**⼀、整体认知APM系统**

**提问**：

同学们是否有遇到过这样的场景：业务运营⼈员反映、会员很多功能访问很慢。你做为会员

系统的负责⼈，接到反馈后做了如下事情：

1. 检查线上11台机器各硬件负载指标均正常

2. 查询线上11台会员系统⽇志，出现⼤量请求Timeout

3. 检查3台mySql数据库，访问⾮常慢，并且连接已占满

4. 进⼀步查明，⼤量连接都在执⾏⼀个会员注册的Sql语句，找到真凶。

**⼤家预估⼀下以上4个过程如果不借助任何⼯具，都以⼈⾁⽅式进⾏，需要多⻓时间？**

【学员发散式回答问题】

**那以上步骤怎么去优化呢？**

**第1：**采⽤Zabbix 对线上机器各性能指标进⾏集中监控。**第2：**使⽤ELK对线上⽇志集中存储查询。

**第3：**使⽤Anemometer 集中监控mysql 语句

以上3个功能合起来就是我们的APM系统，它对公司的意义，还需要我在啰嗦吗？当然⼀个完

整的不此之些，APM还可以进⾏代码级定位。什么叫代码级定位?

我们继续上述例⼦，上述问题解决了么？并没有，因为这个Sql语句不是慢查询，且这时的会

员注册流量并不⼤，原因⼜是什么呢？只能翻开代码找。这个时间可能⽐前⾯四个步骤花的

时间还要⻓。⽽代码级定位可直接定位⾄问题所在源码。

**APM 简介**

APM全称*Application Performance Management(应⽤性能管理),APM致⼒于监控和管理*

*应⽤软件性能和可⽤性。通过监测和诊断复杂应⽤程序的性能问题，来保证软件应⽤程序的*

*良好运⾏*

APM 发展的三个阶段

**第⼀阶段:** 以⽹络为中⼼，⽹速即应⽤速度，APM 主要指基础组件的性能监控

**第⼆阶段**: 以 IT 部件/组件为中⼼，监控围绕着部件/组件健康状态，基础设施可⽤性，伴随

IT 基础架构组件发布

**第三阶段:** 以 IT 应⽤核⼼和业务交易为中⼼，IT 架构⾼度复杂性，专注 IT 应⽤，⾯向⽤

**各互联⽹公司APM解决⽅案**

1. **google** :dapper

2. **twitter:** zipkin （开源）

3. **淘宝：**鹰眼

4. **京东：**hydra （开源）

5. **⼤众点评：**CAT （开源）

6. **⼩⽶：**open-falcon （开源）

**⼆、现有监控系统解决⽅案对⽐**

**Zabbix 与open-falcon**

主⼯功能与特点:

