大型可扩展的web基础架构

     mysql 的最佳实践

**目录：**

小型网络参考架构

     大小和拓扑

     InnoDB

     HA的实现--MySQL复制

中型网络参考架构

     大小和拓扑

     管理MySQL复制

     linux heartbeat

大型网络参考架构

     大小和拓扑

数据refinery(处理？提炼)

     MySQL集群的会话管理和电子商务

社交网络的架构(超大型)

     大小和拓扑

     分片

完美的MySQL server

     MySQL server

     MySQL集群

增值部分的参考体系结构

     MySQL企业监视器

     MySQL企业备份

     MySQL集群管理

     oracle VM模板和MySQL企业版

     MySQL支持咨询和培训

##################################

**摘要**：

............

在本白皮书中，我们提出了四个参考架构基于最佳实践 从与这个星球上最成功的网页内容合作开发的。 我们选择了常见的大多数网页内容四部分组成（**用户认证 /会话管理，内容管理，电子商务和分析**），并定义 最佳的部署架构为每个。 该参考架构进行分类 由“小”，“中”，“大”和“超大”（社交网络）网站的基础上，上浆 并适合每个环境的可用性需求。 白皮书总结了服务器和存储上的建议 配置需要支持的参考架构

##################################

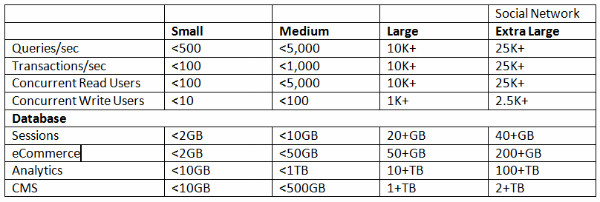
**高可用的注意事项：**

更高程度的可用性显著减少 的停机时间，并通过部署系统具有冗余的更高水平，并取得 容错度。 然而，更大的冗余提高了系统的总成本，由于需求 更多的硬件和软件，以及要求以更大的投资在IT人员， 流程和服务来部署和管理更复杂的环境

对于高可用性的业务需求分析和对理解 所附的成本使该被平衡，以满足待开发的最佳解决方案 该组织在其财务和资源约束的需要。

##############

**网站大小的定义：**



##################################

**小型网络架构的定义：**

每秒请求 <500

每秒事物<100

并发读用户<100

并发写用户<10

数据库大小：

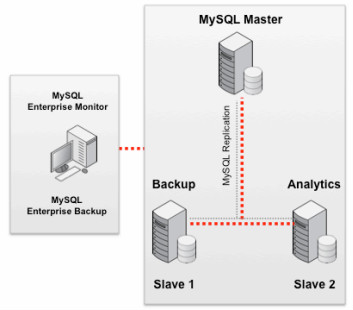
     sessions：<2G

     eCommerce：<2G

     Analytics：<10G

     Content Management：<10G

小型网络拓扑图：



     在这种拓扑结构中，单个MySQL主服务器被部署到支持所有的应用程序， 包括会话管理，电子商务，内容管理和搜索。

     为了确保MySQL主能够将资源集中服务于网络 应用程序，数据库复制到两个slave，一个处理备份，另一个分析处理（注意：MySQL企业备份可以执行在线“热”，非阻塞的MySQL数据库的备份。 完整备份可以执行所有的 InnoDB数据，MySQL在线的情况下，不中断查询或更新，从而 无需使用专用的slave）。

**InnoDB：**

     InnoDB适用于大多数的小型网络架构

     从MySQL 5.5的发布开始，InnoDB已经成为了默认的存储引擎。 InnoDB的 是完全符合ACID，提供快速、零丢失的崩溃恢复功能。 InnoDB的支持高并发应用程序与行级锁和 MVCC（多版本并发控制）的支持，额外还支持外键和约束

**小型架构的拓展：**

     一般的会话管理技术像浏览器和Web服务器为基础的解决方案，常见的会话管理技术都可以满足要求，如果访客流量很低，应用程序并不需要管理大型或复杂的会话数据。 然而，随着应用程序数量和流量的增加，所以这样做将需要在服务器上支持的会话。 加上大小不同的数据，可扩展性和性能问题的复杂性也可能会很快出现。

     使用MySQL比传统的会话管理解决方案来存储这些会话 变量，可能会导致更好的整体性能。 因此，如果服务快速发展，建议会话管理应用程序通过MySQL管理。

     随着Web服务的负载增长，分配内存和调整每个应用程序的硬件资源变得越来越复杂。 因此这种架构是建议当负载很轻时使用。 随着负载的增加，可管理性变得越来越复杂。 基于上述原因，如果预期出现高增长，建议使用者开始中型网络的配置，后面的文件，它提供了更大的容量空间， 更灵活的业务变化和改进的可用性。

**HA的实现：复制**

     MySQL复制已经广泛应用， 它很简单，为用户迅速创建自己的数据库的多个副本，以向外扩展超出了容量限制的单一实例，使他们能够服务于快速增长的数据库工作负载。 MySQL复制的详细讨论，提出在中等网络中介绍。

