数据驱动的智能运维

饶琛琳@日志易

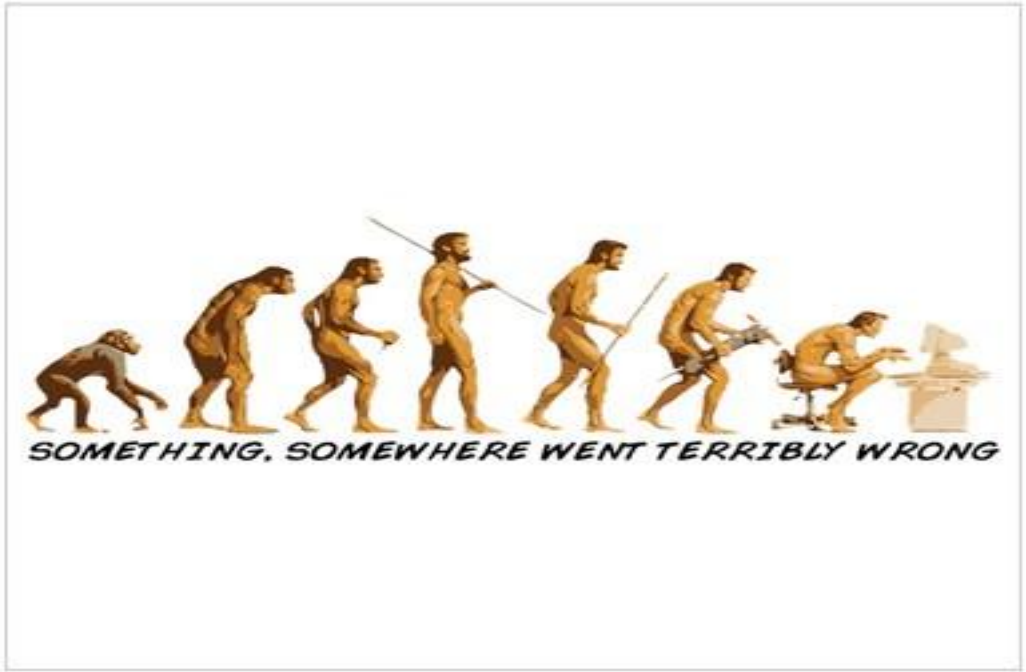
日志易 rizhiyi.com

个人简介

- 日志易产品总监
- 前·新浪微博系统架构师
- Weibo: @ARGV

- 著有：
- 《网站运维技术与实践》
- 《ELKstack权威指南》
- 译有：
- 《Puppet 3 Cookbook》
- 《Learning Puppet 4》

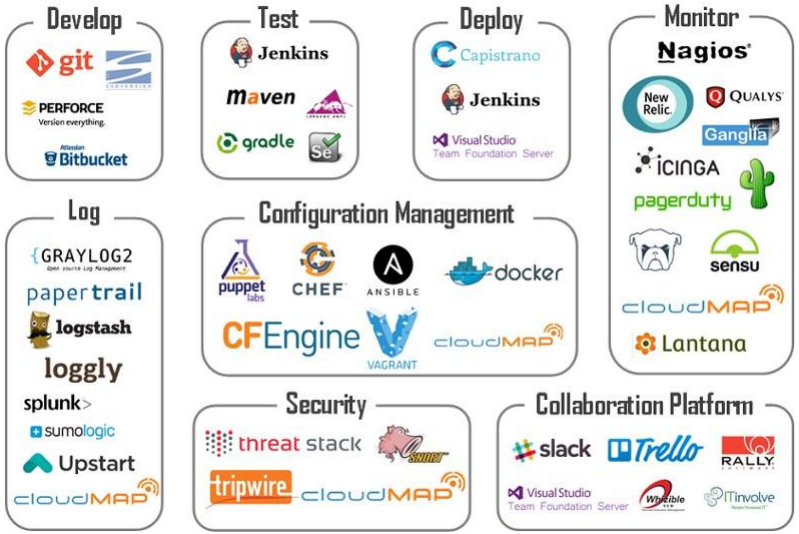
SA, SE, OP, DevOps,SRE?



感谢Dave O'Connor："sort of Devops is to SRE as HBase is to BigTable." https://www.quora.com/Is-SRE-Googles-version-of-DevOps

所以，你如何 DevOps-ing?

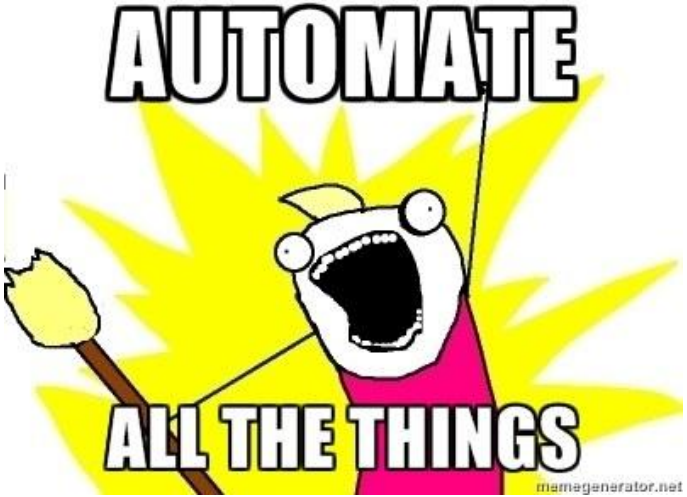
- 用这些DevOps-ready 工具?



日志易 rizhiyi.com

所以，你如何 DevOps-ing?

- 还是自动化、自动化、自动化？



70% 的微博故障原因是变更操作！你呢？

日志易 rizhiyi.com

数据驱动的运维操作

- 我们需要充分的数据来证明，下一步变更是有必要的，安全的。
- 现在，请出示你的：
  - alerts
  - reports
  - metrics
  - exceptions
  - …

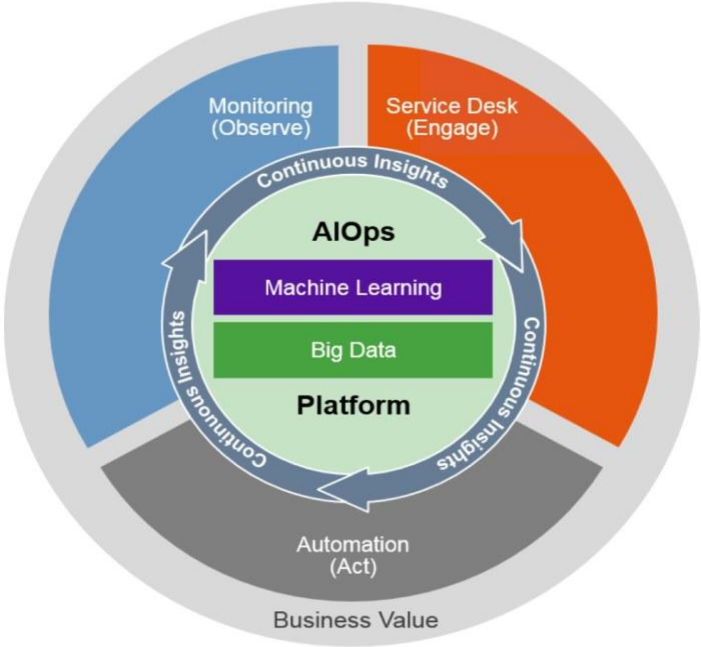
日志易 rizhiyi.com

数据驱动的运维操作

- 监控软件大多是采样式的。
- 采样意味着监控的评定是一种模糊估算，是去除了细节的大趋势上的判断。
- 在通过监控做到了总体稳定的初级目标以后，有必要通过全量数据分析的方式，对细节做更 明确、更高效的诊断和优化。
- 随着技术的发展，大数据的兴起，靠数据来驱动运维，也成为可能。

日志易 rizhiyi.com

如何驱动？



- Gartner 2016.04:
- 2019年，全球有25%的企业将搭建好自己的AIOps平台，而这个数字目前是不到5%。

日志易 rizhiyi.com

What's inside AIOps?

