

# Linux服务器构建与运维管理

## 第05章：使用MariaDB实现数据库服务

阮晓龙

13938213680 / ruanxiaolong@hactcm.edu.cn

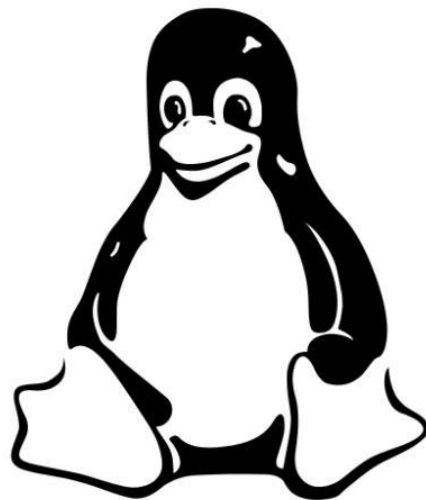
<http://linux.xg.hactcm.edu.cn>  
<http://www.51xueweb.cn>

河南中医药大学信息管理与信息系统教研室  
信息技术学院网络与信息系统科研工作室  
河南中医药大学医疗健康信息工程技术研究所

2022.9

# 提纲

- MariaDB 简介
  - MariaDB 介绍
  - 安装 MariaDB
- 管理 MariaDB
  - 使用 phpMyAdmin 实现 Web 方式管理
  - 使用 MySQL WorkBench 实现运维管理
  - 使用 Navicat Monitor 监控数据库服务器
- 高可用
  - 数据库集群
  - 通过主从集群实现 MariaDB 的高可用



# 1.MariaDB简介

## 1.1 数据库服务

- 数据库是长期存储在计算机内、有组织、可共享的数据集合。
- 数据库中的数据按照一定的数据模型组织和存储，具有较小的冗余度、较高的数据独立性和易用性。
- 数据库按照关系模型分为关系型数据库和非关系型数据库两种。
  - 关系型数据库是指采用了关系模型来组织数据的数据库，其以行和列的形式存储数据，其存储的数据格式可以直观地反映实体间的关系。关系模型可以简单理解为二维表格模型，而关系型数据库就是由二维表及其之间的关系组成的数据组织。
  - 非关系型数据库不遵循关系型数据库提供的关系模型，而是使用针对特定存储数据类型而优化的存储模型，主要包括键值存储数据库、列存储数据库、文档型数据库、图形数据库、时间序列数据库等。它们在支持的数据类型以及如何查询数据方面往往更加具体。例如，时间序列数据库针对基于时间的数据序列进行了优化，而图形数据库则针对实体之间的加权关系进行了优化。



表 5-1 广泛应用的关系型数据库管理系统

序号	名称	优点	缺点
1	MySQL	性能卓越服务稳定，很少出现异常宕机 体积小、易于维护、安装及维护成本低 支持多种操作系统 提供多种 API 接口	不易于扩展 部分开源
2	Oracle SQL	可移植性好，能在所有主流平台上运行 安全性高，获得最高认证级别的 ISO 标准认证 性能最高，保持着开放平台下 TPC-D 和 TPC-C 世界记录。 支持多种工业标准，支持 ODBC、JDBC、OCI 等连接 完全向下兼容	对硬件的要求高 价格昂贵 操作比较复杂，管理维护麻烦
3	PostgreSQL	遵循 BSD 协议完全开源 源代码清晰、易读性高、易于二次开发 支持丰富的数据类型 支持多进程，并发处理速度快 具有强大的查询优化器，可以进行很复杂的查询处理	对于简单而繁重的读取操作，PostgreSQL 性能较低 缺乏报告和审计工具
4	SQL Server	Windows 操作系统的兼容性很好 强壮的事务处理功能，采用各种方法保证数据的完整性 支持对称多处理器结构、存储过程，并具有自主的 SQL 语言丰富的文档和社区帮助	价格较贵 仅支持 Windows 操作系统



# 1.MariaDB简介

## 1.1 数据库服务

### □ 国产数据库:



# 1.MariaDB简介

## 1.2 MariaDB简介

- MariaDB的目标是完全兼容MySQL，包括API和命令行，使之能轻松成为MySQL的替代品。
  - 随着Oracle公司的收购，MySQL成为了Oracle旗下的数据库产品。MySQL的发展进入缓慢期，再加上其它的种种原因，其更新越来越慢，并存在闭源的可能，目前仍然保存有开源版本。
  - MySQL之父Michael Widenius宣布开创MariaDB数据库管理系统。
  - MariaDB开发团队由MySQL原有的核心成员构成，遵循GNU GPLv2协议，保持开源并且无单独商业版本。
  - 目前是大多数云产品和Linux发行版的默认产品。