##################################

**中型网络架构的定义：**

每秒请求 <5000

每秒事物<1000

并发读用户<5000

并发写用户<100

数据库大小：

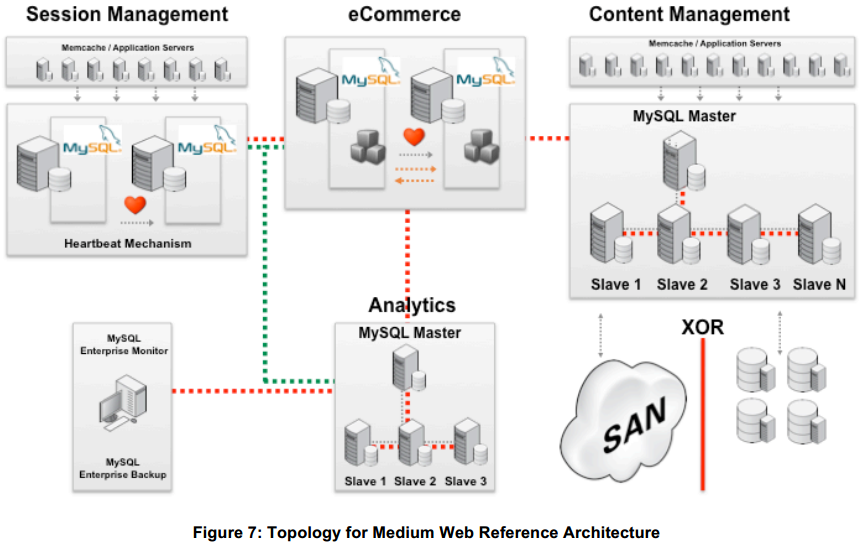
     sessions：<10G

     eCommerce：<10G

     Analytics：<500G

     Content Management：<500G

中型网络拓扑图：



相比前面说的小的网络参考架构，中等网络拓扑的核心功能是会话管理，电子商务，内容管理和分析 他们自己的服务器和存储基础架构，使每个部件都能被单独部署，管理和独立缩放。

如果用户希望自己的Web服务随着时间的推移而扩展来支持将来的负载的话，建议他们开始由最初就部署这个中等拓扑。  这方法还提供了更简单的演变和体系结构的可管理性，以应对随着工作负载不断变化和增长超出了最初的设计和商业预期。

对于应用服务器和MySQL服务器的比例，良好的经验法则是，每个 MySQL服务器能支持8台应用服务器，应用程序在读密集的环境下可以增加更多的 slave服务器来向外扩展。 对于 PHP的应用，更多的应用服务器通常是必需的，并且对于 Java，一般少一些。

随着连接到MySQL 服务器的负载增加，需要向外扩展 (scale-out)基础设施的各个部分。 独立复制的 MySQL主从服务器为每个核心功能的开发人员和 DBA在应对MySQL 作为基础设施增长需求的时候提供更大的灵活性。

**会话管理与电子商务**

这两种工作使用默认的InnoDB存储引擎提供事务支持和崩溃恢复。 **为了提供高可用性，既也可以使用 Linux Heartbeat 与半同步 MySQL复制或基于OS 的解决方案，如 DRBD（两者都在后面的“Achieving HA”一节中讨论），以及 MySQL企业备份。**

在典型的网络环境中，会话状态一般维持 45分钟至1 小时在一个专门的数据库分区中，通过滚动分区来快速删除旧的会话数据 通过删除受影响的表。

对于数据挖掘和商业智能相关的程序，会话和电子商务数据被存储在分析数据库中为离线生成报告做准备。

随着网络上正致力于提高基于用户个性化体验历史上的浏览和购买行为，所以会话数据正变得越来越重要。随着越来越大的会话数据被实时的保存和完成，所以工作量会变成性能密集型，同时要求非常高的可用性，以确保无缝的客户体验。  在这些情况下，它使用MySQL Cluster替换 InnoDB存储引擎。

作为一个为设计目标为99.999％可用性的实时数据库， MySQL集群也取代现有的heatbeating机制和 Memcached的层。Again ，这是MySQL的咨询团队与 Oracle分析的应用程序和确定最佳选择存储引擎。  MySQL集群是在讨论本文的“大型网络 ”参考体系结构一章。

**内容管理**

内容管理应用程序的可扩展性是至关重要的，因为这是的一个核心部分web服务。 MySQL复制是用来提供**读取的可扩展性**，每个MySQL的master通常连接到20 - 30个slave。 在常规的内容管理工作量下，每个从机应能支持多达3000个并发用户。

这些度量仅是参考，它依赖于许多因素，包括：

-读取和写入卷(volumes)

-传播和网络流量的负载均衡

-使用缓存机制

分布式文件系统，如**MogileFS**经常被用来索引的物理资产内容管理系统，与元数据中的InnoDB存储在每个资产由MySQL管理的表。 内容资产本身-图像，视频，文件等-不是存储在数据库中，而是在该文件系统。

Distributed file systems such as MogileFS are often used to index the physical assets of

the content management system, with the metadata for each asset stored within InnoDB

tables managed by MySQL. The content assets themselves –images, videos,

documents, etc. –are not stored in the database, but rather within the file system.

对于物理存储，内容资产可以存储在SAN（存储区域网络）或分布在连接到每个服务器的本地存储设备。因为SAN可能存在单点问题（SPOF：Single Point of Failure），因此建议只使用中档和高端产品，因为这些设备中通常提供高可用的机制。 不推荐使用NAS（网络附加存储）或NFS（网络文件系统）。

如果数据是被分布的存放在本地存储设备，那么保证索引和元数据的高可用很重要。 通常可以部署以下这些解决方案：Linux Heartbeat与半同步MySQL复制或OS为基础的解决方案像DRBD。

**Memcached:**

在中等参考架构中，Memcached与MySQL一块部署来支持缩放会话和内容管理服务。 Memcached是一个分布式的内存缓存层，并已广泛的在一些最大的网站与MySQL一起来使用来实现高可扩展性，可以实现更快的页面加载速度和更有效地利用现有的数据库资源。

一旦源数据被缓存在高速缓存中，对于相同数据请求就使用缓存副本，而不是从原始源数据库中获取数据，减少数据库层的负载。 Memcached不仅可以缓存数据和结果集，而且还可以缓存其他的对象，如预编译的HTML。

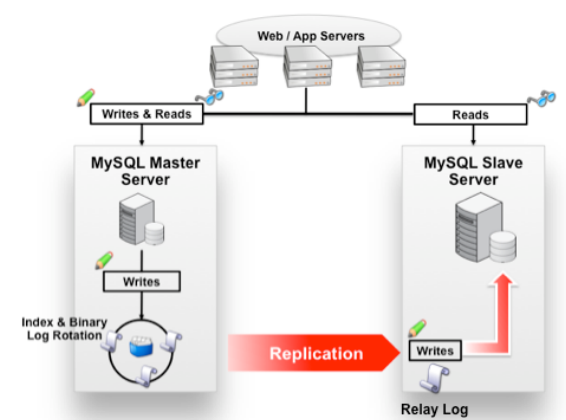
Memcached的可以安装在专用服务器上或者同一地点的网络上的应用程序或数据库服务器上。 Memcached的可以逐步向外扩展的按需方式。因为memcached的可扩展到支持几十个甚至几百个节点的最小开销，空闲的内存资源可以进一步扩展其他的应用程序。 memcached的被设计成**一个非阻塞的基于事件的服务器**并且没有特殊的网络或互联的要求。

