



.....

# MySQL高可用

.....

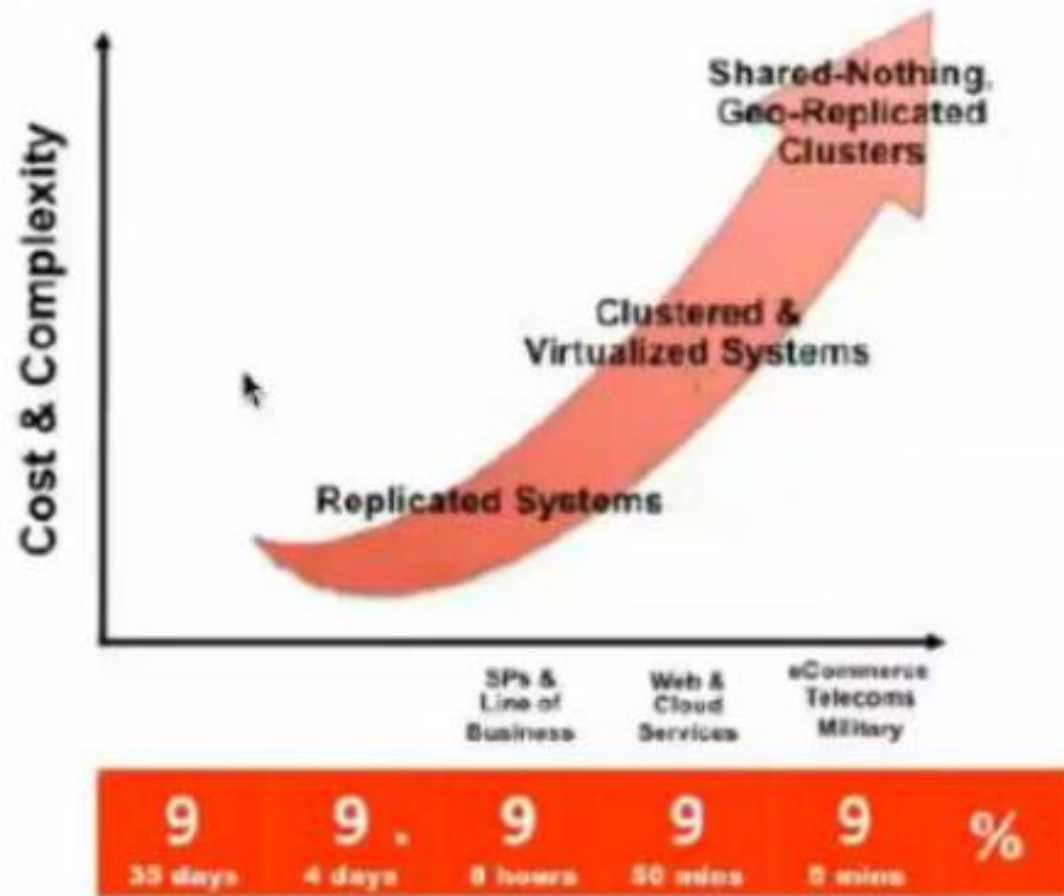
黄钧阳  
2016年11月21日

# 什么是高可用

高可用性**H.A. (High Availability)**指的是通过尽量缩短因日常维护操作（计划）和突发的系统崩溃（非计划）所导致的停机时间，以提高系统和应用的可用性。它与被认为是不间断操作的容错技术有所不同。**HA**系统是目前企业防止核心计算机系统因故障停机的最有效手段。

# 高可用与成本的关系

## 高可用与成本



# SLA宕机时长

Percentile target	Max downtime per year
90%	36 days
99%	3.65 days
99.5%	1.83 days
99.9%	8.76 hours
99.99%	52.56 minutes
99.999%	5.26 minutes
99.9999%	31.5 seconds

# 高可用级别

	Level of availability
Simple replication	98 to 99.9+ %
Master-Master with MMM manager	99%
Heartbeat/SAN (depends on SAN)	99.5% to 99.9%
Heartbeat/DRBD	99.9%
NDB Cluster	99.999%

# 导致宕机的原因

- ❖ 1. 硬件故障
- ❖ 2. 软件或MySQL出现bug
- ❖ 3. 意外的错误操作
- ❖ 4. 各种攻击

宕机原因很多，但**HA**不能解决所有问题！

# 如何实现高可用

- ❖ **1.服务器高可用**
  - ❖ **2.网络高可用**
  - ❖ **3.存储高可用**
-

# HA的两种方式

**1.Active/Active(primary/primary)**

**2.Active/Passive(primary/secondary)**



# 几种HA方案介绍

**1.MMM**

**2.LVS+Keepived+M/M**

**3.Heartbeat+DRBD**

**4.PXC**

**5.MHA**

**6.NDB Cluster**

**7.中间层实现**

---

# MMM介绍

**MMM (Master-Master replication manager for MySQL)** 是一套支持双主故障切换和双主日常管理的脚本程序。**MMM**使用**Perl**语言开发，主要用来监控和管理**MySQL Master-Master**（双主）复制，虽然叫做双主复制，但是业务上同一时刻只允许对一个主进行写入，另一台备选主上提供部分读服务，以加速在主主切换时刻备选主的预热，可以说**MMM**这套脚本程序一方面实现了故障切换的功能，另一方面其内部附加的工具脚本也可以实现多个**slave**的**read**负载均衡。

---

# MMM介绍

**MMM**提供了自动和手动两种方式移除一组服务器中复制延迟较高的服务器的虚拟**ip**，同时它还可以备份数据，实现两节点之间的数据同步等。由于**MMM**无法完全的保证数据一致性，所以**MMM**适用于对数据的一致性要求不是很高，但是又想最大程度的保证业务可用性的场景。对于那些对数据的一致性要求很高的业务，非常不建议采用**MMM**这种高可用架构。

