# 一．概述

当前控制系统的日志模块是基于文件系统来进行存储和访问的，在开发过程和用户使用过程中遇到文件管理分散，性能优化困难，编程复杂度高等问题。考虑兼顾目前现状，解决以上问题，尝试在对现有系统不进行结构调整的情况下，对日志系统进行改进升级。

# 二．系统

1. 系统概述

创建一个Window服务程序，定期从文件中导入日志到MongoDB的日志表中，构建一个Web服务，对外开放日志的基本查询。

1. 系统结构

# 三．详细设计

1. Data Transfer Service
2. 功能

DTS基于WCF，采用HTTP协议，构建window服务，定期导入Event到MongoDB中，且提供简单的设置，修改服务设置。

1. 导入数据任务可可视化，可管理
2. 设置：采用Restful对参数进行设定

间隔：定期执行的时间

策略：从新往旧/从旧往新/指定文件

1. 表结构

Events（EventID，Message）

存储控制系统产生的Event

Tasks（id, FilePath, LineNumber,IsActive,CreateDate,ModifyDate）

Task为导入任务，每一次导入就是一个任务

Log\_Setting(type,value,modifydate)

修改一些导入设置参数的记录

Log\_Task(id, filepath, fromlines,tolines, begindate, enddate)

每次导入任务的执行情况日志

Log\_Query(id,querydate,querytimespan)

每次执行查询后的情况日志

1. 接口

查询导入历史记录

查询任务执行情况

1. NodeJs Web Server
2. 功能

提供基于HTTP协议的接口，JSON的数据格式，对外提供Events的查询

1. 接口

Querylog(begindate,enddate,type)

Querylog(begindate,enddate,type,msg)

1. 优化

分页

# 四．开发环境

1. 开发工具

NodeJS, Sails, WebStorm，MongoVIE

1. 项目管理

NPM

1. 集成单元测试
2. 源码管理

GitHub，SourceTree

# 五．搭建环境

* 1. 源码管理
  2. MongoDB
  3. Sails（WebStorm）