**图论科技贺宏宏整理**

**一、Google 开源的项目**

看看有没有感兴趣的，排名顺序按照 Github ★Star 数排列。

0、机器学习系统 TensorFlow ★Star 62533

TensorFlow 是谷歌的第二代机器学习系统，按照谷歌所说，在某些基准测试中，TensorFlow的表现比第一代的DistBelief快了2倍。TensorFlow 内建深度学习的扩展支持，任何能够用计算流图形来表达的计算，都可以使用TensorFlow。任何基于梯度的机器学习算法都能够受益于TensorFlow的自动分 化（auto-differentiation）。通过灵活的Python接口，要在TensorFlow中表达想法也会很容易。（详情：https://github.com/tensorflow/tensorflow）

1、material-design-icons ★Star 30315

Google 开源了 Material Design 系统图标包其中的 750 个字形。该系统图标包含常用的图标，如用于媒体播放、通讯、内容编辑、连接等等。在 Web 应用，安卓和 iOS 设计均适用。（详情：http://google.github.io/material-design-icons/）

2、前端开发工具组MDL ★Star 27873

MDL (Material Design Lite) 是 Google 推出的网站前端开发工具组。Material Design Lite (MDL)可以让你添加一个 Material Design 的外观和感觉到你的静态内容网站，不依赖于任何的 JavaScript 框架和库。MDL 可以优化跨设备的使用体验，可以在旧版的浏览器进行平滑的切换，提供非常快速的访问体验。（详情：https://github.com/google/material-design-lite）

3、Web 前端框架 Angular ★Star 25524

Angular 是一款十分流行且好用的 Web 前端框架，目前由 Google 维护。官方已将 Angular 2 和之前的版本 Angular.js 分开维护（两者的 GitHub 地址和项目主页皆不相同）。渐进式 Web 应用，借助现代化 Web 平台的力量，交付 app 式体验。高性能、离线化、零安装。（详情：https://github.com/angular/angular）

4、容器集群管理系统 Kubernetes ★Star 24599

Kubernetes 是来自 Google 云平台的开源容器集群管理系统。基于 Docker 构建一个容器的调度服务。该系统可以自动在一个容器集群中选择一个工作容器供使用。其核心概念是 Container Pod。（详情：https://github.com/kubernetes/kubernetes）

5、数据描述语言 protobuf ★Star 18447

Protocol Buffers (ProtocolBuffer/ protobuf )是Google公司开发的一种数据描述语言，类似于XML能够将结构化数据序列化，可用于数据存储、通信协议等方面。现阶段支持C++、JAVA、Python等三种编程语言。同XML相比，Protocol buffers在序列化结构化数据方面有许多优点（详情：https://github.com/google/protobuf）

6、Java 常用库 Guava ★Star 17267

该项目是 Google 的一个开源项目，包含许多 Google 核心的 Java 常用库。（详情：https://github.com/google/guava）

7、Spark ★Star 13378

Spark 是谷歌公司推出的一款基于 Chrome 浏览器的开发环境。提供一组可重用的 UI 组件。采用 Dart 开发。（安装方法：https://github.com/dart-lang/spark/tree/master/ide）

8、RPC 框架 GRPC ★Star 10363

GRPC 是一个高性能、开源和通用的 RPC 框架，面向移动和 HTTP/2 设计。目前提供 C、Java 和 Go 语言版本，分别是：grpc, grpc-java, grpc-go. 其中 C 版本支持 C, C++, Node.js, Python, Ruby, Objective-C, PHP 和 C# 支持.GRPC 基于 HTTP/2 标准设计，带来诸如双向流、流控、头部压缩、单 TCP 连接上的多复用请求等特。这些特性使得其在移动设备上表现更好，更省电和节省空间占用。（详情：https://github.com/grpc/grpc）

9、高质量压缩图片算法工具Guetzli ★Star 9959

Guetzli，是一个针对数码图像和网页图像的 JPEG 编码器，能够通过产生更小的 JPEG 文件来达到更快的在线体验，并且同时保持与当前浏览器，图像处理应用和 JPEG 标准的兼容性。Google 称 Guetzli 创建高质量的 JPEG 图像文件的大小比当前的压缩方法要再小 35％。上图为 16x16 像素样本，是挂在蓝天下的一根电话线，传统 JPEG 算法经常会遇到的失真状况。左边是未压缩的原图，中间为较小尺寸的 libjpeg，右边是失真更少的 Guetzli 。（详情：https://github.com/google/guetzli）

10、k/v数据库 Leveldb ★Star 9799

Leveldb是一个google实现的非常高效的kv数据库，目前的版本1.2能够支持billion级别的数据量了。 在这个数量级别下还有着非常高的性能，主要归功于它的良好的设计。特别是LSM算法。（详情：https://github.com/google/leveldb）

11、自动化网络审查工具 Lighthouse ★Star 9400

Lighthouse 是一个开源的自动化工具，用于改进网络应用的质量。 可以将其作为一个 Chrome 扩展程序运行，或从命令行运行。 当为 Lighthouse 提供一个要审查的网址，它将针对此页面运行一连串的测试，然后生成一个有关页面性能的报告。可以参考失败的测试，看看可以采取哪些措施来改进应用。（详情：https://github.com/GoogleChrome/lighthouse）

12、Gson ★Star 9261

Gson 是 Google 提供的用来在 Java 对象和 JSON 数据之间进行映射的 Java 类库。可以将一个 JSON 字符串转成一个 Java 对象，或者反过来。（详情：https://github.com/google/gson）

13、最小系统加载工具 systemjs ★Star 8356

systemjs 是一个最小系统加载工具，用来创建插件来处理可替代的场景加载过程，包括加载 CSS 场景和图片，主要运行在浏览器和 NodeJS 中。它是 ES6 浏览器加载程序的的扩展，将应用在本地浏览器中。通常创建的插件名称是模块本身，要是没有特意指定用途，则默认插件名是模块的扩展名称。（详情：https://github.com/systemjs/systemjs）

14、序列化库 FlatBuffers ★Star 7991

FlatBuffers 是一个 Java 的序列化库，用于游戏和其他内存受限的应用。FlatBuffers 可以让你直接访问序列化后的数据，无需解压并进行解析的过程。同时提供很强的向前和向后兼容性。FlatBuffers 支持 C++ 和 Java 语言，无需依赖第三方库支持。（详情：https://github.com/google/flatbuffers）

15、Xi 文本编辑器 Xi Editor ★Star 8000

Xi Editor 是 Google 开源的一款用 Rust 语言编写的文本编辑器，最初是为 Mac OS X 构建的，使用 Cocoa 作为用户界面，已有计划适配其它平台。所有编辑操作都可以在 16ms 之内提交并处理。文本绘制使用最好的技术（如 Mac 上的 Core Text，Windows 上的 DirectWrite 等），并完全支持 Unicode。（详情：https://github.com/google/xi-editor）

16、Grumpy ★Star 7760

Grumpy 是一个 Python to Go 源代码翻译编译器和运行时，旨在替代 CPython 2.7。 关键的区别是它将 Python 源代码编译为 Go 源代码，然后将其编译为本机代码，而不是字节码。这意味着 Grumpy 没有 VM。编译的 Go 源代码是对 Grumpy 运行时的一系列调用，Go 库服务与 Python C API 类似的目的（尽管不直接支持 C API）。（详情：https://github.com/google/grumpy）

17、Javascript 编译器 Traceur ★Star 7300

Traceur 是一个来自 Google 的 Javascript 编译器，通过它可以体验一些很新并且有趣的 Javascript 语言特性，这些多数是还没有被当前浏览器实现的 ECMAScript 标准或草案，比如：数组比较、类、模块、迭代器、方法参数默认值、Promise等。（详情：https://github.com/google/traceur-compiler）

18、媒体播放器 ExoPlayer ★Star 7152

ExoPlayer 是 Android 上一个应用级的媒体播放器。它为 Android MediaPlayer 的 API 在播放本地或在线的视频与音频上提供了一个候选。ExoPlayer 支持一些 Android MediaPlayer API 无法提供的特性，包括 DASH 和 SmoothStreaming 自适应回放，持久的高速缓存和自定义渲染器。不像 MediaPlayer API，EXOPlayer 很容易定制和扩展，而且它可以通过 Play Store 更新升级。（详情：https://github.com/google/ExoPlayer）