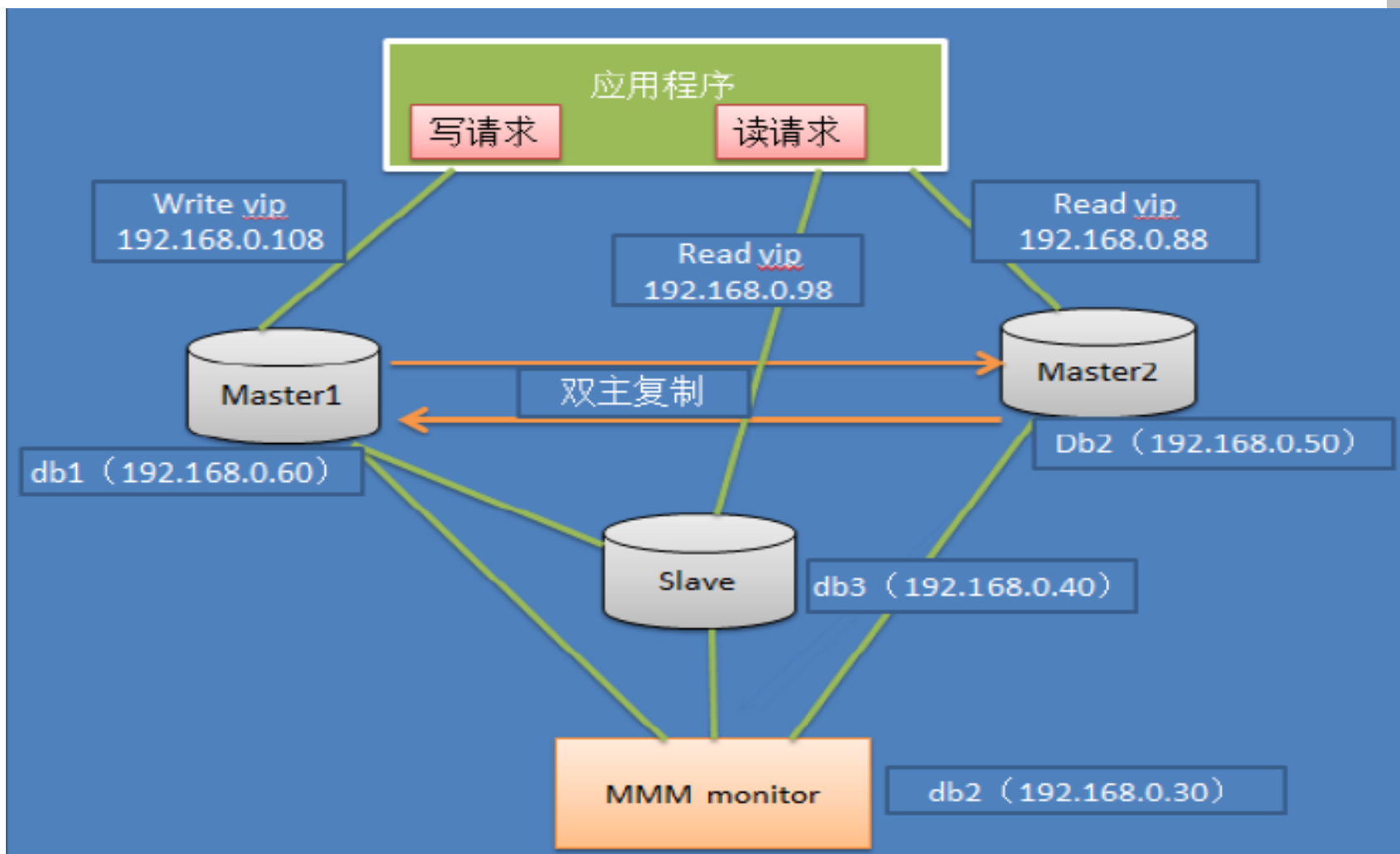
**MMM**项目来自 **Google**:

**<http://code.google.com/p/mysql-master-master>**

官方网站为: **<http://mysql-mmm.org>**

---

# MMM架构



# 安装和配置

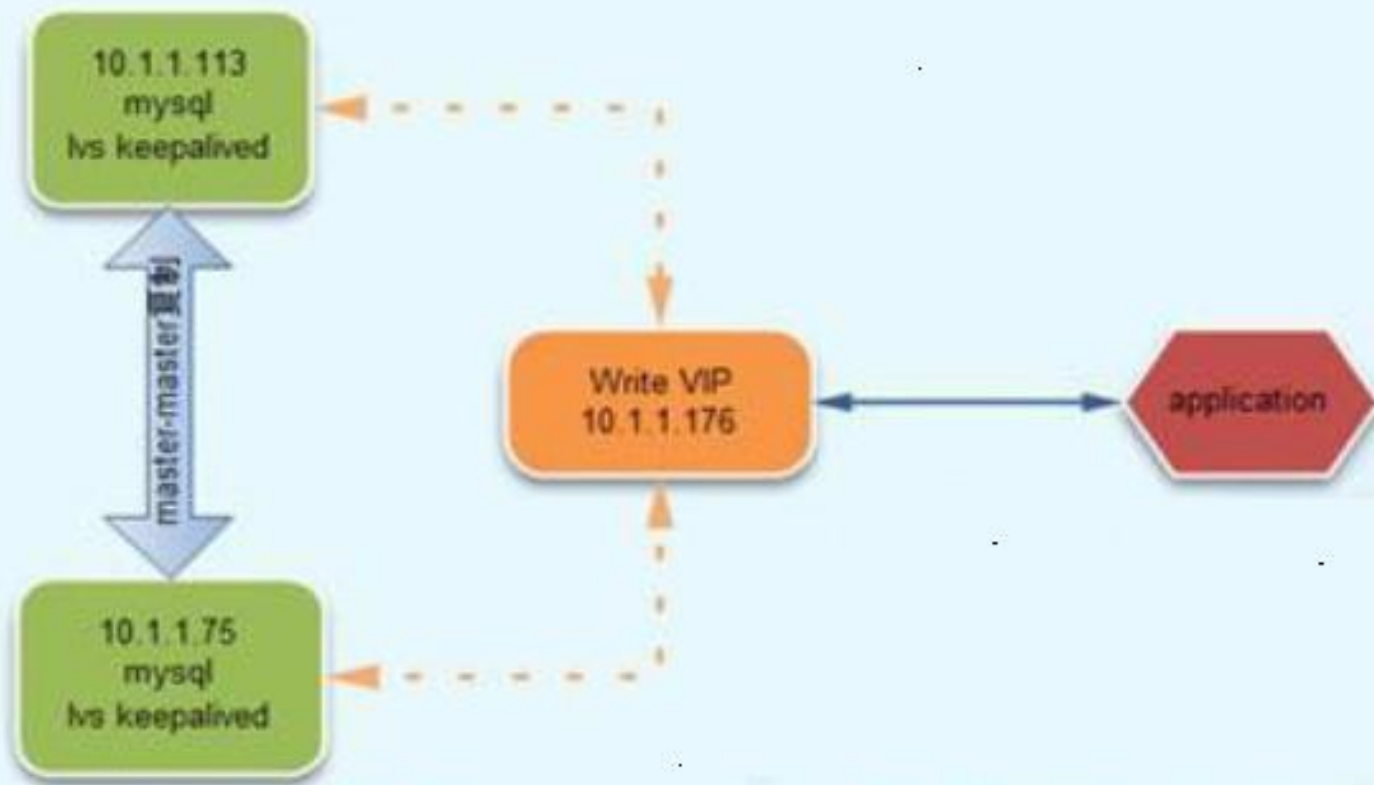
- ❖ <http://mysql-mmm.org/mmm2:guide>
- ❖ 见辅助资料 《MySQL高可用方案之MMM》