***实现HA - MySQL复制***

正如前面的文章介绍，复制一个数据库从一个物理位置或系统到另一个系统（通常是从master到slave）。 这通常用于增加可用性和数据库的可伸缩性，通过上面的体系结构，用户通常执行备份操作或在在slave上运行分析查询，从而减轻master的负载。

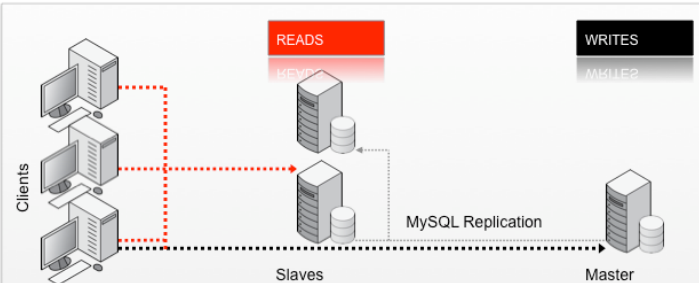
原生的MySQL支持单向，异步复制作为标准功能数据库。MySQL 5.5及以上的，半同步复制功能被增加。 根据配置的不同，您可以选择复制所有的数据库，指定的数据库，甚至指定一个数据库内的表。

MySQL复制的工作原理是只要有一台服务器作为master，而一个或多个服务器充当slave服务器。 主服务器将记录对数据库的更改。 一旦这些变化都被记录后，随后被发送并应用到slave（s）。



**图*8：MySQL复制支持HA和Read Scalability “Out of the Box”***

复制往往是应用于一个向外扩展的实施，使该请求简单的向从服务器“读”的数据， 涉及交易写入的执行在主服务器上。这通常不仅有一个性能提升，更可以有效的利用资源。



**图9： *缩放读取与MySQL复制***

使用MySQL复制可以向外扩展提供很高的读的能力对于在数据库上执行的读的操作。slave只花费很少的主服务器开销，不难发现有高达30个slave服务器在比较大的网站部署。 这是高度依赖于流量模式的工作模式，所以它是最好的MySQL的咨询团队和甲骨文合作开发的最合适的拓扑结构。This is highly dependent on the traffic patterns of the workload, and so it is best to work with the MySQL Consulting team at Oracle to develop the most appropriate topology

复制也是提供高可用性（HA）MySQL数据库的最常见方式。 更新从master复制到slave服务器，出现故障的时候例如死机或者维护目的就可以把故障转移到从服务器上。 故障转移可在应用程序中实现或利用各种机制，这不是本文的内容。单独复制不提供故障切换机制。 故障转移解决方案在后面讨论。

当MySQL被配置为使用默认的异步复制时，主机写入事件到其二进制日志，但不知道从何时或是否已经检索和处理过它们了。 如果主机崩溃，交易很可能没有成功复制到任何slave。 因此，从master故障切换到slave的情况下，可能会导致丢失最新提交的服务器交易。

在新的**MySQL 5.5是使用半同步复制从而提高了数据支持完整性**，并可以替代内置的异步复制。

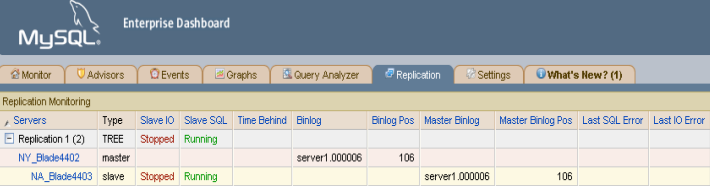
当数据从**master用半同步复制复制到slave时，提交成功返回，所以已知的是在至少两个地方存在的数据**（在master和至少一个slave）。 执行事务的线程提交后master块是完整的，并等待直到至少一个半同步slave确认，它已收到了交易的所有事件记录到**中继日志**，或直到发生超时。 请注意，**master不必等到slave把语句执行了，只要知道它已经接收到中继日志即可。**

半同步复制确实有一定的性能影响，因为提交的事物需要等待slave确认而且slave一般慢。 这是权衡了数据的完整性得到的结果。放缓的量至少是在TCP / IP往返时间发送提交从站并等待接收从器件的确认。The amount of slowdown is at least the TCP/IP roundtrip time tosend the commit to the slave and wait for the acknowledgment of receipt by the slave. 这意味着半同步复制工作最适合用于服务器物理上在同一地点，或者在通信快速的网络条件下。

***管理MySQL复制***

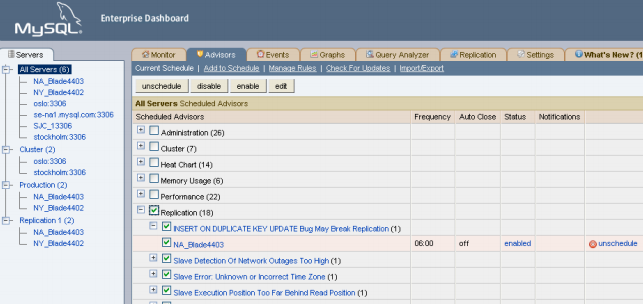
众多的MySQL客户使用的MySQL企业监控器，作为部分MySQL企业版和MySQL Cluster电信级版本，其基于GUI界面来监视他们的复制拓扑。 随着网络的属性成长和master和slave数量的增加，这个工具已经被证明可以节省显著的时间和简化管理MySQL的复制的操作。