19、代码构建工具Bazel ★Star 6641

Bazel 是 Google 的一款可再生的代码构建工具。它主要是用于构建 Google 的软件,处理出现在谷歌的开发环境的构建问题,比如说:大规模数据构建问题,共享代码库问题,从源代码构建的软件的相关问题。支持多种语言并且跨平台，还支持自动化测试和部署、具有再现性（Reproducibility）和规模化等特征。（详情：https://github.com/bazelbuild/bazel）

20、创作艺术和谱写曲子的机器智能 magenta ★Star 6563

Google Brain团队的一组研究人员发布了一个项目Project Magenta，其主要目标是利用机器学习创作艺术和谱写曲子。Project Magenta使用了 TensorFlow系统，研究人员在GitHub上开源了他们的模型和工具。（详情：https://github.com/tensorflow/magenta）

21、自动生成命令行界面的内容库 Python Fire ★Star 6500

Python Fire 是 Google 开源的一个可从任何 Python 代码自动生成命令行接口（CLI）的库。Python Fire 是一种在 Python 中创建 CLI 的简单方法；是开发和调试 Python 代码的有用工具；能够使 Bash 和 Python 之间的转换更为容易；并且通过使用你需要导入和创建的模块和变量来设置 REPL，使得使用 Python REPL 更容易

（详情：https://github.com/google/python-fire）

22、软硬件框架AnyPixel.js ★Star 5906

AnyPixel.js 是 Google 开源的一个软件和硬件框架，可以用来构建各种由“像素”构成的展示，每个像素可以是任何一种可交互的实体对象，如 LED 灯、气球等。（详情：https://github.com/googlecreativelab/anypixel）

23、物联网工具Physical Web ★Star 5899

Physical Web 是由 Chrome 团队主导的一个项目, 意在用 URL 连接世界, 方便用户接受数据。在Web世界中，各种URL可以说是链接的基础，也是去中心化的，所以“The Physical Web”要做的就是让每个智能设备用URL来标识自己，然后用户按照自己的需要通过URL和设备进行交互。这样一来，你使用智能设备的体验就和在网站上使 用各种超链接差不多了。（详情：https://github.com/google/physical-web）

24、Docker 容器性能分析工具cAdvisorc ★Star 5853

Advisor 是谷歌公司用来分析运行中的 Docker 容器的资源占用以及性能特性的工具。cAdvisor 是一个运行中的守护进程用来收集、聚合、处理和导出运行容器相关的信息，每个容器保持独立的参数、历史资源使用情况和完整的资源使用数据。当前支持 lmctfy 容器和 Docker 容器。（详情：https://github.com/google/cadvisor）

25、C++单元测试工具 ★Star 5755

Google的开源C++单元测试框架Google Test，简称gtest 是一个非常的不错单元测试框架。支持跨平台以及包括 Windows CE 和 Symbian 在内的一些手机操作系统。（详情：https://github.com/google/googletest）

26、IndexedDB 关系查询引擎 lovefield ★Star 5532

lovefield 是建立在 IndexedDB 上的关系查询引擎。它提供了类似 SQL 的语法，并且可以跨浏览器工作（目前支持 Chrome 37 及以上版本，Firefox 31 及以上版本，IE 10 及以上版本）。（详情：https://github.com/google/lovefield）

27、 Java 生成器源代码集合 Auto ★Star 5295

Auto 是 Java 生成器源代码集合，Java 有许多机械、重复、未经测试的代码，而且有时会出现一些微妙的 Bug 。Auto 项目是自动执行这些类型的任务的代码生成器的集合，他们可以无 Bug 创建你要编写的代码。（详情：https://github.com/google/auto）

28、Python 代码格式化的工具 ★Star 5197

YAPF 是 Google 开发的一个用来格式化 Python 代码的工具。（详情：https://github.com/google/yapf）

29、基于 TensorFlow 的神经网络库 Sonnet ★Star 5000

Sonnet 库使用面向对象的方法，允许创建定义一些前向传导计算的模块。模块用一些输入 Tensor 调用，添加操作到图里并返回输出 Tensor。其中一种设计选择是通过在随后调用相同的模块时自动重用变量来确保变量分享被透明化处理。 该库兼容 Linux/Mac OS X 和 Python 2.7。TensorFlow 的版本必须至少为 1.0.1。Sonnet 支持 TensorFlow 的 virtualenv 安装模式，以及 nativ pip 安装。（详情：https://github.com/deepmind/sonnet）

30、无损压缩算法Brotli ★Star 4822

Brotli 是一个通用目的的无损压缩算法，它通过用变种的 LZ77 算法，Huffman 编码和二阶文本建模进行数据压缩，是一种压缩比很高的压缩方法。在压缩速度上跟 Deflate 差不多，但是提供了更密集的压缩。（详情：https://github.com/google/brotli）

31、负载均衡系统 Seesaw ★Star 4286

Seesaw 是 Google 开源的一个基于 Linux 的负载均衡系统。Seesaw 包含基本的负载均衡特性，同时支持一些高级的功能，诸如：anycast, Direct Server Return (DSR), 支持多个 VLANs 和集中式配置。同时其设计的宗旨是易于维护。（详情：https://github.com/google/seesaw）

32、Gumbo ★Star 4060

Gumbo 是 Google 的一款用C语言实现的HTML5解析库，无需任何外部依赖。（详情：https://github.com/google/gumbo-parser）

33、GXUI ★Star 3803

GXUI 是 Google 出品的一个跨平台 GO 语言的 UI 框架。（详情：https://github.com/google/gxui）

34、iOS UI 测试自动化框架 EarlGrey ★Star 3721

遵从开源Apache授权的EarlGrey，由Objective-C编写而成。Google也确实用的它来测试自家的iOS app。EarlGrey可支持和模拟回溯至iOS 8的设备，有关该工具的详细文档请戳这里。除了EarlGrey，其它iOS UI测试自动化框架还包括Calabash和Sauce Labs的Appium。（详情：https://github.com/google/EarlGrey）

35、Blockly ★Star 3520

Blockly 是一个基于 Web 的可视化编程工具，只需要拖动几个图形就可以编程, 完全不需要打字. 根据项目FAQ介绍, 这个新语言的主要目的是为web应用提供宏(或脚本编 程)的支持. 可以把生成的脚本输出成javascript, python 等.已经有几个利用Blockly的demo.（详情：https://github.com/google/blockly）

36、go-github ★Star 2707

go-github 是 Google 对 Github 的开放 API 进行 Go 语言封装的一个项目。（详情：https://github.com/google/go-github）

37、抓取bug工具 error-prone ★Star 2708

error-prone 用来抓取 Java 编译时的错误。通常使用编译器只能做静态类型的检查。但使用此工具，能够进行编译器的类型分析，能够检测并抓取到编译过程中的 Bug，能够大大的节约开发者的时间。（详情：https://github.com/google/error-prone）

38、ROS 系统支持的 SLAM 库 Cartographer ★Star 2700

Cartographer，是Google开源的一个ROS系统支持的2D和3D SLAM（simultaneous localization and mapping）库。SLAM 算法结合来自多个传感器（比如，LIDAR、IMU 和 摄像头）的数据，同步计算传感器的位置并绘制传感器周围的环境。在产业界和学术界常见的传感器配置上，Cartographer 能实时建立全局一致的地图。（详情：https://github.com/googlecartographer/cartographer）

39、安卓电量分析工具 Battery Historian ★Star 2579

Battery Historian 是一个通过分析安卓 "bugreport" 进程文件来统计电量消费情况。它允许应用程序开发人员在时间线上可视化系统和应用程序级事件，具有平移和缩放功能，在设备上完全充电后可以轻松查看各种汇总统计信息，并能选择应用程序，检查影响应用程序特定电池的指标。 它还允许两个错误报告的A / B比较，突出显示关键电池相关指标的差异。（详情：https://github.com/google/battery-historian）

