

# 一些好用的开源监控工具汇总

高效开发运维 InfoQ 3天前



编辑 | 张婵 - 高效开发运维

监控系统是整个 IT 架构中的重中之重，小到故障排查、问题定位，大到业务预测、运营管理，都离不开监控系统，可以说一个稳定、健康的 IT 架构中必然会有一个人可信赖的监控系统。

但是，难道监控就只是监控？多年来，对于监控的术语一直都有很多困惑，一些很糟糕的工具也宣称能够以一种格式完成所有事情。

在 DevOps 和云原生时代，今年，“可观察性”（Observability）被引入到了 IT 领域，其首先表现为 CNCF-Landscape 出现了 Observability 分组。从该分组的内容看包含了监控，日志，Tracing 等领域的项目。可观察性与监控有什么不同？简单说来，后者是前者的一个子集。监控关注系统的失败因素，从而定义出系统的失败模型。它的核心是运维，是监控设施。而可观察性除了关注失败之外，其核心是研发，是应用，是对系统的一种自我审视。是站在创造者的角度去探究系统应如何恰当的展现自身状态。一个由外向内，一个由内向外。

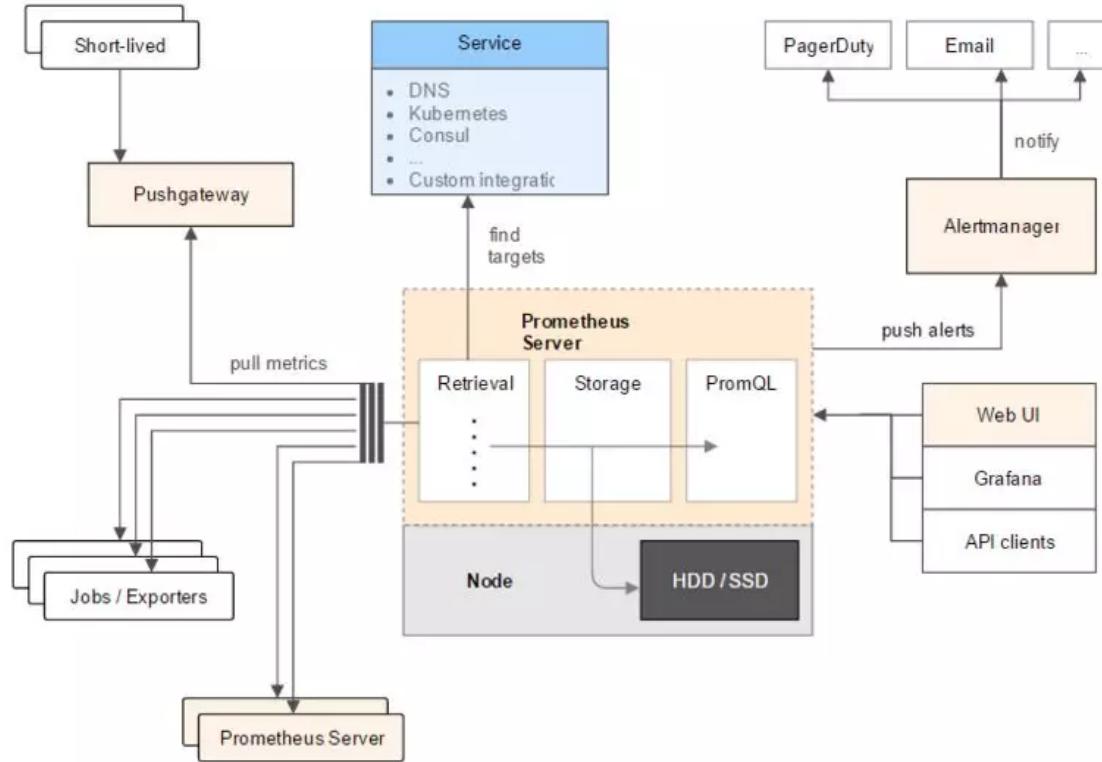
观察工具包括：度量聚合（Metrics aggregation）（主要是时序数据），日志聚合（Log aggregation），告警 / 可视化（Alerting/visualizations），分布式追踪（Distributed tracing）。

## 监控工具

## Prometheus

Prometheus 是云原生应用程序最受认可的时间序列监控解决方案，由 CNCF 托管，使用 Go 语言开发，是 Google BorgMon 监控系统的类似实现。

Prometheus 使用的是 Pull 模型，Prometheus Server 通过 HTTP 的 pull 方式到各个目标拉取监控数据。它使用局部配置来描述要收集的端点和收集所需的间隔。每个端点都有一个客户端收集数据并在每次请求时更新该表示（或者客户端是配置的）。收集此数据并将其保存在本地磁盘上的高效存储引擎中。存储系统使用每个度量标准的仅附加文件。



Prometheus 包含一种高级表达式语言，用于选择和显示名为 PromQL 的数据。此数据可以通过 REST API 以图形或表格显示或由外部系统使用。表达式语言允许用户创建回归，分析实时数据或趋势历史数据。标签也是用于过滤和查询数据的绝佳工具。标签可以与每个度量标准名称相关联。

Prometheus 附带 AlertManager 来处理警报。AlertManager 允许进行警报聚合以及更复杂的流量以限制发送警报的时间。假设在开关关闭的同时 10 个节点突然出问题，你可能不需要发送有关这 10 个节点的告警，因为接到报警的每个人在开关修好之前可能无法执行任何操作。使用 AlertManager，可以仅向网络团队发送有关开关告警，并在其中包含其他可能受影响系统的信息；也可以向系统团队发送电子邮件。