- CPU负荷

- 内存使⽤

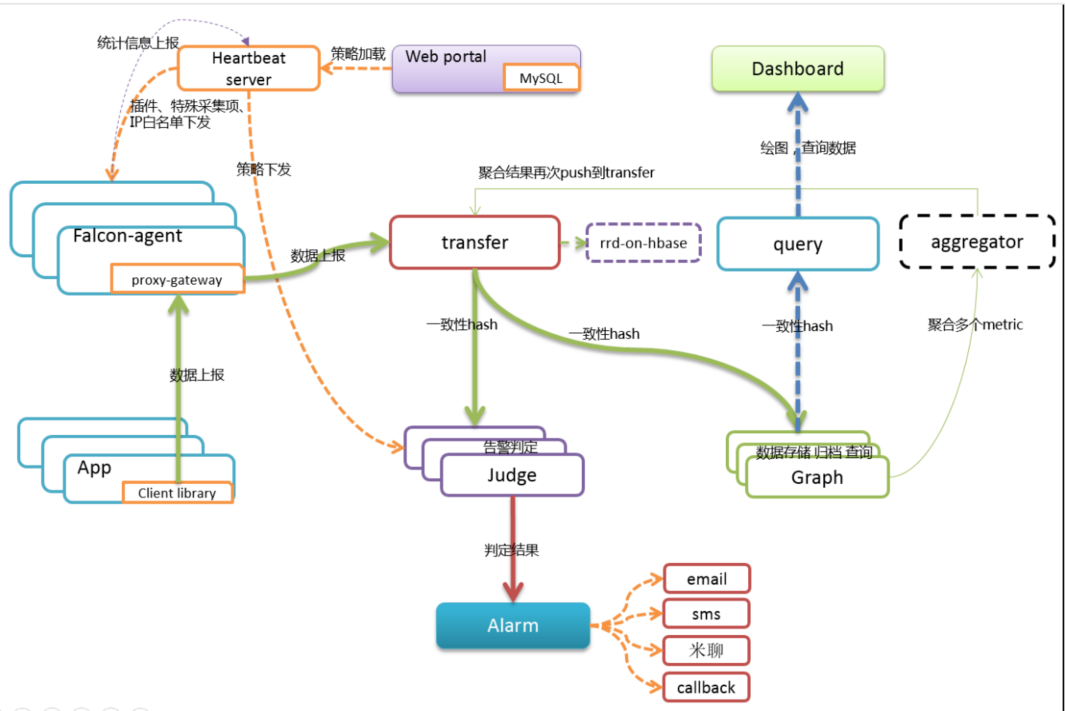
- 磁盘使⽤

- ⽹络状况

- 端⼝监视- ⽇志监视



**Open-falcon 架构图**



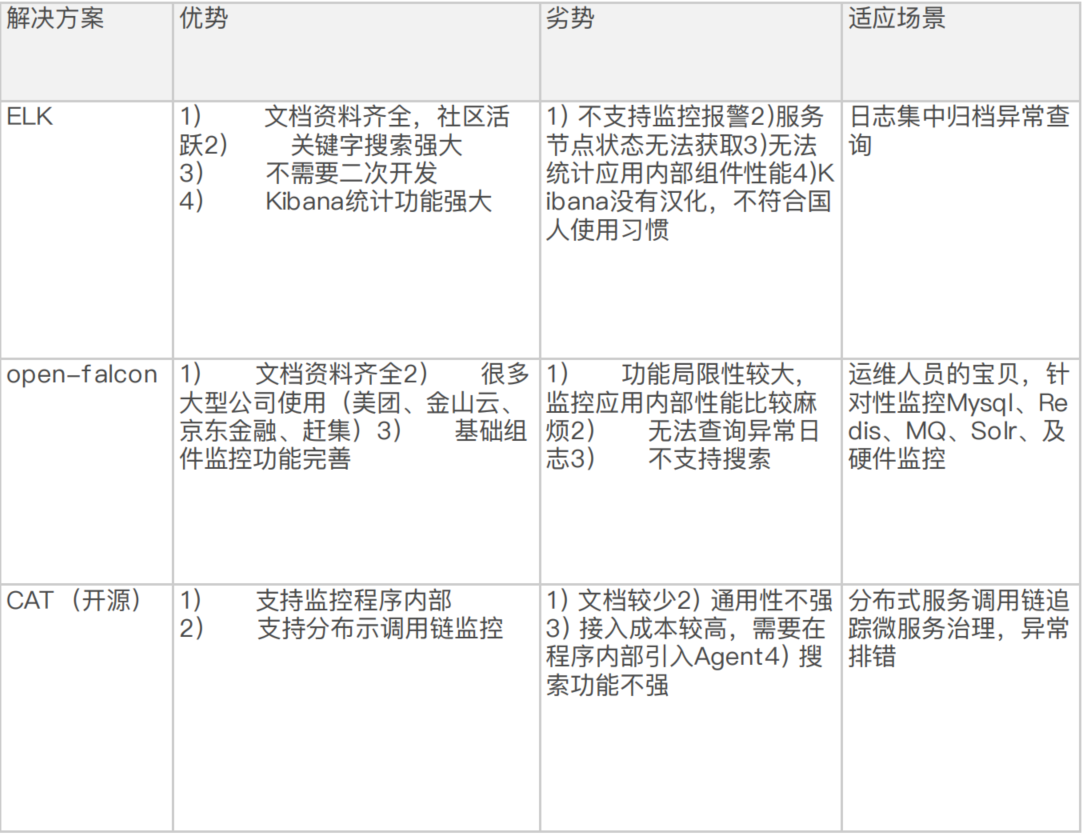
单纯服务节点监控与报警，以上两个开源软件都在⼀定程度上可以⽀持，但做为APM还缺失