- 三大作用：
- 更灵活、更易用的访问和分析数据；
- 能分析过去散落在各组件中未利用上的业务数据和上下文；
- 快速的探索和实验平台，提供独特的洞擦力

日志易 rizhiyi.com

What's inside AIOps?

- 『随着日志文件的容量和类型的增长，对日志数据进行分析、追踪潜在的问题、发现错误变 得越来越难，尤其是在多日志相关性分析出现之后。即便在最佳状态下，也需要经验丰富的 操作人员跟踪事件链、过滤噪音，并最终诊断出导致复杂问题产生的根本原因』
- -- 埃森哲咨询

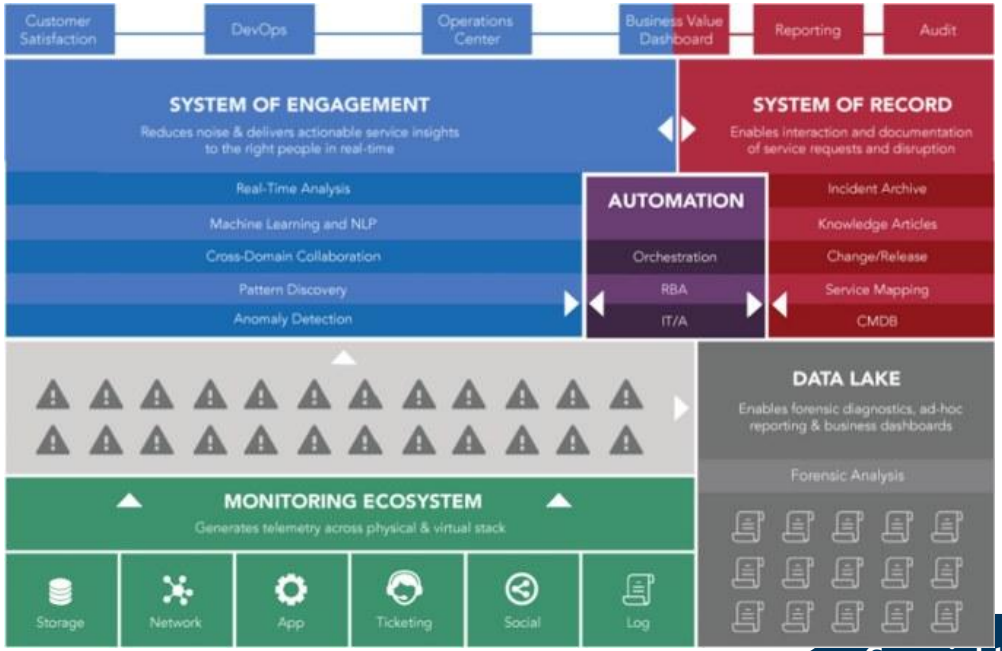
日志易 rizhiyi.com

What's inside AIOps?

- 两个方向：
- 大数据和机器学习技术，实现以数据为中心的可用性和性能分析；
- 将以数据为中心的的方法扩展到其他ITOA学科，比如SIEM和业务分析。

What's inside AIOps?

- 从『系统组成』看AIOps架构：数据湖、自动化系统、记录系统、交互系统和监控生态圈



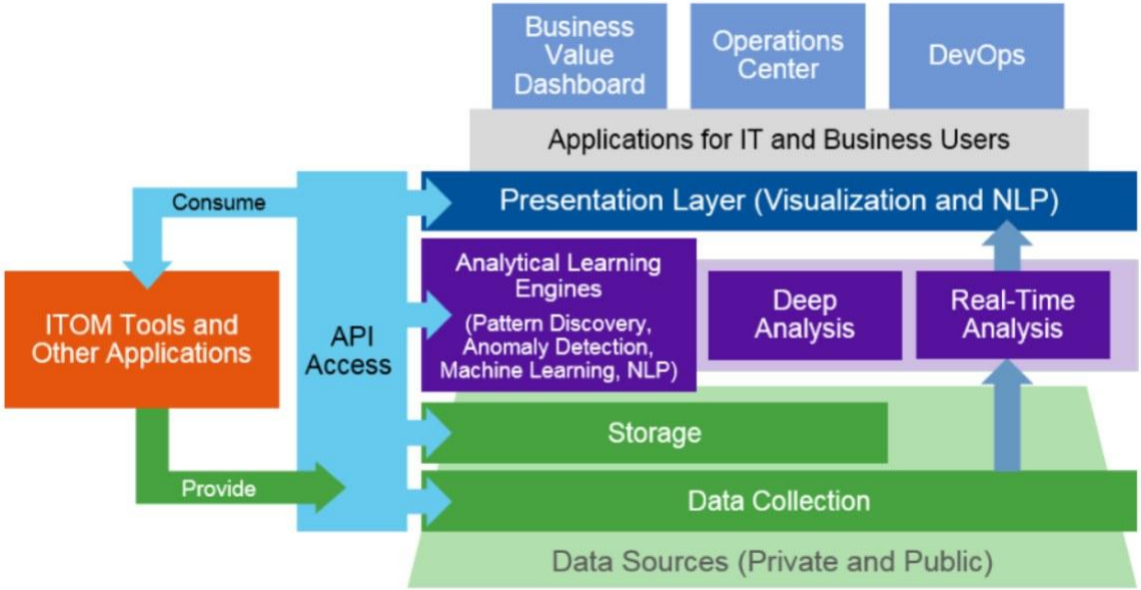
日志易 rizhiyi.com

What's inside AIOps?