件（而不是页面），以便他们知道这些节点已关闭，除非系统在开关修复后没有恢复，否则他们不需要响应。如果发生这种情况，则 AlertManager 将重新激活那些被开关警报抑制的警报。

## Graphite

Graphite 是一款用 Python 写的开源企业级监控绘图工具，可以在廉价机硬件上运行。Graphite 可以实时收集、存储、显示时间序列类型的数据，它由三个软件组件组成：

- carbon - 基于 Twisted 的进程，用于监听并接收数据；
- whisper - 专门存储时序数据的小型数据库，在设计上类似于 RRD；
- graphite webapp - 基于 Django 的网页应用程序，可以从 whisper 数据库获取时间序列数据并且进行展示。



Graphite 架构图

Graphite 是一个基于推送的系统，通过让应用程序推送数据到 Graphite 的 Carbon 组件中，从应用程序接收数据。Carbon 将此数据存储在 Whisper 数据库中，Graphite Web 组件读取 Carbon 它的和数据库，允许用户在浏览器中绘制数据图或通过 API 提取数据。一个非常酷的功能是能够将这些图形导出为图像或数据文件，以便将它们轻松嵌入到其他应用程序中。

Graphite 的另一个有趣功能是能够存储与时序指标相关的任意事件。可以在 Graphite 中添加和跟踪应用程序或基础架构部署，这允许运维人员或开发人员对问题进行故障排除，能获得正在调查的异常行为环境中更多的背景信息。

Graphite 监控上手指南 : <http://www.infoq.com/cn/articles/graphite-intro>

## InfluxDB

---

InfluxDB 是一个相对较新的时序数据库，使用 Go 语言编写，无需外部依赖，安装配置非常方便，适合构建大型分布式系统的监控系统。

其设计目标是实现分布式和水平伸缩扩展。

InfluxDB 的一些主要特征：

- 无结构 (无模式)：可以是任意数量的列
- 可以设置 metric 的保存时间
- 支持与时间有关的相关函数 (如 min、max、sum、count、mean、median 等)，方便统计
- 支持存储策略：可以用于数据的删改。(influxDB 没有提供数据的删除与修改方法)
- 支持连续查询：是数据库中自动定时启动的一组语句，和存储策略搭配可以降低 InfluxDB 的系统占用量。
- 原生的 HTTP 支持，内置 HTTP API
- 支持类似 sql 语法
- 支持设置数据在集群中的副本数

- 支持定期采样数据，写入另外的 measurement，方便分粒度存储数据
- 自带 web 管理界面，方便使用 (登入方式：<http://< InfluxDB-IP >:8083>)

## OpenTSDB

---

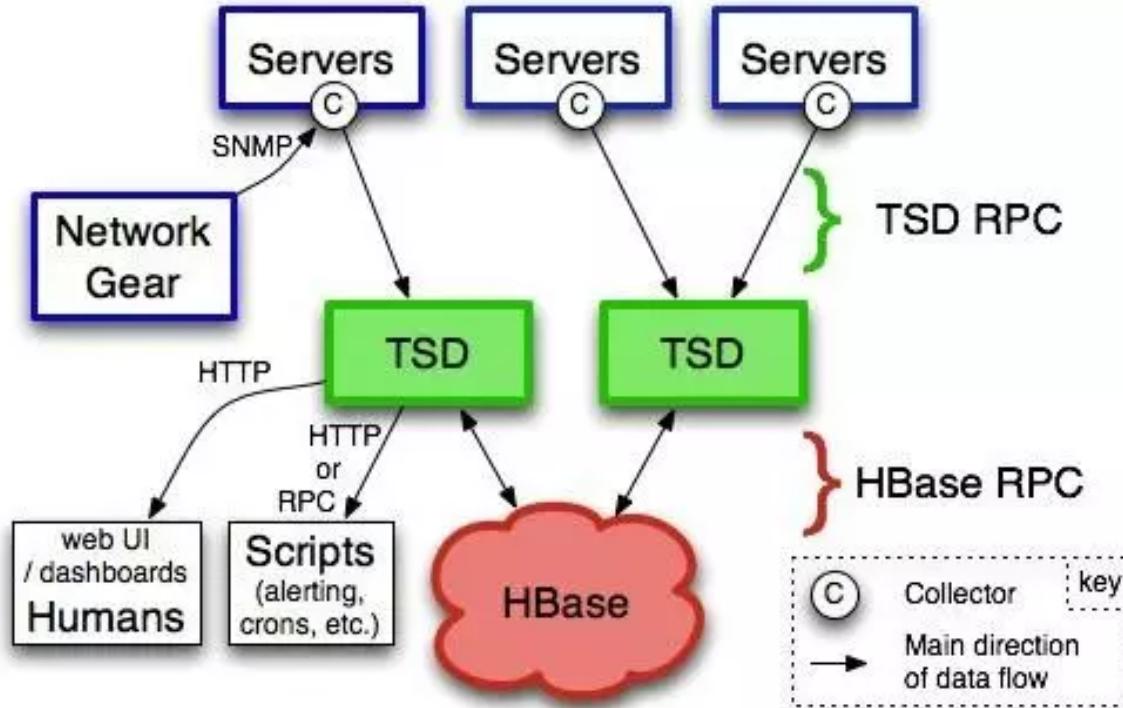
OpenTSDB 是基于 Hbase 的分布式的，可伸缩的时序数据库，确切地说，它只是一个 HBase 的应用。OpenTSDB 主要用途就是做监控系统；譬如收集大规模集群（包括网络设备、操作系统、应用程序、环境状态）的监控数据并进行存储，查询。

