**logCrawler 核心日志爬找设计文档**

**V2.1**

**二〇一五年八月四日**

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **修订说明** | **修订人** |
| 2015-8-4 | 1.0 | 初稿 | 刘剑群 |
| 2015-8-12 | 2.0 | 增加数据抽取展示部分 | 刘剑群 |
| 2015-8-17 | 2.1 | 增加集合设计，用于后期展示部分 | 刘剑群 |
|  |  |  |  |

1. **存在问题**

目前，支撑查障，全人工查找，尤其是对于一些有规律性、异常并且大量的报文，全靠人工根据经验去巡检，不仅耗时，而且巡检后的结论，准确性不搞，也未必覆盖触发的诱因。由此急需一种程序，自动、高效、智能的去协助生产支撑人员、开发人员查询故障，提取异常日志信息。重复劳作的事务让工具做，让开发、支撑人员更关注于分析处理故障上。

1. **解决事项**
2. 爬找关键词查障类
   * 超时接口报文请求监测，异常导出到异常日志文件
   * 根据输入关键字，查找相关联的所有日志输出
   * 根据输入关键字，统计数量。用于报文请求数量统计

例如，可统计本日志中，某个订单接口的

* + 根据输入的平台包参数组，获取相关联的所有处理日志

例如，查找订单接口 SCS0001 ,401 参数组4001参数组为000000 ，爬出相关规则的请求流水报文。

* + 两级关键字查找。爬寻出相关联的所有日志输出

例如，查询 签约接口CUM0012 下的所有流水请求日志中，再过滤抽取包含ORA 异常的日志。

备注：考虑到生产查障的需要，此操作不涉及数据库操作，全文扫描获取所需。

1. 统计展示类
   * 文本/数据统计

* 把统计的各类关键数据，拆分成字段，经过过滤分析，存储进数据库表T\_SCS\_SERVICE\_LOG（核心服务日志表）。
* 根据已有数据，统计出每日接口使用量（内部接口调用需区分，不符合累计），平均耗时等数据信息。
  + 报表图形化统计显示

根据上文T\_SCS\_SERVICE\_LOG 导入的每日数据，图表方式显示。可分类按照 时间、天数、月份等不同时间段，统计接口平均时长、接口某一阶段数量、响应码和超时的接口。

* + 应用服务器、总集群 接口平均时长、最长耗时、最短耗时
  + 支持可配置。例如1s 、1min 某台机器的接口数量等
  + 报表展示机器、总集群的QPS、QPM、QPH
  + 可生成报告，展示关键数据

1. 增量实时爬取

此部分需要协同监控组，开如何进行智能分工。

1. **工作计划**

详细请见后面的附录。

需求确认人：刘剑群

需求开发人： 林子恒/严锦木

1. **数据库选型与设计**
2. 数据库选型-MongoDB

由于存储的是日志关键信息的数据，数据/结构具有以下特点：

1. 数据量极大，每天多台服务器，几十万的数据量
2. 数据插入后，对更新无要求，但会多次进行不同的要求，进行查找

故综上所述，采用NoSQL 数据库系列中的MongoDB 数据库，能满足本次数据库所需。有以下几点优势：

* 弱一致性。更加能保证用户的访问速度
* 对海量数据的高效率存储和访问的需求
* 学习成本较低。易上手、易部署，操作类似Mysql。
* 社区活跃，免费开源。
* 支持python、ruby、C++、java 等语言
* 非关系型、分布式、水平可拓展等拓展性。

1. 数据类型表述代码映射关系表

表 数据库类型定义表

|  |  |  |  |
| --- | --- | --- | --- |
| 定义 | 代码 | 类型描述 | 示例 |
| 空类型 | NULL | null类型表示空值或不存在的字段 | {“one”:null} |
| 布尔类型 | BOOL | true /false | {“one”:false} |
| 64位浮点数 | NUMBER | 32、64位浮点数、整数都转义为64位浮点 | {“one”:2.55} |
| 字符串 | STRING | 符号、字符串都转义为字符串 | {“one”:”hello”} |
| 对象ID | OBJECTID |  | ObjectId("4eae239f63520362e051e7fd") |
| 日期 | DATE | 使用的时候加上new | {“one”:new Date()} |
| 正则表达 | REGEX | 正则表达式按照JavaScript规范 | {“one”:\w\s} |
| 代码 | JS | 格式按照javascript格式 | {“one”：function:{/\*xxx\*/}} |
| 数组 | ARRAY | 数组内部还可嵌套数组 | {“a”:[“a”,”b”,[“c”,”d”]]} |
| 文档 | DOC | 嵌套文档 | {“x”:{“name”:”JQun”,”age”:27}} |