## LVS方案介绍

**Lvs+keepalived**作为目前比较流行的高可用解决方案，**lvs**提供负载均衡，**keepalived**作为故障转移，提高系统的可用性。但是一般的**MySQL**高可用为了实现**MySQL**数据的一致性，一般都是采用单点写入，本方案采用**keepalived**中的**sorry\_server**来实现写入数据库为单点的需求。本方案实现的功能是当网络有问题、**MySQL**有问题、服务器宕机、**keepalived**服务停止后，服务器能自动跳转到备用机，当主服务器服务启动起来后会自动切换回来

# LVS方案架构

MM+LVS+KEEPALIVED单写入高可用方案



# LVS方案安装和配置

见辅助资料 《MySQL高可用方案之LVS》

---



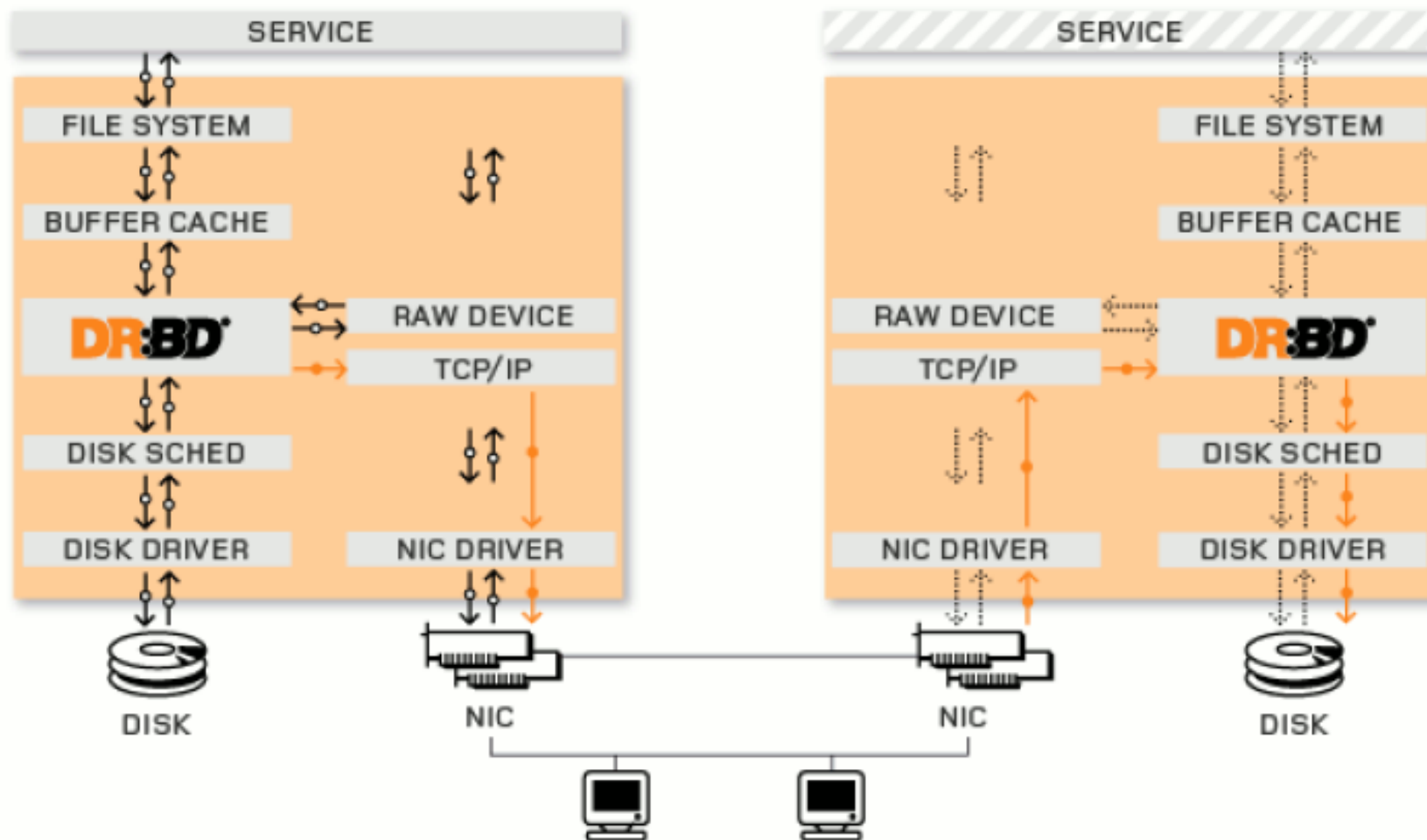
# heartbeat+DRBD介绍

## 什么是DRBD?

**Distributed Replicated Block Device(DRBD)**是由内核模块和相关脚本而构成，用以构建高可用性的集群。其实现方式是通过网络来镜像整个设备。您可以把它看作是一种网络**RAID1**。

**DRBD**负责接收数据，把数据写到本地磁盘，然后发送给另一个主机。另一个主机再将数据存到自己的磁盘中。其他所需的组件有集群成员服务，如**TurboHA** 或 心跳连接，以及一些能在块设备上运行的应用程序。

# heartbeat+DRBD介绍

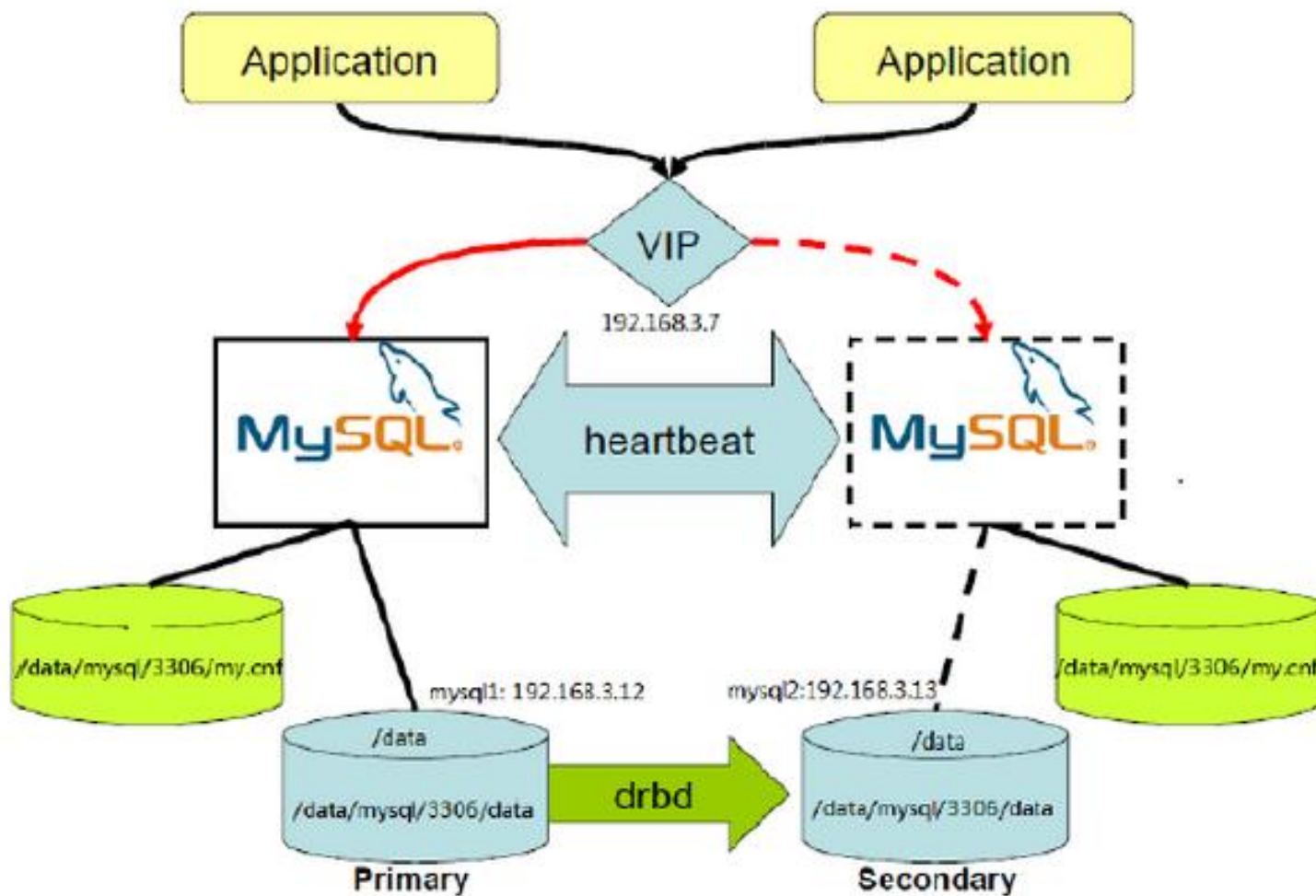


# heartbeat+DRBD介绍

**heartbeat**是**Linux-HA**的全称是**High-Availability Linux**的一个开源项目，这个开源项目的目标是：通过社区开发者的共同努力，提供一个增强**linux**可靠性（**reliability**）、可用性（**availability**）和可服务性（**serviceability**）（**RAS**）的群集解决方案。