OpenTSDB 可以动态的增加 metrics，灵活支持任何语言的收集器，极大的方便了运维人员，降低了开发和维护成本。

存储到 OpenTSDB 的数据，是以 metric 为单位的，metric 就是一个监控项，譬如服务器的话，会有 CPU 使用率、内存使用率这些 metric；

OpenTSDB 使用 HBase 作为存储，由于有良好的设计，因此对 metric 的数据存储支持到秒级别；

OpenTSDB 支持数据永久存储，即保存的数据不会主动删除；并且原始数据会一直保存（有些监控系统会将较久之前的数据聚合之后保存）



## 日志聚合工具

一些日志记录规则：

- 包括时间戳
- 格式为 JSON
- 请勿记录无意义的事件
- 请记录所有应用程序错误

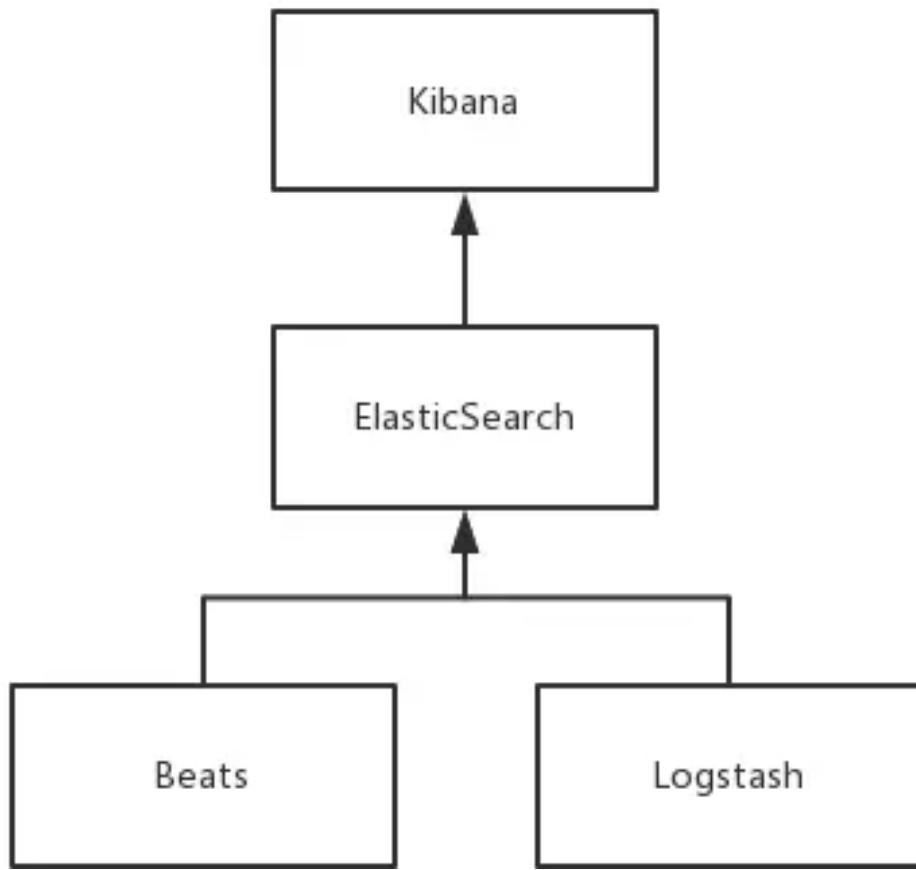
- 可以有日志警告
- 打开日志记录
- 以可读的形式写消息
- 请勿在生产中记录信息数据
- 请勿记录无法阅读或反应的任何内容

## ELK

ELK 是 Elasticsearch , Logstash 和 Kibana 的缩写，在实时数据检索和分析场合，三者通常是配合共用，是市场上最受欢迎的开源日志聚合工具。它被 Netflix , Facebook , Microsoft , LinkedIn 和 Cisco 使用。这三个组件都是由 Elastic 开发和维护的。Elasticsearch 本质上是一个 NoSQL , 以 Lucene 搜索引擎实现。 Logstash 是一个日志管道系统，可以提取、转换数据并将其加载到像 Elasticsearch 这样的商店中。 Kibana 是 Elasticsearch 之上的可视化层。

几年前出现了数据收集器 Beats , 能简化数据传输到 Logstash 的过程。用户可以安装 Beat , 能导出 NGINX 日志或 Envoy 代理日志，以便在 Elasticsearch 中有效使用，无需了解每种类型日志的正确语法。

在安装生产级 ELK 堆栈时，可能会包含 Kafka , Redis 和 NGINX 等其他部分。此外，Logstash 通常可以用 Fluentd 替换。这个系统操作起来很复杂，早期有很多问题导致了很多抱怨。这些问题很大程度上都被修复了，但它仍然是一个复杂的系统，所以对于较小的操作，你可能不想尝试它。



ELK 堆栈还通过 Kibana 提供了出色的可视化工具，但它缺乏警报功能。Elastic 在付费 X-Pack 插件中提供警报功能，但开源系统中没有内置任何功能。Yelp 为这个问题提供了名为 ElastAlert 的解决方案，可能还有其他类似的工具。这个额外的软件非常强大，但它进一步增加了 ELK 堆栈的复杂性。