MySQL企业仪表板可以更容易地向外扩展，实现了高可用的MySQL复制，并且提供了先进的自动检测，分组，记录和监控所有MySQL复制的主/从层次关系。 修改和补充现有的复制拓扑结构也能自动检测和维护，数据库管理员提供即时可视性的实施更新。 这有助于减少数据库管理员新MySQL复制的学习曲线难度来完成具体的高可用性和向外扩展的环境 。

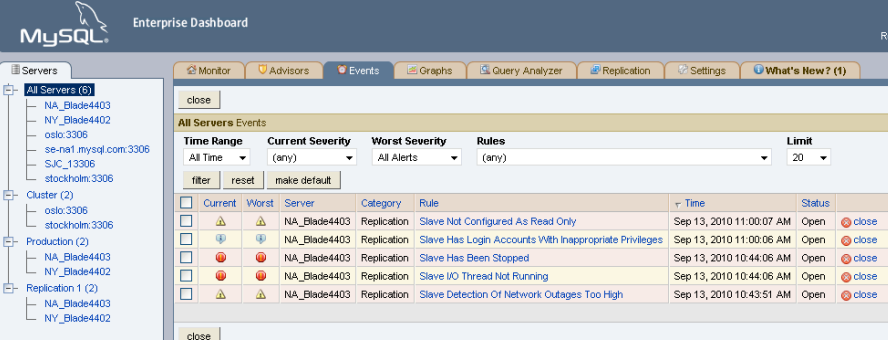


**图*10：MySQL企业监控器-复制仪表板***

复制监视器提供了一个综合的，实时查看到所有主/从拓扑结构的性能和可用性。与MySQL企业监控器-复制顾问(Replication Advisor)配合，复制监视器可以帮助DBA能够主动识别和纠正复制相关的问题，以避免代价高昂的停机。



**图*11：MySQL企业监控器-复制顾问***



**图*12：MySQL企业监控器-复制事件***

复制顾问Replication Advisor可以识别问题并发出警报，DBA可以使用复制监视器的警示内容来研究受影响的状态主或从。 使用复制监视器，并从专家的意见复制顾问，他们可以查看**当前主/从状态和视图指标（如slave的I / O，slave的SQL线程，落后master的时间，master的二进制日志的位置，最后一个错误，等等）**是相关的诊断并解决任何问题。 该复制监视器，设计并实现节省DBA的时间和维持收集脚本，巩固和监视类似MySQL的复制状态和诊断数据。

***Linux Heartbeat***

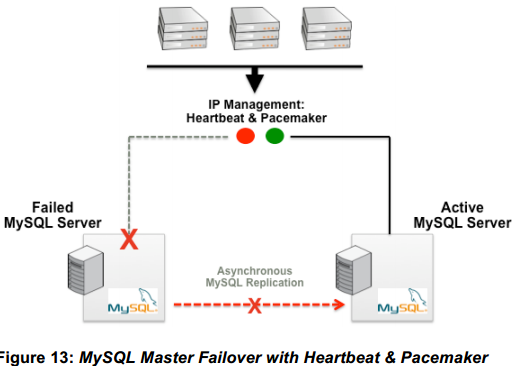
其中最常见的解决方案来解决故障检测和故障切换的服务器集群方案是Linux Heartbeat(用于集群通信)和集群资源管理如Pacemaker或OpenAIS。 Linux的Heartbeat不是唯一的自动故障转移解决方案。 Solaris Cluster的广泛部署在运行Solaris操作系统的环境中。 对于这些参考架构，我们重点放在Linux上的Heartbeat。

Linux的Heartbeat是将消息按固定的时间间隔定期两发送到两个或多个节点之间。 如果一个消息从另一个节点没有接收到的固定时间间隔的消息，则假定该节点已经失效，故障切换动作也是由群集资源管理器发起。

Heartbeat和资源管理器初始化的时候，一个主机被选为主节点。 当Heartbeat在主节点上启动时，一个虚拟IP地址被分配到master的网络接口。 这个接口将会被外部程序，应用程序和用户将要访问的节点的地址。This interface will be the mechanismin which external processes, applications and users will access the node

在出现故障的情况下，发生故障的主机上的资源被禁用，并且资源会被另一台主机代替。 此外，故障主机的虚拟IP地址（VIP）将被集群重定向到替代故障设备的新主机。

下图说明了一个主MySQL服务器的Linux Heartbeat的故障转移和Pacemaker自动重定向应用到从服务器。



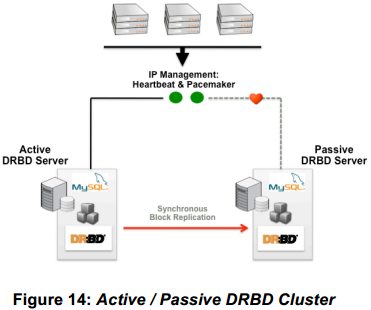
**图*13：MySQL的master的故障切换使用Heartbeat和pacemaker***

要创建一个完全事务安全的“双机热备”的配置，一些用户也部署DRBD（分布式复制块设备）配合Heartbeat。

DRBD是利用一个开源的linux内核块设备复制到两个系统，通常是在主动/被动之间的数据镜像配置。

如下面的图中，MySQL被配置为存储数据在DRBD块设备，一台服务器作为主，并让可提供服务的第二台主机在发生故障时立即替换。 Heartbeat管理两个服务器上的接口并且在故障的情况下自动配置次级（被动）服务器替换 主服务器(活动)服务器。

