

本科毕业设计开题报告

1. 项目背景

酷家乐网站（www.kujiale.com）是隶属于群核云计算的装修设计网站，装修用户可以在酷家乐快速完成户型图绘制、获得免费户型风水报告、户型改造建议。用户还可以利用酷家乐的设计工具自行设计新家，并获得新家的渲染效果图。图片渲染过程非常耗时，公司采用几十台渲染服务器并行计算方式大幅度削减渲染时间。

家装渲染的过程中，需要家具、门窗、吊顶等 3D 模型和表现模型表面纹理的三角面片，这些素材文件已经积累有数 TB 的容量，而且每天还会新增 GB 级别的新素材。公司使用 swift 集群作为这些模型文件和纹理图片的存储方案。

Swift 是一个提供 RESTful HTTP 接口的对象存储系统，主要为网络提供应用服务。对象存储提供 Key-Value 方式的 RESTful 数据读写接口，并且常以网络服务形式提供数据的访问。它采用扁平化的数据组织结构，抛弃了嵌套的文件夹，从而避免维护庞大的目录树。这种扁平数据组织和 K/V 访问方式更能满足数据管理的需求。

IDC（Internet Data Center）机房，是企业互联网应用的发展而推出的一种优良的为企事业单位提供专业互联网服务的数据中心。原先公司有一个机房（本地机房），机房内有一个 swift 集群作为渲染素材（图片以及模型文件）的存储方案。渲染素材小组有新增渲染素材时在本地机房的 swift 集群中上传、替换或删除素材。

随着公司业务的快速增长，渲染服务器持续增多，机房数量新增至两个，很快将增至三个甚至更多。每个机房分别配备一个 swift 集群，原则上每个 swift 集群都存储所有的渲染素材，对于外部机房 swift 集群中的素材，我们需要提供一种机制把本地 swift 集群的素材同步过去。

2. 目标和任务

2.1. 问题

目前，应对多机房素材同步问题，我们采取的方法是定期使用 python 脚本实现跨机房同步。该 Python 脚本会检查本地资源增量，并将增量素材同步到其他机房的 swift 集群。但是这种人工调用脚本同步素材的方式会产生一系列问题：

第一，使用脚本同步方式不能满足自动化需求，随着业务增长，素材增长加快，工作量会迅速增大。脚本方式耗时耗力，而且出错率高。

第二，使用脚本同步方式存在高延时问题，目前每天使用脚本同步一次，这样新

增素材需要一天后才能同步到其他机房。对于个别同步失败需要重新同步的素材，素材上传者不能接受同步状态反馈有一天的延迟。

第三，外地机房渲染服务器在当地 swift 集群得不到素材时，会申请从本地机房服务器拿，因为网络传输的高延时，渲染时间会大大增加。

第四，跨机房的传输在外网进行，接口也暴露于外网环境，使用脚本同步无法提供安全机制的扩展。

2.2. 任务目标

面对以上临时解决方案出现的一系列问题，决定开发 IDC 资源备份服务系统 (IdcResource)，它可实现以下功能：

第一，IdcResource 提供统一的 API，提供同步资源、获取同步状态及进度等功能。素材上传到本地机房 swift 的时候自动调用 API，同步至外部机房。

第二，同步在上传新增素材时自动完成，上传者也能及时得到同步状态的反馈。IdcResource 还负责对同步状态的侦测，如果同步失败，IdcResource 在设定可重复次数内自动重新同步素材。

第三，外地机房服务器在 swift 集群中寻找不到所需素材时，不再直接向本地机房的 swift 集群申请传输素材，而是向 IdcResource 发出缺失素材的同步请求，由 IdcResource 负责完成同步，这样外地机房其他渲染服务器只需从其本地 swift 集群中获得素材。

第四，使用白名单限制 ip 访问权限，利用数字签名阻止中间人攻击等方法解决安全问题。

此外，idcResource 有良好的可扩展性。不久之后，公司机房将要新增至三个，每个外部机房都可以部署一个 idcResource。内部机房向任何一个 idcResource 发出同步情况后，已经同步的节点自动向其他 idcResource 同步，实现自动化多节点同步。

3. 可行性分析

为了控制项目实施风险，使项目目标更加合理，计划更加明确，我们对此项目进行了如下各方面的可行性分析。

3.1. 技术分析

Spring 框架：

Spring 是一个开源框架，它由 Rod Johnson 创建，是为了解决企业应用开发的复杂性而创建的。Spring 的用途不仅限于服务器端的开发，从简单性、可测试性和松耦合的角度而言，任何 Java 应用都可以从 Spring 中受益。

Swift 存储：

Swift 是一个提供 RESTful HTTP 接口的对象存储系统，主要为互联网应用服务用于服务器端的开发。

3.2. 环境和资源分析

在本项目开发的环境和条件约束下分析得到结论，我们对本项目的成果有较大的把握。

本项目在公司数位有长期开发经验的高级工程师带领下进行，并且得到浙大计算机学院陈越教授的指导。并且本人有较丰富的 Java 编程经验，有足够的相关语言及工作代码量，对系统架构设计和代码细节编写能较好的驾驭。