ELK Stack 在最近两年迅速崛起，和传统的日志处理方案相比，ELK Stack 具有如下几个优点：

- 处理方式灵活。Elasticsearch 是实时全文索引，不需要像 storm 那样预先编程才能使用；

- 配置简易上手。Elasticsearch 全部采用 JSON 接口，Logstash 是 Ruby DSL 设计，都是目前业界最通用的配置语法设计；
- 检索性能高效。虽然每次查询都是实时计算，但是优秀的设计和实现基本可以达到全天数据查询的秒级响应；
- 集群线性扩展。不管是 Elasticsearch 集群还是 Logstash 集群都是可以线性扩展的；
- 前端操作炫丽。Kibana 界面上，只需要点击鼠标，就可以完成搜索、聚合功能，生成炫丽的仪表板。

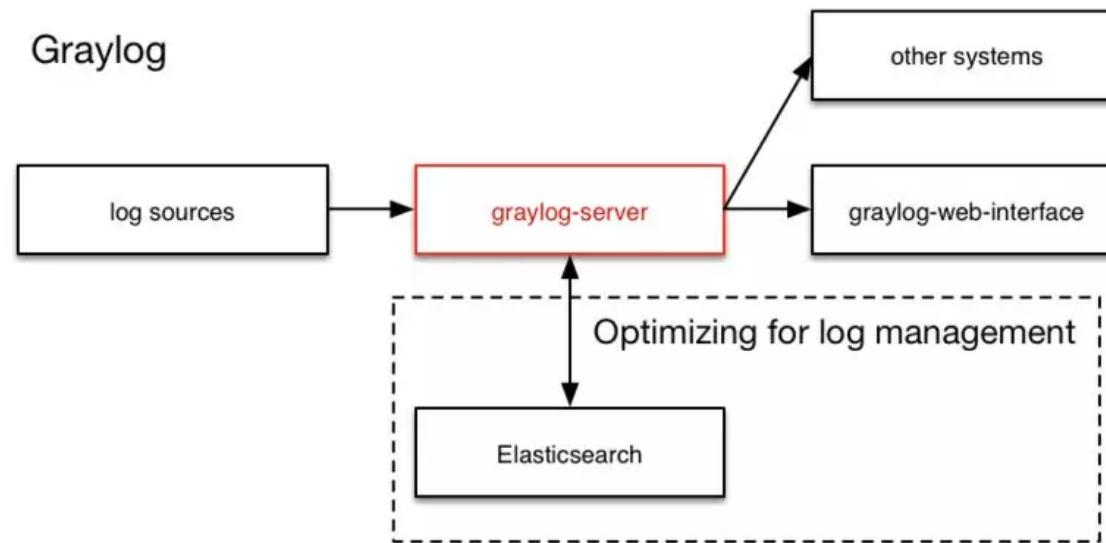
## Graylog

Graylog 是强大的日志管理、分析工具，基于 Elasticsearch, Java 和 MongoDB，这使得它像 ELK 堆栈一样运行起来很复杂，甚至更加复杂。但是，Graylog 开源版本带有内置的警报，以及其他一些值得注意的功能，如流式传输，消息重写和地理定位。

流式传输功能允许数据在处理时能实时路由到特定 Stream。使用此功能，用户可以在一个 Stream 中查看所有数据库错误，在另一个 Stream 中查看 Web 服务器错误。当添加新项目或超过阈值时，告警甚至可以基于这些 Stream。延迟可能是日志聚合系统的最大问题之一，Graylog 中的 Streams 中消除了这个问题，一旦日志进入，无需处理即可通过 Stream 路由到其他系统。

消息重写功能使用开源规则引擎 Drools，允许根据用户定义的规则文件来评估所有传入消息。该文件可以丢弃消息（称为黑名单），添加或删除字段，以及修改信息。

Graylog 最酷的功能可能是地理位置功能，它支持在地图上绘制 IP 地址。这样功能相当常见，Kibana 中也有这个功能，但 Graylog 中增加了很多价值，特别是你想将它用作 SIEM 系统时。地理定位功能在 Graylog 的开源版本中提供。



