## Maatkit

Maatkit is a toolkit for users, developers, and administrators of open-source databases. It is high-quality, formally tested software that is well documented and has an active open-source developer community. It has some unusual goals: all tools are standalone, with minimal dependencies and no need for installation. You can simply download them and use them. Documentation is embedded in the tools themselves. There are many more nice attributes, too.

MySQL Utilities provides a collection of command-line utilities that are used for maintaining and administering MySQL servers, including:

Admin Utilities (Clone, Copy, Compare, Diff, Export, Import, User Management)

Replication Utilities (Setup, Configuration, Verification)

General Utilities (Disk Usage, Redundant Indexes, Manage Meta & Audit Data)

And more...............

## Librarian-puppet

Librarian-puppet is a bundler for your puppet infrastructure. You can use librarian-puppet to manage the puppet modules your infrastructure depends on, whether the modules come from the Puppet Forge, Git repositories or just a path.

## Librarian-Chef

Librarian-Chef is a bundler for your Chef-based infrastructure repositories.

It is a tool that helps you manage the cookbooks that your chef-repo depends on. Here are some more details.

Librarian-Chef is a bundler for infrastructure repositories using Chef. You can use Librarian-Chef to resolve your infrastructure's cookbook dependencies, fetch them, and install them into your infrastructure repository.

## Bower

Bower works by fetching and installing packages from all over, taking care of hunting, finding, downloading, and saving the stuff you’re looking for. Bower keeps track of these packages in a manifest file, bower.json. How you use packages is up to you. Bower provides hooks to facilitate using packages in your tools and workflows.

## CoffeeScript

CoffeeScript is a little language that compiles into JavaScript. Underneath that awkward Java-esque patina, JavaScript has always had a gorgeous heart. CoffeeScript is an attempt to expose the good parts of JavaScript in a simple way.

## AWStats

AWStats是在Sourceforge上发展很快的一个基于Perl的WEB日志分析工具。相对于另外一个非常优秀的开放源代码的日志分析工具Webalizer，AWStats的优势在于：

1.界面友好：可以根据浏览器直接调用相应语言界面（有简体中文版）

2.基于Perl：并且很好的解决了跨平台问题，系统本身可以运行在GNU/Linux上或Windows上（安装了ActivePerl后）；分析的日志直接支持Apache格式 (combined)和IIS格式(需要修改)。Webalizer虽然也有Windows平台版，但目前已经缺乏 维护；AWStats完全可以实现用一套系统完成对自身站点不同WEB服务器：GNU/Linux/Apache和Windows/IIS服务器的统一统计

## ntpd、ntpdate的区别

使用之前得弄清楚一个问题，ntpd与ntpdate在更新时间时有什么区别。ntpd不仅仅是时间同步服务器，他还可以做客户端与标准时间服务器进行同步时间，而且是平滑同步，并非ntpdate立即同步，在生产环境中慎用ntpdate，也正如此两者不可同时运行。

时钟的跃变，对于某些程序会导致很严重的问题。许多应用程序依赖连续的时钟——毕竟，这是一项常见的假定，即，取得的时间是线性的，一些操作，例如数据库事务，通常会地依赖这样的事实：时间不会往回跳跃。不幸的是，ntpdate调整时间的方式就是我们所说的”跃变“：在获得一个时间之后，ntpdate使用settimeofday(2)设置系统时间，这有几个非常明显的问题：

第一，这样做不安全。ntpdate的设置依赖于ntp服务器的安全性，攻击者可以利用一些软件设计上的缺陷，拿下ntp服务器并令与其同步的服务器执行某些消耗性的任务。由于ntpdate采用的方式是跳变，跟随它的服务器无法知道是否发生了异常（时间不一样的时候，唯一的办法是以服务器为准）。

第二，这样做不精确。一旦ntp服务器宕机，跟随它的服务器也就会无法同步时间。与此不同，ntpd不仅能够校准计算机的时间，而且能够校准计算机的时钟。

第三，这样做不够优雅。由于是跳变，而不是使时间变快或变慢，依赖时序的程序会出错（例如，如果ntpdate发现你的时间快了，则可能会经历两个相同的时刻，对某些应用而言，这是致命的）。

因而，唯一一个可以令时间发生跳变的点，是计算机刚刚启动，但还没有启动很多服务的那个时候。其余的时候，理想的做法是使用ntpd来校准时钟，而不是调整计算机时钟上的时间。