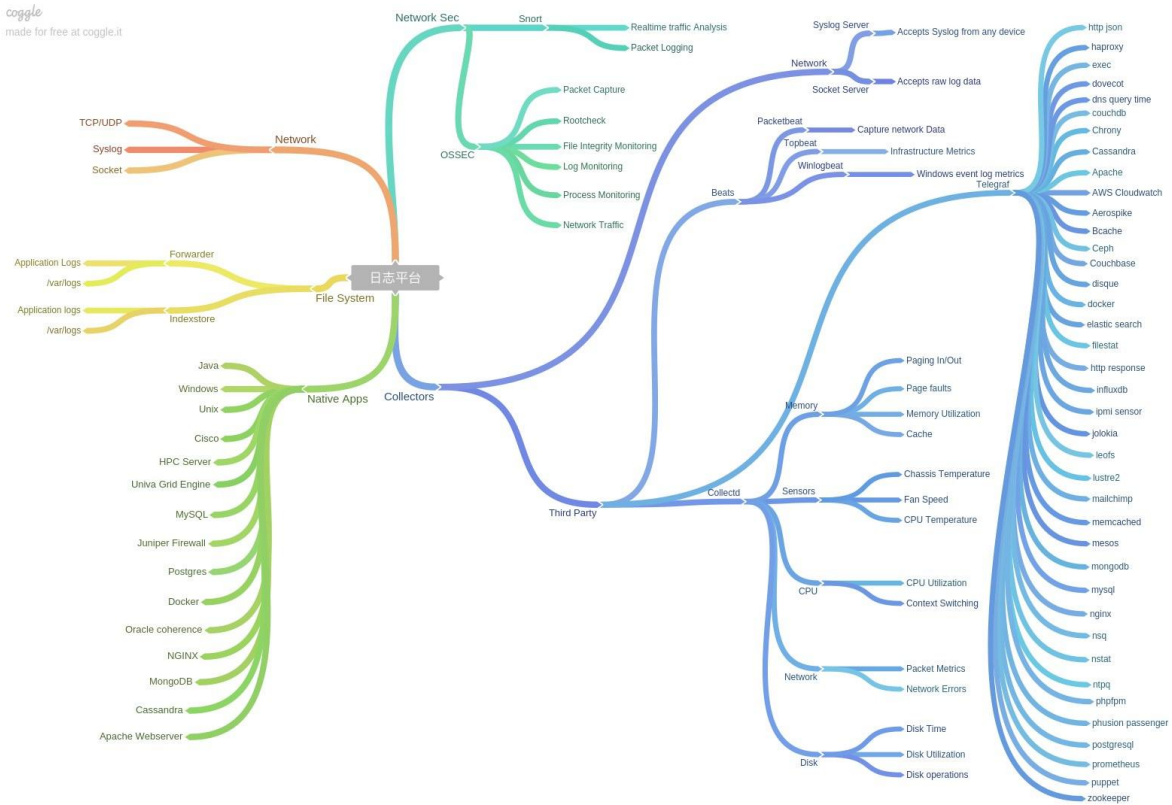
- 监控系统: 硬件和虚拟平台的检测，管理服务质量；(StatsD, CollectD)
- 记录系统: 问题记录和知识库积累，并与CMDB关联；(Jira, GitLab)
- 自动化系统: 自动执行固化事件的解决脚本；(Puppet, Saltstack, Ansible)
- 交互系统: 降噪和实时分发信息到真正负责的人，以及一些早期检测和修复；(Nagios, Zabbix, Zenoss)
- 数据湖: 诊断、即时图表和仪表盘。保存你所有可能会用到的日志，用于深度分析

What's inside AIOps?

- 从『数据流向』看AIOps架构：采集、存储、分析、可视化。

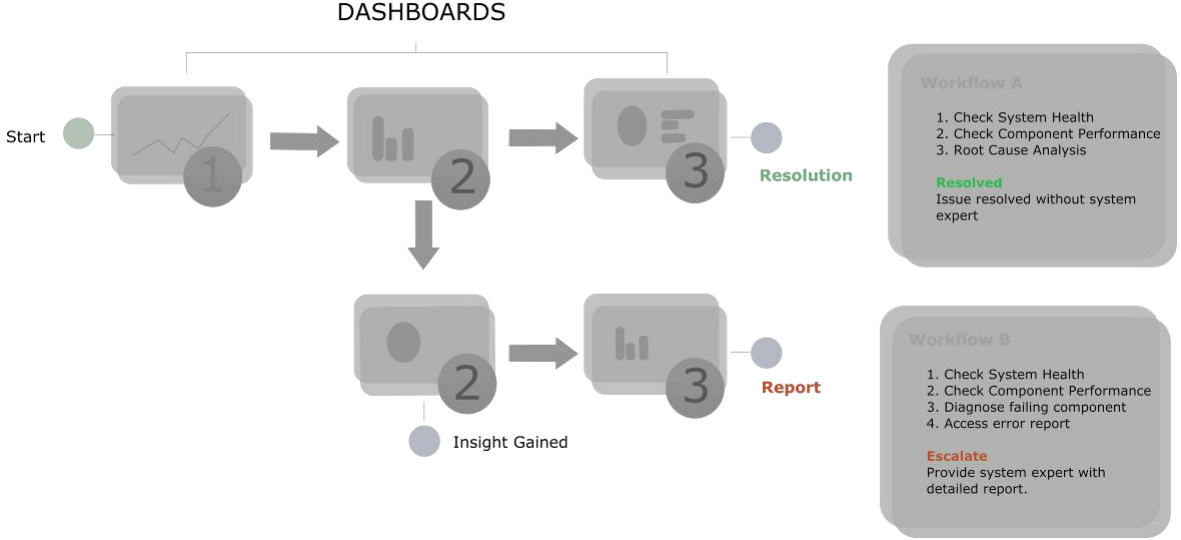


数据接入


coggle
made for free at coggle.it
日志平台
Collectors
Network
TCP/UDP
Syslog
Socket
File System
Forwarder
Application Logs
/var/logs
Indexstore
Application logs
/var/logs
Native Apps
Java
Windows
Unix
Cisco
HPC Server
Univa Grid Engine
MySQL
Juniper Firewall
Postgres
Docker
Oracle coherence
NGINX
MongoDB
Cassandra
Apache Webserver
Network Sec
Snort
Realtime traffic Analysis
Packet Logging
OSSEC
Packet Capture
Rootcheck
File Integrity Monitoring
Log Monitoring
Process Monitoring
Network Traffic
Third Party
Beats
Packetbeat
Capture network Data
Topbeat
Infrastructure Metrics
Winlogbeat
Windows event log metrics
Network
Syslog Server
Accepts Syslog from any device
Socket Server
Accepts raw log data
Collectd
Memory
Paging In/Out
Page faults
Memory Utilization
Cache
Sensors
Chassis Temperature
Fan Speed
CPU Temperature
CPU
CPU Utilization
Context Switching
Network
Packet Metrics
Network Errors
Disk
Disk Time
Disk Utilization
Disk operations
Telegraf
http json
haproxy
exec
dovecot
dns query time
couchdb
Chrony
Cassandra
Apache
AWS Cloudwatch
Aerospike
Bcache
Ceph
Couchbase
disque
docker
elastic search
filestat
http response
influxdb
ipmi sensor
jolokia
leofs
lustre2
mailchimp
memcached
mesos
mongodb
mysql
nginx
nsq
nstat
ntpq
phpfpm
phusion passenger
postgresql
prometheus
puppet
zookeeper