40、通用编码器&解码器框架 seq2seq ★Star 2433

seq2seq 是 Google 开源的一款用于 TensorFlow 的通用编码器&解码器框架（encoder-decoder framework），可用于机器翻译、文本摘要、会话建模、图像描述等。（详情：https://github.com/google/seq2seq）

41、Web应用 Tracing Framework ★Star 2371

Tracing Framework 是一套库、工具，用于跟踪和调查复杂的 Web 应用。它就可以帮你发现代码的性能问题，并且帮你打造十分流畅的 60 FPS Web 应用。它目前只能用于特定的应用场合，并不是为应用于所有场景而设计，如果你在使用过程中遇到了问题，请呈递你的 Bug。（详情：https://github.com/google/tracing-framework）

42、深度概率编程语言 Edward ★Star 2395

Edward 是一个用于概率建模、推理和评估的 Python 库。它是一个用于快速实验和研究概率模型的测试平台，其涵盖的模型范围从在小数据集上的经典层次模型到在大数据集上的复杂深度概率模型。Edward 融合了以下三个领域：贝叶斯统计学和机器学习、深度学习、概率编程。（详情：https://github.com/blei-lab/edward）

43、JavaScript 库 Closure Library ★Star 2257

Closure Library 是一个强大的 JavaScript 库，用于复杂的大规模的 Web 应用。它被许多 Google Web 应用使用，比如 Gmail 和 Google Docs。（详情：https://github.com/google/closure-library）

44、大型微服务系统管理工具 Istio ★Star 2291

Istio 是一个由谷歌、IBM 与 Lyft 共同开发的开源项目，旨在提供一种统一化的微服务连接、安全保障、管理与监控方式。Istio 项目能够为微服务架构提供流量管理机制，同时亦为其它增值功能（包括安全性、监控、路由、连接管理与策略等）创造了基础。这款软件利用久经考验的 Lyft Envoy 代理进行构建，可在无需对应用程序代码作出任何发动的前提下实现可视性与控制能力。（详情：https://github.com/istio/istio）

45、远程现场取证系统 GRR ★Star 2100

GRR 是 Google 开发的远程现场事件取证系统。GRR 由一个代理（客户端）和服务器端组成，客户端可以部署在一个任务系统中，服务器可以管理客户端，跟客户端进行交互。（详情：https://github.com/google/grr）

46、Android 的悬浮操作栏 Hover ★Star 2053

Hover 是一个用于 Android 的悬浮操作栏。该菜单栏可以作为一个 service 启动，从而跨进程使用，悬浮在桌面或者其他应用程序的前面，当然，这里需要申请权限。 Hover 仍处在开发阶段，还有很多代码清理工作需要做，也就是说，Hover 现在仅仅处于可用状态。（详情：https://github.com/google/hover）

47、模块化深度学习系统 Tensor2Tensor ★Star 2000

Tensor2Tensor 是一个模块化和可扩展的库和二进制文件， 能够帮助人们为各种机器学习程序创建最先进的模型，可应用于多个领域，如翻译、语法分析、图像信息描述等，大大提高了研究和开发的速度。（详情：https://github.com/tensorflow/tensor2tensor）

48、Google 模糊测试服务 OSS-Fuzz ★Star 1781

OSS-Fuzz 能够针对开源软件进行持续的模糊测试，它的目的是利用更新的模糊测试技术与可拓展的分布式执行相结合，提高一般软件基础架构的安全性与稳定性。OSS-Fuzz 结合了多种模糊测试技术/漏洞捕捉技术（即原来的libfuzzer）与清洗技术（即原来的 AddressSanitizer），并且通过 ClusterFuzz 为大规模可分布式执行提供了测试环境。（详情：https://github.com/google/oss-fuzz）

49、优化搜索工具or-tools ★Star 1771

or-tools 是 Google 的优化搜索工具。Google 优化工具包括：约束编程解决方案；为线性规划和混合整数规划解决方案提供简单统一的接口，包括 CBC, CLP, GLOP, GLPK, Gurobi, SCIP, 和 Sulum；背包算法；图算法 (最短路径，线性和分配，最小费用流，最大流)（详情：https://github.com/google/or-tools）

50、加密库安全测试套件 Wycheproof ★Star 1722

Wycheproof 是谷歌开源的加密库测试项目，它包含一系列安全测试，用来检测加密库（cryptographic libraries）软件是否存在已知的攻击漏洞。（详情：https://github.com/google/wycheproof）

51、Common Lisp Koans ★Star 1695

Common Lisp Koans（lisp-koans）是一个语言学习练习程序，类似 ruby koans，python koans 等等。Common Lisp Koans 主要是帮助学习一些 lisp 规范特性和改进，可以学习到大量的 Common Lisp 语言特性。（详情：https://github.com/google/lisp-koans）

52、 3D 图形开源压缩库 Draco ★Star 1600

Draco 是一种库，用于压缩和解压缩 3D 几何网格（geometric mesh）和点云（point cloud）。换句话说，它显著缩小了 3D 图形文件的大小，同时对 3D 图形的观看者来说又根本不严重影响视觉效果。它还旨在改善 3D 图形的压缩和传输。Draco 是作为 C++ 源代码发布的，可以用来压缩 3D 图形，另外还发布了处理编码数据的 C++ 和 Javascript 解码器。（详情：https://github.com/google/draco）

53、 JavaScript 库 Shaka Player ★Star 1354

Shaka Player 它实现了 DASH 客户端的功能。它的播放功能基于 HTML5 video、MediaSource Extensions，和 Encrypted Media Extensions 。一般的 DASH 客户端功能很难实现。（详情：https://github.com/google/shaka-player）

54、TensorFlow 深度学习库 Fold ★Star 1192

TensorFlow Fold 是用于创建使用结构化数据的 TensorFlow 模型库，其中计算图的结构取决于输入数据的结构。 TensorFlow Fold 使得处理不同数据尺寸和结构的深度学习模型更容易实现。（详情：https://github.com/tensorflow/fold）

55、Google 地球企业版 Earth Enterprise ★Star 1161

Earth Enterprise 是 Google Earth Enterprise 的开源版本，是一个提供构建和托管自定义 3D 地球模型和 2D 地图的地理空间应用，旨在让社区继续改进和推进该项目。（详情：https://github.com/google/earthenterprise）

56、图像差异比较库 Butteraugli ★Star 1100

Butteraugli 是用来评判两个图像之间的相似度。通过识别图像之间一些最受关注的差异点并给出相似度分值。这个项目的一个主要动机是对差异受体的不同颜色的位置和密度的统计，特别是蓝色的低密度锥窝。另一个动机来自于更准确的神经节细胞建模，特别是抑制频率空间。目前该项目只提供了 C++ 接口。（详情：https://github.com/google/butteraugli）

57、Go 语言实现的网络协协议栈 Netstack ★Star 1016

Netstack，使用 Go 语言编写的网络协议栈。通过安装 tun\_tcp\_echo demo 尝试在 Linux 上使用 。（详情：https://github.com/google/netstack）

58、钢琴二重奏 AI aiexperiments-ai-duet ★Star 883

“钢琴二重奏”的 A.I. Duet。该项目会在你弹出了几个音符之后，通过 AI 自动计算来帮你“补完”旋律的重奏部分。A.I. Duet 运用了人工智能技术，通晓音符的“编码规则”。（详情：https://github.com/googlecreativelab/aiexperiments-ai-duet）

59、端到端加密系统 E2EMail ★Star 704

E2EMail 是一个实验性质的端到端加密系统。E2EMail由Google开发，内置JavaScript内部开发的JavaScript加密库。它提供了一种通过Chrome扩展程序将OpenPGP集成到Gmail中的方法。消息的明文单独保留在客户端上。（详情：https://github.com/e2email-org/e2email）

# 二、阿里巴巴开源产品列表

代码：[**https://github.com/alibaba**](https://github.com/alibaba)