图片来源：<https://testerhome.com/topics/3026>

Graylog 吸引人的地方：

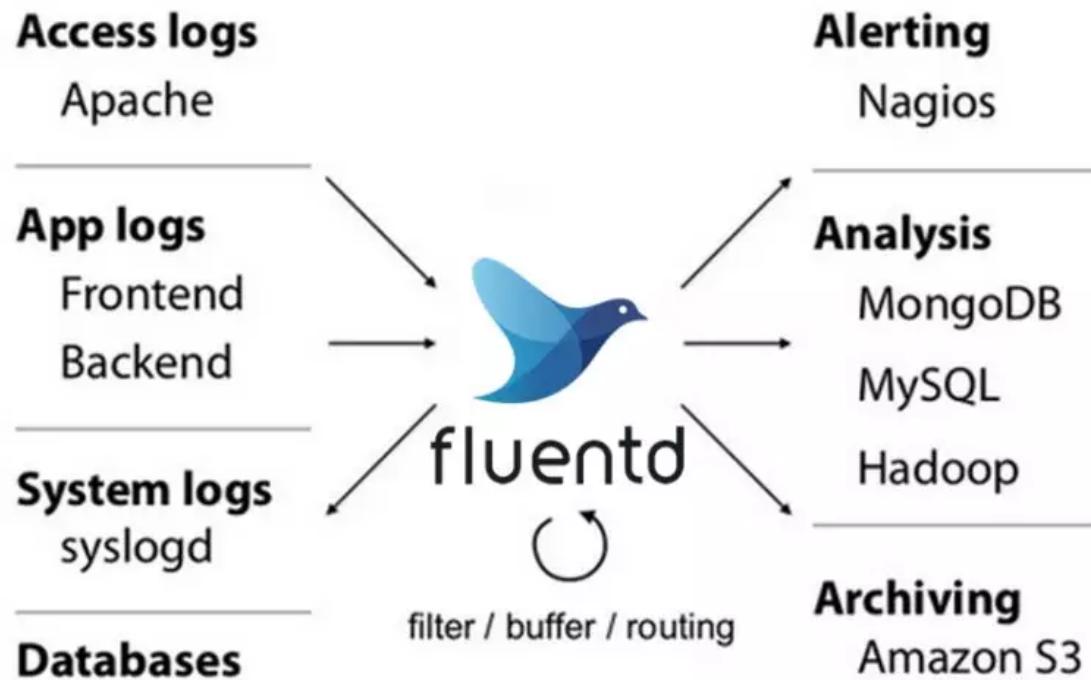
- 一体化方案，安装方便，不像 ELK 有 3 个独立系统间的集成问题。
- 采集原始日志，并可以事后再添加字段，比如 `http_status_code`, `response_time` 等等。

- 自己开发采集日志的脚本，并用 curl/nc 发送到 Graylog Server，发送格式是自定义的 GELF，Fluentd 和 Logstash 都有相应的输出 GELF 消息的插件。自己开发带来很大的自由度。实际上只需要用 inotifywait 监控日志的 modify 事件，并把日志的新增行用 curl/netcat 发送到 Graylog Server 就可。
- 搜索结果高亮显示，就像 google 一样。
- 搜索语法简单，比如： source:mongo AND reponse\_time\_ms:>5000，避免直接输入 elasticsearch 搜索 json 语法。
- 搜索条件可以导出为 elasticsearch 的搜索 json 文本，方便直接开发调用 elasticsearch rest api 的搜索脚本。

Graylog 开源版官网：<https://www.graylog.org/>

## Fluentd

Fluentd 是一个完全开源免费的 log 信息收集软件，支持超过 125 个系统的 log 信息收集，用 C 和 Ruby 编写，被 CNCF 接受为孵化项目，并得到了 AWS 和 Google Cloud 的推荐。在许多安装中，Fluentd 已成为 Logstash 的常见替代工具，充当本地聚合器，用于收集所有节点日志并发送到中央存储系统。但 Fluentd 不是一个日志集成系统。



图片来源：<http://www.muzixing.com/pages/2017/02/05/fluentdru-men-jiao-cheng.html>

Fluentd 使用强大的插件系统，有超过 500 个插件可供使用，可快速轻松地集成不同的数据源和数据输出，涵盖你的大部分用例。

Fluentd 内存要求低（仅几十兆字节），有高吞吐量，因此是 Kubernetes 环境中的常见选择。在像 Kubernetes 这样的环境中，每个 Pod 都有一个 Fluentd side-car，内存消耗将随着每个新 Pod 的创建而线性增加。使用 Fluentd 将大大降低系统利用率。

## 告警和可视化工具

---

通过名称就可以知道告警和可视化工具的用途，告警和可视化系统专注于理解其他系统的输出。这就是他们被归为一组的原因。可视化和警报工具可以将系统输出结构化地表示出来。

### 告警和可视化常见类型

首先，我们要弄清楚哪些不是告警。如果响应人员无法对问题采取任何措施，那么告警就不应该发送。这种情景包括发送给多个人，但只有少数人可以响应的的告警。

例如，如果操作员每天从警报系统收到数百封电子邮件，他将忽略来自警报系统的所有电子邮件，只有在遇到问题，客户发送电子邮件或老板打电话时才会回应真实事件。在这种情况下，警报已失去其意义和用途。

警报不应该是一连串的信息或状态更新。它们只是许多系统称为警报的数据点，代表了一些应该被知晓但没有被响应的事件。这些信息通常是警报工具的可视化系统的一部分，而不是触发实际通知的事件。

