# 可选组件

## 全链路跟踪

### 简介

****请求链路追踪，故障快速定位****：可以通过调用链结合业务日志快速定位错误信息。

****可视化****：各个阶段耗时，进行性能分析。

****依赖优化****：各个调用环节的可用性、梳理服务依赖关系以及优化。

****数据分析，优化链路****：可以得到用户的行为路径，汇总分析应用在很多业务场景

### 安装

#### skywalking 环境搭建

#### 软件包版本

1、apache-skywalking-apm-6.6.0.tar.gz

将以上三个软件包上传至服务器/usr/local/src/路径下

下载地址

<http://39.100.254.140:12011/loit-Infrastructure-doc/loit-initproject-doc/tree/master/3%E3%80%81other/%E9%9B%86%E6%88%90loit%E9%97%A8%E6%88%B7/skywalking>/apache-skywalking-apm-6.6.0.tar.gz

#### 安装Skywalking服务

cd /usr/local/src  
tar -zxvf apache-skywalking-apm-6.6.0.tar.gz  
mv apache-skywalking-apm-incubating skywalking  
mv skywalking/ /HWStor/usr/local/

vim /HWStor/usr/local/skywalking/config/application.yml

**存储选择h2 或者 ES相应配置文件如下：**

1、存储h2

配置如下：

storage:

h2:

driver: ${SW\_STORAGE\_H2\_DRIVER:org.h2.jdbcx.JdbcDataSource}

url: ${SW\_STORAGE\_H2\_URL:jdbc:h2:mem:skywalking-oap-db}

user: ${SW\_STORAGE\_H2\_USER:sa}

1. 修改存储h2为elasticSearch存储

修改配置如下：

storage:

# h2:

# driver: ${SW\_STORAGE\_H2\_DRIVER:org.h2.jdbcx.JdbcDataSource}

# url: ${SW\_STORAGE\_H2\_URL:jdbc:h2:mem:skywalking-oap-db}

# user: ${SW\_STORAGE\_H2\_USER:sa}

elasticsearch:

# nameSpace: ${SW\_NAMESPACE:""}

clusterNodes: ${SW\_STORAGE\_ES\_CLUSTER\_NODES:192.168.251.146:9200}

indexShardsNumber: ${SW\_STORAGE\_ES\_INDEX\_SHARDS\_NUMBER:2}

indexReplicasNumber: ${SW\_STORAGE\_ES\_INDEX\_REPLICAS\_NUMBER:0}

# Batch process setting, refer to https://www.elastic.co/guide/en/elasticsearch/client/java-api/5.5/java-docs-bulk-processor.html

bulkActions: ${SW\_STORAGE\_ES\_BULK\_ACTIONS:2000} # Execute the bulk every 2000 requests

bulkSize: ${SW\_STORAGE\_ES\_BULK\_SIZE:20} # flush the bulk every 20mb

flushInterval: ${SW\_STORAGE\_ES\_FLUSH\_INTERVAL:10} # flush the bulk every 10 seconds whatever the number of requests

concurrentRequests: ${SW\_STORAGE\_ES\_CONCURRENT\_REQUESTS:2} # the number of concurrent requests

vim /HWStor/usr/local/skywalking/webapp/webapp.yml  
修改配置如下：

server:  
 port: 8081  
  
collector:  
 path: /graphql  
 ribbon:  
 ReadTimeout: 10000  
 # Point to all backend's restHost:restPort, split by ,  
 listOfServers: 192.168.251.146:12800

启动：

cd /HWStor/usr/local/skywalking/bin/  
sh startup.sh

下载

<http://39.100.254.140:12011/loit-Infrastructure-doc/loit-initproject-doc/tree/master/3%E3%80%81other/%E9%9B%86%E6%88%90loit%E9%97%A8%E6%88%B7/skywalking>/skywalking-webapp.jar

**替换掉/usr/local/loit/soft/skywalking/webapp 下的**

**skywalking-webapp.jar (此jar做了样式修改)**

#### 附：启动／关闭命令

# 关闭 skywalking web

ps -ef|grep skywalking-webapp

kill -9 pid

# 关闭 OAPServer

ps -ef|grep skywalking.oap.server.starter.OAPServerStartUp

kill -9 pid

# 启动skywalking

cd /usr/local/loit/soft/skywalking/bin/

sh startup.sh

11800 -> 12001 client

12800-> 12002 collection

8080 > 12000 web

### 客户端使用

#### 客户端代理：agent

实际开发时候，每一个jar包获取应用都应该单独使用一个agent，  
所以将agent这个目录拷贝到各自对应的jar包路径下。  
  
核心部分的目录信息如下:

├── activations  
├── config  
│ └── agent.config  
├── logs  
│ └── skywalking-api.log  
├── optional-plugins  
├── plugins  
└── skywalking-agent.jar

其中，config/agent.config是最重要的，需要修改的核心参数如下所示  
  
# 应用名称，当前代理的应用名称，用于UI界面分类和展示  
agent.service\_name=${SW\_AGENT\_NAME:Your\_ApplicationName}  
# 收集器的地址，这个根据实际情况设置，上述`Collector`在哪台服务器启动，ip就设置为多少。

collector.backend\_service=${SW\_AGENT\_COLLECTOR\_BACKEND\_SERVICES:192.168.251.146:11800}

