Atitit 项目启动慢对分析与解决

# 原因可能

Spring 已经增加类lazy ini

Mybatis lazyini

## 从启动到第一条语句 22s花费了。。。

2s spring load

2s 19:23:34,116-[TS] DEBUG main org.mybatis.spring.mapper.ClassPathMapperScanner

10s org.springframework.beans.factory.support.DefaultListableBeanFactory

5s spring ..aop springframework.beans.factory.support.DefaultListableBeanFactory

将日志级别调至Debug级别

没错，这是我们需要做的第一步最重要的步骤，开启单测，把日志打到一个文件里，从头撸到尾，究竟你这加载的一分半钟究竟干嘛了~

一些Spring配置初始化

根据ComponentScan扫路径的Class类，加入待注入候选列表

根据Mapper扫路径的Class类，加入待注入候选列表

对扫出来的Mapper创建MapperFactoryBean

创建注入@Configuration里@Bean注解的Bean的BeanDefinitions

预加载一些Bean

一些组件例如，PostProcessor、Advisor初始化

一些中间件例如数据库、缓存、消息队列加载

扫描的Bean的初始化，依赖注入。。

Bean的PostConstruct开始跑

....//可能不是很全，列举了其中一部分

## 设置<beans default-lazy-init=*"true"*

## 去除aop与事务配置 减少了5s

新建里spirng test cfg..去除aop与事务

## 关闭扫描，， 手动白名单

<!-- 扫描service（service层注入） lazyInit = true

<context:component-scan base-package="application.service" />

-->

但是因为autowire对太多类，貌似不行启动不类。。

## @Lazy 无效

10 @Component

11 @Scope("singleton")

12 //@Lazy(true)

13 //@Lazy(value=true)

14 @Lazy//@Lazy注解可以用在类上, 还可以用在普通方法上,还可以用在构造方法上,还可以用在参数上,还可以用在属性上. 但是只用在类上有效果.其他地方没效果

15 public class DefaultCache implements Cache{