告警可以分为两类：内部中断和外部中断。在这个模型中，系统性能降级被视为中断，因为通常不知道每个用户的影响有多严重。

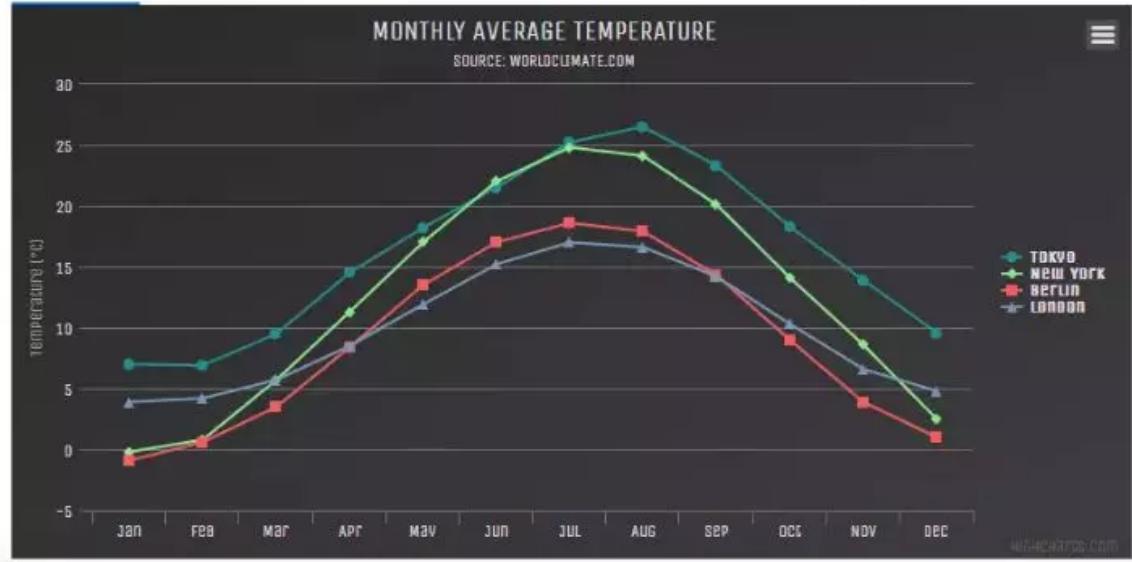
内部中断的优先级低于外部中断，但仍需要快速响应。内部中断通常涉及公司员工使用的内部系统或仅对公司员工可见的应用程序组件。

外部中断包括任何会立即影响客户的系统中断。这些不包括影响系统更新发布的系统中断，但包括面向客户的应用程序故障，数据库中断和网络分区等，还包括可能对用户没有直接影响的工具故障。