**图14： *主动/被动集群DRBD***



**####################################**

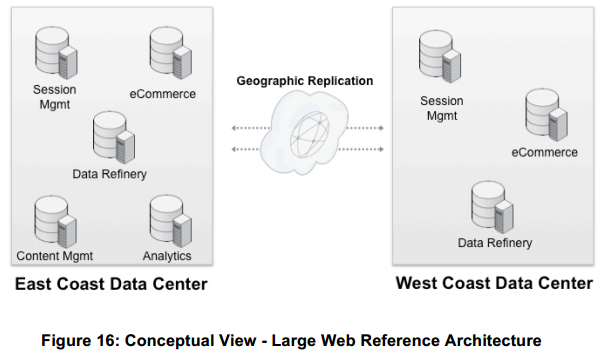
**大型Web参考架构**

***大小和拓扑结构***

大型网络参考架构定义了下面的大小：

**图15：调整大型网络参考架构**

下列为建议的结构，并详细视图的拓扑结构，支持大型的工作量。



**图16：概念视图-大型网络参考架构**

  在概念图中，我们可以看到，MySQL数据库使用MySQL地理复制技术，镜像复制到了一个远程数据中心集中使用，因此提供了灾难恢复能力。 地理的复制通常也部署到物理定位数据更靠近用户的地方，以便减少地理延迟的影响。

通常情况下，会话管理和电子商务数据库将是首要考虑进行复制的。 IP管理可确定用户的位置和他们与数据中心的距离，这是物理上最接近于它们。Data Refinery也是可复制的典型。

大型网络的参考架构是建立在最佳实践上的参考架构。 主/从复制用于​​向外扩展(scale-out)该网站的内容管理和分析组件。 内容管理架构将能够扩展到支持100K +并发用户在大约有30台slave的情况下，again dependent on traffic patterns of the workload再次依赖于传输模式的工作量。

内容管理系统可以在SAN上或在分布式本地存储上实现。 详情请参阅中等架构中的“内容管理”部分网络参考架构。需要的技术考虑的讨论评价最好的部署平台时作出 怎么翻译。Please refer back to the “Content Management” section of the Medium Web Reference Architecture for a discussion on the technology considerations that need to be made when evaluating the best deployment platform

**Data Refinery**

In addition to the existing workloads, a Data Refinery component is added for higher

volume analytics and increasedcontent management demands.

is, but in this instance, they are first replicated to the Data

including the content management system where it performs data cleansing tasks to

modify or delete dirty data. It then builds complete data sets and data warehouse

dimensions for loading into the Analytics database,

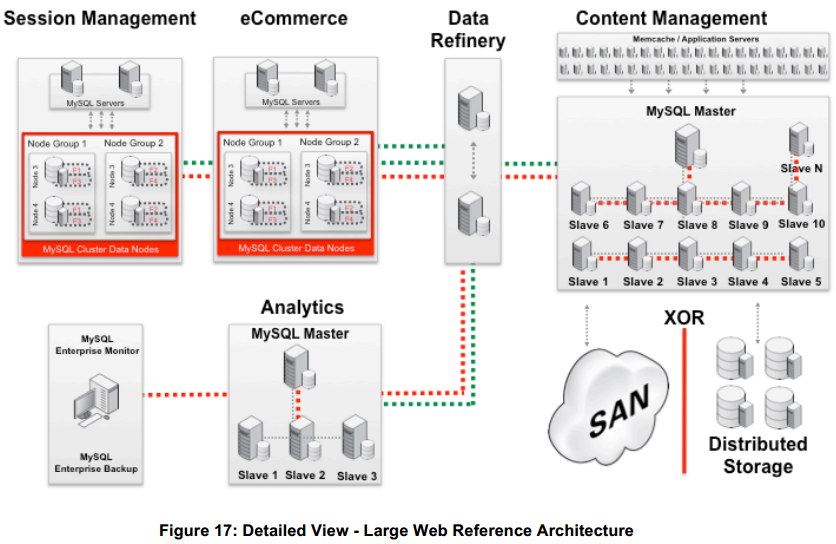
The Data Refinery can also be used to cleanse data before it is loaded into the content

management system

除了 ​​现有的工作负荷，一个数据炼油厂组件被添加为更高

量分析，增加的内容管理需求。

对于与中型参考架构，会话管理和电子商务的数据被用于分析，但在这种情况下，它们首先被复制到data refinery。 该Data Refinery汇总从其他网络基础设计收集来的数据，包括内容管理系统content management system，其中它执行数据清理任务，修改或删除脏数据。 然后它建立完整的数据集和数据仓库然后加载到分析数据库，It then builds complete data sets and data warehouse dimensions for loading into the Analytics database 。Data Refinery也可用于在数据装入内容管理系统之前清洗。



**图17：详细查看-大型网络参考架构**

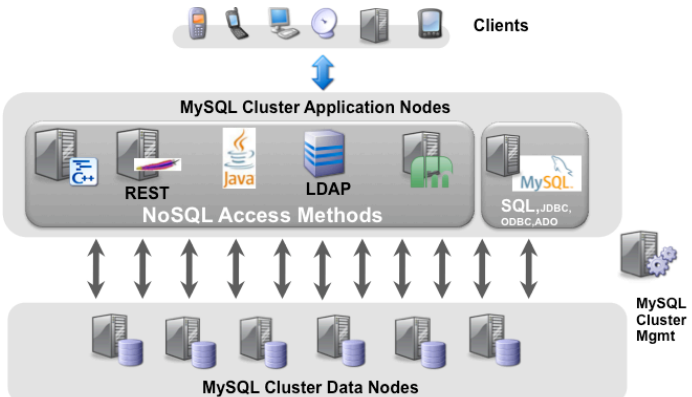
***MySQL cluster的会话管理和电子商务***

为了处理大型网络的更高的性能和可用性要求，会话管理和电子商务数据库部署在MySQL的集群中。 用4个数据节点，它可以支持6000会话（page hits）每秒，每个页面的命中(page hits)产生8 - 12的数据库操作。

通过使用MySQL集群的横向扩展(scale-out)能力，它可以结合会话管理和电子商务数据库扩展到一个更大的集群。

MySQL集群是一个实时的，事务数据存储相结合的高度灵活性的低TCO(总体拥有成本)开源关系数据库。 设计目标是一个”shared-nothing”的无单点故障的分布式架构，MySQL集群设计提供99.999％可用性的事务型Web环境。

MySQL cluster的实时设计提供可预测的，毫秒响应时间，支持每秒几万事务的能力。 支持内存与基于磁盘的数据，自动数据分区负载平衡和补充的能力，节点的集群上运行零停机时间，允许的线性可扩展性的数据库，用来处理最不可预知的基于网络的工作。