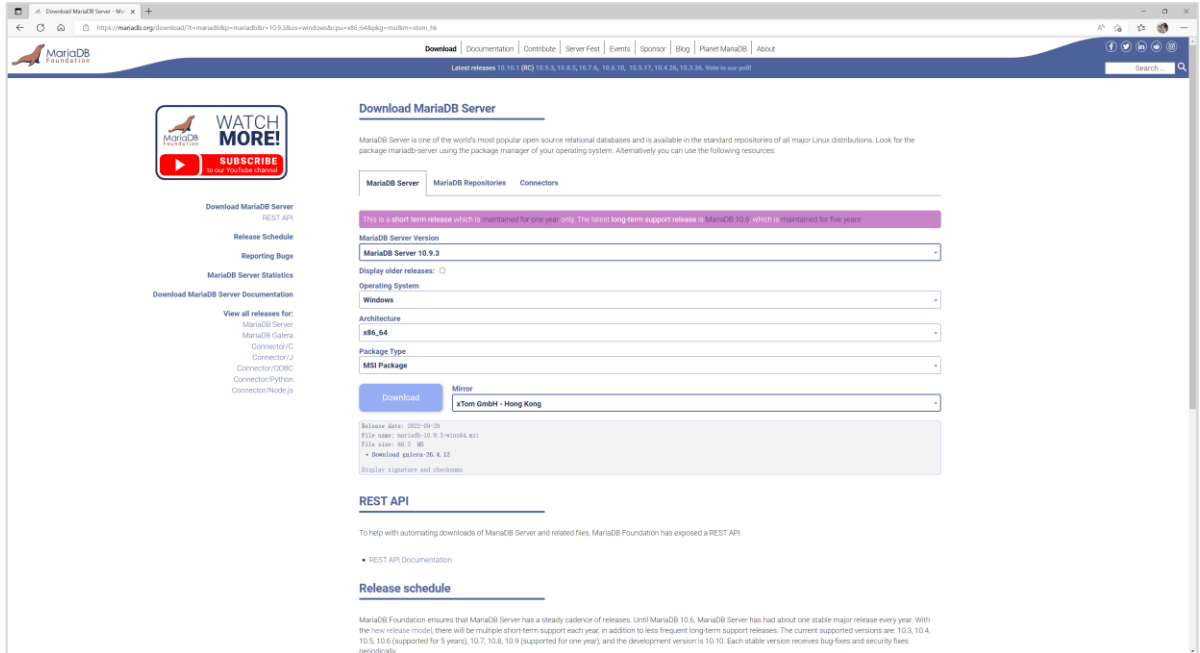


# 1.MariaDB简介

## 1.2 MariaDB简介

- MariaDB的主要特性如下。
  - 支持多种数据类型
  - 支持存储过程
  - 支持Windows操作系统、Linux操作系统
  - 支持灵活的权限和密码验证，并支持基于主机的验证
  - 支持主从集群
  - 提供事务型和非事务型存储引擎
  - 提供C、C++、Java、Perl、PHP、Python、Ruby等编程语言的API
  - 支持ODBC、JDBC等连接
  - 提供mysqladmin、mysqlcheck、mysqldump、mysqlimport等实用工具