**taobao-kernel**

<http://kernel.taobao.org/>

<https://github.com/alibaba/taobao-kernel>

淘宝内核是淘宝内核开发团队基于RHEL6官方内核自己定制的内核，他与RHEL6系列内核的主要区别包括：  
1. 修改了一些编译脚本和错误，使得这个内核可以在RHEL5系列的系统中编译和使用，从而能够让广大用户既能够拥有RHEL5系列的稳定，同时享受到2.6.32系列内核诸多新功能和优化，并省去了重新安装操作系统的麻烦。  
2. 修复了一些Redhat内核中的bug。在使用Redhat系列内核时，经常会遇到一些内核bug等，我们秉承公开的原则会及时向Redhat反馈(通过bugzilla的方式，详细见【1】)，但是Redhat代码更新的过程很漫长，导致我们的问题无法及时得到解决，也没有立即可用的版本，于是我们会把这些修复加入淘宝内核中(patches.taobao目录下的部分patches)，等Redhat合并了fix以后再从淘宝内核中摘除。  
3. 增加了一些我们认为对淘宝业务有很大收益的特性比如netoops，bigalloc等(patches.taobao目录下的部分patches)，这些特性如果等待Redhat去实现需要更长的时间，所以只能自己来做。

**---------------------------------**

**AliJVM**

[http://jvm.taobao.org/index.php?title=首页](http://jvm.taobao.org/index.php?title=%E9%A6%96%E9%A1%B5)

阿里有几万台Java应用服务器，上千名Java工程师、及上百个Java应用。为此，核心系统研发部专用计算组的工作之一是专注于OpenJDK的优化及定制，根据业务、应用特点及开发者需要，提供稳定，高效和深度定制的JVM版本：TaobaoJVM。  
AliJVM基于OpenJDK HotSpot VM，是国内第一个优化、定制且开源的服务器版Java虚拟机。目前已经在淘宝、天猫上线，全部替换了Oracle官方JVM版本，在性能，功能上都初步体现了它的价值。

性能优化\*Performance Tuning（针对淘宝x86平台的专用优化）  
定制\*Customization（根据淘宝业务需求）  
Bug修复\*Bug fixing

[基于OpenJDK深度定制的淘宝JVM（TaobaoVM）](http://book.51cto.com/art/201504/472732.htm)

其实从严格意义上来说，在提升Java虚拟机性能的同时，却严重依赖物理CPU类型。也就是说，部署有TaobaoVM的服务器中，CPU全都是清一色的Intel CPU，且编译手段采用的是Intel C/CPP Compiler进行编译，以此对GC性能进行提升。除了优化编译效果外，TaobaoVM还使用了crc32指令实现JVM intrinsic降低JNI的调用开销  
  
在提升Java虚拟机性能的同时，却严重依赖物理CPU类型。也就是说，部署有TaobaoVM的服务器中，CPU全都是清一色的Intel CPU，且编译手段采用的是Intel C/CPP Compiler进行编译，以此对GC性能进行提升。除了优化编译效果外，TaobaoVM还使用了crc32指令实现JVM intrinsic降低JNI的调用开销  
  
除了在性能优化方面下足了功夫，TaobaoVM还在HotSpot的基础之上大幅度扩充了一些特定的增强实现。比如创新的GCIH（GC invisible heap）技术实现off-heap，这样一来就可以将生命周期较长的Java对象从heap中移至heap之外，并且GC不能管理GCIH内部的Java对象，这样做最大的好处就是降低了GC的回收平率以及提升了GC的回收效率，并且GCIH中的对象还能够在多个Java虚拟机进程中实现共享。其他扩充技术还有利用PMU hardware的Java profiling tool和诊断协助功能等。

**---------------------------------**

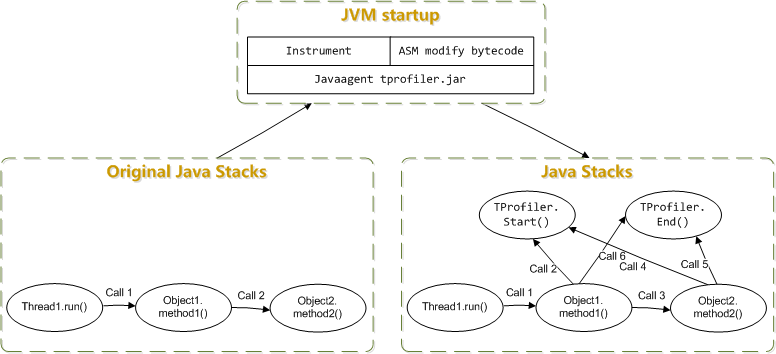
**TProfiler**

TProfiler是一个可以在生产环境长期使用的性能分析工具

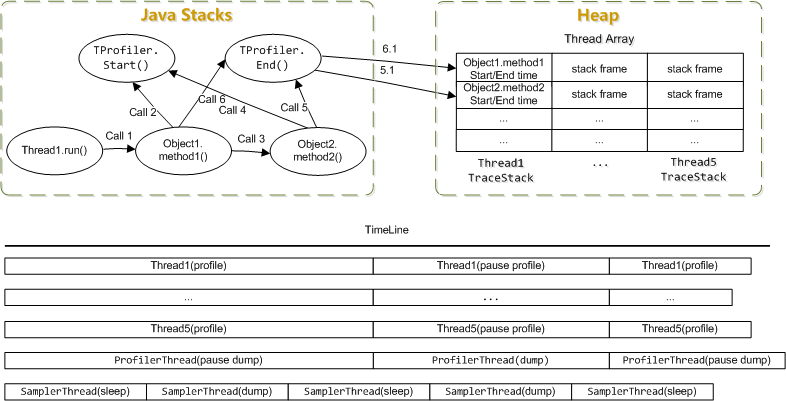
<https://github.com/alibaba/TProfiler>

TProfiler是一个可以在生产环境长期使用的性能分析工具.它同时支持剖析和采样两种方式,记录方法执行的时间和次数,生成方法热点 对象创建热点 线程状态分析等数据,为查找系统性能瓶颈提供数据支持.  
  
TProfiler在JVM启动时把时间采集程序注入到字节码中,整个过程无需修改应用源码.运行时会把数据写到日志文件,一般情况下每小时输出的日志小于50M.

字节码修改:

****

运行实现原理

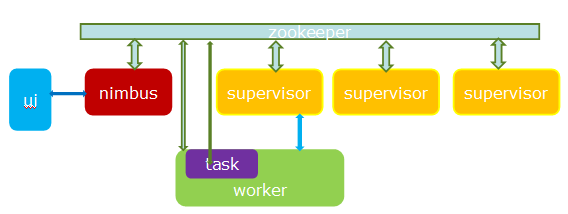


**---------------------------------**

**Jstom**

<https://github.com/alibaba/jstorm>

jstorm可以看作是storm的java增强版本，除了内核用纯java实现外，还包括了thrift、python、facet ui。从架构上看，其本质是一个基于zk的分布式调度系统。



更多参考此文：[Jstom简介](http://luoshi0801.iteye.com/blog/2168848)

------------------------------------------------

**tengine**

<http://tengine.taobao.org/>

Tengine是由淘宝网发起的Web服务器项目。它在Nginx的基础上，针对大访问量网站的需求，添加了很多高级功能和特性。



------------------------------------------------

**Dubbo**

<http://dubbo.io/>

DUBBO是一个分布式**服务框架**，致力于提供高性能和透明化的RPC远程服务调用方案，是阿里巴巴SOA服务化治理方案的核心框架，每天为2,000+个服务提供3,000,000,000+次访问量支持，并被广泛应用于阿里巴巴集团的各成员站点。

[用户指南和使用背景](http://dubbo.io/User+Guide-zh.htm#UserGuide-zh-%E8%83%8C%E6%99%AF)

随着互联网的发展，网站应用的规模不断扩大，常规的垂直应用架构已无法应对，分布式服务架构以及流动计算架构势在必行，亟需一个治理系统确保架构有条不紊的演进。

阿里内部并没有采用Zookeeper做为注册中心，而是使用自己实现的基于数据库的注册中心，即：Zookeeper注册中心并没有在阿里内部长时间运行的可靠性保障，此Zookeeper桥接实现只为开源版本提供，其可靠性依赖于Zookeeper本身的可靠性。