**图18：MySQL集群架构消除任何单点故障**

MySQL集群是低成本的支持向外扩展的商用服务器，包括三个不同类型的节点。

**数据节点**是群集的主要节点，提供了以下功能：

•数据存储和管理基于内存和磁盘

•支持自动和用户手动的数据分区，即使新节点被添加到一个正在运行的群集

•数据节点的数据复制同步使用**两阶段确认协议**

•支持事务和数据检索

•自动故障转移(fail-over)

•失败后重新同步

通过在shared-nothing架构分发数据，并同步实施两阶段提交协议的复制，如果一个数据节点发生故障，将总有至少有一个附加的数据节点存储的信息相同。 这允许交易不间断地进行支付。 数据节点可以也可在线添加，允许数据的容量和处理的快速扩展，而不会影响应用程序的可用性。

**应用节点**负责将应用程序连接到数据节点。 这可以利用应用程序中嵌入的高性能NDB（C + +）或Java的形式的API，或者更常见的是，他们连接到MySQL服务器，提供一个SQL接口到集群中存储数据。 因此，多个应用程序可以同时使用一组丰富的接口与任何访问MySQL集群中的数据更新基础数据即时访问任何应用程序访问集群。 应用节点也可以在线添加不中断服务集群。

**管理节点**负责管理集群，使集群提供其它节点的配置信息。 管理节点都在使用启动时有一个系统的重新配置。 管理节点可以停止并重新启动，而不会影响正在进行的执行的数据和应用程序节点。默认情况下，管理节点还提供**仲裁服务**，在一些事件网络故障导致“脑裂”，或者集群表现出“网络分区”的情况下。

通过这种分布式架构，其中的依赖已经被最小化，应用程序继续运行，数据保持一致，对数据，应用程序，即使任何一个，或管理节点失败。

MySQL集群电信级版本包含了MySQL Cluster Manager来简化创建群集操作，通过自动化常见的管理任务和管理。更多信息，白皮书的“增值部分”一节中。！

要了解更多关于MySQL集群架构，请参考MySQL集群。架构和新功能白皮书发表于：http://www.mysql.com/why-mysql/white-papers/mysql\_wp\_cluster7\_architecture.php

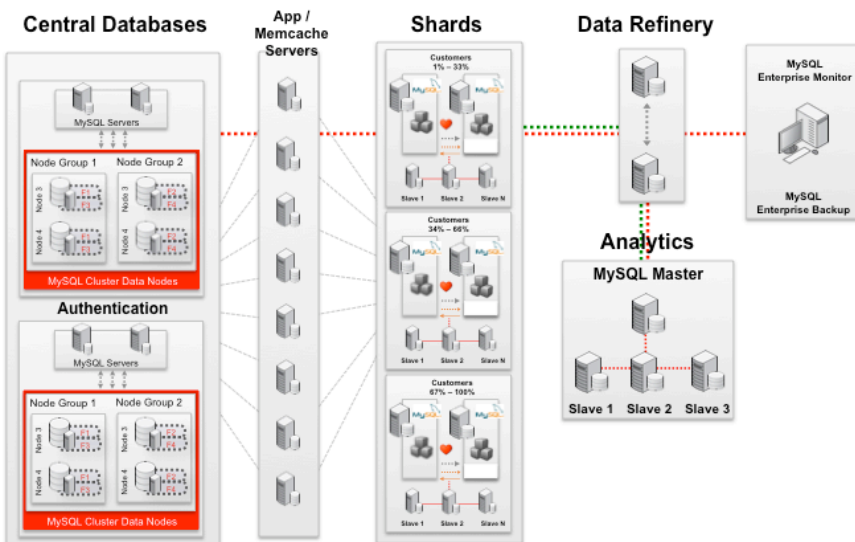
**社交网络（特大）参考架构**

***大小和拓扑结构***

之前的表格中有

**图19：调整为社交网络参考架构**

下图显示了推荐的拓扑结构，以支持“社交网络”



**图20：拓扑社交网络参考架构**

该社交网络参考架构重新使用了中型和大型参考架构中的硬件，包括基础设施内的工作负载，memcached 以减轻数据库服务器的负载，和Data Refinery来进行数据筛选，汇总等。

**MySQL cluster是用于用户身份验证，并提供查找机制使应用程序用来直接读取和写入到相应的分区**或为一个以上的键“shared分片”，用于查找用户数据​​库。

***分片***

**分片**是常用的大型网络**增加写入性能和可拓展性的手段**。 在高流量的Web 2.0环境，如社交网络大部分的内容都是由用户自己生成的，从而要求高水平的**写**可扩展性，处理快速变化的数据。但我们必须记住，**读仍然占着Web 2.0环境的主导地位**，通常每个记录将被读在更新应用之前。It must be remembered that readswill still predominate in Web 2.0-type environments as typically each record will be read before an update is applied.同样要强调的是数以百万计的用户仍然可以使用网络提供应用程序服务，而无需任何数据库拆分的。

**分片-也称为应用程序分区**-涉及到的应用程序划分数据库分成更小的数据集和跨多个服务器分发它们。(involves the application dividing the database into smaller data sets and distributing them across multiple servers)这方法能够非常符合成本效益的可扩展性作为低成本的商用服务器可易于部署时的容量需求增长。 此外，查询处理影响只有一个较小的子集的数据，从而提高性能。

应用程序需要能够”感知分片”这样写才能被正确引导到正确的分片，JOIN操作才能最小化。 如果最佳实践能得到应用，那么**分片将成为扩展关系型数据库支持基于Web的写密集型工作负载的有效途径**。

针对单个列(键)的散列被证明是分片数据的最有效手段。 在社交网络参考架构，我们通过分片用户的数据​​库用户ID，所以一个分片处理用户总数的三分之一，第二个分片处理用户的第二部分，第三部分和其余的是存储在第三子库。

**这种方法最初提供最合乎逻辑的分片机制，使得它简单向外扩展来支持服务的增长**。 然而，很可能是**一些碎片将比其他碎片包含更多活跃用户，这意味着碎片可能在未来需要重新平衡**。 应用程序和数据库需要有足够的灵活性，以适应变化的架构。

在参考架构中，为每个分片的高可用性是由操作系统提供集群技术，如DRBD与Heartbeat，每个主服务器复制到从服务器上实现读的可扩展性。 不同于网络参考架构，社会网络架构管理上的分片本身的会话状态。Unlike the Web reference architectures, the Social Networking architecture manages session state on the shards themselves.