动态的知识积累

- 仪表盘动态钻取流程设计，帮助无场景知识积累的人快速了解事件根源。



日志易 rizhiyi.com

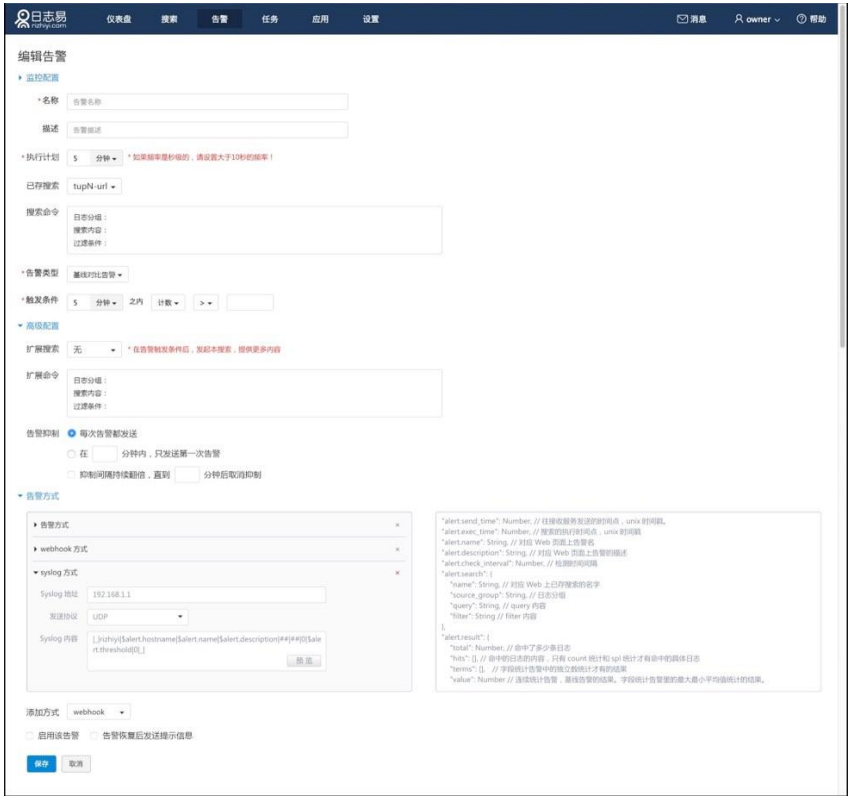
告警交互系统

- "Alertmanager ... takes care of ***deduplicating, grouping,*** and ***routing*** them to the correct receiver integrations such as email, PagerDuty, or OpsGenie. It also takes care of ***silencing*** and ***inhibition*** of alerts."
- -- prometheus.io

日志易 rizhiyi.com

告警交互系统

- 扩展搜索
- 风暴静默
- 模板渲染



日志易 rizhiyi.com

告警交互系统

- 有一个spl告警，告警名称为『单域名平均响应时间大于2s』，搜索条件是： * | bucket timestamp span=1h as ts | stats avg(request_time) as avg_ by ***domain***, ts。触发条件为 avg_ 大于 2。
- 这时候我想着如果能在告警出来的时候，顺带把request_time:>10 AND upstream_resp_time:<1的部分日志也附带上，可能会更有助于我们判断故障。
- 我们可以把这个搜索也保存下来，然后在『扩展搜索』上选择这个搜索。然后填写这样的告 警邮件内容模板：

日志易 rizhiyi.com

邮件内容的django模板

- 告警名称：{{ alert.name }}
- 触发条件：
{{ alert.strategy.trigger.compare_desc_text }}
- 可能导致本次服务异常的可疑访问如下：
- {% for hit in alert.result.extend_hits %}
- {{ hit.raw_message }}
- {% endfor %}
- 建议您优先排查这部分。

日志易 rizhiyi.com

邮件内容

- 告警名称：单域名平均响应时间大于2s
- 触发条件：avg_的值大于2
- 可能导致本次服务异常的可疑访问如下：
- 219.134.34.124 - - [15/Jun/2016:14:21:06.588 +0800] "GET /index/login/ HTTP/1.1" 200 1938 "-" "-" "-" 13.21 0.030
- 219.134.34.124 - - [15/Jun/2016:14:21:06.588 +0800] "GET /index.jsp HTTP/1.1" 200 10326 "-" "-" "-" 10.22 0.301
- 219.134.34.124 - - [15/Jun/2016:14:21:06.588 +0800] "GET /index/login/ HTTP/1.1" 200 1938 "-" "-" "-" 14.20 0.103
- 建议您优先排查这部分。