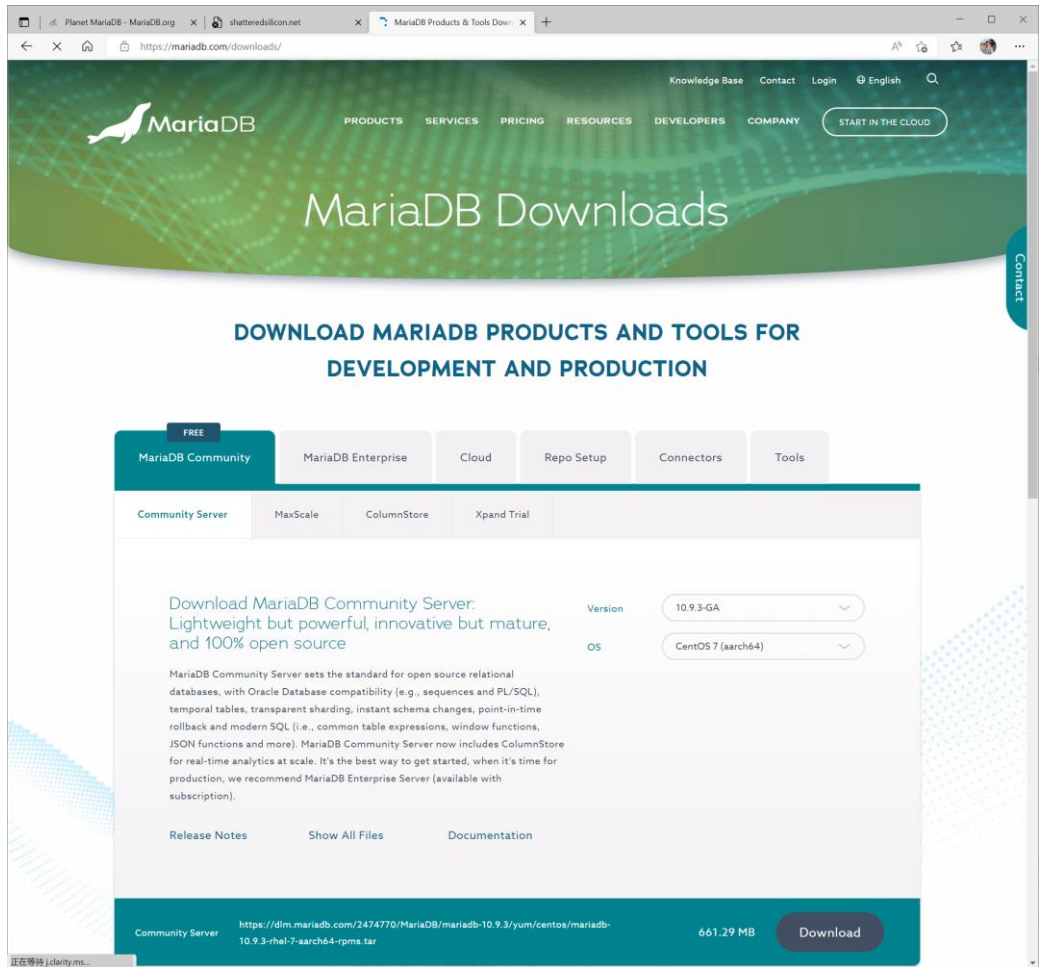




https://mariadb.org







https://mariadb.com



# 1.MariaDB简介

## 1.3 MariaDB的管理工具

- MariaDB在安装时内置了一些常用的管理工具，使用它们可以快速、便捷的管理MariaDB，主要工具如下。
  - mysqladmin
    - mysqladmin是用于执行管理操作的客户端工具。可以用来检查服务器配置和状态、创建和删除数据库等。
  - mysqlcheck
    - mysqlcheck可用于检查、修复、优化、分析数据表。
  - mysqldump
    - mysqldump用于对数据库进行备份。
  - mysqlimport
    - mysqlimport用于将sql文件导入到指定数据库中。



# 1.MariaDB简介

## 1.4 任务1

### 任务1：安装MariaDB

步骤1：创建虚拟机并完成CentOS的安装

步骤2：完成虚拟机的主机配置、网络配置及通信测试

步骤3：通过在线方式安装MariaDB

步骤4：启动MariaDB服务

步骤5：查看MariaDB运行信息

步骤6：配置mariadb服务为开机自启动



# 1.MariaDB简介

## 1.4 任务1

### 任务1：安装MariaDB

步骤7：使用MariaDB客户端初始root权限

步骤8：使用MariaDB客户端管理数据库





操作视频 / 现场演示

## ✓ 任务1: 安装MariaDB

### ■ 任务目标

- 完成MariaDB的安装与配置
- 实现数据库、数据表的创建





## 命令指南 / 操作引导

1. # Configure MariaDB
2. touch /etc/yum.repos.d/MariaDB.repo
3. echo [mariadb] > /etc/yum.repos.d/MariaDB.repo
4. echo name = MariaDB >> /etc/yum.repos.d/MariaDB.repo
5. echo baseurl = https://mirrors.aliyun.com/mariadb/yum/10.4/centos8-amd64 >> /etc/yum.repos.d/MariaDB.repo
6. echo module\_hotfixes=1 >> /etc/yum.repos.d/MariaDB.repo
7. echo gpgkey= https://mirrors.aliyun.com/mariadb/yum/RPM-GPG-KEY-MariaDB >> /etc/yum.repos.d/MariaDB.repo
8. echo gpgcheck=1 >> /etc/yum.repos.d/MariaDB.repo
  