由于分片是在应用层实现的，它的最佳实践很重要取决于应用程序和数据库的设计。

通过分片和MySQL复制的部署，社交网站如Facebook，Flickr后，LinkedIn和Twitter已经能够扩展满足他们要求的关系数据管理指数。

**完美的MySQL服务器**

MySQL是支持在20个不同的平台，包括多CPU架构和操作系统。 下节列出了最常见的平台上已被证明为提供最佳的性能，可扩展性和可用性。

***MySQL服务器***

当挑选服务器大小时，它大小和特点很重要取决于您当前运行的数据库大小并估计未来的大小要求。 同样重要的是要记住用于从属服务器可扩展性应该是和master一样强大。 推荐规格如下：

-8 - 16个x86-64位CPU核心（MySQL 5.5及以上）。

-4 - 8 个x86 -64位CPU核心（MySQL 5.1和更早的版本）。

-3 - 10倍比活跃数据更多的RAM。

-Linux和Solaris或Windows操作系统。

-最少的4个硬盘驱动器。 8 - 16个磁盘将提高I / O密集型应用的性能。

-硬件RAID带电池备份缓存。

-RAID 10的建议。 RAID 5是合适的，如果工作量是读密集型。

-2个网络接口卡和2个电源模块冗余。

***MySQL集群***

由于MySQL集群，推荐根据服务器配置的分布式特性来部署，和一个MySQL服务器运行的MyISAM或InnoDB不是完全一样。

**应用节点（MySQL服务器）建议**

-4 - 16个x86-64位CPU内核

-内存最低4GB。**内存不是在这一层的关键**，主要用于连接器和缓冲池。

-2个网卡和2个电源模块冗余。

**数据节点推荐**

-8个x86-64位CPU内核。 使用尽可能高的主频，可能因为这将使消息处理更快

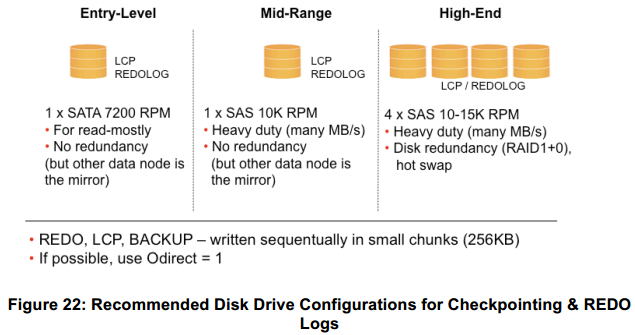
-**每个服务器内存=数据库大小\*副本数量\* 1.25（数据冗余+索引驱动数据节点的整体内存需求）/数据节点的数量**

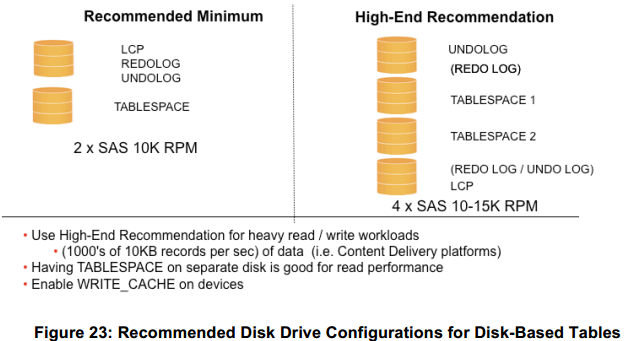
例如：10GB数据库\* 2副本\* 1.25 / 2个数据节点= 12.5GB的每个节点的数据RAM。

-Linux和Solaris或Windows操作系统。

-2个网络接口卡和2个电源模块冗余。

-磁盘驱动器的建议如下。

**图22：建议的磁盘驱动器配置为检查点和重做日志**

 **图23：建议的磁盘驱动器配置为基于磁盘的表**

这是推荐部署专用网络上的数据和应用节点（即IP地址开始于10.0.1.0），用1个千兆以太网作为最低的运输。 该MySQL的服务器将随后也被连接到公共网络。

提供防止网络故障，建议配置每个服务器绑定冗余的网络接口卡（NIC）连接到冗余

开关。 如果多个数据节点都部署到一台主机（即SUN甲骨文CMT系统），它是推荐绑定四个或更多的NIC在一起以处理增加的网络带宽的要求。

**参考的增值部分架构(自动翻译)**

上面定义的参考架构包括增值部分交付的部分商业MySQL的产品。 这些都为讨论：

**MySQL企业监控器**

MySQL企业监控器持续监控你的MySQL服务器，并提醒用户在它们影响你的系统之前解决潜在的问题。 这就像有一个“虚拟DBA助理”在你旁边并建议你最佳做法，以消除安全隐患，提高复制优化性能和更多。 因此，你的开发人员，数据库管理员和系统管理员的生产力显著改善。

加上MySQL查询分析器可以帮助您通过监控查询提高应用程序性能表现，而导致放缓准确地针对SQL代码。 查询是呈现在所有的MySQL服务器的聚合视图，以便您可以过滤特定的查询问题和分析最昂贵的代码。 与MySQL查询分析器，您可以积极发展中提高SQL代码，并持续监测和调整查询在生产。

**MySQL企业备份**

MySQL企业备份执行在线“热”，无阻塞你的MySQL备份数据库。 完整备份可以在所有InnoDB数据来进行，而MySQL是在线，无需中断查询或更新。 此外，增量备份，支持其中只有数据已经从以前的备份操作更改备份。 另外部分备份支持时，只有特定的表或表空间需要进行备份。