其中**Heartbeat**就是**Linux-HA**项目中的一个组件，也是目前开源**HA**项目中最成功的一个例子，它提供了所有 **HA** 软件所需要的基本功能，比如心跳检测和资源接管、监测群集中的系统服务、在群集中的节点间转移共享 **IP** 地址的所有者等，自**1999**年开始到现在，**Heartbeat**在行业内得到了广泛的应用，也发行了很多的版本，可以从**Linux-HA**的官方网站[www.linux-ha.org](http://www.linux-ha.org)下载到**Heartbeat**的最新版本。

---

# heartbeat+DRBD架构



# heartbeat+DRBD安装和配置

见辅助材料 《MySQL高可用方案之  
**heartbeat+DRBD**》

---

# 高可用方案-PXC

## ❖ Percona Xtradb Cluster介绍

**Percona XtraDB Cluster** 听起来像是某种新概念的**Cluster**，但其实它就是**Percona** 公司把自己的**XtraDB**（**InnoDB**的加强版）与**CodeShip**的**Galera Cluster**合体的发行版，与**SkySQL**的**MariaDB Galera Cluster**类似。

**Percona XtraDB Cluster** 的好处是集成了用于检查**Cluster**健康状态的脚本和服务，可以更方便的与**HAProxy**配合组成完整的负载均衡系统。

# Percona Xtradb Cluster介绍

**Percona XtraDB Cluster**是针对**MySQL**用户的高可用性和扩展性解决方案，基于**Percona Server**。其包括了**Write Set REplication**补丁，使用**Galera 2.0**库，这是一个针对事务性应用程序的同步多主机复制插件。

