

Spring Cloud 配置

小马哥 (mercyblitz)

主要议题

- **Bootstrap 配置属性**
- **分布式配置**
- **动态配置属性Bean**
- **健康指标**
- **管理端点**

Bootstrap 配置属性

Bootstrap 配置属性

- Bootstrap 配置文件
- 调整 Bootstrap 配置文件路径
- 覆盖远程配置属性
- 自定义 Bootstrap 配置
- 自定义 Bootstrap 配置属性源

Bootstrap 配置属性

- Bootstrap 配置文件
- 调整 Bootstrap 配置文件路径
- 覆盖远程配置文件属性
- 自定义 Bootstrap 配置
- 自定义 Bootstrap 配置属性源

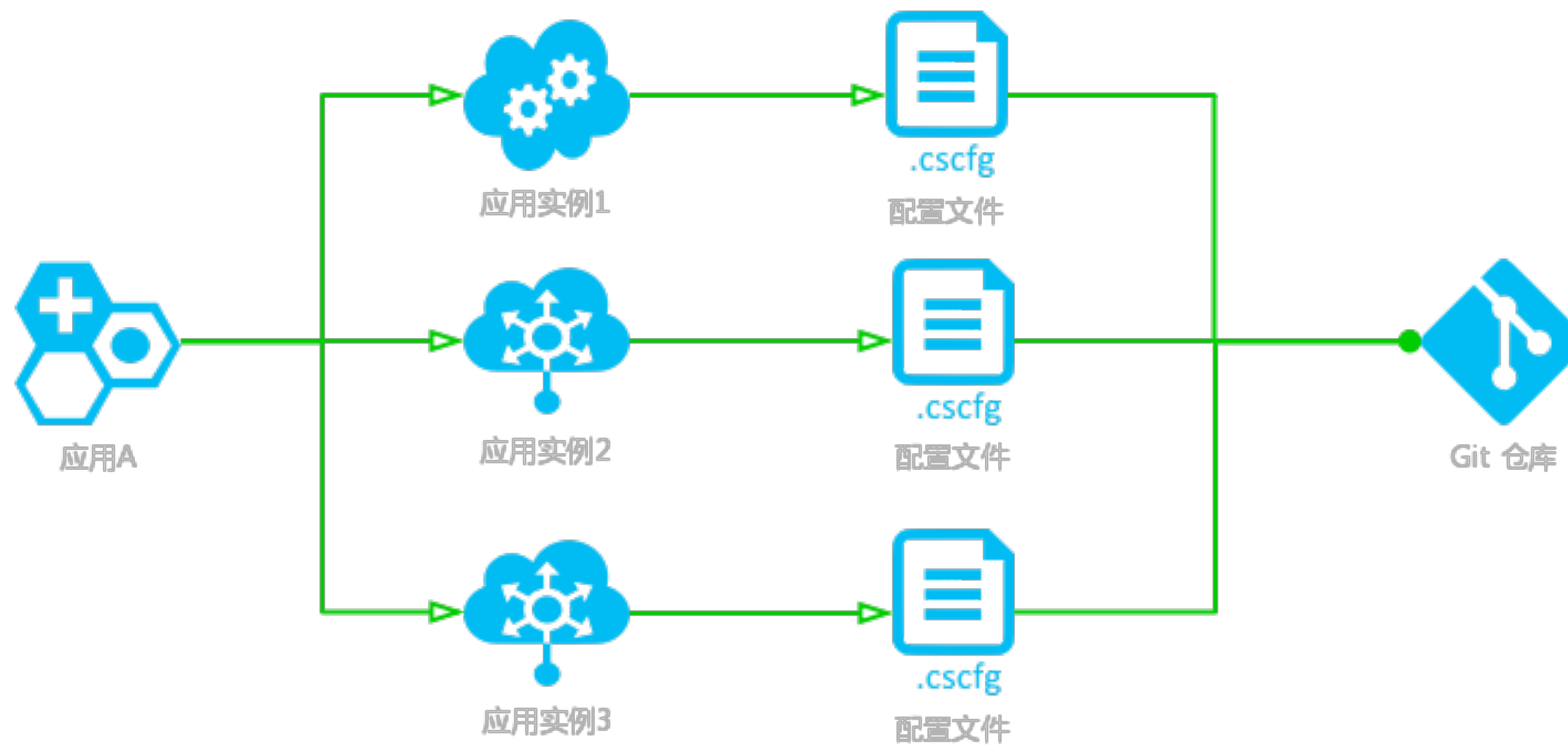
分布式配置

分布式配置

- 分布式配置架构
- Spring Cloud 配置服务器
- 服务端 Environment 仓储
- Spring Cloud 配置客户端

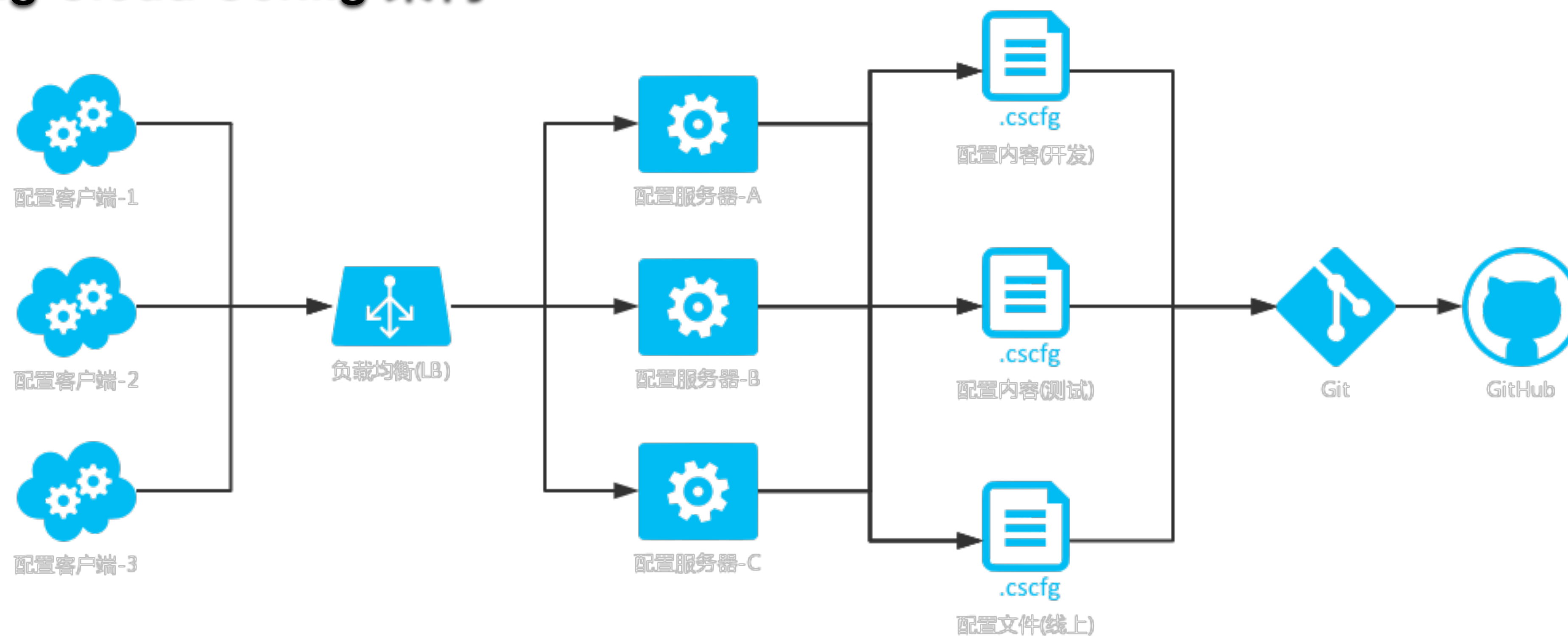
分布式配置架构

- 传统架构



分布式配置架构

- Spring Cloud Config 架构



Spring Cloud 配置服务器

- Spring Cloud Config Server

Spring Cloud 配置服务器提供分布式、动态化集中管理应用配置信息的能力

- 构建 Spring Cloud 配置服务器

- @EnableConfigServer

服务端 Environment 仓储

- **EnvironmentRepository**

Spring Cloud 配置服务器管理多个客户端应用的配置信息，然而这些配置信息需要通过一定的规则获取。Spring Cloud Config Sever 提供 EnvironmentRepository 接口供客户端应用获取，其中获取维度有三：

- **{application}** : 配置客户端应用名称，即配置项：spring.application.name
- **{profile}**: 配置客户端应用当前激活的Profile，即配置项：spring.profiles.active
- **{label}**: 配置服务端标记的版本信息，如 Git 中的分支名

服务端 Environment 仓储

- 实战演示
 - 基于文件系统实现
 - `spring.cloud.config.server.git.uri = ${user.home}/configs`
 - 基于Git 版本控制
 - `spring.cloud.config.server.git.uri = https://github.com/mercyblitz/tmp.git`

服务端 Environment 仓储

- 理解服务端配置映射

`/application/profile[/label]`

`/application-profile.yml`

`/label/application-profile.yml`

`/application-profile.properties`

`/label/application-profile.properties`

Spring Cloud 配置客户端

- **Spring Cloud Config Client**

Spring Cloud 配置客户端提供连接 Spring Cloud 服务端，并且获取订阅的配置信息。

- **配置Spring Cloud 配置客户端**

- 创建 bootstrap.yml 或者 bootstrap.properties
- 配置 spring.cloud.config.* 信息

动态配置属性Bean

动态配置属性Bean

- @RefreshScope
- /refresh Endpoint
- ContextRefresher