了⽇志集中搜索与查询的功能。这就需要另外⼀套技术来补充。

**Elastic Stack**



**解决⽅案对⽐：**



**注：**

open-falcon ⽂档中⼼:http://book.open-falcon.org/zh/intro/index.html

Elastic 官⽹：https://www.elastic.co/cn/products

以上⽅案的缺陷：⽆法做到灵活的监控应⽤内部

三、APM系统架构实战

架构⽬标：

1. 功能性需求

a. 系统和服务指标收集

b. ⽇志集中存储与搜索

c. 应⽤内部监控

d. 图表展示

e. 监控指标报警

2. ⾮功能性需求

a. ⾼性能：⽀持TB级别监控⽇志收集

b. ⾼可⽤：整个系统⽆核⼼单点

c. 强伸缩：能根据公司规模灵活调整

架构⽅案描述：

整套架构在ELK基础上进⾏扩展升级，基主要分为⽇志信息采集、传输、存储与展现四部

分。

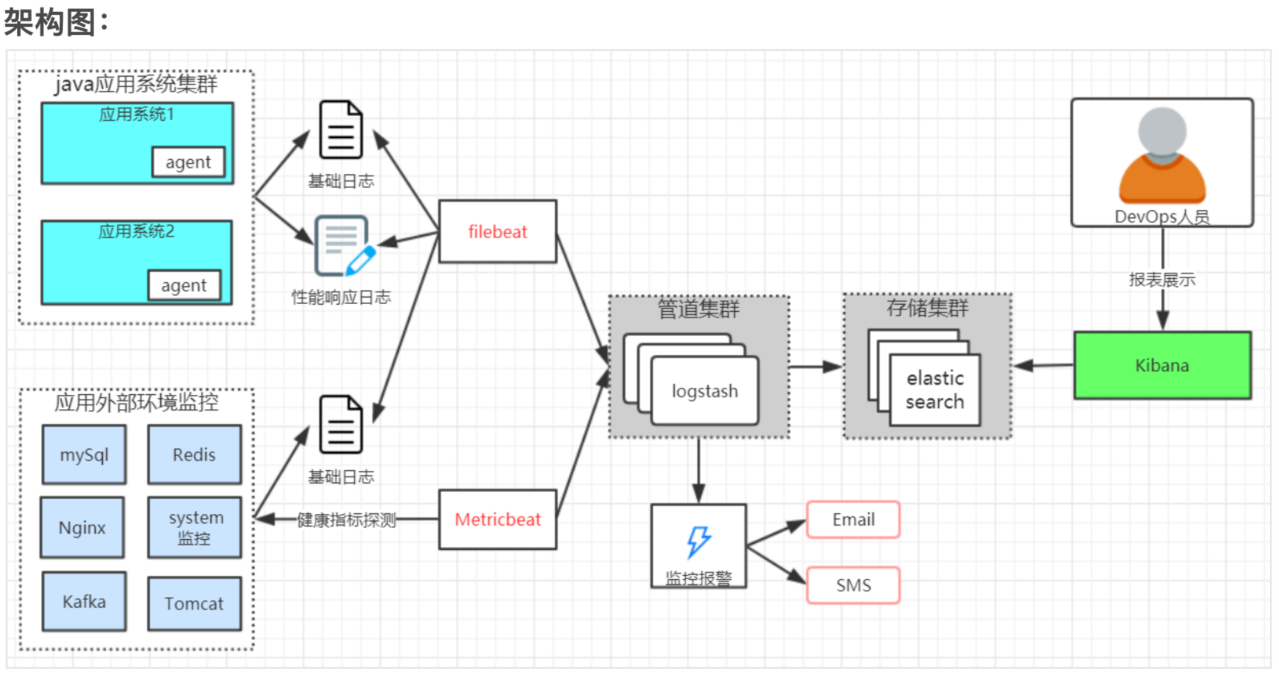
采集：javaagent 采集应⽤内部性能指标并输出⾄⽇志、FileBeat采集⽂本⽇志、

Metricbeat采集系统性能指标。

传输：beats、logstash、kafka

存储：Elasticsearch

展现：Kibana



应⽤内部性能采集说明： 基于字节码插桩技术监控应⽤内部性能指标如：Http响应时间、服务响应、异常、SQL 语句、外部接⼝响应。该⽅式特点是：即可以精准监控应⽤内部，⼜不会对应⽤造成侵⼊从 ⽽简化上线推⼴成本。

注：字节码插桩技术指使⽤javaagent与 javassist动态重构Class字节已插⼊监控指令。

应⽤外部环境采集说明： 基于metricbeat 主动采集系统信息如： CPU 使⽤率、内存、⽂件系统、磁盘 IO 和⽹络 IO 统计数据，以及获得如同系统上 top 命令类似的各个进程的统计数据。此外基于 metricbeat 中提供的插件去主动采集 mySql、Redis、Nginx等状态信息。

监控报警说明： 如果报警条件⽐较简单，可以直接基于logstash当中logstash-output-http 插件输送数 据流⾄⾃实现报警中⼼实现报警。

如果报警条件复杂则可基于logstash-output-zabbix输送 数据流⾄ zabbix系统触发报警。

问题1：⽇志太多，磁盘存储不够⽤怎么办？ 1、设定⽇志过期时间，写脚本定期清理Elasticsearch 当中过期数据。 2、在logstash中设置过滤器，过滤debug 和info ⽇志。

问题2：logstash 和Elasticsearch 是分开部署的么，不想要这么复杂的架构怎么办？ 可以直接去掉logstash ，数据直接从beats 发送⾄logstash

问题3：⽇志信息吞吐量太⼤怎么办？ 这种情况只能在整个传输管道中间加⼊⼀层 Broker作缓冲Redis或Kafak。当然这样做将 会使整个系统变得更复杂，众多节点⾃身的健康状况维护也变得困难。

具体看下图。