MySQL企业备份从具有完全的向后兼容性的完整备份恢复您的数据。一致点的即时恢复（PITR）使DBA来执行还原到一个特定的点时间。 使用MySQL备份和二进制日志，数据库管理员还可以执行细粒度的前滚恢复特定交易。 部分还原允许目标表或表空间的恢复。 另外，DBA可以还原备份到一个单独的位置，或者创建克隆用于快速复制设置或管理。

MySQL企业备份支持创建压缩备份文件，通常减少备份尺寸从70％到超过90％的实际数据库文件的大小相比-降低存储和其他成本。

**MySQL集群管理器**

MySQL集群管理器简化MySQL集群的创建和管理电信级版本通过自动化常见管理任务。 其结果是，数据库管理员（DBA）和系统管理员的生产力，使他们能够专注于支持业务战略的IT计划，更迅速回应不断变化的用户需求。 在数据库的停机时间，在同一时间的风险而以前从手动配置错误导致，会显著减少。！

作为一个例子，管理操作要求MySQL簇的滚动重启数据库先前要求46手动命令并消耗2.5的DBA时间小时现在可以使用单个命令执行的，并完全自动化，有利于降低：集群管理复杂性和开销。！

•自定义脚本的管理命令。！

•停机时间，通过配置和变更管理的自动化风险流程。

您可以通过浏览了解更多关于MySQL集群管理器

www.mysql.com /集群/立方米

**Oracle VM模板的MySQL企业版**

在Oracle VM模板，可以快速部署并消除手动配置通过提供预安装和预配置的虚拟MySQL企业版5.5在Oracle Linux和Oracle VM软件映像运行，在生产中使用的认证。

在Oracle VM模板为MySQL企业版是最快，最简单，最可靠的方式来提供虚拟化的MySQL实例使用户能够满足为Web和基于云的服务的爆炸性需求通过启用：

**-更快的部署** ，以大大减少了安装和配置周期两个初始部署，并提供复制的横向扩展环境

**-提高了可靠性** ，使用具有预安装和预配置的软件堆栈已经进行了广泛的质量保证和生产使用的认证。

**-更高的正常运行时间** ，利用虚拟化技术来帮助减少停机时间造成的故障以及计划的维护操作。

您可以了解更多通过浏览到：

http://www.mysql.com/why-mysql/virtualization/index.html

**MySQL支持，咨询和培训**

**MySQL支持**

Oracle提供全天候，MySQL的全球支持。 MySQL的支持团队是由

经验丰富的MySQL开发，谁是数据库专家和理解问题，并

挑战你的脸。 与Oracle标准支持服务，您可以降低总成本和风险

拥有你的MySQL数据库，提高您的IT投资回报，并

优化您的IT解决方案的商业价值。 MySQL的支持是包含在

2

基于由4×MySQL簇数据节点，2个MySQL服务器的MySQL集群配置

SQL节点和2个MySQL簇管理节点跨越单个服务器实现的（8×总）。

总的操作包括以下命令：1个集群状态初步检查，8×的ssh

每台服务器的命令; 8×每进程停止命令;的配置文件4×SCP（2×mgmd＆2×

mysqld的）; 8×每个进程的启动命令，8×检查启动和重新加入过程; 8×过程

完成验证; 1个验证整个集群的完成。 总命令计数不包括

每个配置文件中手动编辑。

3

基于一个DBA重新启动4×MySQL Cluster数据节点，每个节点的数据6GB，并执行10,000

每秒http://www.clusterdb.com/mysql-cluster/mysql-cluster-data-node-restart-times/操作

认购的最终用户，以及可单独向独立软件开发商的商业执照

与原始设备制造商。 Oracle标准支持服务为MySQL包括以下特点：

•24×7支持生产

•无限支持事件

•知识库

•维护版本，bug修复，补丁和更新

•MySQL的咨询支持

您可以了解更多的 http://www.mysql.com/support/

**MySQL的咨询**

Oracle提供了一个全方位的MySQL的咨询服务。 从体系结构和设计

对于新项目，通过性能优化，以优化现有的应用程序，或

从另一种数据库迁移到MySQL，我们为您提供经济实惠的解决方案。

您可以了解更多的http://www.mysql.com/consulting/

**MySQL的培训**

Oracle提供了一套全面的MySQL的培训课程，让你在一个竞争优势

建设世界一流的数据库解决方案。 专为数据库管理员各级和开发，覆盖

数据库和应用程序设计，包括课程致力于动态发展

Web应用程序。

<http://www.mysql.com/training/>

**结论**

设计，管理和扩展的基于Web的架构是很难的。 技术创新都发生在一个更大的速度比以往任何时候，用户需求的变化更为迅速。

对新服务的需求，加上爆炸中的数据，需要加以管理创建任何组织希望利用网络来显著障碍推动他们的业务。

凭借在本白皮书中确定的最佳实践，我们希望至少有提供一个起点，使您能够构建下一个网页的现象！

**其他资源**

MySQL企业版：数据库。 监控。 支持。

http://www.mysql.com/why-mysql/white-papers/mysql\_wp\_enterprise\_ready.php

MySQL查询分析器：概述

http://www.mysql.com/why-mysql/white-papers/mysql\_wp\_queryanalyzer.php

白皮书- MySQL集群为Web和电子商务应用：

http://www.mysql.com/why-mysql/white-papers/mysql\_wp\_Cluster\_For\_OnlineApps.php

设计和实现可扩展的应用程序使用的Memcached和MySQL

http://www.mysql.com/why-mysql/white-papers/mysql\_wp\_memcached.php

缩放Web服务与MySQL簇：另类视角的MySQL的&memcached的

http://www.mysql.com/why-mysql/white-paper/ mysql\_wp\_cluster\_ScalingWebServices.php

点播网络研讨会- 5个简单步骤入门MySQL集群：

http://www.mysql.com/news-and-events/on-demand-webinars/display-od-566.html