## 可视化

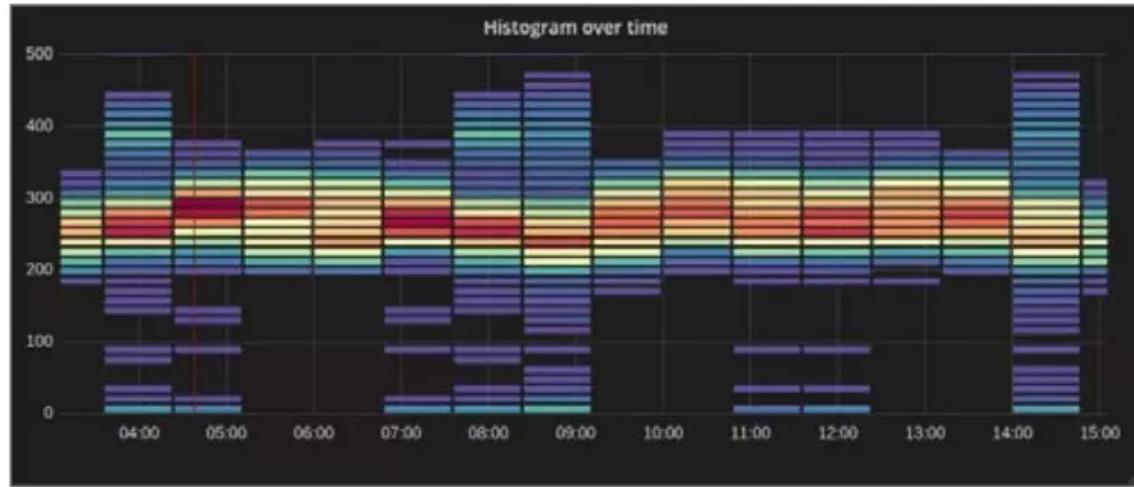
常见的可视化和理解系统的方案有：

**折线图**：折线图可能是最常见和最普遍的可视化。随着时间的推移，折线图可以很好地理解系统。也可以堆叠折线图以显示关系。例如，你可能希望单独查看每个服务器上的请求，但也可以聚合查看。



**热图**：另一种常见的可视化是热图，配合直方图查看很有用。这种类型的可视化类似于条形图，但可以在条形图中显示表示整体度量标准的不同百分位数的渐变。

例如，你可能正在查看请求延迟，并希望快速了解所有请求的总体趋势和分布。热图对此非常有用，可以通过颜色快速浏览每个部分的数量。



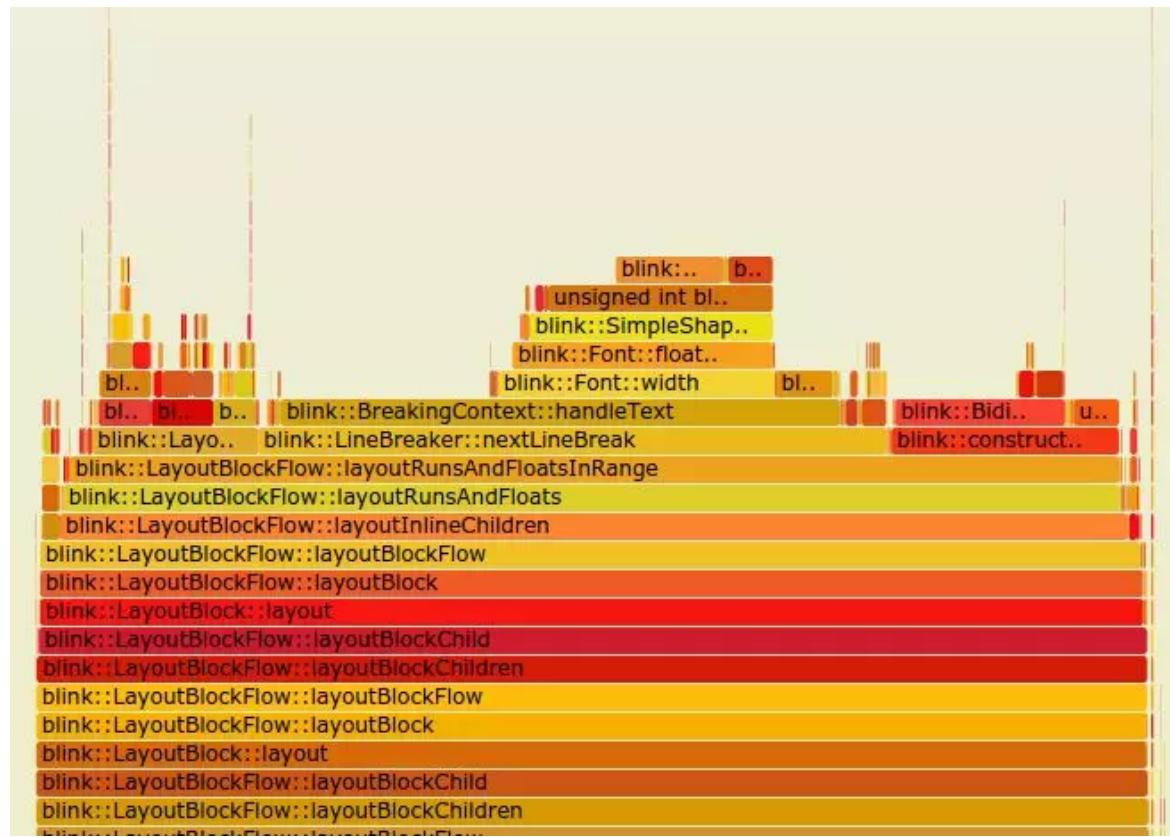
**gauge** : gauge 是单值计量可视化。gauge 的面貌可以是半圆表或全圆表。您可以自定义内线和外线的厚度以达到所需的设计美学效果。测量仪和文本的颜色可根据一组规则完全自定义。



**火焰图**：火焰图是基于 perf 结果产生的 SVG 图片，用来展示 CPU 的调用栈。

y 轴表示调用栈，每一层都是一个函数。调用栈越深，火焰就越高，顶部就是正在执行的函数，下方都是它的父函数。

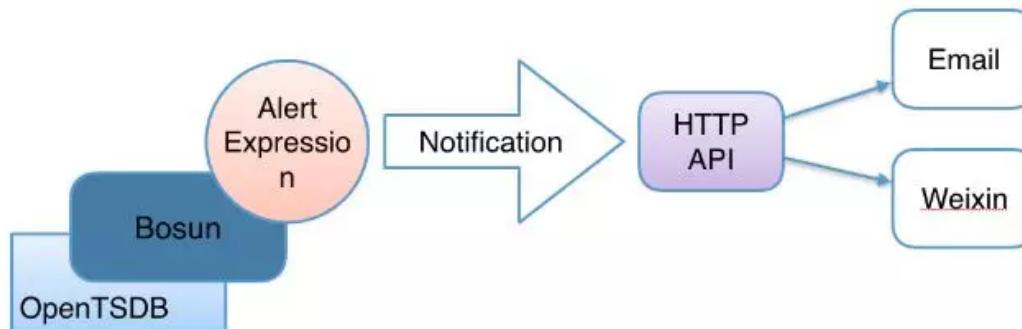
x 轴表示抽样数，如果一个函数在 x 轴占据的宽度越宽，就表示它被抽到的次数多，即执行的时间长。注意，x 轴不代表时间，而是所有的调用栈合并后，按字母顺序排列的。



## 告警工具

### Bosun

Bosun 是一个新型的监控和告警系统，由 Stack Exchange 团队打造，使用 golang 编写，支持定义复杂的告警规则，支持 OpenTSDB、Graphite、Logstash-Elasticsearch 等数据源。Bosun 将是 zabbix、nagios 的有力竞争者。



图片来源：<http://blog.tangyingkang.com/post/2016/12/06/bosun-alert-guide/>

Bosun 的一个非常巧妙的功能是它可以让你根据历史数据测试警报。Bosun 还具有一些常见的功能，如显示简单的图形和创建警报。Bosun 通过表达式语言来查询监控指标，可以看成是一个简易的编程语言

集，这在使它变得灵活强大的同时，也令它略显复杂。

## Cabot

Cabot 是一个免费开源的轻量级监控报警服务，集合了 PagerDuty，Server Density，Pingdom 和 Nagios 所具备的一些最佳功能，但是没有这些工具复杂，也不如它们成本高。

Cabot 的架构和 Bosun 类似，都不收集数据。原生支持 Graphite 和 Jenkins，比较少见。

Cabot 提供了一个 Web 界面，允许监控服务（例如“Stage Redis 服务器”，“生产 ElasticSearch 集群”），并在服务发生故障时向值班团队发送电话，短信或电子邮件警报，连一行代码都不需要你写。

Cabot 的报警可以基于：

- graphite 收集的监控数据；
- url 的响应内容和状态码；
- jenkins 编译任务的状态；

而不需要实现和维护一个全新的数据收集器系统。

## 可视化工具

## Grafana

Grafana 是用于可视化大型测量数据的开源程序，使用 Go 语言开发，功能齐全，有着好看的仪表盘和图表，可用来做日志的分析与展示曲线图（如 API 的请求日志），支持多种 backend，如 ElasticSearch、InfluxDB、OpenTSDB 等等，最常用于网络基础设施和应用分析，具有热插拔控制面板和可扩展的数据源，