[Dubbo与Zookeeper、SpringMVC整合和使用（负载均衡、容错）](http://doc.okbase.net/congcong68/archive/112508.html)

Zookeeper作为Dubbo服务的注册中心，Dubbo原先基于数据库的注册中心，没采用Zookeeper，Zookeeper一个分布式的服务框架，是树型的目录服务的数据存储，能做到集群管理数据 ，这里能很好的作为Dubbo服务的注册中心，Dubbo能与Zookeeper做到集群部署，当提供者出现断电等异常停机时，Zookeeper注册中心能自动删除提供者信息，当提供者重启时，能自动恢复注册数据，以及订阅请求。

---------------------

**Druid**

<http://www.oschina.net/p/druid/>

Druid是一个JDBC组件，它包括三部分：   
  
DruidDriver 代理Driver，能够提供基于Filter－Chain模式的插件体系。   
DruidDataSource 高效可管理的数据库连接池。   
SQLParser   
Druid可以做什么？   
1) 可以监控数据库访问性能，Druid内置提供了一个功能强大的StatFilter插件，能够详细统计SQL的执行性能，这对于线上分析数据库访问性能有帮助。   
2) 替换DBCP和C3P0。Druid提供了一个高效、功能强大、可扩展性好的数据库连接池。   
3) 数据库密码加密。直接把数据库密码写在配置文件中，这是不好的行为，容易导致安全问题。DruidDruiver和DruidDataSource都支持PasswordCallback。   
4) SQL执行日志，Druid提供了不同的LogFilter，能够支持Common-Logging、Log4j和JdkLog，你可以按需要选择相应的LogFilter，监控你应用的数据库访问情况。   
扩展JDBC，如果你要对JDBC层有编程的需求，可以通过Druid提供的Filter-Chain机制，很方便编写JDBC层的扩展插件。

[在线文档](http://tool.oschina.net/apidocs/apidoc?api=druid0.26)

-----------------------------

**fastjson**

<http://www.oschina.net/p/fastjson>

fastjson 是一个性能很好的 Java 语言实现的 JSON 解析器和生成器，来自阿里巴巴的工程师开发。

主要特点：  
快速FAST (比其它任何基于Java的解析器和生成器更快，包括jackson）  
强大（支持普通JDK类包括任意Java Bean Class、Collection、Map、Date或enum）  
零依赖（没有依赖其它任何类库除了JDK）

------------------------------

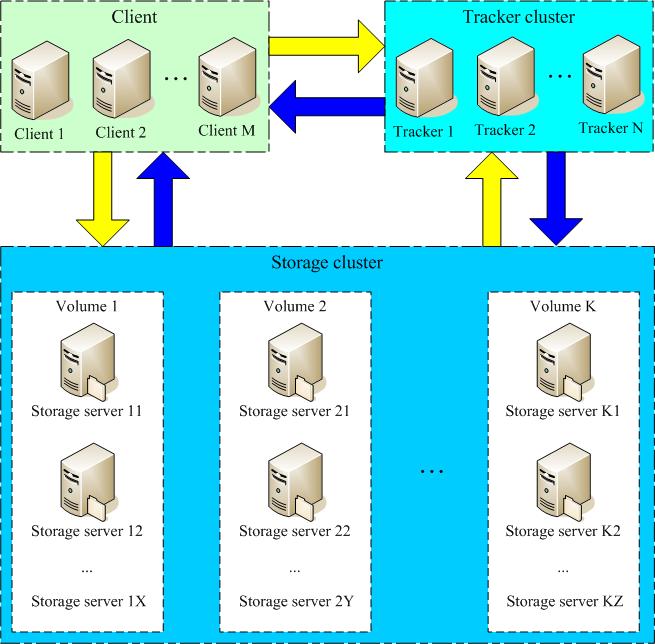
**FastDFS**

<http://bbs.chinaunix.net/forum-240-1.html>

FastDFS是一个开源的分布式文件系统，对文件进行管理，功能包括：文件存储、文件同步、文件访问（文件上传、文件下载）等，解决了大容量存储和负载均衡的问题。特别适合以文件为载体的在线服务，如相册网站、视频网站等等。  
FastDFS服务端有两个角色：跟踪器（tracker）和存储节点（storage）。跟踪器主要做调度工作，在访问上起负载均衡的作用。  
存储节点存储文件，完成文件管理的所有功能：存储、同步和提供存取接口，FastDFS同时对文件的meta data进行管理。

<http://www.oschina.net/p/fastdfs>

<https://sourceforge.net/projects/fastdfs/>



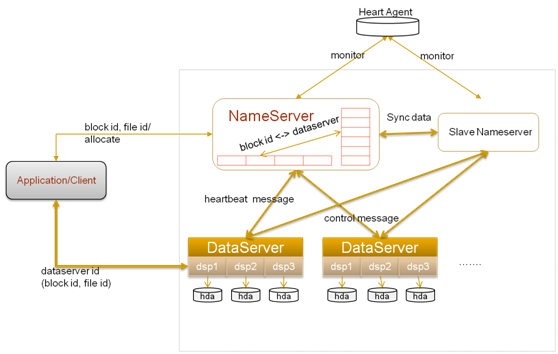
-----------------------

**TFS**

<https://github.com/alibaba/tfs>

TFS是淘宝针对海量非结构化数据存储设计的分布式系统，构筑在普通的Linux机器集群上，可为外部提供高可靠和高并发的存储访问。高可扩展、高可用、高性能、面向互联网服务。

<http://tfs.taobao.org/>



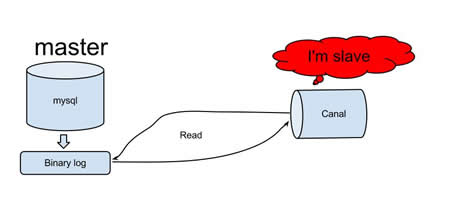
TFS为淘宝提供海量小文件存储，通常文件大小不超过1M，满足了淘宝对小文件存储的需求，被广泛地应用在淘宝各项应用中。它采用了HA架构和平滑扩容，保证了整个文件系统的可用性和扩展性。同时扁平化的数据组织结构，可将文件名映射到文件的物理地址，简化了文件的访问流程，一定程度上为TFS提供了良好的读写性能。  
TFS的总体结构  
一个TFS集群由两个!NameServer节点（一主一备）和多个!DataServer节点组成。这些服务程序都是作为一个用户级的程序运行在普通Linux机器上的。  
在TFS中，将大量的小文件(实际数据文件)合并成为一个大文件，这个大文件称为块(Block), 每个Block拥有在集群内唯一的编号(Block Id), Block Id在!NameServer在创建Block的时候分配, !NameServer维护block与!DataServer的关系。Block中的实际数据都存储在!DataServer上。而一台!DataServer服务器一般会有多个独立!DataServer进程存在，每个进程负责管理一个挂载点，这个挂载点一般是一个独立磁盘上的文件目录，以降低单个磁盘损坏带来的影响。