日志易 rizhiyi.com

智能运维平台实现要点

- 非结构化数据的处理
- 多模块关联追踪分析
- 动态阈值的异常检测
- 平台服务的资源管控

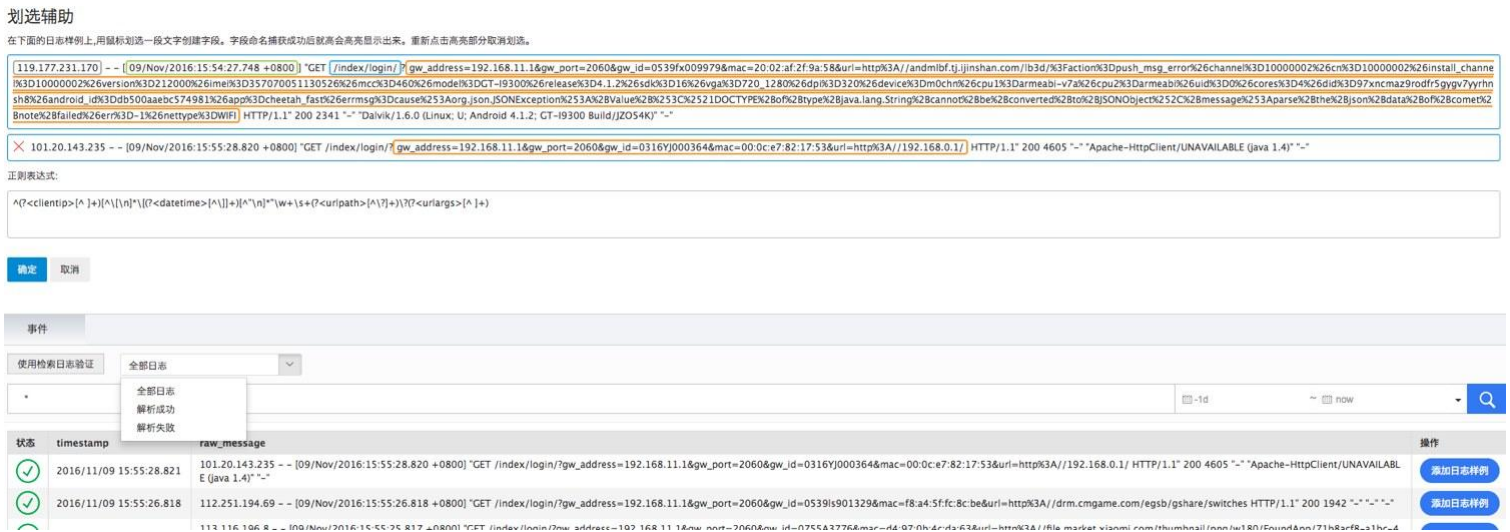
非结构化数据的处理

- 平台运维≠业务模块开发。90%的日志不会结构化

日志易 rizhiyi.com

非结构化数据的处理

- 鼠标拖拽命名，通过机器学习算法自动生成正则

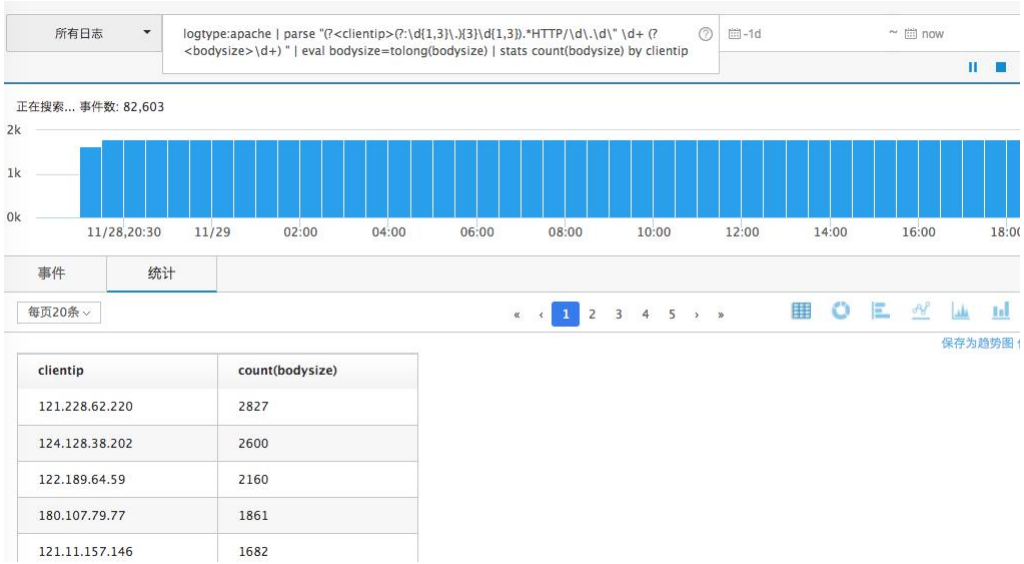


非结构化数据的处理

- 古典方案：
- Hadoop的MapReduce离线批处理
- 现代方案：
- ELK的预先处理，将非结构化数据转变为半自由的结构化数据
- 后现代方案：
- 在搜索运行时，对非结构化数据做临时性的必要结构化处理统计

非结构化数据的处理

- 查询时字段的提取和统计



clientip	count(bodysize)
121.228.62.220	2827
124.128.38.202	2600
122.189.64.59	2160
180.107.79.77	1861
121.11.157.146	1682

日志易 rizhiyi.com

非结构化数据的处理

- 矛盾：
- 刚刚用算法避免了用户学正则，转身又还是要用户在搜索的时候手写正则来提取了？

- 解决办法：
- 通过算法生成的正则，自动运用到用户搜索的数据上？

- 难点：
- 正则的质量无法保证。
- 性能性能性能！重说三……

日志易 rizhiyi.com

多模块下的关联分析

- 幸运的人：
- 统一框架下，依赖于基础库的改造，实现比较方便的全局唯一ID关联；
- 不幸的人：
- 别说多模块之间的调用关系，连自己调用的模块谁写的都不清楚。谁给你加ID？

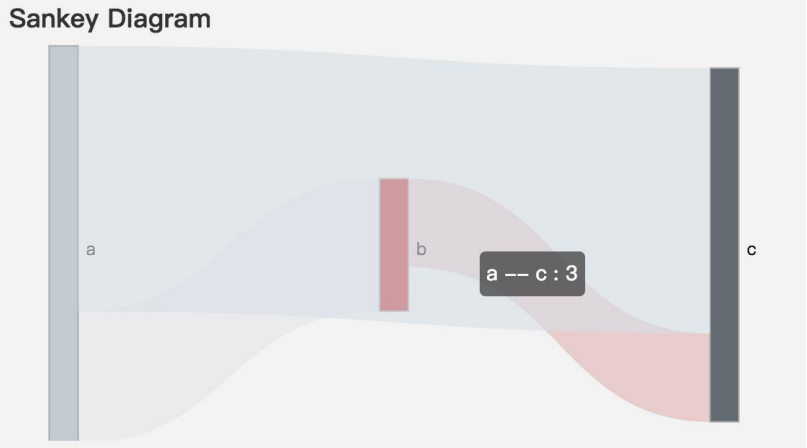
日志易 rizhiyi.com

多模块下的关联分析

{"timestamp":1491985634000, "sid":1, "module":"a"}
{"timestamp":1491985634000, "sid":2, "module":"a"}
{"timestamp":1491985634002, "sid":1, "module":"b"}
{"timestamp":1491985634003, "sid":1, "module":"c"}
{"timestamp":1491985634003, "sid":2, "module":"c"}
{"timestamp":1491985634004, "sid":3, "module":"a"}
{"timestamp":1491985634005, "sid":3, "module":"b"}

- 通过聚类模式，将时序数据经过多ID串联，找到复杂拓扑下的请求链关联。

多模块关联分析可视化



通过如下spl统计：

```
* | transaction sid with states a,b,c in module results by flow
```

可以得到如下统计结果：

fromstate	tostate	_duration
a	b	2
a	c	3
b	c	1
a	b	1

然后再加一层统计：

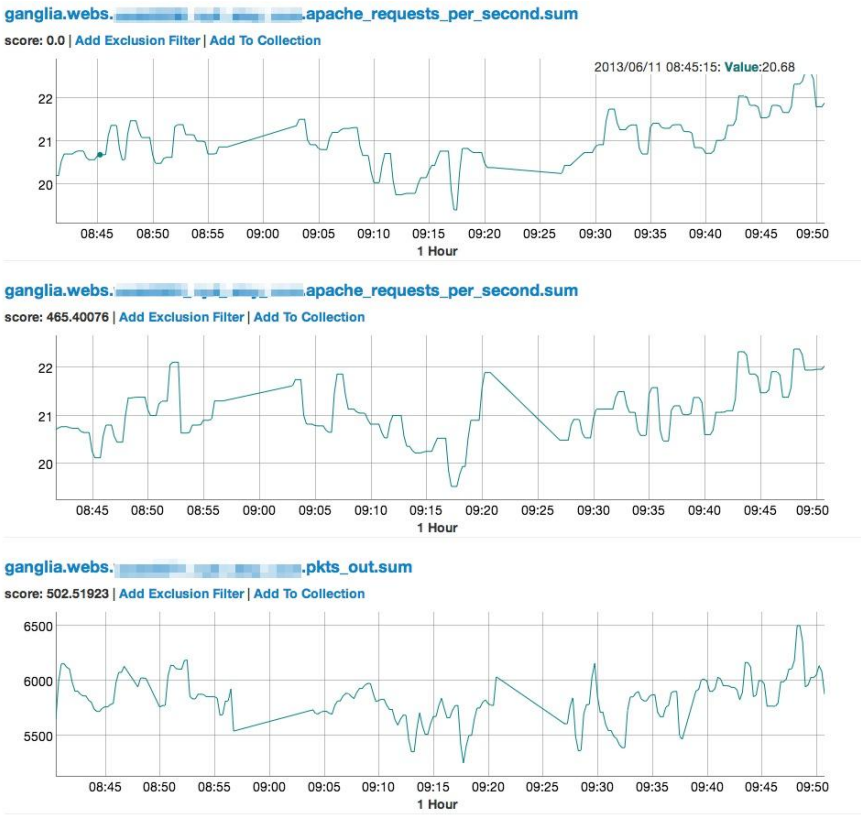
```
* | transaction sid with states a,b,c in module results by flow | stats count(), avg(_duration) by fromstate, tostate
```

