

falcon 系统设计

为什么设计falcon

- 负载
- 规则配置
- 数据清理

设计目标

3

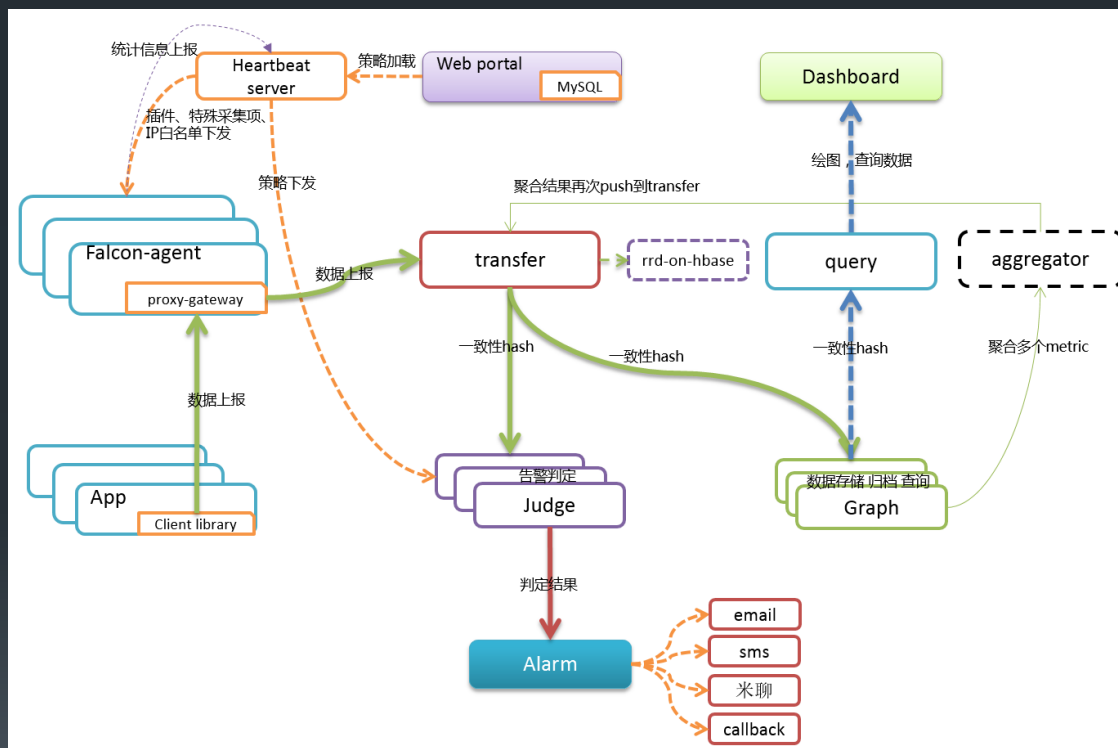
- 水平可扩展
- 数据易维护
- 采集数据无状态
- 数据的生产 / 消费分离



实现

Architecture

5



- 每个采样数据都带有完整的描述信息

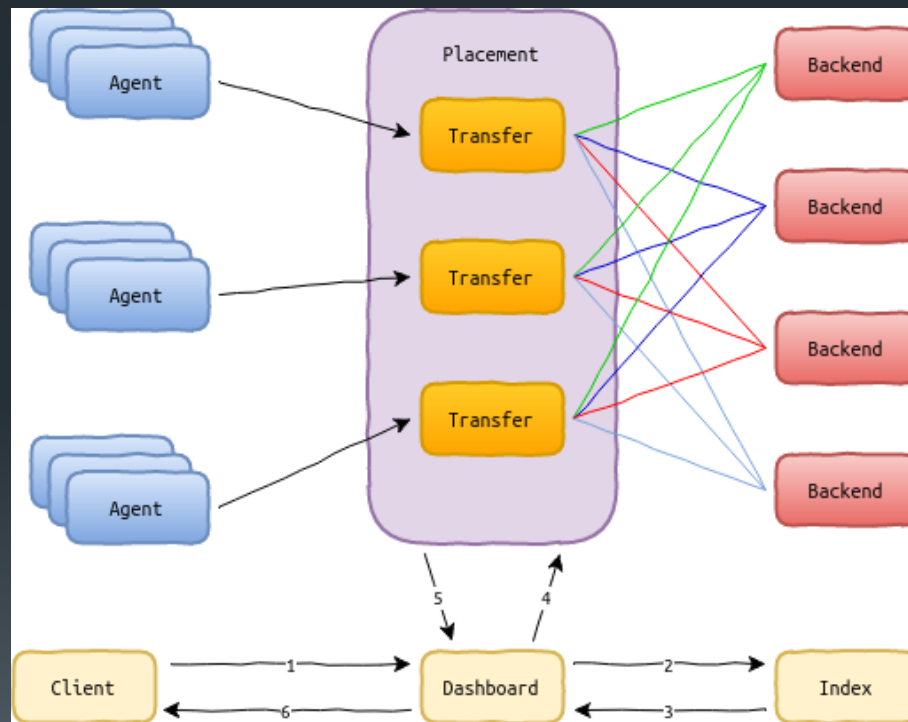
```
{  
  metric: load.1min,  
  endpoint: open-falcon-host,  
  tags: srv=falcon,idc=aws-sgp,group=az1  
  value: 1.5,  
  timestamp: 1489549579,  
  counterType: GAUGE,  
  step: 60  
}
```

传输

7

falcon的通讯方式，较好的解决了之前的性能瓶颈,小米目前的监控系统，每秒传输的数据采样点在850,000左右

- placement(PUT,GET)
 - 使用一致性hash实时计算出采样点的位置信息
- index(LIST)
 - 通过tag，metric查询采样数据时，需要使用索引，查询出数据的hash值



存储

8

以小米目前的负载，每天会产生547G的原始数据，会给存储和检索带来较大压力

- 默认使用rrdtool作为存储
 - 实时归档,查询时的数据不需要整理
 - 数据文件大小固定，不需要维护
- 支持tsdb
- 历史数据的查询
 - rrd 整理后存储，没有保存原始数据
 - tsdb, hbase
- 由于使用了hash分片，数据无法直接聚合
 - Cluster

单位	秒	天	月	年
183byte	148.34M	12.22T	366.70T	4.36P
8byte	6.48M	547.17G	16.03T	195.04T

查询

9

<http://perfcounter.pt.xiaomi.com/screen/17012?cols=4>

- web browser -> dashboard -> query -> graph
- 20 picture
- 8 line / picture
- 280 point / line
- 160 request
- 44800 point
- 耗时0.7~1.2s



报警的策略通过匹配后绑定到相应的数据样本上，每当有数据更新时，与之相关的策略被执行，满足报警条件，将会触发策略中的报警行为

- 策略配置 portal
- 匹配触发 judge
- 报警发送 alarm
- 没有数据时的报警 nodata
 - 动作是响应式的，需要事件触发

TODO

11

- 所有模块自动发现，中心化配置
- 数据的冗余,恢复,迁移
- 更丰富的查询&判断
- 模块&插件支持

References

12

- <http://open-falcon.org>
- <http://github.com/open-falcon>
- <http://oss.oetiker.ch/rrdtool>
- https://en.wikipedia.org/wiki/Consistent_hashing



Thank You