9. # Install MariaDB
10. yum install MariaDB-server -y
  
11. # Start mariadb
12. systemctl start mariadb
  
13. # Look MariaDB
14. # systemctl status mariadb | head -10
15. systemctl status mariadb
16. # Configuration mariadb
17. systemctl enable mariadb
18. systemctl is-enabled mariadb
  
19. # Use the initial root permission for the MariaDB client
20. mysql -e "set password = password('centos@mariadb#123');"
  
21. # Use MariaDB
22. mysql -e "create database firstdb;"
23. mysql -e "show databases;"
24. mysql -e "use firstdb;create table test\_table(id int(11),name varchar(20),sex enum('0','1','2'),primary key (id));"
25. mysql -e "use firstdb;show tables;"



## 2.管理MariaDB

---



Web方式管理



客户端方式管理



运维监控





## 2.管理MariaDB

- ❑ phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.
- ❑ phpMyAdmin comes with a wide range of documentation and users are welcome to update our wiki pages to share ideas and howtos for various operations. The phpMyAdmin team will try to help you if you face any problem; you can use a variety of support channels to get help. To ease usage to a wide range of people, phpMyAdmin is being translated into 72 languages and supports both LTR and RTL languages.
- ❑ phpMyAdmin is a mature project with a stable and flexible code base; you can find out more about the project and its history and the awards it earned.
- ❑ The phpMyAdmin project is a member of Software Freedom Conservancy. SFC is a not-for-profit organization that helps promote, improve, develop, and defend Free, Libre, and Open Source Software (FLOSS) projects.

<https://www.phpmyadmin.net>





## 2.管理MariaDB

### □ Features

- Intuitive web interface
- Support for most MySQL features:
  - browse and drop databases, tables, views, fields and indexes
  - create, copy, drop, rename and alter databases, tables, fields and indexes
  - maintenance server, databases and tables, with proposals on server configuration
  - execute, edit and bookmark any SQL-statement, even batch-queries
  - manage MySQL user accounts and privileges
  - manage stored procedures and triggers
- Import data from CSV and SQL
- Export data to various formats: CSV, SQL, XML, PDF, ISO/IEC 26300 - OpenDocument Text and Spreadsheet, Word, LATEX and others
- Administering multiple servers
- Creating graphics of your database layout in various formats
- Creating complex queries using Query-by-example (QBE)
- Searching globally in a database or a subset of it
- Transforming stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link
- And much more...



## phpMyAdmin 演示服务器

您正在使用演示服务器。您可以在这里做任何事，但请不要修改 root、debian-sys-maint 和 pma 用户。详情请见 demo.phpmyadmin.net。

## 常规设置

服务器连接排序规则: utf8mb4\_unicode\_ci

[更多设置](#)

## 外观设置

语言 - Language: 中文 - Chinese simplified

主题: pmahomme

## 数据库服务器

- 服务器: phpMyAdmin demo - MariaDB (192.168.30.22 via TCP/IP)
- 服务器类型: MariaDB
- 服务器连接: SSL未被使用
- 服务器版本: 10.4.8-MariaDB-1:10.4.8+maria~stretch-log - mariadb.org binary distribution
- 协议版本: 10
- 用户: root@192.168.30.20
- 服务器字符集: UTF-8 Unicode (utf8mb4)

## 网站服务器

- nginx/1.14.2
- 数据库客户端版本: libmysql - mysqlnd 7.4.4
- PHP 扩展: mysqli curl mbstring
- PHP 版本: 7.4.4