则可以得到最终统计结果：

fromstate	tostate	count()	avg(_duration)
a	b	2	1.5
a	c	1	3
b	c	1	1

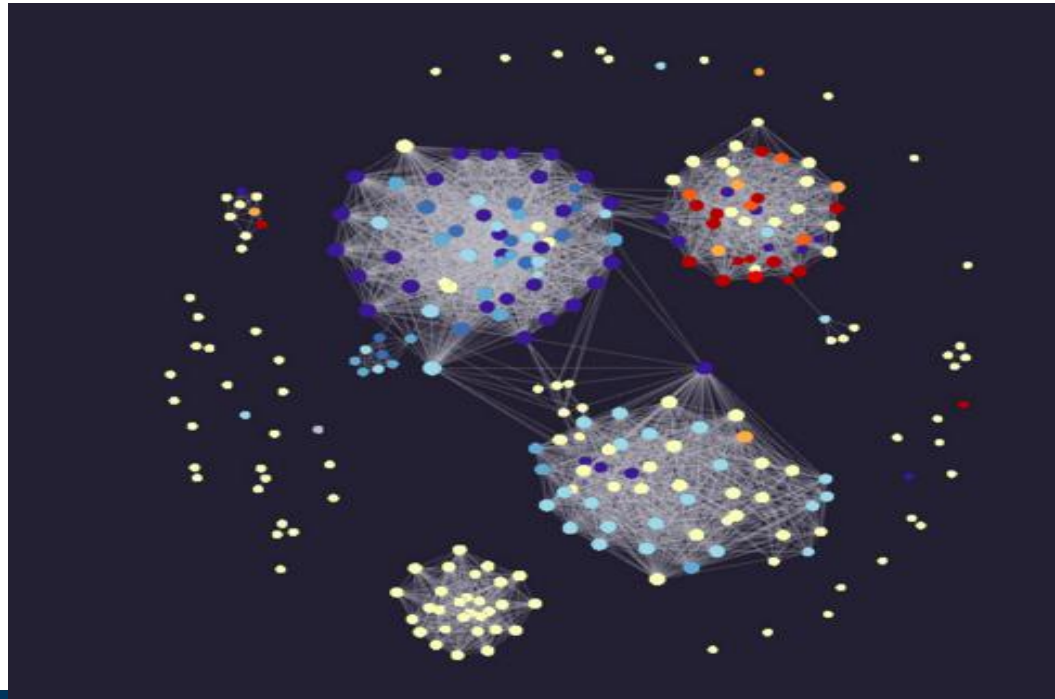
基于聚类算法的RCA

- 对不同系统的相关性，
- 采用距离算法做聚类。
- Etsy的Oculus系统：
  - 欧氏距离
  - FastDTW



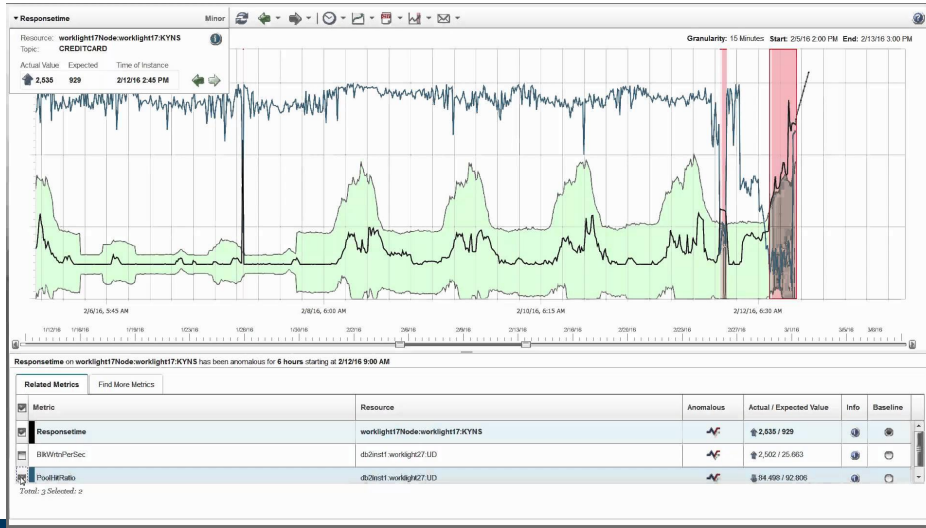
基于聚类算法的RCA

- Uber的argos系统。
- nodes之间的weight决定了重要性。



基于聚类算法的RCA

- IBM的predictive insight
  - granger cause
- If past values of A and B can predict future value of B better than past values of B alone,Then, time series A granger cause time series B



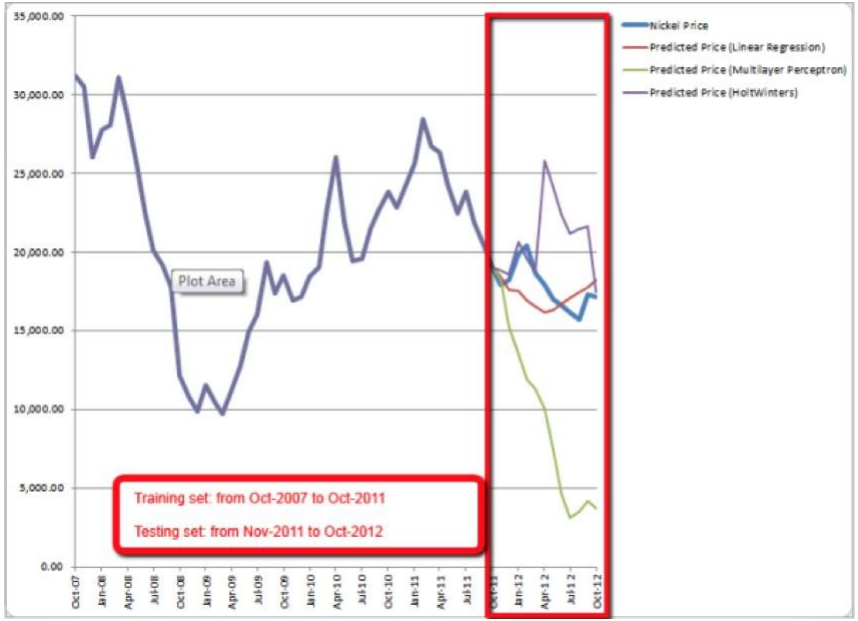
日志易 rizhiyi.com

异常检测

- 到底什么叫异常检测：
  - rare统计？
  - 同比环比？
  - Box？
  - Histogram？
- 算法驱动的异常检测：
  - 时序预测：趋势？季节？多样本校验？
  - 多元预测：降维？