------------------------

**canal和outer**

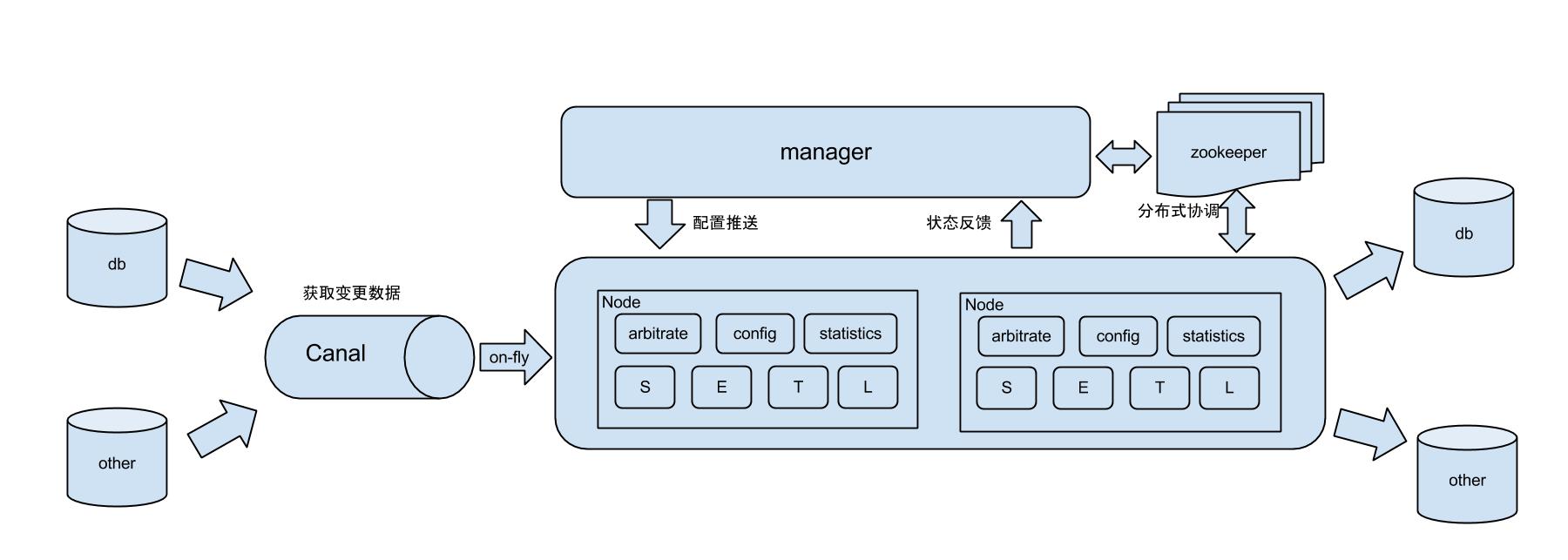
canal

canal 是阿里巴巴mysql数据库binlog的增量订阅&消费组件。  
定位： 基于数据库增量日志解析，提供增量数据订阅&消费，目前主要支持了mysql  
  
原理相对比较简单：  
canal模拟mysql slave的交互协议，伪装自己为mysql slave，向mysql master发送dump协议  
mysql master收到dump请求，开始推送binary log给slave(也就是canal)  
canal解析binary log对象(原始为byte流)



--------

otter 基于Canal开源产品，获取数据库增量日志数据。准实时同步到本机房或异地机房的mysql/oracle数据库. 一个分布式数据库同步系统。



[分布式数据库同步系统otter(解决中美异地机房)](http://agapple.iteye.com/blog/1929370)

杭州和美国异地机房的需求，同时为了提升用户体验，整个机房的架构为双A，两边均可写，由此诞生了otter这样一个产品。

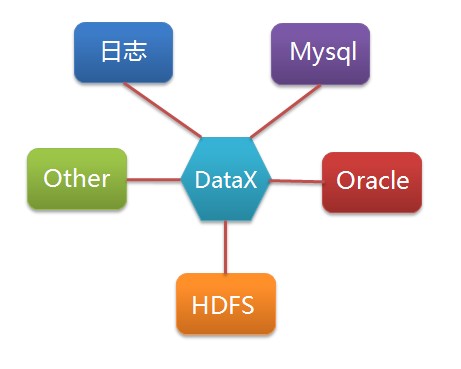
1.  异构库同步  
   a.  mysql ->  mysql/oracle.  (目前开源版本只支持mysql增量，目标库可以是mysql或者oracle，取决于canal的功能)  
2.  单机房同步 (数据库之间RTT < 1ms)  
   a. 数据库版本升级 b. 数据表迁移 c. 异步二级索引  
3.  跨机房同步 (比如阿里巴巴国际站就是杭州和美国机房的数据库同不，RTT > 200ms，亮点)  
4.  双向同步  
    a.  避免回环算法  (通用的解决方案，支持大部分关系型数据库) b.  数据一致性算法   (保证双A机房模式下，数据保证最终一致性，亮点)  
5.  文件同步  
    a.  站点镜像  (进行数据复制的同时，复制关联的图片，比如复制产品数据，同时复制产品图片).

-------------------

**DataX**

<https://github.com/alibaba/DataX>

DataX 是阿里巴巴集团内被广泛使用的离线数据同步工具/平台，实现包括 MySQL、Oracle、HDFS、Hive、OceanBase、HBase、OTS、ODPS 等各种异构数据源之间高效的数据同步功能。DataX是一个在异构的数据库/文件系统之间高速交换数据的工具，实现了在任意的数据处理系统(RDBMS/Hdfs/Local filesystem）之间的数据交换，由淘宝数据平台部门完成。 



-----------------

**LVS**

LVS是Linux Virtual Server的简写，意即Linux虚拟服务器，是一个虚拟的服务器集群系统。

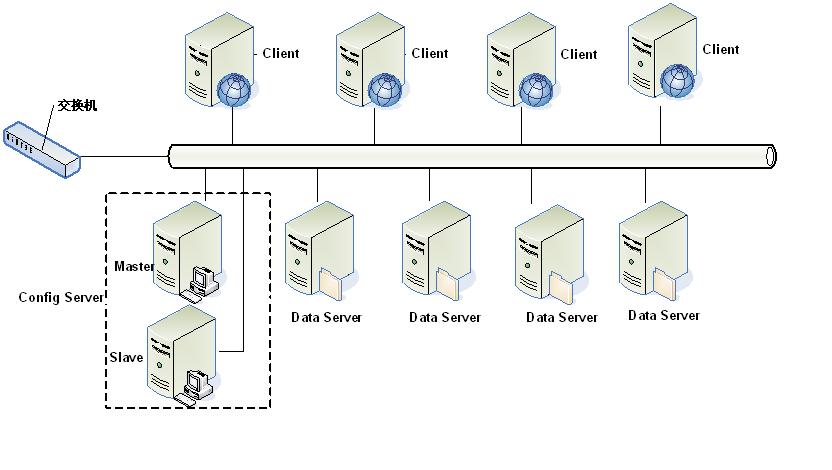
<http://www.linuxvirtualserver.org/>   <https://github.com/alibaba/LVS>

---------------

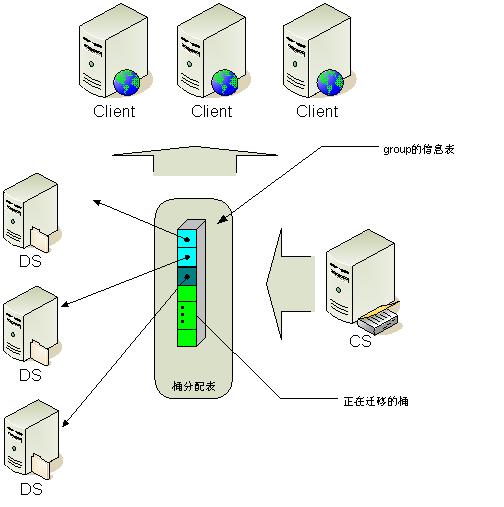
**Tair**

<http://code.taobao.org/p/tair/src/>

Tair是淘宝自主开发的一个分布式key/value存储系统。



tair 是淘宝自己开发的一个分布式 key/value 存储引擎. tair 分为持久化和非持久化两种使用方式. 非持久化的 tair 可以看成是一个分布式缓存. 持久化的 tair 将数据存放于磁盘中. 为了解决磁盘损坏导致数据丢失, tair 可以配置数据的备份数目, tair 自动将一份数据的不同备份放到不同的主机上, 当有主机发生异常, 无法正常提供服务的时候, 其于的备份会继续提供服务.  
tair 的总体结构  
tair 作为一个分布式系统, 是由一个中心控制节点和一系列的服务节点组成. 我们称中心控制节点为config server. 服务节点是data server. config server 负责管理所有的data server, 维护data server的状态信息. data server 对外提供各种数据服务, 并以心跳的形式将自身状况汇报给config server. config server是控制点, 而且是单点, 目前采用一主一备的形式来保证其可靠性. 所有的 data server 地位都是等价的.



tair 的分布采用的是一致性哈希算法, 对于所有的key, 分到Q个桶中, 桶是负载均衡和数据迁移的基本单位. config server 根据一定的策略把每个桶指派到不同的data server上. 因为数据按照key做hash算法, 所以可以认为每个桶中的数据基本是平衡的. 保证了桶分布的均衡性, 就保证了数据分布的均衡性.