## phpMyAdmin

- 版本信息: 5.1.0-dev, 最新稳定版本: 5.0.2
- Git 修订版本: 58eee2e 来自 master 分支  
由 Mauricio Meneghini Fauth 提交于 2020-05-28 02:46:12
- 文档
- 官方主页
- 贡献
- 获取支持
- 更新列表
- 授权

phpMyAdmin 演示服务器: 当前为 Git master 分支 RELEASE\_5\_0\_2-1771-g58eee2e371 版。

## 控制台

按 Ctrl+Enter 键执行查询



## 2.管理MariaDB

### 2.2 MySQL Workbench

- MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.
  - Design
    - MySQL Workbench enables a DBA, developer, or data architect to visually design, model, generate, and manage databases. It includes everything a data modeler needs for creating complex ER models, forward and reverse engineering, and also delivers key features for performing difficult change management and documentation tasks that normally require much time and effort.
  - Develop
    - MySQL Workbench delivers visual tools for creating, executing, and optimizing SQL queries. The SQL Editor provides color syntax highlighting, auto-complete, reuse of SQL snippets, and execution history of SQL. The Database Connections Panel enables developers to easily manage standard database connections, including MySQL Fabric. The Object Browser provides instant access to database schema and objects.

<https://www.mysql.com/products/workbench>



## 2.管理MariaDB

- MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.
  - **Administer**
    - MySQL Workbench provides a visual console to easily administer MySQL environments and gain better visibility into databases. Developers and DBAs can use the visual tools for configuring servers, administering users, performing backup and recovery, inspecting audit data, and viewing database health.
  - **Visual Performance Dashboard**
    - MySQL Workbench provides a suite of tools to improve the performance of MySQL applications. DBAs can quickly view key performance indicators using the Performance Dashboard. Performance Reports provide easy identification and access to IO hotspots, high cost SQL statements, and more. Plus, with 1 click, developers can see where to optimize their query with the improved and easy to use Visual Explain Plan.

<https://www.mysql.com/products/workbench>



## 2.管理MariaDB

- MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.
  - Database Migration
    - MySQL Workbench now provides a complete, easy to use solution for migrating Microsoft SQL Server, Microsoft Access, Sybase ASE, PostgreSQL, and other RDBMS tables, objects and data to MySQL. Developers and DBAs can quickly and easily convert existing applications to run on MySQL both on Windows and other platforms. Migration also supports migrating from earlier versions of MySQL to the latest releases.

<https://www.mysql.com/products/workbench>



Community  
Edition  
社区版本

MySQL Workbench  
Community Edition

Commercial  
Edition  
商业版本

MySQL Workbench  
Standard Edition

MySQL Workbench  
Enterprise Edition

The Community Edition is available free of charge.

The Commercial Edition provides additional Enterprise features, such as access to MySQL Enterprise Backup, MySQL Firewall, and MySQL Audit.

For a complete comparison, see <https://www.mysql.com/products/workbench/features.html>