异常检测

- 不同算法的预测表现：
  - 指数平滑
  - 多层感知
  - 线性回归



日志易 rizhiyi.com

时序预测的开源选择

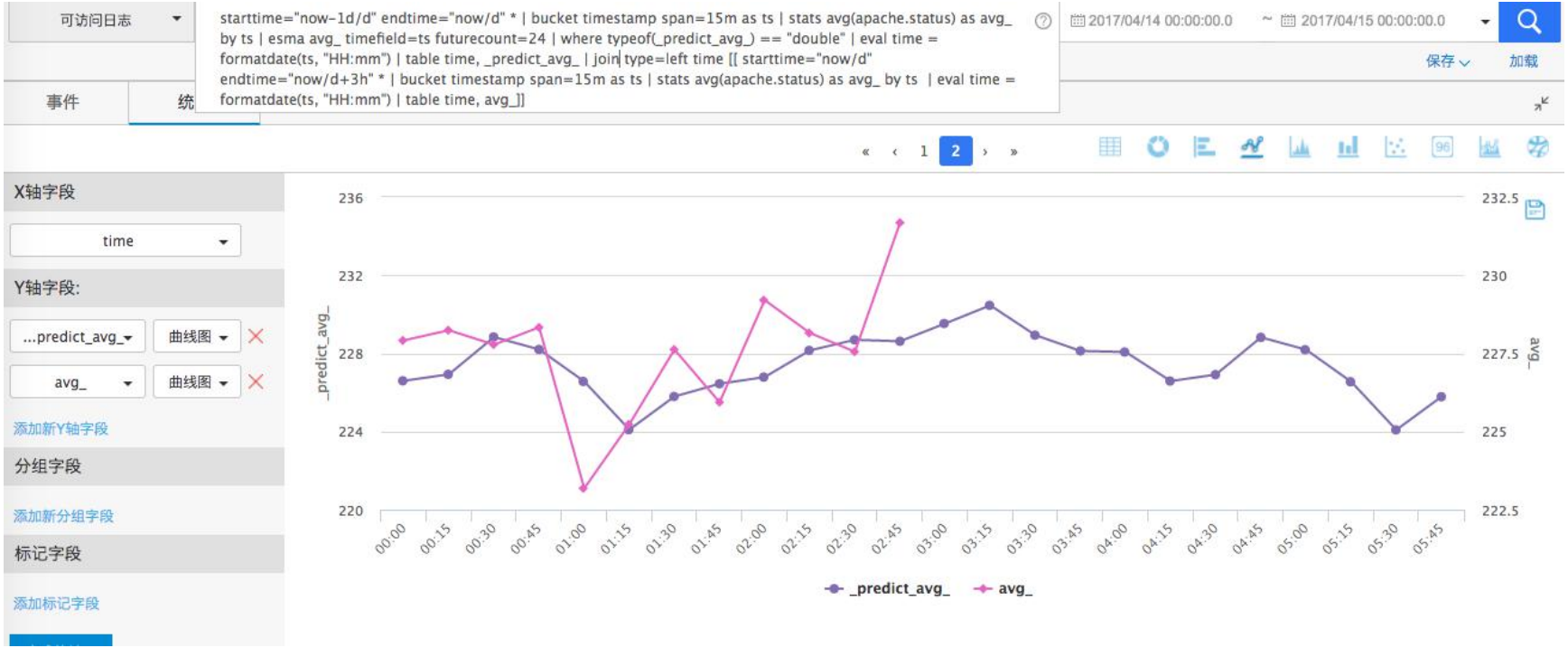
- Facebook在今年2月刚开源的Prophet库。(R/Python)
- Yahoo!在去年开源的egads库。(Java)
- Twitter在去年开源的anomalydetection库。(R)
- Netflix在2015年开源的Surus库。(Pig，基于PCA)
- Etsy在2013年开源的skyline库。(python)
- Numenta在2013年开源的NuPIC库。(python，基于HTM)
- RRDtool在1997年实现的HWPREDICT。(C，基于holt-winters)
- 。。。

日志易 rizhiyi.com

IT环境下的时序预测

- 一个稳定的IT环境中，时序数据通常具有趋势性，甚至季节性。
  - Simple exponential smoothing
  - Double exponential smoothing (Holt’s linear trend)
  - Seasonal triple exponential smoothing (Holt Winters)
- 人工调节α，β，γ三个参数，工作量太大。
  - best model select（Akaike information criterion）
  - best smoothing parameter optimize（通过Nelder-Mead simplex非线性优化算法，获得最小的MSE）