NTPD 在和时间服务器的同步过程中，会把 BIOS 计时器的振荡频率偏差——或者说 Local Clock 的自然漂移(drift)——记录下来。这样即使网络有问题，本机仍然能维持一个相当精确的走时。

## FFmpeg

FFmpeg是一套可以用来记录、转换数字音频、视频，并能将其转化为流的开源计算机程序。它包括了目前领先的音/视频编码库libavcodec。 FFmpeg是在Linux下开发出来的，但它可以在包括Windows在内的大多数操作系统中编译。这个项目是由Fabrice Bellard发起的，现在由Michael Niedermayer主持。可以轻易地实现多种视频格式之间的相互转换，例如可以将摄录下的视频avi等转成现在视频网站所采用的flv格式。

## nload

nload is a console application which monitors network traffic and bandwidth usage in real time. It visualizes the in- and outgoing traffic using two graphs and provides additional info like total amount of transfered data and min/max network usage.

## network shares

network shares, network file systems, and SANs i

## Atomikos

Atomikos TransactionsEssentials 是一个为Java平台提供增值服务的并且开源类事务管理器

XA协议采用两阶段提交方式来管理分布式事务。XA接口提供资源管理器与事务管理器之间进行通信的标准接口。XA协议包括两套函数，以xa\_开头的及以ax\_开头的

## MySQL-mmm

MySQL-mmm： MySQL Master-Master 架構常被用在 SQL query 相依性低的情況，像是 counter 常使用的 INSERT INTO ... ON DUPLICATE KEY UPDATE a = a + 1 不會因為 out-of-order 而造成問題。而 MySQL MMM 算是其中一套寫得比較好的 MySQL Master-Master 架構管理工具。

## Percona XtraDB Cluster

Percona XtraDB Cluster is an active/active high availability and high scalability open source solution for MySQL® clustering. It integrates Percona Server and Percona XtraBackup with the Galera library of MySQL high availability solutions in a single package which enables you to create a cost-effective MySQL high availability cluster - See more at: http://www.percona.com/software/percona-xtradb-cluster#sthash.rIiITGXs.dpuf

## Galera Cluster

Galera Cluster for MySQL is a true Multimaster Cluster based on synchronous replication. It is an easy-to-use, high-availability solution, which provides high system up-time, no data loss and scalability for future growth.

## MariaDB Galera

MariaDB Galera Cluster is a synchronous multi-master cluster for MariaDB. It is available on Linux only, and only supports the XtraDB/InnoDB storage engines (although there is experimental support for MyISAM

## Gearman vs RabbitMQ

I would say that Gearman is better for queuing "jobs" and RabbitMQ is better for queuing "data". Of course, they are both really the same thing, but the way it works out for me is that if you are trying to "fan out" work to be done, and the workers can work independently, Gearman is the better way to do it. But if you are trying to feed data from a lot of sources down into fewer data consumers, RabbitMQ is the better solution.

The history of RabbitMQ, as something that allowed Twitter to take bursty loads of messages, and feed them into crusty old SMS gateways that could keep only one connection open, were rate limited, and didnt have retries, is illustrative of the kind of problems that RabbitMQ is good at solving.

## Buildroot

Buildroot is a set of Makefiles and patches that makes it easy to generate a complete embedded Linux system. Buildroot can generate any or all of a cross-compilation toolchain, a root filesystem, a kernel image and a bootloader image. Buildroot is useful mainly for people working with small or embedded systems, using various CPU architectures (x86, ARM, MIPS, PowerPC, etc.) : it automates the building process of your embedded system and eases the cross-compilation process.

## Ansible

Ansible 提供一种最简单的方式用于发布、管理和编排计算机系统的工具，你可在数分钟内搞定。

Ansible 是一个模型驱动的配置管理器，支持多节点发布、远程任务执行。默认使用 SSH 进行远程连接。无需在被管理节点上安装附加软件，可使用各种编程语言进行扩展。

## Dokku

Dokku: Docker powered mini-Heroku. The smallest PaaS implementation you've ever seen.

## Hipache

Hipache (pronounce hɪ'pætʃɪ) is a distributed proxy designed to route high volumes of http and websocket traffic to unusually large numbers of virtual hosts, in a highly dynamic topology where backends are added and removed several times per second. It is particularly well-suited for PaaS (platform-as-a-service) and other environments that are both business-critical and multi-tenant.