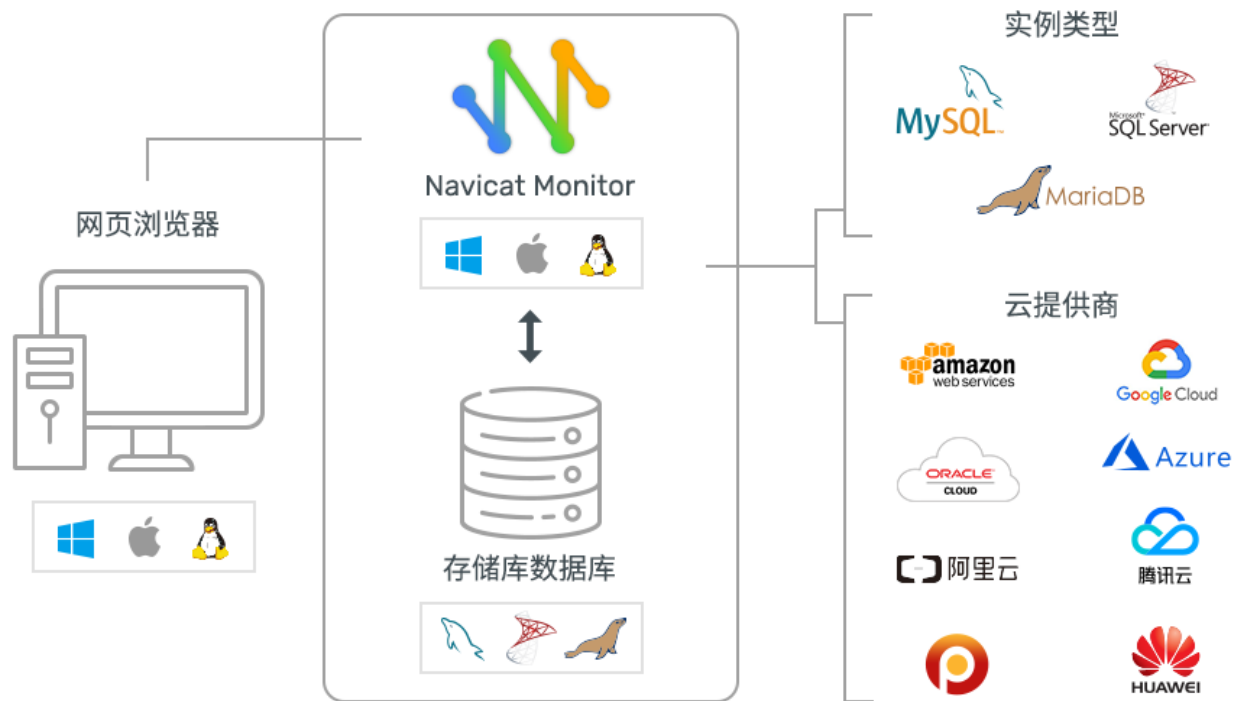




## 2.管理MariaDB

- Navicat Monitor 是一套安全、简单而且无代理的远程服务器监控工具。它具有强大的功能使你的监控发挥最大效用。受监控的服务器包括 MySQL、MariaDB 和 SQL Server, 并与 Amazon RDS、Amazon Aurora、Oracle Cloud、Microsoft Azure、阿里云、腾讯云和华为云等云数据库兼容。
- Navicat Monitor 是一款基于服务器的软件, 可以使用网页浏览器从任何地方访问。通过网络访问, 可以全天候轻松无缝地掌握全球服务器的状况。
- Navicat Monitor的核心功能
  - **实时性能监测**
    - Navicat Monitor 包含一组丰富的实时和历史图表, 可深入了解服务器统计的详细信息。它提供了有关每个服务器负载和性能的详细查看, 包括其可用性、磁盘使用率、网络 I/O、表锁等。
    - 除了可以轻松跟踪服务器之间的偏差和流量外, 还可研究可能的解决方案并调整服务器设置。
  - **无代理架构**
    - Navicat Monitor 应用无代理架构监控MySQL、MariaDB 和 SQL Server, 并定期收集指标。
    - 通过 SSH 或 SNMP 收集进程指标, 如 CPU 负载、RAM 使用率以及各种其他资源。
    - Navicat Monitor 可以安装在任何本地电脑或虚拟机上, 并且不需要在被监控的服务器上安装任何软件。
    - Navicat Monitor 需要一个存储库来存储用于历史分析的警报和指标。存储库数据库可以是现有的 MySQL、MariaDB、PostgreSQL、SQL Server 或 Amazon RDS 实例。

<http://www.navicat.com.cn/products/navicat-monitor>





## 2.管理MariaDB

### 2.4 任务2

任务2：使用phpMyAdmin实现Web方式管理

任务3：使用MySQL WorkBench实现运维管理

任务4：使用Navicat Monitor监控数据库服务器



## 2.管理MariaDB

### 2.4 任务2

#### 任务2：使用phpMyAdmin实现Web方式管理

步骤1：明确phpMyAdmin安装要求

步骤2：完成LAMP环境配置

步骤3：验证系统环境是否满足部署要求

步骤4：获取phpMyAdmin程序

步骤5：配置Apache发布网站

步骤6：安全配置



## 2.管理MariaDB

### 2.4 任务2

#### 任务2：使用phpMyAdmin实现Web方式管理

步骤7：使用phpMyAdmin连接MariaDB服务器

步骤8：创建数据库

步骤9：创建数据表

步骤10：数据操作（查询、插入、修改、删除等）

步骤11：数据库备份与恢复（数据导入、数据导出）

步骤12：数据库监控





操作视频 / 现场演示



- ✓ 任务2：使用phpMyAdmin实现Web方式管理
  - 任务目标
    - 完成phpMyAdmin的部署
    - 实现Web方式管理MariaDB数据库