IT环境下的时序预测



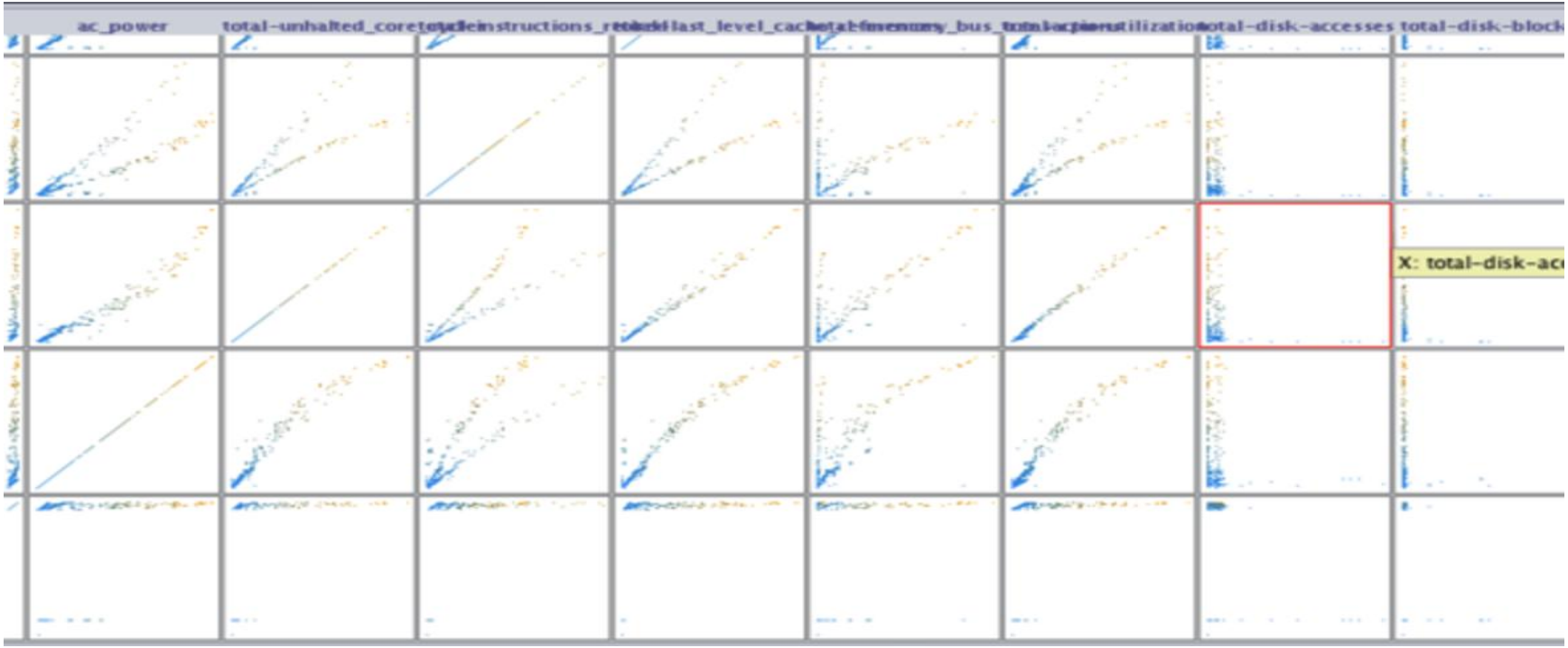
日志易 rizhiyi.com

多元预测示例

- 服务器电力未来是否需要扩容？单纯靠电力自己的趋势意义不大。
- 尽量收集更多的服务器指标，尝试做多元预测。

1. 完整性校验
2. 降维
3. 多元预测

PCA降维



多种算法的预测效果对比

算法	RAE	耗时(s)
kNN	7.1043%	31.65
线性回归	19.2408%	0.38
MLP神经网络	9.7343%	0.39
M5P树回归	6.4732%	0.42
随机森林	6.2794%	1.65

日志易 rizhiyi.com

平台服务的资源管控

- 平台服务的『多租户』特性，也是AIOps用『大数据』统一管理多个隔离的『小数据』的办 法。

- 资源：
  - 逻辑资源，包括用户所能读写的数据、告警、报表资源管理；
  - 物理资源，包括用户所能利用的CPU、IO、MEM资源管理。

- 听起来像是Docker或者k8s的关键词？
- 日志分析系统既有海量数据不便切分迁移的难点，又有资源隔离控制的需求。

日志易 rizhiyi.com

搜索的任务管理

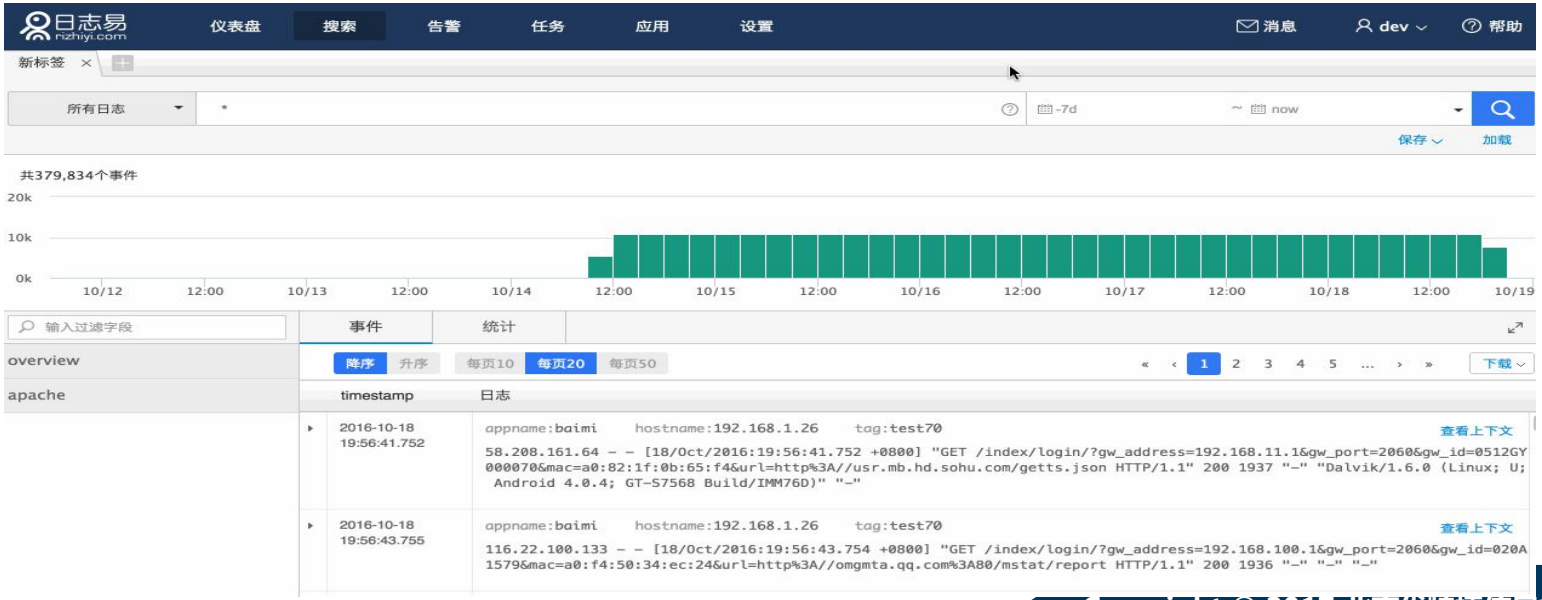
- 从纯粹的实时分布式，变成任务式。
- 任务管理的几个思路：
  1. 对超限的kill掉。
  2. Cgroup类资源限制。
  3. CPU时间分片调度。
  4. 基于IaaS的集群切分。

不同思路的优缺点

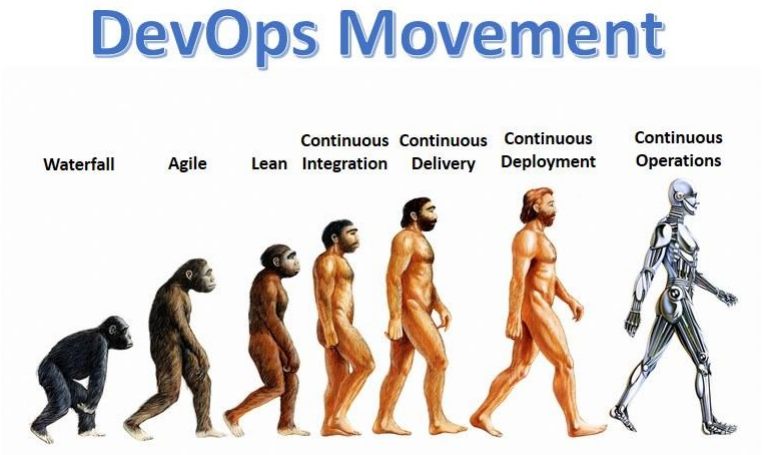
思路	优点	缺点
kill	较简单	正常的大任务永远都执行不了
cgroup	可以利用OS特性	所有任务都更慢
时间片	尽可能充分利用资源	开发量较大
IaaS	都有现成方案	跨集群无法关联，管理复杂度大

平台服务的资源管控

- 公平调度下的任务执行引擎。
- 未来可以根据多用户的角色扩展更多的调度算法。权重、并发等等。



Let's move to AIOps.



日志易 rizhiyi.com

欢迎加入！

- DevOps
- Machine Learning
- Search Engine

飞马网
WWW.FMI.COM.CN

中生代技术
FRESHMAN TECHNOLOGY

优云 | IT大咖说 知识分享平台

演讲完毕，谢谢大家！