**Percona Server**是**MySQL**的改进版本，使用 **XtraDB** 存储引擎，在功能和性能上较 **MySQL** 有着很显著的提升，如提升了在高负载情况下的 **InnoDB** 的性能，为 **DBA** 提供了一些非常有用的性能诊断工具，另外有更多的参数和命令来控制服务器行为。

---

# Percona Xtradb Cluster介绍

**Percona XtraDB Cluster**提供了：

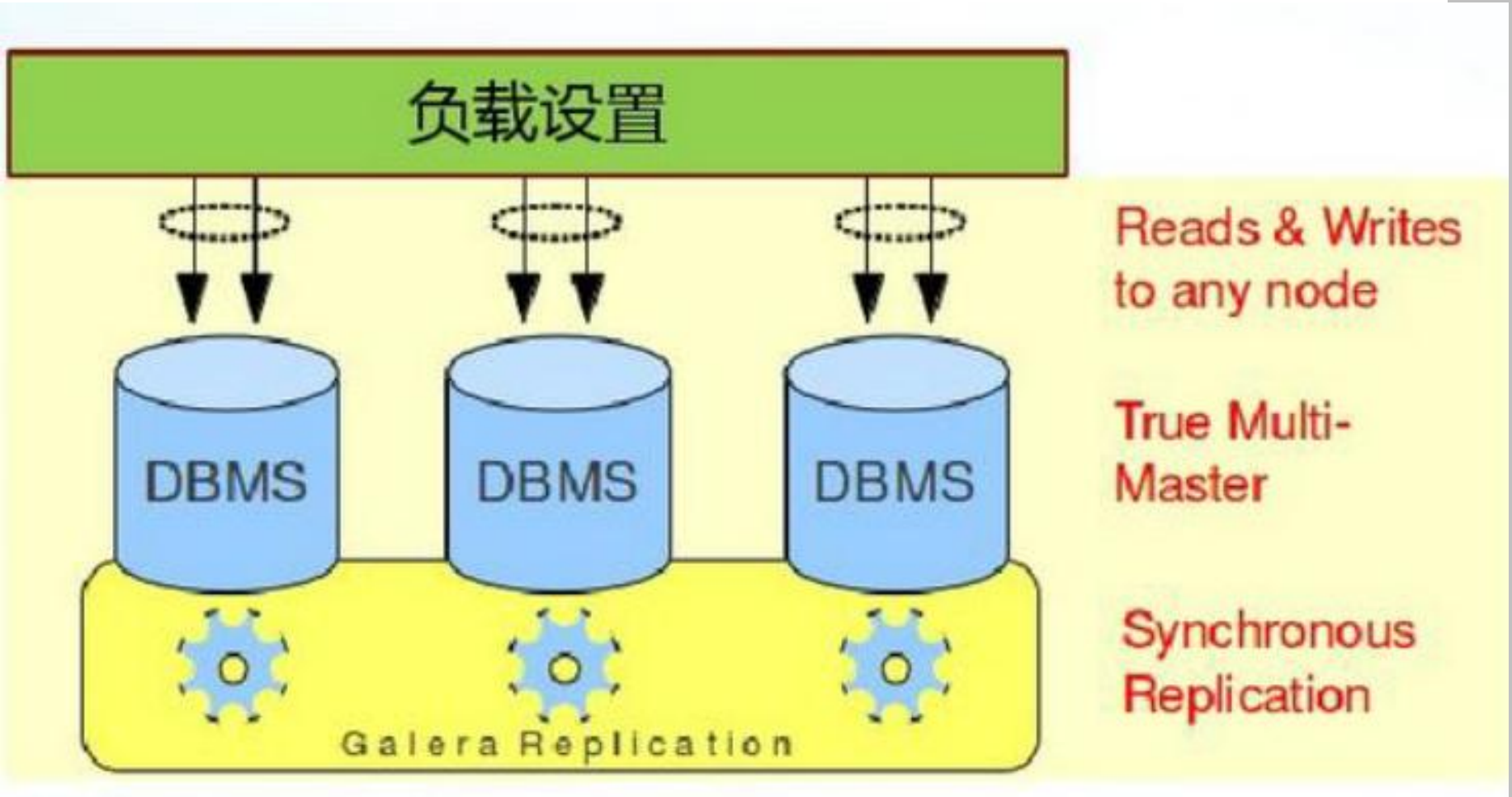
- 1.同步复制，事务可以在所有节点上提交。
- 2.多主机复制，你可以写到任何节点。
- 3.从（**slave**）服务器上的并行应用事件，真正的“并行复制”。
- 4.自动节点配置。
- 5.数据一致性，不再有未同步的从服务器。