开发11800、12800端口：

firewall-cmd --list-ports  
firewall-cmd --zone=public --add-port=11800/tcp --permanent  
firewall-cmd --reload

设置好参数后，对于 Java 应用，添加核心的-javaagent进行启动  
java -javaagent:agent/skywalking-agent.jar -jar xxx.jar  
  
idea里启动，添加VM启动参数，例如：

-javaagent:.../agent/skywalking-agent.jar   
-Dskywalking.agent.service\_name=test\_etl-local   
-Dskywalking.collector.backend\_service=192.168.251.146:11800

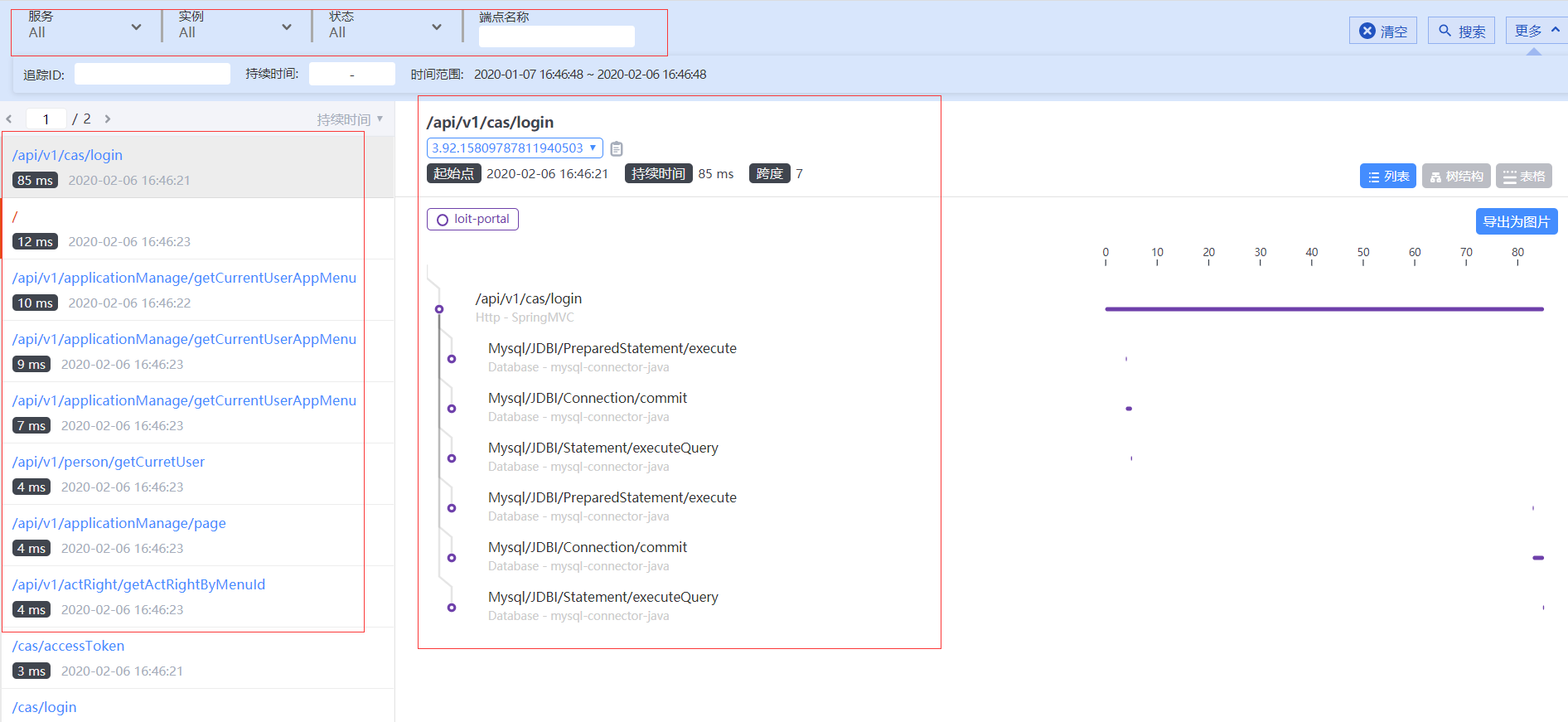
#### 启动客户端

启动脚本中添加如下

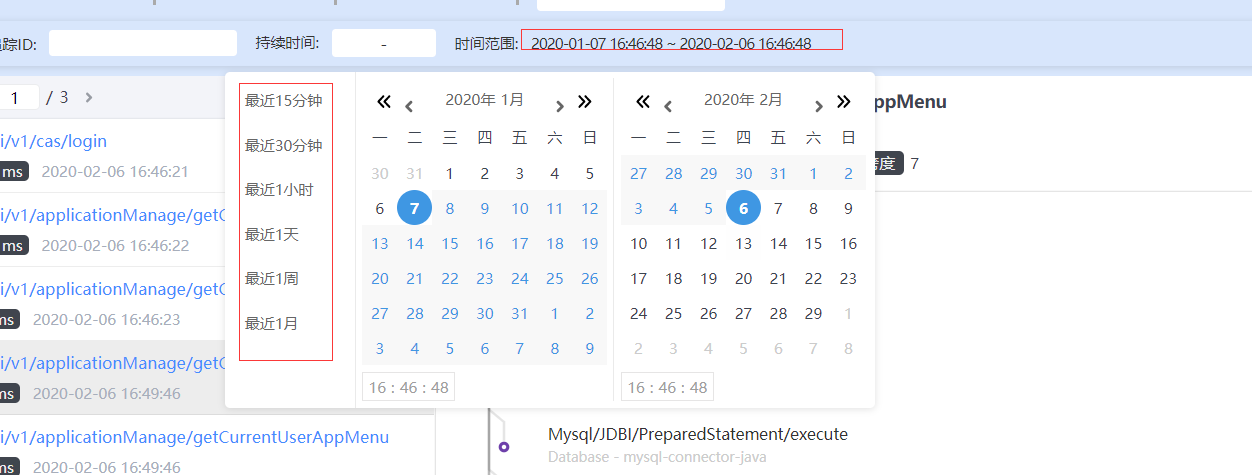
