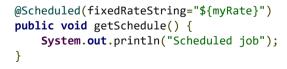

I find it easy once done for my project.
Change `fixedRate` to `fixedRateString` and put the property key in `double quotes` like this:

16

```
@Scheduled(fixedRateString="${myRate}")
public void getSchedule() {
    System.out.println("Scheduled job");
}
```

edited Sep 26 '18 at 16:14   answered Jun 13 '18 at 11:23

In my application I use the annotation `PropertySource` on my config class:

`@PropertySource("application-${spring.profiles.active}.yml")`

`spring.profiles.active` returns the active profile (dev, test, etc). My properties file name is application-dev.yml

The annotation `@Scheduled` works with property injection. Dont forget the annotation with prefix configuration on your class.

edited Aug 12 '16 at 23:33   answered Aug 12 '16 at 16:36

Fred Barclay   Dalila Serpa
814   12   22   **11**