# 统一配置中心需求说明书

## 需求

为了更好的解决分布式环境下多台服务实例的配置统一管理问题，本文提出了一套完整的分布式配置管理解决方案（简称为disconf[4]，下同）。首先，实现了同构系统的配置发布统一化，提供了配置服务server，该服务可以对配置进行持久化管理并对外提供restful接口，在此基础上，基于zookeeper实现对配置更改的实时推送，并且，提供了稳定有效的容灾方案，以及用户体验良好的编程模型和WEB用户管理界面。其次，实现了异构系统的配置包管理，提出基于zookeeper的全局分布式一致性锁来实现主备统一部署、系统异常时的主备自主切换。通过在百度内部以及外部等多个产品线的实践结果表明，本解决方案是有效且稳定的。

## 技术分析

在一个分布式环境中，同类型的服务往往会部署很多实例。这些实例使用了一些配置，为了更好地维护这些配置就产生了配置管理服务。通过这个服务可以轻松地管理成千上百个服务实例的配置问题。

王阿晶提出了基于zooKeeper的配置信息存储方案的设计与实现, 它将所有配置存储在zookeeper上，这会导致配置的管理不那么方便，而且他们没有相关的源码实现。淘宝的diamond是淘宝内部使用的一个管理持久配置的系统，它具有完整的开源源码实现，它的特点是简单、可靠、易用，淘宝内部绝大多数系统的配置都采用diamond来进行统一管理。他将所有配置文件里的配置打散化进行存储，只支持KV结构，并且配置更新的推送是非实时的。百度内部的BJF配置中心服务[3]采用了类似淘宝diamond的实现，也是配置打散化、只支持KV和非实时推送。

同构系统是市场的主流，特别地，在业界大量使用部署虚拟化（如JPAAS系统，SAE，BAE）的情况下，同一个系统使用同一个部署包的情景会越来越多。但是，异构系统也有一定的存在意义，譬如，对于“拉模式”的多个下游实例，同一时间点只能只有一个下游实例在运行。在这种情景下，就存在多台实例机器有“主备机”模式的问题。目前国内并没有很明显的解决方案来统一解决此问题。