**Percona XtraDB Cluster**完全兼容**MySQL**或**Percona Server**，包括：

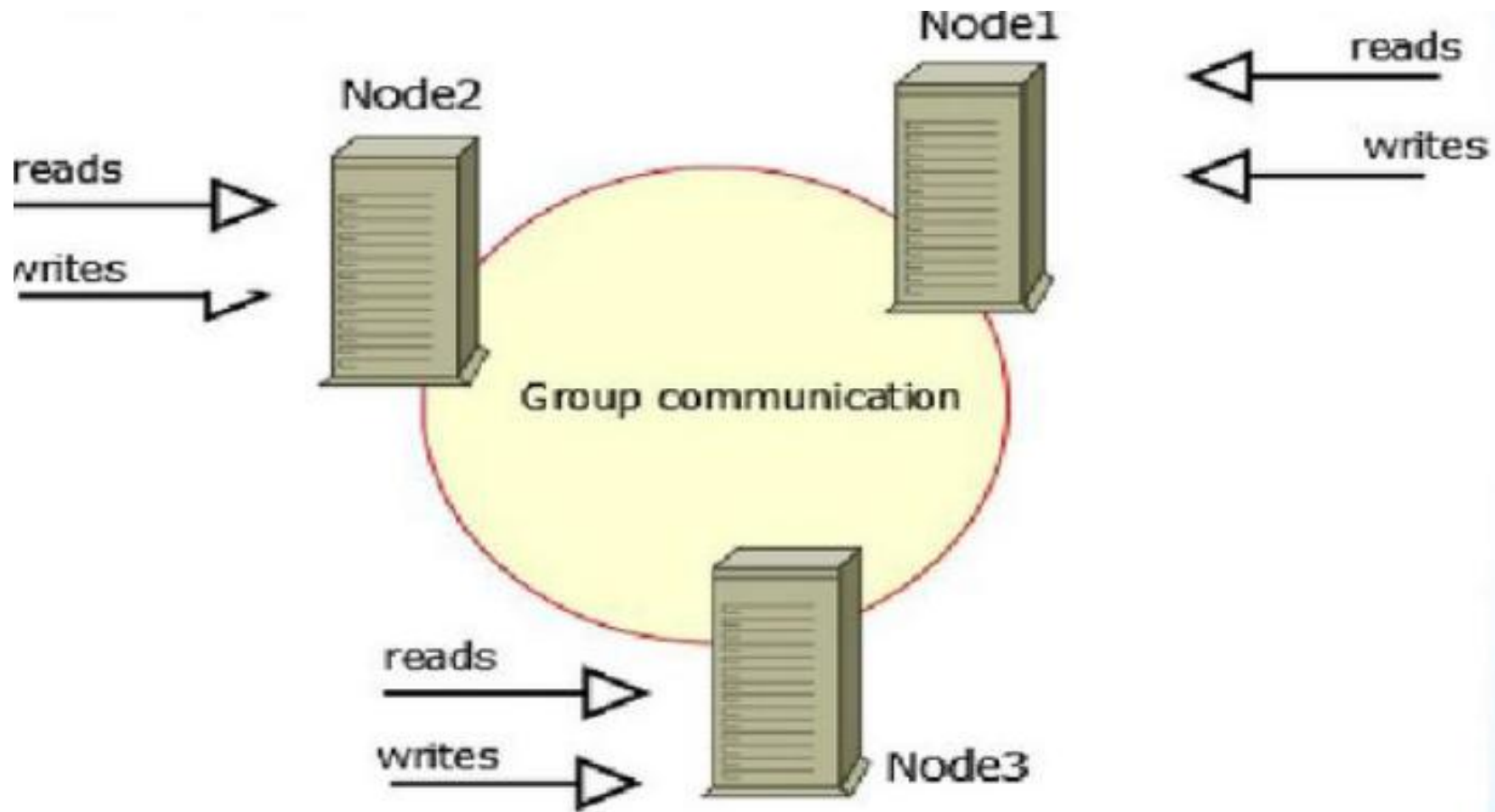
- 1.数据兼容。**Percona XtraDB Cluster**可在由**MySQL**或**Percona Server**创建的数据库上使用。
- 2.应用程序兼容。如果要使用**Percona XtraDB Cluster**，你的应用程序基本不需要作任何更改。



# Percon Xtradb Cluster架构



# Percon Xtradb Cluster架构



# PXC的安装和配置

见辅助材料 《MySQL高可用方案之PXC》

---

# MHA

**MHA**是一位日本**MySQL**大牛用**Perl**写的一套**MySQL**故障切换方案，来保证数据库系统的高可用。在宕机的时间内（通常**10—30**秒内），完成故障切换，部署**MHA**，可避免主从一致性问题，节约购买新服务器的费用，不影响服务器性能，易安装，不改变现有部署。支持在线切换，从当前运行**master**切换到一个新的**master**上面，只需要很短的时间（**0.5-2**秒内），此时仅仅阻塞写操作，并不影响读操作，便于主机硬件维护。

