# QUARTZ - JOB SCHEDULING

Posizione del file job\_context.xml

C:\Users\sabatinija\Desktop\4Price\workspace\Branches\CARREFOUR\config\produzione\frontend1\ job\_context.xml

Definisco il job (classe che implementa l’interfaccia **JOB**):

<bean id="loadImportData\_Job" class="org.springframework.scheduling.quartz.JobDetailFactoryBean">

<property name="jobClass" value="**com.valueteam.customizations.task.job.LoadImportDataJob**" />

<property name="durability" value="true" />

</bean>

Definisco la CRON Expression relativa al Job:

(<https://www.baeldung.com/cron-expressions>)

<bean id="trigger\_2" class="org.springframework.scheduling.quartz.CronTriggerFactoryBean">

<property name="jobDetail" ref="loadImportData\_Job" />

<property name="cronExpression" value="0 0 22 \* \* ?" />

</bean>

Attivo il JOB (job attivi con bean\_id di riferimento):

<!-- Scheduler -->

<bean id="VtPriceScheduler" class="org.springframework.scheduling.quartz.SchedulerFactoryBean">

<property name="jobFactory">

<bean class="com.valueteam.industry.utils.AutowiringSpringBeanJobFactory"/>

</property>

<property name="triggers">

<list>

<!-- <ref local="trigger\_1" />-->

<ref local="trigger\_2" />

<ref local="trigger\_3" />

<!--ref local="trigger\_4" /-->

<!--ref local="trigger\_5" /-->

<!--ref local="trigger\_6" /-->

<!-- <ref local="trigger\_7" />-->

<ref local="trigger\_8" />

<ref local="trigger\_9" />

<ref local="trigger\_10" />

<!--<ref local="trigger\_11" />-->

<!--<ref local="trigger\_12" />-->

<ref local="trigger\_16" />

<ref local="trigger\_18" />

<ref local="trigger\_19" />

<ref local="trigger\_31" />

</list>

</property>

<property name="autoStartup" value="true" />

<property name="configLocation" value="classpath:quartz.properties" />

</bean>

</beans>

/\*\*

\*

\* @author Ruffolo

\*/

public class LoadImportDataJob implements Job {

static final Logger logger = LogManager.getLogger(LoadImportDataJob.class);

@Autowired

private InterfaceService interfaceService;

@Autowired

private TraceService traceService;

@Autowired //NTT\_20180219\_DM

private CatalogService catalogService; //NTT\_20180219\_DM

@Override

public void execute(JobExecutionContext jec){

logger.info("------ START -> execute job ------");

try{

//Unico modo per far digeririre l’autowired

InterfaceFullController interfaceFull = new InterfaceFullController();

interfaceFull.setInterfaceService(interfaceService);

interfaceFull.setTraceService(traceService);

interfaceFull.setCatalogService(catalogService); //NTT\_20180219\_DM

//Metodo per lanciare il job

**interfaceFull.taskManagerLoad(-1L, InterfaceFullController.K\_DEFAULT\_TASKS);**

}

catch(Exception e){

logger.error("",e);

}

logger.info("------ END -> execute job ------");

}

}