1. 数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| mongodb | | | | |
| 数据库：tmdb ；集合：colUnitlogs | | | | |
| 集合用途：基础单元数据，为后期各种拓展统计、数据筛选做基础数据。 | | | | |
| NO. | NAME | DESC | TYPE | Remark |
|  | SEQ\_ID | 流水号 | STRING | 日志流水号 |
|  | SEQ\_REQU | 请求报文 | STRING |  |
|  | SEQ\_RESP | 返回报文 | STRING |  |
|  | MAC\_IP | 应用服务器地址 | STRING |  |
|  | SERVICE\_NAME | 接口名 | STRING |  |
|  | BEGIN\_DATE | 开始处理时间 | DATE |  |
|  | END\_DATE | 结束时间 | DATE |  |
|  | TMSERVICE\_PID | 服务进程ID | NUMBER |  |
|  | LEVEL\_ID | 父节点日志级别 | NUMBER | 0：默认一级  1：内部第二层历遍  以此类推 |
|  | FILE\_NAME | 文件名 | STRING |  |
|  | STAT | 启用状态 | STRING | S0A:启用  S0X：弃用 |
|  | MARK1 | 备注1 | STRING |  |
|  | MARK2 | 备注2 | STRING |  |
|  | USED\_TIME | 报文用时 | NUMBER | 保持浮点数2位 |
|  | BATCH\_SEQID | 批次号 | STRING |  |
|  | \_id | 默认唯一键 | OBJECTID |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| mongodb | | | | |
| 数据库：tmdb ；集合：colServiceSlice | | | | |
| 用途：按照每日时间切片，统计每秒的请求数和响应数。 | | | | |
| 数据特例：  SERVICE\_NAME=all，表示针对所有接口切片的统计  MAC\_IP = all，表示针对总集群所有机器的接口负载情况 | | | | |
| NO. | NAME | DESC | TYPE | Remark |
|  | SERVICE\_NAME | 接口名 | STRING |  |
|  | TimeSlice | 时刻 | TIME | 一天切割成24X60X60时间点 |
|  | RequestCount | 请求接口数量 | NUMBER |  |
|  | ResponseCount | 返回接口数量 | NUMBER |  |
|  | STAT | 启用状态 | STRING | S0A:启用  S0X：弃用 |
|  | BATCH\_SEQID | 批次号 | STRING |  |
|  | MAC\_IP | 应用服务器地址 | STRING |  |
|  | REQ\_AVG | 请求平均时长 | NUMBER |  |
|  | REQ\_MAX | 请求耗时最大值 | NUMBER |  |
|  | REQ\_MIN | 请求耗时最小值 | NUMBER |  |
|  | \_id | 默认唯一键 | OBJECTID |  |

1. **详细设计**

表 操作命令

|  |  |
| --- | --- |
| 操作命令 | 描述 |
| -h|-help | 帮助文档 |
| -v|-version | 版本号 |
| -type export –f all | 导入当前文件夹所有文件到数据库 |
| -type export -f [pathfile] | 导入制定文件到数据库 |
| -type rule –k [keyword] |[keyword] –f all | 根据关键字，爬找文件夹下所有日志（keyword|keyword 表示同个流水号下 包含的关键字文件，不局限与一行） |
| -type rule –k [keyword] |[keyword]–f [pathfile] | 根据关键字，爬找相关联所有日志 |
| -type rule –t[timeout value] –s all -f [pathfile] | 针对所有接口，查找所有接口，爬找所有超时的接口 |
| -type rule –t[timeout secondvalue] –s [serv\_name] -f [pathfile] | 根据接口名，查找相关联，爬找所有超时的接口 |

1. **展示**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **一期进度** | 15% |  | | | | | |
| **本周重点工作** | 里程碑需求确认；自测工具类图设计定稿、接口文档 | | | | | | |
| **需配合事项** | 支撑组：在开发环境搭建mongodb数据库和pymongo 包。（87.9 已搭建好） | | | | | | |
| **需求计划** | **里程碑产物** | **内容** | **工作日** | **开始时间** | **结束时间** | **状态** | **描述** |
| 搭建日志工具框架；文档元数据导入mongoDB中 | 需求说明文档/规划 | 2 | 8月 7日周五 | 8月 11日周二 | 完成 |  |
| 需求详细设计(类图、接口文档) | 2 | 8月12日周三 | 8月13日周四 | 完成 |  |
| * 研发实现 | 3 | 8月14日周五 | 8月18日周二 | 进行中 |  |
| * 互测（Excel案例） | 1 | 8月19日周三 | 8月19日周三 |  |  |
| * 代码互审/修复 | 1 | 8月20日周四 | 8月20日周四 |  |  |
| 动态分析源日志，导入符合规范日志 | 需求分析/详细设计 | 1 | 8月14日周五 | 8月19日周三 | 进行中 | 同步 |
| 研发实现 | 2.5 | 8月24日周一 | 8月26日周三 |  |  |
| 互测（Excel案例） | 0.5 | 8月26日周三 | 8月26日周三 |  |  |
| 代码互审/修复 | 1 | 8月27日周四 | 8月27日周四 |  |  |
| mongoDB 集合操作，以界面化展示数据.一期 | 需求分析/详细设计 | 3 | 8月14日周三 | 8月18日周二 | 进行中 | 同步 |
| 研发实现 | 3.5 | 8月24日周一 | 8月26日周三 |  |  |
| 互测（Excel案例） | 0.5 | 8月26日周三 | 8月26日周三 |  |  |
| 代码互审/修复 | 1 | 8月27日周四 | 8月27日周四 |  |  |
| 外部使用/培训/投产 | 操作手册 | 1 | - | - | 未开启 |  |
| 测试验收 | 0.5 | - | - |  |  |

表 核心进度表