Hipache was originally developed at dotCloud, a popular platform-as-a-service, to replace its first-generation routing layer based on a heavily instrumented nginx deployment. It currently serves production traffic for tens of thousands of applications hosted on dotCloud. Hipache is based on the node-http-proxy library.

Distributed systems and service discovery are also interesting to me, and I am working to learn about new implementations such as CoreOS' etcd or Hashicorp's Consul, among others

## OpenStack Swift

OpenStack Swift 最初是由 Rackspace 公司开发的高可用分布式对象存储服务，并于 2010 年贡献给 OpenStack 开源社区作为其最初的核心子项目之一，为其 Nova 子项目提供虚机镜像存储服务。Swift 构筑在比较便宜的标准硬件存储基础设施之上，无需采用 RAID（磁盘冗余阵列），通过在软件层面引入一致性散列技术和数据冗余性，牺牲一定程度的数据一致性来达到高可用性和可伸缩性，支持多租户模式、容器和对象读写操作，适合解决互联网的应用场景下非结构化数据存储问题。

## Consul

Consul, a solution for service discovery and configuration. Consul is completely distributed, highly available, and scales to thousands of nodes and services across multiple datacenters.

## CoreOS' etcd

CoreOS' etcd：A highly-available key value store for shared configuration and service discovery. etcd is inspired by Apache ZooKeeper and doozer, with a focus on being:

Simple: curl'able user facing API (HTTP+JSON)

Secure: optional SSL client cert authentication

Fast: benchmarked 1000s of writes/s per instance

Reliable: properly distributed using Raft

简单的做法是添加启动脚本到/etc/init.d/rc.local等，让openerp 随系统启动而运行。此类方法只在系统启动时运行，但万一程序在运行中崩溃，您可能要等到用户发现不能使用了，才去重启服务器。下面请出今天的主角： supervisor (http://supervisord.org/)

## Supervisor

Supervisor is a client/server system that allows its users to monitor and control a number of processes on UNIX-like operating systems.

## Kubernetes:

建于Docker之上的Kubernetes可以构建一个容器的调度服务，其目的是让用户透过Kubernetes集群来进行云端容器集群的管理，而无需用户进行复杂的设置工作。系统会自动选取合适的工作节点来执行具体的容器集群调度处理工作。其核心概念是Container Pod（容器仓）。一个Pod是有一组工作于同一物理工作节点的容器构成的。这些组容器拥有相同的网络命名空间/IP以及存储配额，可以根据实际情况对每一个Pod进行端口映射。此外，Kubernetes工作节点会由主系统进行管理，节点包含了能够运行Docker容器所用到的服务；可以说是谷歌云计算服务器Compute Engine的一个扩展

## Zookeeper

Zookeeper 分布式服务框架是 Apache Hadoop 的一个子项目，它主要是用来解决分布式应用中经常遇到的一些数据管理问题，如：统一命名服务、状态同步服务、集群管理、分布式应用配置项的管理等的一个子项目，它主要是用来解决分布式应用中经常遇到的一些数据管理问题，如：统一命名服务、状态同步服务、集群管理、分布式应用配置项的管理等

## Doozer

Doozer 是高可用的，完整一致性的用于小量、极端重要的数据的存储。当数据变化时，它立刻通知接入的客户端（不缓存），对于那些很少更新，但是希望更新发生时实时性高的客户端来说是非常理想的。Doozer 对于名字服务、主数据库选取和多个设备之间的配置数据同步很适合。了解什么时候应该使用它？下面，有详细的信息。

## OpenNebula

OpenNebula是一款为云计算而打造的开源工具箱。它允许你与Xen, KVM或VMware ESX一起建立和管理私有云，同时还提供Deltacloud适配器与Amazon EC2相配合来管理混合云。除了像Amazon一样的商业云服务提供商，在不同OpenNebula实例上运行私有云的Amazon合作伙伴也同样可以作为远程云服务供应商。

## noVNC

noVNC is a HTML5 VNC client that runs well in any modern browser including mobile browsers (iPhone/iPad and Android).

Many companies/projects have integrated noVNC including Ganeti Web Manager, OpenStack, OpenNebula, and LibVNCServer. See the Projects and Companies wiki page for a more complete list with additional info and links.

## Jinja2

Jinja2 is a full featured template engine for Python. It has full unicode support, an optional integrated sandboxed execution environment, widely used and BSD licensed.

## pipsi

pipsi is a wrapper around virtualenv and pip which installs scripts provided by python packages into separate virtualenvs to shield them from your system and each other.