-----------------------------------------

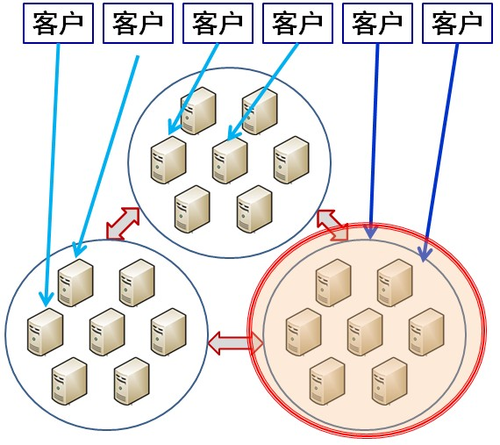
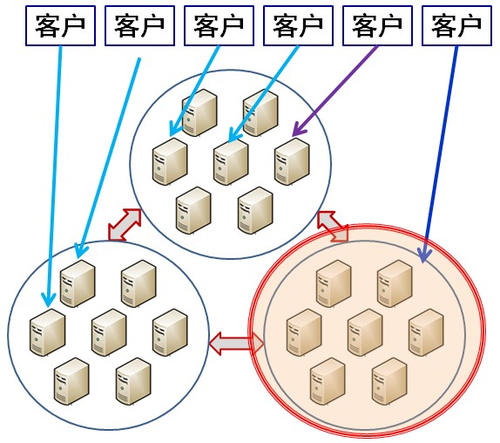
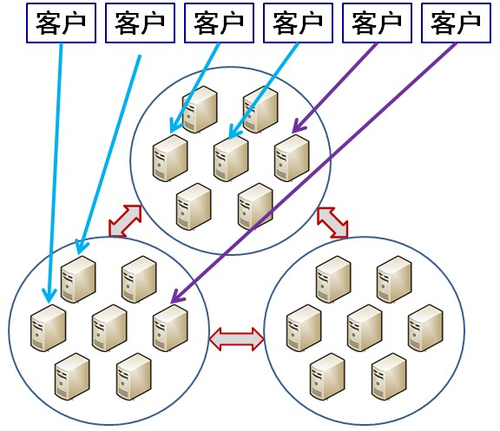
**OceanBase**

OceanBase是阿里巴巴集团研发的可扩展的关系数据库，实现了数千亿条记录、数百TB数据上的跨行跨表事务。

<http://code.taobao.org/p/OceanBase/wiki/index/>

OceanBase的目标是支持数百TB的数据量以及数十万TPS、数百万QPS的访问量。无论是数据量还是访问量，即使采用非常昂贵的小型机甚至是大型机，单台关系数据库系统都无法承受。而经过对在线业务数据的分析，我们发现虽然其数据量十分庞大，例如几十亿条、上百亿条甚至更多记录，但最近一段时间（例如一天）的修改量往往并不多，通常不超过几千万条到几亿条，因此，OceanBase决定采用单独的更新服务器来记录最近一段时间的修改增量，而以前的数据保持不变。每次查询都需要把基准数据和增量数据融合后返回给客户端。  
OceanBase将写事务集中在独立的更新服务器上，可以避免复杂的分布式事务，高效地实现跨行跨表事务。更新服务器上的修改增量通过定期合并操作融合多台基准数据服务器中，从而避免其成为瓶颈，实现了良好的扩展性。

[揭秘阿里服务互联网金融的关系数据库——OceanBase](http://www.csdn.net/article/2015-04-02/2824402)

[[](http://www.csdn.net/article/2015-04-02/2824402)](http://www.csdn.net/article/2015-04-02/2824402)

OceanBase则是“多活”设计，即多个库（3个，5个等）每个都可以有部分读写流量，升级时先把要升级的库的读写流量切走，升级后先进行数据对比，正常后逐步引入读写流量(白名单，1%，5%，10%......)，一切正常并运行一段时间后再升级其他的库。

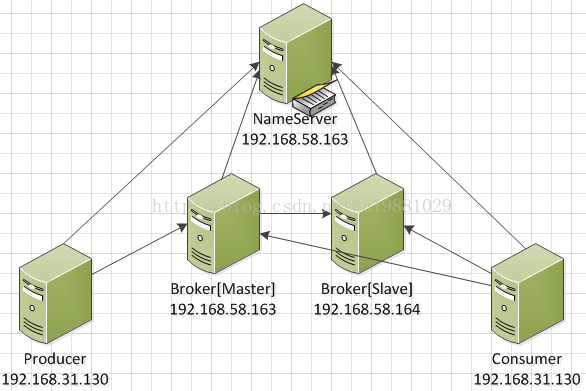
----------------------

**RocketMQ**

<https://github.com/alibaba/RocketMQ>

[阿里RocketMQ Quick Start](http://blog.csdn.net/a19881029/article/details/34446629)

RocketMQ单机支持1万以上的持久化队列，前提是足够的内存、硬盘空间，过期数据数据删除（RocketMQ中的消息队列长度不是无限的，只是足够大的内存+数据定时删除）



[RocketMQ与Kafka对比（18项差异）](https://github.com/alibaba/RocketMQ/wiki/rmq_vs_kafka)【深度比较非常好】

Kafka无限消息堆积，高效的持久化速度吸引了我们，但是同时发现这个消息系统主要定位于日志传输，对于使用在淘宝交易、订单、充值等场景下还有诸多特性不满足，为此我们重新用Java语言编写了RocketMQ，定位于非日志的可靠消息传输（日志场景也OK），目前RocketMQ在阿里集团被广泛应用在订单，交易，充值，流计算，消息推送，日志流式处理，binglog分发等场景。

数据可靠性，性能对比，单机支持的队列数，消息投递实时性，消费失败重试，消息顺序，定时消息，分布式事务消息，消息查询，消息回溯，消费并行度，消息轨迹，开发语言友好性，Broker端消息过滤，消息堆积能力  
  
  
RocketMQ支持异步实时刷盘，同步刷盘，同步复制，异步复制  
Kafka使用异步刷盘方式，异步复制/同步复制  
  
  
Kafka单机写入TPS约在百万条/秒，消息大小10个字节  
RocketMQ单机写入TPS单实例约7万条/秒，单机部署3个Broker，可以跑到最高12万条/秒，消息大小10个字节。  
  
  
Kafka单机超过64个队列/分区，Load会发生明显的飙高现象，队列越多，load越高，发送消息响应时间变长。Kafka分区数无法过多的问题  
RocketMQ单机支持最高5万个队列，Load不会发生明显变化  
  
  
Kafka使用短轮询方式，实时性取决于轮询间隔时间，0.8以后版本支持长轮询。  
RocketMQ使用长轮询，同Push方式实时性一致，消息的投递延时通常在几个毫秒。  
  
  
Kafka消费失败不支持重试。  
RocketMQ消费失败支持定时重试，每次重试间隔时间顺延  
例如充值类应用，当前时刻调用运营商网关，充值失败，可能是对方压力过多，稍后再调用就会成功，如支付宝到银行扣款也是类似需求。  
这里的重试需要可靠的重试，即失败重试的消息不因为Consumer宕机导致丢失。  
  
  
消息查询对于定位消息丢失问题非常有帮助，例如某个订单处理失败，是消息没收到还是收到处理出错了。对于定位消息丢失问题非常有帮助，例如某个订单处理失败，是消息没收到还是收到处理出错了。  
  
  
消息回溯最典型业务场景如consumer做订单分析，但是由于程序逻辑或者依赖的系统发生故障等原因，导致今天消费的消息全部无效，需要重新从昨天零点开始消费，那么以时间为起点的消息重放功能对于业务非常有帮助。

-----------------------

**Tsar**

tsar是淘宝自己开发的一个采集工具，主要用来收集服务器的系统信息（如cpu，io，mem，tcp等），以及应用数据（如squid haproxy nginx等）。  
收集到的数据存储在磁盘上，可以随时查询历史信息，输出方式灵活多样，另外支持将数据存储到mysql中，也可以将数据发送到nagios报警服务器。