在有高可用，数据一致性要求的系统上，**MHA** 提供了有用的功能，几乎无间断的满足维护需要。

---

# MHA特性

- 1.主服务器的自动监控和故障转移
  - 2.交互式主服务器故障转移
  - 3.非交互式的主故障转移
  - 4.在线切换主服务器
-

# MHA所需要的前提条件

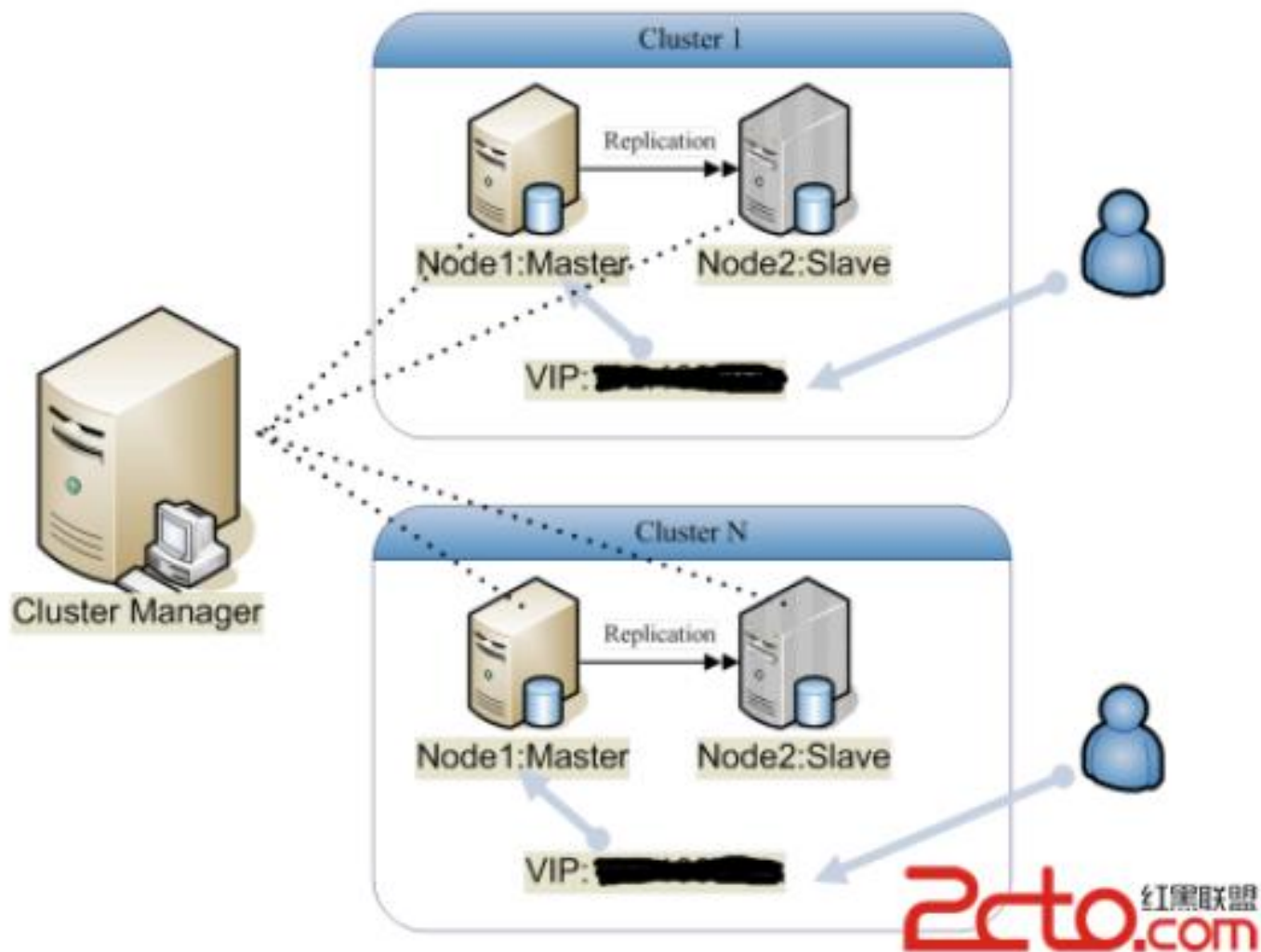
**1.SSH**公钥验证

**2.Linux**系统

---

# MHA架构

常规状态



# MHA的安装和配置

见辅助材料 《MySQL高可用方案之MHA》

---



## 高可用方案-NDB

# NDB Cluster介绍

**MySQL** 群集是 **MySQL** 适合于分布式计算环境的高可用、高冗余版本。它采用了 **NDB Cluster** 存储引擎，允许在 **1** 个群集中运行多个 **MySQL** 服务器。在 **MySQL 5.0** 及以上的二进制版本中，以及与最新的 **Linux** 版本兼容的 **RPM** 包中提供了该存储引擎。

# NDB Cluster介绍

**MySQL** 群集是一种技术，该技术允许在无共享的系统中部署“内存中”和“磁盘中”数据库的 **Cluster**。通过无共享体系结构，系统能够使用廉价的硬件，而且对软硬件无特殊要求。此外，由于每个组件有自己的内存和磁盘，不存在单点故障。**MySQL Cluster** 由一组计算机构成，每台计算机上均运行着多种进程，包括 **MySQL** 服务器，**NDB Cluster** 的数据节点，管理服务器，以及（可能存在的）专门的数据访问程序。

管理服务器(**MGM**节点)负责管理 **Cluster** 配置文件和 **Cluster** 日志。**Cluster** 中的每个节点从管理服务器检索配置数据。当数据节点内出现新的事件时，节点将关于这类事件的信息传输到管理服务器，然后，将这类信息写入 **Cluster** 日志。

# NDB Cluster介绍

**MySQL** 群集的数据节点之间的通信是不加密的，并且需要高速的带宽，所以建议把群集建立在一个高速局域网内，不建议跨网段、跨公网的部署这种系统体系。

**MySQL** 群集分为三种节点：管理节点，数据节点和**SQL**节点。

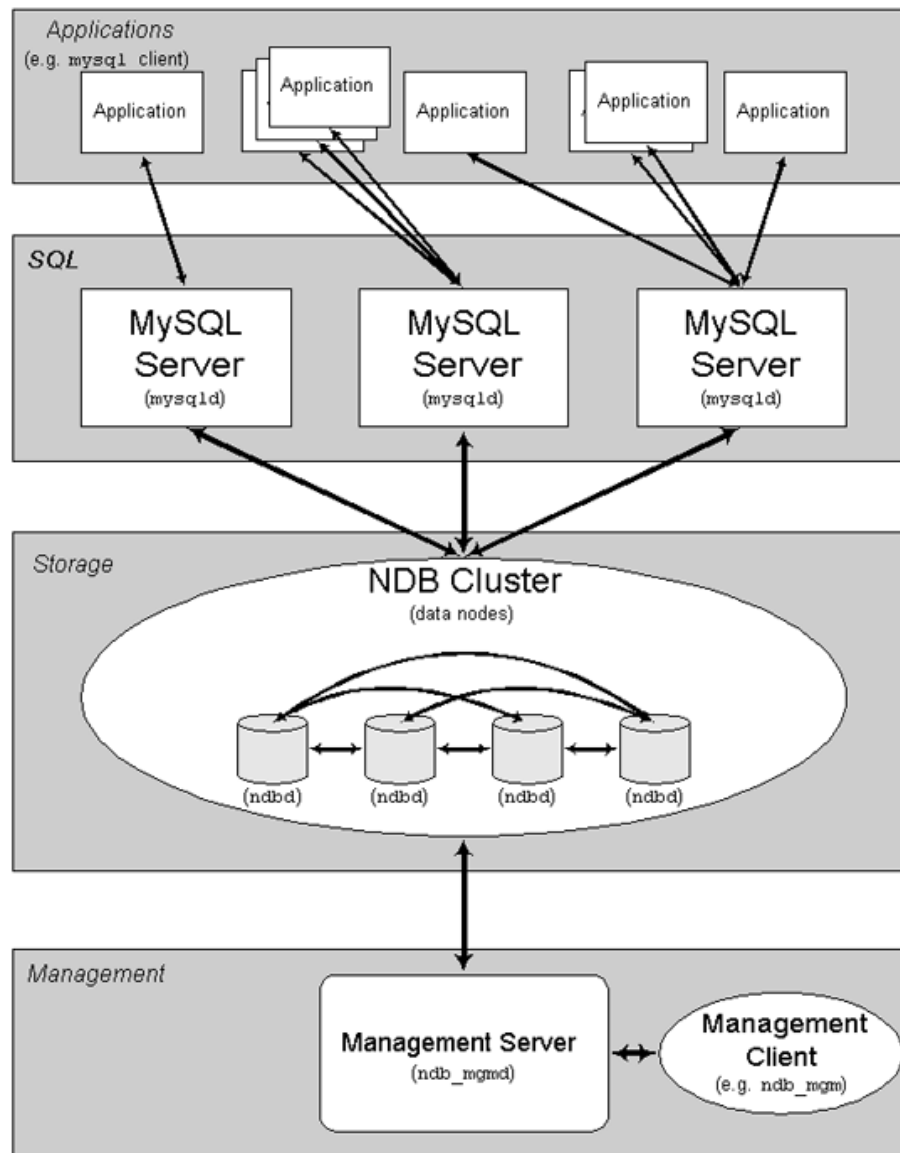
管理节点：主要用于管理各个节点，能够通过命令对某个节点进行重启、关闭、启动等操作。也能够监视全部节点的工作状态。