群核云计算公司拥有自己架设的多台服务器，提供虚拟机让我在上面部署需要的服务和软件系统进行测试。IdcResource 基础功能完成后就能直接部署到专用 IDC 服务器上应用，可以在实际应用过程中发现 bug，调试并修复 bug，同时可根据实际的应用需求开发更多的功能。

本项目从今天一月份开始实施，现已完成一半以上，已经解决框架设计、确定业务逻辑。之后安全方面的功能扩展，也能通过 Spring 框架已有的模块实现。因此，在毕设时间内，我们足以完成剩余的项目工作。

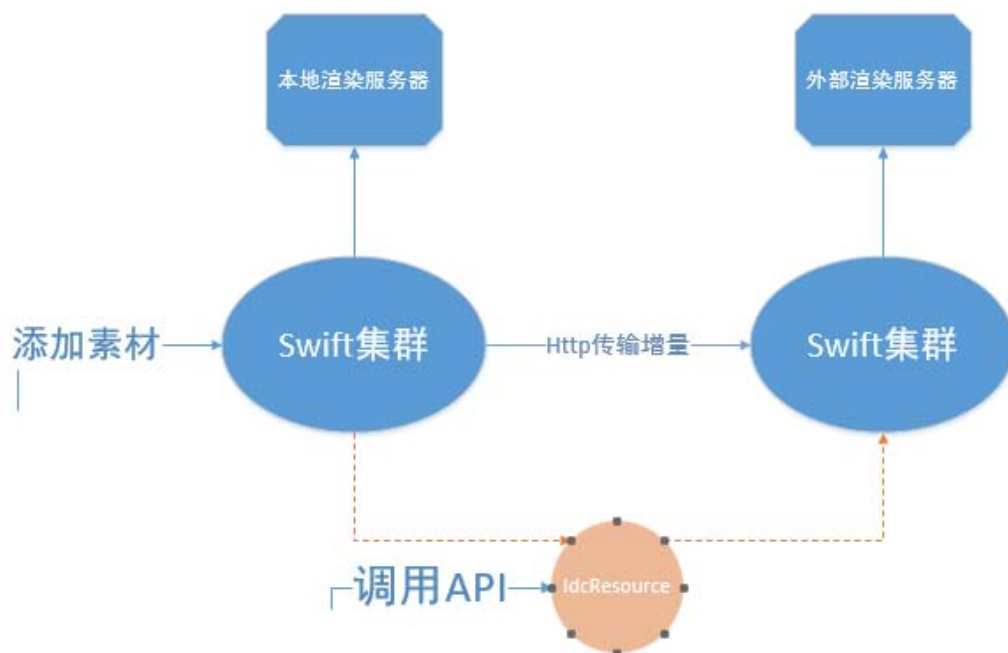
综上所述，我们的项目方案是可行的。

4. 初步技术方案和关键技术考虑

4.1. 初步设计方案

技术选择和整体构架设计已由公司中的同事和企业导师完成。

如下图所示，用户调用 IdcResource 提供的 API，实现不同机房 swift 集群的素材同步。



4.2. 关键技术

Spring 框架:

Spring 框架是一个开源的 Java 平台，它为容易而快速的开发出耐用的 Java 应用程序提供了全面的基础设施。与公司其他项目相同，本项目也在 Spring 框架开发，极大简化了开发难度。

IOC 和 AOP:

控制反转模式（也称作依赖性介入）的基本概念是：不创建对象，但是描述创建它们的方式。在代码中不直接与对象和服务连接，但在配置文件中描述哪一个组件需要哪一项服务。容器（在 Spring 框架中是 IOC 容器）负责将这些联系在一起。在典型的 IOC 场景中，容器创建了所有对象，并设置必要的属性将它们连接在一起，决定什么时间调用方法。

面向切面的编程:

即 AOP，是一种编程技术，它允许程序员对横切关注点或横切典型的职责分界线的行为（例如日志和事务管理）进行模块化。

Java 多线程编程:

每个素材从 1M 和 100M 不等，跨机房 http 传输需要一定的时间。如果单线程模式，每个素材等待传输结束返回成功同步状态后再进行下一个素材同步，将会非常耗时。Java 平台提供了一套广泛而功能强大的 API、工具和技术，其中包括对多线程编程内置的支持。一个多线程程序包含两个或多个能并发运行的部分。

Spring MVC 拦截器:

拦截器是指通过统一拦截从浏览器发往服务器的请求来完成功能的增强，可以用来解决一些如权限验证问题等的共性问题。我们使用 Spring MVC 框架带有的拦截器，实现接口的白名单功能。

5. 预期工作结果

我们的项目将要完成一个功能完善的在分布式计算环境下实现资源同步的系统。它提供完整的 API，包括发送同步请求、查询同步进度等。系统内部实现同步失败后的重试机制，并且返回同步状态给接口调用者。

在不久之后新增至三个以上机房后，内部机房素材想任何一个 idcResource 发出同步请求后系统自动在多个外部机房间实现同步。

安全方面，利用白名单和为 idcResource 接口增加签名认证方式来限制 API 的调用权限和防止中间人攻击。

6. 进度计划

日期	内容	进度
2016.1~2016	了解公司工作流程，搭建工作环境；	已完成