<http://code.taobao.org/p/tsar/wiki/index/>

<https://github.com/alibaba/tsar>

[淘宝开发的系统监控工具 Tsar 开源](http://blog.jobbole.com/38249/)

-----------------------

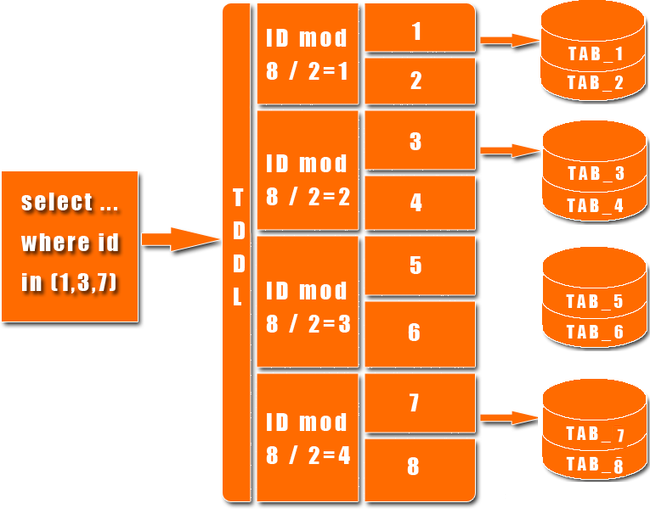
**TDDL**

淘宝分布式数据层

<https://github.com/alibaba/tb_tddl>

<http://www.oschina.net/p/tddl>

数据到了百亿级别的时候，任何一个库都无法存放了，于是分成2个、4个、8个、16个、32个……直到1024个、2048个。  
这时候，数据查询的中间件对上层来说，必须像查询一个数据库一样来查询数据，还要像查询一个数据库一样快（每条查询在几毫秒内完成），TDDL就承担了这样一 个工作。在外面有些系统也用DAL（数据访问层） 这个概念来命名这个中间件。



[剖析淘宝TDDL（TAOBAO DISTRIBUTE DATA LAYER）](http://www.tuicool.com/articles/nmeuu2)

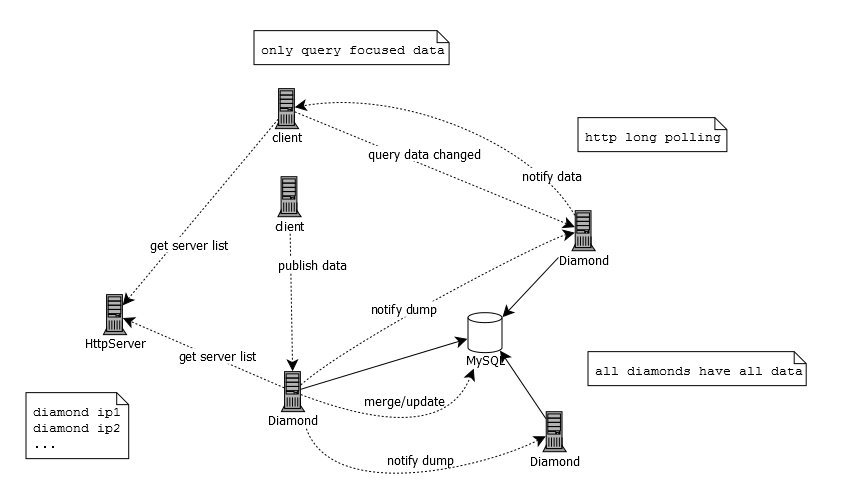
TDDL 必须要依赖 diamond 配置中心（ diamond 是淘宝内部使用的一个管理持久配置的系统，目前淘宝内部绝大多数系统的配置）。

-----------------------

**Diamond**

[淘宝分布式配置管理服务Diamond](http://blog.csdn.net/kevinlynx/article/details/40017109/)  原文地址：<http://codemacro.com/2014/10/12/diamond/>

zookeeper的一种应用就是分布式配置管理(基于ZooKeeper的配置信息存储方案的设计与实现)。百度也有类似的实现：disconf。  
Diamond则是淘宝开源的一种分布式配置管理服务的实现。Diamond本质上是一个Java写的Web应用，其对外提供接口都是基于HTTP协议的，在阅读代码时可以从实现各个接口的controller入手。



Diamond中的数据是简单的key-value结构。应用方订阅数据则是基于key来订阅，未订阅的数据当然不会被推送。数据从类型上又划分为聚合和非聚合。因为数据推送者可能很多，在整个分布式环境中，可能有多个推送者在推送相同key的数据，这些数据如果是聚合的，那么所有这些推送者推送的数据会被合并在一起；反之如果是非聚合的，则会出现覆盖。  
  
  
数据的来源可能是人工通过管理端录入，也可能是其他服务通过配置管理服务的推送接口自动录入。  
Diamond服务是一个集群，是一个去除了单点的协作集群。  
  
  
Diamond服务集群每一个实例都可以对外完整地提供服务，那么意味着每个实例上都有整个集群维护的数据。  
虽然Diamond去除了单点问题，不过问题都下降到了mysql上。但由于其作为配置管理的定位，其数据量就mysql的应用而言算小的了，所以可以一定程度上保证整个服务的可用性。  
  
  
由于Diamond服务器没有master，任何一个实例都可以读写数据，那么针对同一个key的数据则可能面临冲突。这里应该是通过mysql来保证数据的一致性。每一次客户端请求写数据时，Diamond都将写请求投递给mysql，然后通知集群内所有Diamond实例（包括自己）从mysql拉取数据。当然，拉取数据则可能不是每一次写入都能拉出来，也就是最终一致性。  
  
  
Diamond中没有把数据放入内存，但会放到本地文件。对于客户端的读操作而言，则是直接返回本地文件里的数据。

----------------------

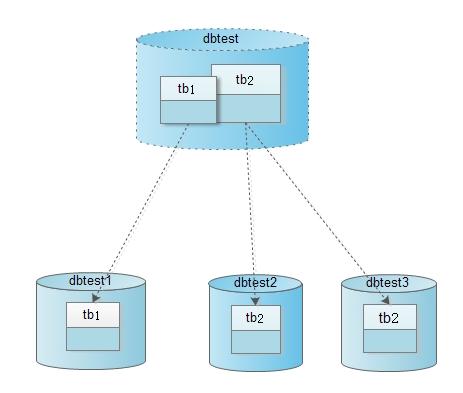
**Cobar**

基于MySQL的分布式数据库服务中间件

<https://github.com/alibaba/cobar>

Cobar是提供关系型数据库（MySQL）分布式服务的中间件，它可以让传统的数据库得到良好的线性扩展，并看上去还是一个数据库,对应用保持透明。  
  
  
产品在阿里巴巴稳定运行3年以上。  
接管了3000+个MySQL数据库的schema。  
集群日处理在线SQL请求50亿次以上。  
集群日处理在线数据流量TB级别以上。

[关系型数据的分布式处理系统 Cobar](http://www.oschina.net/p/cobar)

Cobar是关系型数据的分布式处理系统，它可以在分布式的环境下像传统数据库一样为您提供海量数据服务。以下是快速启动场景：  
  
系统对外提供的数据库名是dbtest,并且其中有两张表tb1和tb2。  
tb1表的数据被映射到物理数据库dbtest1的tb1上。  
tb2表的一部分数据被映射到物理数据库dbtest2的tb2上，另外一部分数据被映射到物理数据库dbtest3的tb2上。  


[Cobar使用文档（可用作MySQL大型集群解决方案）](http://my.oschina.net/ydsakyclguozi/blog/374564)

----------------------

**beatles**

<https://github.com/cenwenchu/beatles>

小规模即时流数据分析集群，分析规则抽象于SQL，计算规则采用MapReduce模式。

--------------------------------

**综合**

[淘宝开放平台技术历程](http://blog.csdn.net/cenwenchu79)

=========================

参考：<http://www.oschina.net/project/alibaba>

**三、腾讯开源项目**

腾讯开源项目列表全部在腾讯公司的网页上：http://code.tencent.com/

**四，其他公司**

百度，网易，京东等公司都有开源项目，大家在网上搜索即可，在这就不一一列举了。