数据节点：主要是对数据的存储，不提供其他的服

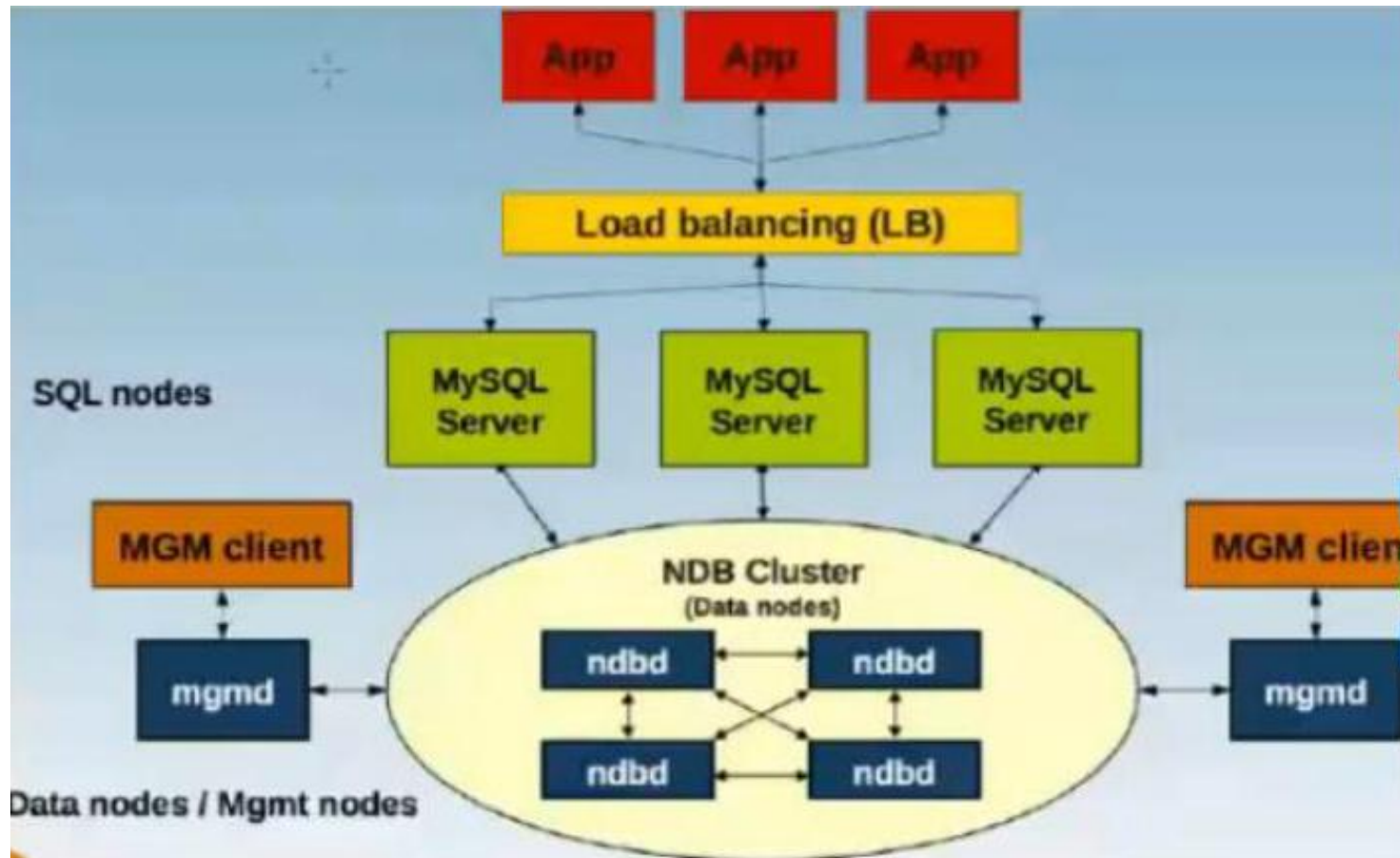
**SQL**节点：主要是对外提供**SQL**功能，类似一台普通的**MySQL Server**。

---

# NDB Cluster架构



# NDB Cluster架构



# NDB Cluster安装和配置

见文档 《MySQL培训（10）\_MySQL高可用方案之cluster安装》

---

# HA方案比较参考

## HA方案比较参考

Requirements	MySQL Replication	MySQL Replication + Heartbeat	MySQL Heartbeat + DRBD	MySQL Cluster
Availability				
Automated IP failover	No	Yes	Yes	No
Automated database failover	No	No	Yes	Yes
Typical failover time	User/script-dependent	Varies	< 30 seconds	< 3 seconds
Automatic resynchronization of data	No	No	Yes	Yes
Geographic redundancy support	Yes	Yes	Yes, when combined with MySQL Replication	Yes, when combined with MySQL Replication
Scalability				
Built-in load balancing	No	No	No	Yes
Supports Read-intensive applications	Yes	Yes	Yes, when combined with MySQL Replication	Yes
Supports Write-intensive applications	No	No	Yes	Yes
Maximum number of nodes per group	One master, multiple slaves	One master, multiple slaves	One active (primary), one passive (secondary) node	255
Maximum number of slaves	Unlimited (reads only)	Unlimited (reads only)	One (failover only)	





# Thank You !

[www.mysql.com/](http://www.mysql.com/)