## Pygments 代码高亮

## Rust

Rust 是 Mozilla 的一个新的编程语言，由web语言的领军人物Brendan Eich（js之父），Dave Herman以及Mozilla公司的Graydon Hoare 合力开发。

创建这个新语言的目的是为了解决一个很顽疾的问题：软件的演进速度大大低于硬件的演进，软件在语言级别上无法真正利用多核计算带来的性能提升。Rust是针对多核体系提出的语言，并且吸收一些其他动态语言的重要特性，比如不需要管理内存，比如不会出现Null指针等等。

## Mahout

Mahout为推荐引擎提供了一些可扩展的机器学习领域的经典算法实现,可以使开发人员更为快捷的创建智能应用程序。

## Spark

Spark 是一种前景无限的大数据分析解决方案，专为使用内存处理的高效集群计算而开发。其目标使用模型包括整合了迭代式算法的模型（也就是说，能够受益于将数据保留在内存之中，而非将其推送到杨恩较高的文件系统的模型）

## hadoop

全量数据处理使用的大多是鼎鼎大名的hadoop或者hive，作为一个批处理系统，hadoop以其吞吐量大、自动容错等优点，在海量数据处理上得到了广泛的使用。但是，hadoop不擅长实时计算，因为它天然就是为批处理而生的，这也是业界一致的共识。否则最近这两年也不会有s4,storm,puma这些实时计算系统如雨后春笋般冒出来啦。先抛开s4,storm,puma这些系统不谈，我们首先来看一下，如果让我们自己设计一个实时计算系统，我们要解决哪些问题。

## Storm

http://storm.apache.org/

是一个开源的分布式实时计算系统，可以简单、可靠的处理大量的数据流。Storm有很多使用场景：如实时分析，在线机器学习，持续计算，分布式RPC，ETL等等。Storm支持水平扩展，具有高容错性，保证每个消息都会得到处理，而且处理速度很快（在一个小集群中，每个结点每秒可以处理数以百万计的消息）。Storm的部署和运维都很便捷，而且更为重要的是可以使用任意编程语言来开发应用。

## Deis

Deis (pronounced DAY-iss) is an open source PaaS that makes it easy to deploy and manage applications on your own servers. Deis builds upon Docker and CoreOS to provide a lightweight PaaS with a Heroku-inspired workflow.

## Flow-chart

Cacoo https://cacoo.com/diagrams/4j7Sc1VluxrVulvG/edit

vs gliffy https://www.gliffy.com/go/html5/launch?app=1b5094b0-6042-11e2-bcfd-0800200c9a66

vs https://www.draw.io/ [online free]

vs Pencil http://pencil.evolus.vn/ [free]

vs http://www.yworks.com/

vs Visual Paradigm http://www.visual-paradigm.com/download/

vs http://creately.com/

vs dia [free]

vs kivio [free]

## ylint

ylint is a tool that checks for errors in Python code, tries to enforce a coding standard and looks for bad code smells. This is similar but nevertheless different from what pychecker provides, especially since pychecker explicitly does not bother with coding style. The default coding style used by Pylint is close to PEP 008 (aka Guido’s style guide). For more information about code smells, refer to Martin Fowler’s refactoring book.

Pylint will display a number of messages as it analyzes the code, as well as some statistics about the number of warnings and errors found in different files. The messages are classified under various categories such as errors and warnings (more below). If you run Pylint twice, it will display the statistics from the previous run together with the ones from the current run, so that you can see if the code has improved or not.

docker-registry

docker-hub manager

VAGRANT\_LOG=info vagrant provision

Hadoop、Spark、Storm

Webmin or ISPconfig or kloxo - VPS Hosting

zabbix vs nagios

nodejs vs iojs

vagrant vs packer

docker vs coreos[rocker]

puppet vs chef vs ansible vs saltstack

gearman vs rabbitmq vs celery

composer vs npm vs cpan vs Carton(perl) vs bundle(ruby) vs lein(lua) vs sbt(scala) vs rbenv(ruby) vs gvm(groovy) vs virtualenv(python) vs vagrant vs docker vs packer(www.packer.io) vs veewee vs bento vs docker vs rocker vs maven vs grandle vs phpmaven vs perlmaven

docker:

1.nsenter

https://github.com/jpetazzo/nsenter

2.docker-enter

https://github.com/Pithikos/docker-enter

windows:

chocolatey

scoop

chocolatey

ansible vs saltstack vs puppet vs chef