使用 grafana 可以直观地设置警报。这意味着你可以查看图表，甚至可以查看由于系统性能下降而应该触发警报的位置，单击要触发警报的图表，然后告诉 Grafana 将警报发送到何处。这是一个非常强大的补充新能，不一定会取代警报平台，但肯定可以增强告警功能。

从本质上说，Grafana 是一个功能丰富的 Graphite-web 替代品，能帮助用户更简单地创建和编辑仪表盘。它包含一个独一无二的 Graphite 目标解析器，从而可以简化度量和函数的编辑。Grafana 快速的客户端渲染默认使用的是 Flot，即使很长的时间范围也可应对，这样用户就可以创建具有智能轴格式（比如线和点）的复杂图表了。

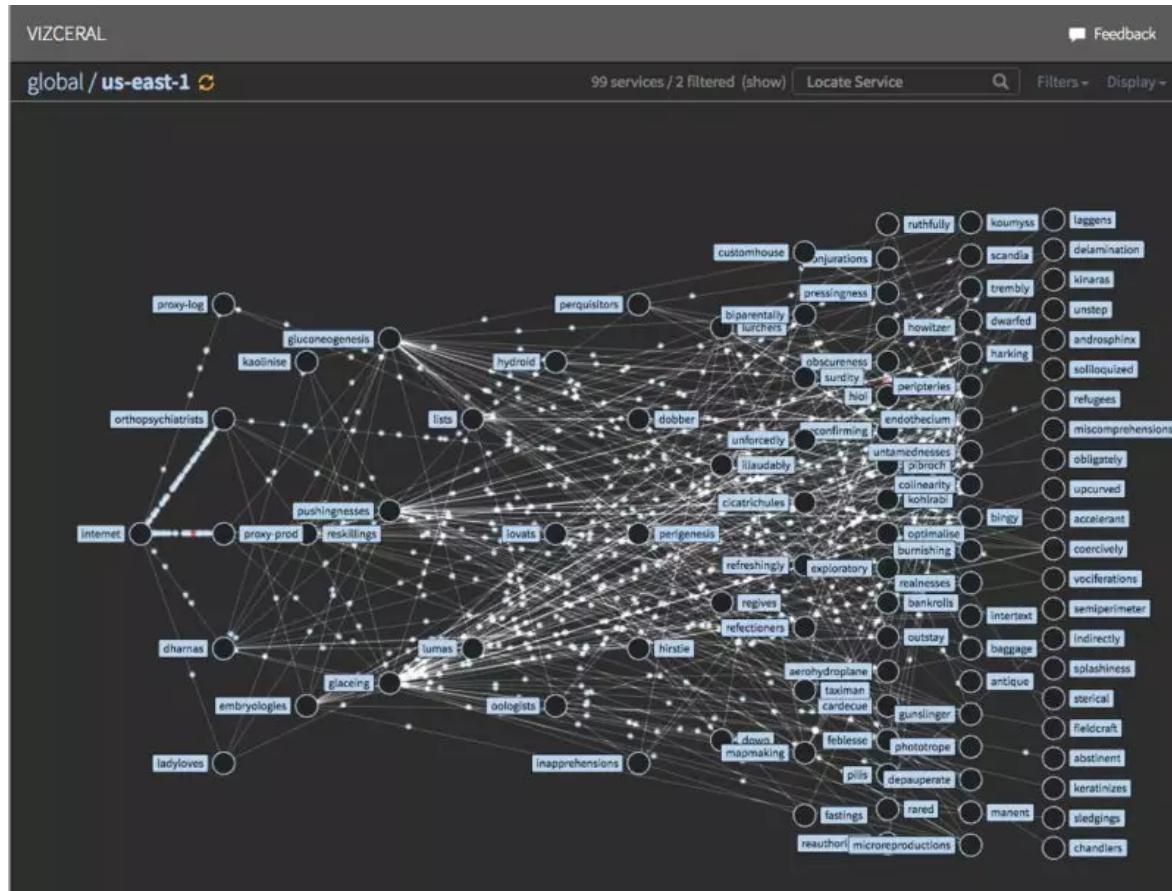


## Vizceral

Vizceral 是 Netflix 发布的一个开源项目，用于近乎实时地监控应用程序和集群之间的网络流量。

Vizceral 是一组采用 WebG 标准实现的动态展示线路图组件，可以实现数据的查看以及交互，分为全局、部分区域、水平三个维度，使数据更为直观明了的展示。

Vizceral 组件可以采取多个流量图，并将生成一个“全局”图，显示所有传入的流量到每个“区域”，支持跨区域通信。



Netflix 区域间流量图

## 参考资料

<https://opensource.com/article/18/8/now-available-open-source-guide-devops-monitoring-tools>

置顶InfoQ公众号



朝闻天下技术事



如何将机器学习应用到公司业务中，有哪些落地案例和踩坑经验可供参考？相关 AI 工具、平台、框架如何做选型？怎样构建一个专门的人工智能团队？

学习来自 Google、Twitter、Netflix、BAT、360、京东、美团、小米等 40+ 机器学习落地案例，还有知识图谱、NLP、语音识别、搜索推荐、计算机视觉、AI 架构等热门技术，干货满满。

AICon 大会 8 折售票火热进行中，团购更优惠。点击 “[阅读原文](#)” 了解更多详情！如有任何问题，可咨询票务小姐姐：18514549229 [ 微信同号 ]



[Read more](#)