nohup /usr/local/java8/jdk1.8.0\_11/bin/java -javaagent:/usr/local/loit/soft/skywalking/agent/skywalking-agent.jar -Dskywalking.agent.service\_name=loit-portal -Dskywalking.collector.backend\_service=39.100.254.140:12001 -Xms768m -Xmx768m -XX:PermSize=256M -XX:MaxPermSize=512M -jar $KILL\_PROCESS\_NAME --spring.profiles.active=test7009 >/var/www/portal-7009/loit-portal.log 2>&1 &

### Dashboard 使用说明

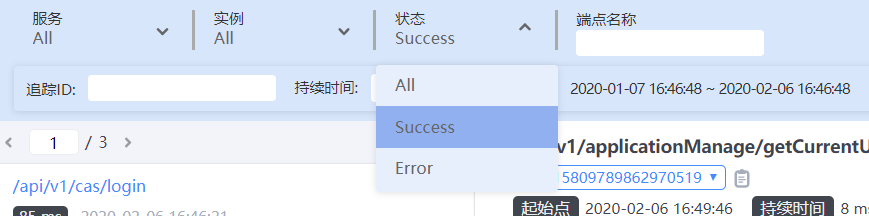
可以根据服务、实例、状态、时间等搜索链路信息



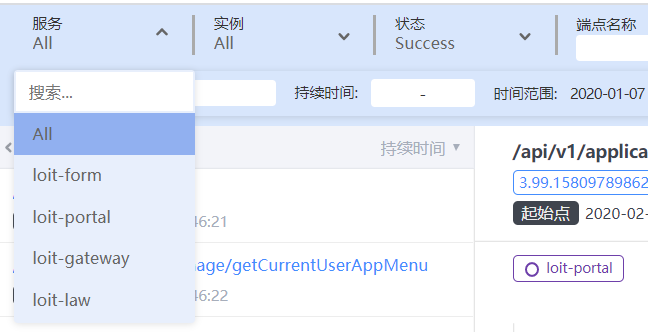
根据时间查询



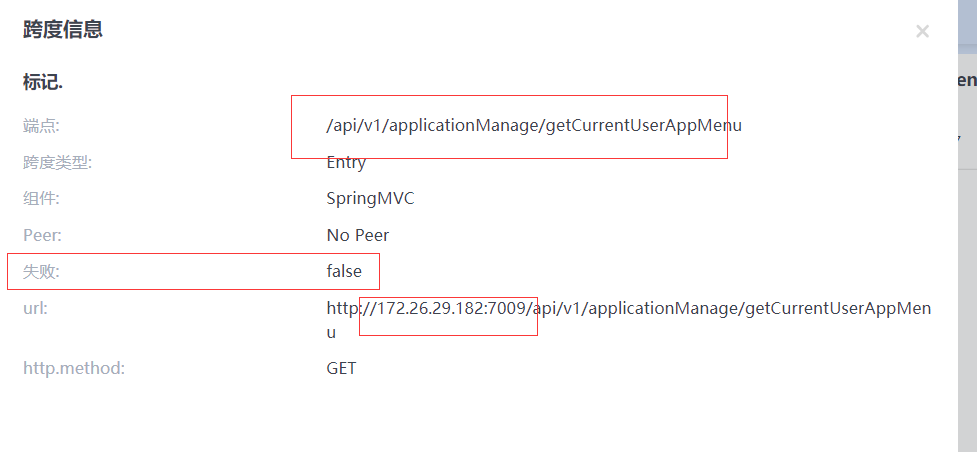
根据状态查询



根据服务查询



点击详情可以查看每个链路调用服务器地址和状态信息



可以查看链路调用过程中sql 执行情况