## 命令指南 / 操作引导

1. # Apache installation and configuration
2. yum install -y httpd
3. systemctl start httpd
4. systemctl enable httpd
5. # Complete the installation of PHP and support modules
6. yum install -y php php-mysqlnd php-json
7. # Verify that the system environment meets deployment requirements
8. httpd -v
9. php -v
10. mysql --version
11. # Install wget
12. yum install -y wget
13. # Get phpMyAdmin
14. wget https://files.phpmyadmin.net/phpMyAdmin/5.0.1/phpMyAdmin-5.0.1-all-languages.tar.gz
15. # Unzip the phpMyAdmin installer to /var/www directory
16. yum install -y tar
17. tar -zxvf phpMyAdmin-5.0.1-all-languages.tar.gz -C /var/www/
18. cd /var/www/
19. mv phpMyAdmin-5.0.1-all-languages phpmyadmin
20. # Set phpmyadmin
21. chown -R apache:apache /var/www/phpmyadmin
22. # Configure Apache
23. sed -i 's/DocumentRoot "\var\www\html"/DocumentRoot "\var\www\phpmyadmin"/g' /etc/httpd/conf/httpd.conf
24. sed -i 's/<Directory "\var\www\html">/<Directory "\var\www\phpmyadmin">/g' /etc/httpd/conf/httpd.conf
25. rm -f /etc/httpd/conf.d/welcome.conf
26. systemctl restart httpd
  
27. # Turn off the firewall
28. systemctl stop firewalld
29. setenforce 0



# 在phpMyAdmin上操作



## 2.管理MariaDB

### 2.5 任务3

#### 任务3：使用MySQL WorkBench实现运维管理

步骤1：配置MariaDB开启远程访问

步骤2：在本地主机安装MySQL WorkBench

步骤3：使用MySQL WorkBench连接MariaDB

步骤4：数据库应用：创建数据库、数据表、插入数据等

步骤5：数据库的导出与导入

步骤6：使用MySQL WorkBench监控MariaDB服务器





操作视频 / 现场演示

- ✓ 任务3：使用MySQL WorkBench实现运维管理
  - 任务目标
    - 在本地主机安装MySQL WorkBench
    - 使用MySQL WorkBench管理MariaDB





# 创建MariaDB远程管理权限



# 在MySQL WorkBench上操作



## 2.管理MariaDB

### 2.6 任务4

#### 任务4：使用Navicat Monitor监控数据库服务器

步骤1：配置MariaDB开启远程访问

步骤2：在本地主机安装Navicat Monitor

步骤3：使用Navicat Monitor监控MariaDB数据库服务器

步骤4：实时监控

步骤5：历史监控与分析

步骤6：综合评估MariaDB数据库服务器的运行性能





操作视频 / 现场演示

- ✓ 任务4：使用Navicat Monitor监控数据库服务器
  - 任务目标
    - 完成Navicat Monitor的安装
    - 使用Navicat Monitor监控MariaDB



# 在Windows上部署Navicat Monitor



# 在Linux上部署Navicat Monitor





## 命令指南 / 操作引导

1. #!/usr/bin/bash
2. # Configure MariaDB YUM
3. touch /etc/yum.repos.d/MariaDB.repo
4. echo [mariadb] > /etc/yum.repos.d/MariaDB.repo
5. echo name = MariaDB >> /etc/yum.repos.d/MariaDB.repo
6. echo baseurl = https://mirrors.aliyun.com/mariadb/yum/10.4/centos8-amd64 >> /etc/yum.repos.d/MariaDB.repo
7. echo module\_hotfixes=1 >> /etc/yum.repos.d/MariaDB.repo
8. echo gpgkey = https://mirrors.aliyun.com/mariadb/yum/RPM-GPG-KEY-MariaDB >> /etc/yum.repos.d/MariaDB.repo
9. echo gpgcheck = 1 >> /etc/yum.repos.d/MariaDB.repo
  
10. # Install MariaDB
11. yum install -y MariaDB-server
12. # Start mariadb
13. systemctl start mariadb
14. # Configure the mariadb service and boot
15. systemctl enable mariadb
16. systemctl is-enabled mariadb
17. #MariaDB Initial root permission
18. mysql -e "set password = password('centos@mariadb#123');"
  
19. #install Navicat Monitor 2.4 On CentOS 8
20. yum install -y wget
21. wget http://download3.navicat.com/monitor2-download/el7-repo/navicatmonitor-2.4.5-0.el7.x86\_64.rpm
22. yum localinstall -y navicatmonitor-2.4.5-0.el7.x86\_64.rpm
  
