Quartz-Monitor设计

郭蕾

目 录

[第1章 Quartz监控管理的需求 3](#_Toc345686302)

[1.1 背景 3](#_Toc345686303)

[1.2 需求 3](#_Toc345686304)

[第2章 Quartz监控管理的设计 3](#_Toc345686305)

[2.1 总体设计 3](#_Toc345686306)

[2.2 Quartz监控管理设计 4](#_Toc345686307)

[2.3 Quartz监控管理的功能设计 4](#_Toc345686308)

[第3章 关键技术 5](#_Toc345686309)

[3.1 JMX API的使用 5](#_Toc345686310)

[3.2 DWZ框架的使用 5](#_Toc345686311)

[第3章 写在最后 5](#_Toc345686312)

# 第1章 Quartz监控管理的需求

## 1.1 背景

在目前的CRM系统中，使用Quartz的地方比较多，而这些定时任务没有一个统一的管理和监控平台，使得我们根本无法灵活去控制定时任务的执行，也无法了解定时任务的状态。Quartz-Monitor类似于Windows中的任务管理器，可以获得运行时的实时监控，查看任务运行状态，动态增加、暂停任务，动态管理任务的触发条件。

## 1.2 需求

Quartz-Monitor需能够管理和监控job的执行情况，并能够动态增加和修改job。

Quartz-Monitor的需求主要有以下几点：

1. 能够管理多个环境下的Quartz。
2. 提供job的基本管理，包括job列表查看、删除job、添加job以及立即执行job和暂停恢复job。
3. 提供对trigger的基本管理，包括trigger列表查看、添加trigger、删除trigger。
4. 提供对Cron Expression表达式的校验。

# 第2章 Quartz监控管理的设计

## 2.1 总体设计

Quartz-Monitor利用Quartz的JMX API来实现对任务的管理，这样将应用和管理平台解耦合，方便集成。前端使用DWZ框架，DWZ简单易学，基于JQuery之上，容易扩展。

## 2.2 Quartz监控管理设计

Quartz-Monitor的设计上，我主要遵从简单、轻量的原则，所以没有做持久化的部分，每次加载都存到内存中。在配置部分，将配置利用xsteam将配置持久化为xml文件，基本满足目前需求。View层使用的是struts2，大家都比较熟悉，以后的扩展也比较容易。

## 2.3 Quartz监控管理的功能设计

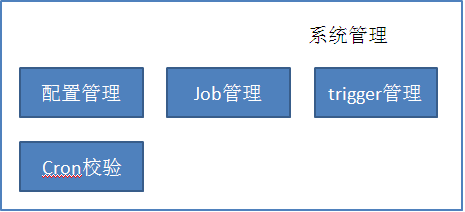


图2-3 Quartz-Monitor功能设计

Quartz-Monitor的功能主要有四个：

1. 配置管理

针对不同环境的Quartz，需要有一个统一的入口去管理这些配置，来满足对不同环境的任务的管理。

1. Job管理

提供对Job的管理和维护功能。Monitor提供对Job的基本管理，包括对其状态、执行时间、基本信息的管理以及提供基于Job的基本操作。

1. Trigger管理

提供对trigger的管理和维护功能。可以查看某个job的trigger信息，并添加和修改trigger。

1. Cron Expression校验

Cron Expression虽然简单却非常容易写错，所以我们提供了对其的校验功能。

# 第3章 关键技术

## 3.1 JMX API的使用

Quartz官方文档中基本没有介绍这部分功能，并且API的说明也比较少，网上的资料几乎没有。我通过junit测试已经基本了解每个API的功能。

## 3.2 DWZ框架的使用

DWZ虽然简单，但是也需要学习成本。期间也遇到了很多问题，我都通过读源码或者网上搜索得以解决，这些积累也写到了我的博客上。

# 第3章 写在最后

经过一周(8天)的开发，我基本完成了Quartz-Monitor的第一个版本，此版本已经可以满足目前业务的所有需求。由于时间有限，系统可能还存在一些不足之处，当然这些问题也是我下个版本要着力改善的。包括：

1. 将gson替换为fastjson。
2. 分页部分完善。
3. 支持排序功能。
4. 代码优化。