

Monitor服务重构设计 -Influxdb

工具

被lixininfo添加，被章耿最后更新于十月 20, 2016

背景

目前监控报表基于 OpenTSDB base on Hbase，性能方面在大规模开启方法监控后无法保证；无法动态从原始1分钟级别的数据集合出更大粒度的数据；自行设计的worker无法及时的算完所需的统计数据；

重构选型influx db，时间序列型数据库，特点是快速、可扩展，数据可自动聚合、动态聚合功能非常棒；

服务数据量估算—monitor数据产生速度

按照上限计算每分钟产生数据：2000（接口）*10（方法）*40（provider count）*80(consumer count) = 64,000,000 6千4百万条；

线上最典型的一个组 productService: 72个provider 2734个consumer 72*2734/4 =57600条

按此计算 需要的插入速度：64,000,000 /60 = 1,066,667条 / 秒 即1百万条每秒才能满足；

HTTP 单线程插入测试：60*10*1440=864000条 插入约需9分钟，即每秒插入速度1440条；（2014-11-01）

性能方面需要再探索下采用UDP或者TCP方式来发送；生产环境需要提升到10w条左右 / 秒比较合适，如单接口不能满足则考虑部署多个 influx DB来散列写；

864000条占硬盘47M，约合每条57个字节；

需要跟进的问题：

- 1）大数据量下的性能，写入、读取、聚合；
- 2）continues query自动聚合数据；
- 3）监控按方法设置开启，客户端发送频率动态可在JSF Web console动态调整；

influx db存储结构设计

Database jsf;

性能流水表

Serie Name:performance_接口名_方法名

column name	类型	含义	示例
time（必有）	数值	时间戳	
host	字符	providerIP	
port	字符	provider port	
reomte	字符	consumer IP	
callTimes	数值	调用次数	
inData	数值	进入的数据包	
outData	数值	流出数据包	
elapsed	数值	耗时	

异常流水表

Serie Name:exception_接口名_方法名

column name	类型	含义	示例
time	数值	时间戳	

column name	类型	含义	示例
host	字符	providerIP	
port	字符	provider port	
exceptionName	字符	异常名称	
num	数值	异常次数	

查询语句设计

select median(callTimes),host,remote from com.jd.HelloService_hello group by time(30m) where time>'2014-11-01 00:00:00' and time <'2014-11-01 12:00:00' and host='172.17.18.1:20880' and remote='172.17.18.1'

聚合某接口某方法 时间范围内 10分钟内的调用次数 指定host 、 remote 以及port
select sum(callTimes) from com.jd.HelloService_hello group by time(10m) where time>'2014-11-01 00:00:00' and time <'2014-11-01 01:00:00' and remote='172.17.18.1' and host='172.17.18.1' and port='20880'

Continuous Queries Example

1	select mean(instantaneous_ops_per_sec) as instantaneous_ops_per_sec,mean(connected_clients) as connected_clients,mean(used_memory) as used_memory,mean(expired_keys) as expired_keys,mean(net_io_in_per_sec) as net_io_in_per_sec,mean(net_io_out_per_sec) as net_io_out_per_sec from /^instance.*/ group by time(2m) into day :series_name	
8	select mean(instantaneous_ops_per_sec) as instantaneous_ops_per_sec,mean(connected_clients) as connected_clients,mean(used_memory) as used_memory,mean(expired_keys) as expired_keys from /^day_instance.*/ group by time(20m) into week :series_name	
9	select mean(instantaneous_ops_per_sec) as instantaneous_ops_per_sec,mean(connected_clients) as connected_clients,mean(used_memory) as used_memory,mean(expired_keys) as expired_keys from /^day_instance.*/ group by time(1h) into month :series_name	
10	select mean(instantaneous_ops_per_sec) as instantaneous_ops_per_sec,mean(connected_clients) as connected_clients,mean(used_memory) as used_memory,mean(expired_keys) as expired_keys from /^day_instance.*/ group by time(12h) into year :series_name	

报表设计

三个报表相关页面，注意此处所说的报表都是指细化到方法级别的监控数据即：interfaceId-method:

default dashboard 页面：展示用户所关心的方法的最近半小时指标（调用次数、方法耗时、异常次数）；

指标页面：默认展示指定接口、指定方法 最近半小时的调用，方法耗时，用户可以设置时间范围、group by的粒度；

统计页面：用户统计一段时间内的调用来源分布，数据量分布、异常来源分布；

	无
--	---