23. #tart Navicat Monitor
24. /etc/init.d/navicatmonitor start
25. setenforce 0
26. systemctl stop firewalld
27. systemctl disable firewalld



# 初始化安装Navicat Monitor





# 使用Navicat Monitor

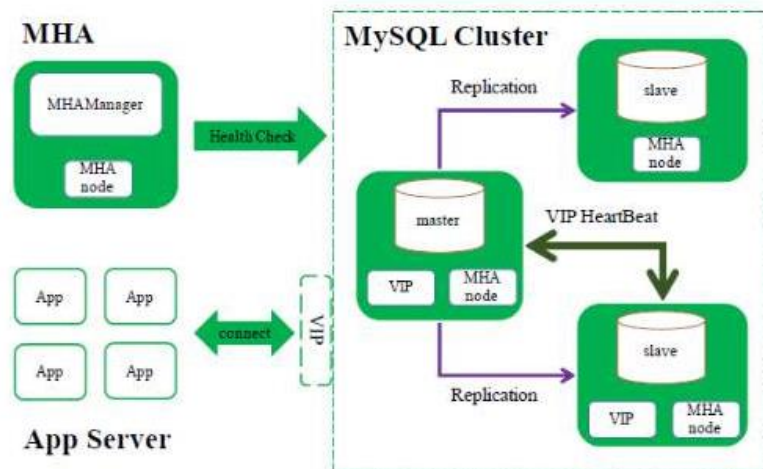
(监控实例添加后需耐心等待)



## 3.高可用

### 3.1 数据库集群

- 数据库集群就是利用两台或者多台数据库服务器：
  - 构成一个虚拟单一数据库逻辑映像，
  - 像单个数据库系统那样，提供透明的数据服务。



## 3.高可用

### 3.1 数据库集群

#### □ 使用数据库集群的优势

##### ■ 高可用性

- 数据库集群可以实现在主服务器上完成所有写入和更新操作，在一个或多个从服务器上完成读操作，以提高性能。

##### ■ 负载均衡

- 在数据库主节点发生故障时，从节点能够自动接管主数据库，从而保证业务不中断和数据完整性。

##### ■ 备份协助

- 数据库备份可能会对数据库服务器产生重大影响，从服务器运行备份能够很好的规避该问题，从服务器执行备份不会影响到主服务器。

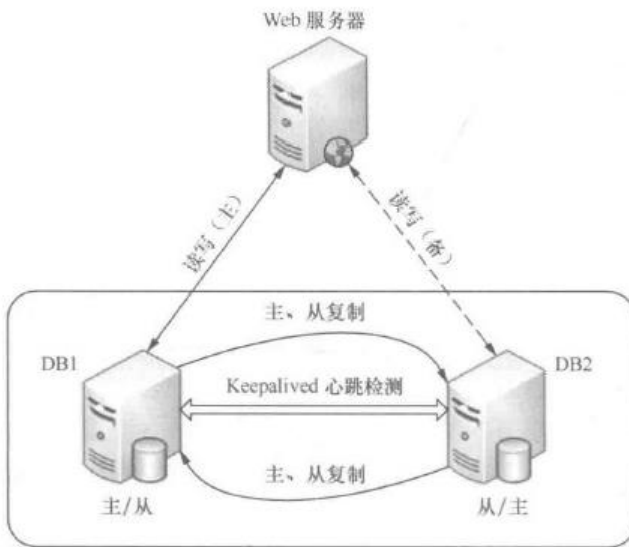


## 3.高可用

### 3.1 数据库集群

#### □ 主从模式的工作原理

- 主数据库开启二进制日志记录，将所有操作作为binlog事件写入二进制日志中。
- 从数据库读取主数据库的二进制日志并存储到本地的中继日志（relay log），然后通过中继日志重现主数据库的操作，从而保持数据的一致性。



## 3.高可用

### 3.2 任务5

#### 任务5：通过主从集群实现MariaDB的高可用

步骤1：创建第一台数据库服务器

步骤2：配置第一台数据库服务器为主节点

步骤3：创建第二台数据库服务器

步骤4：配置第二台数据库服务器为从节点

步骤5：启动主从集群同步服务

步骤6：验证主从集群同步状态



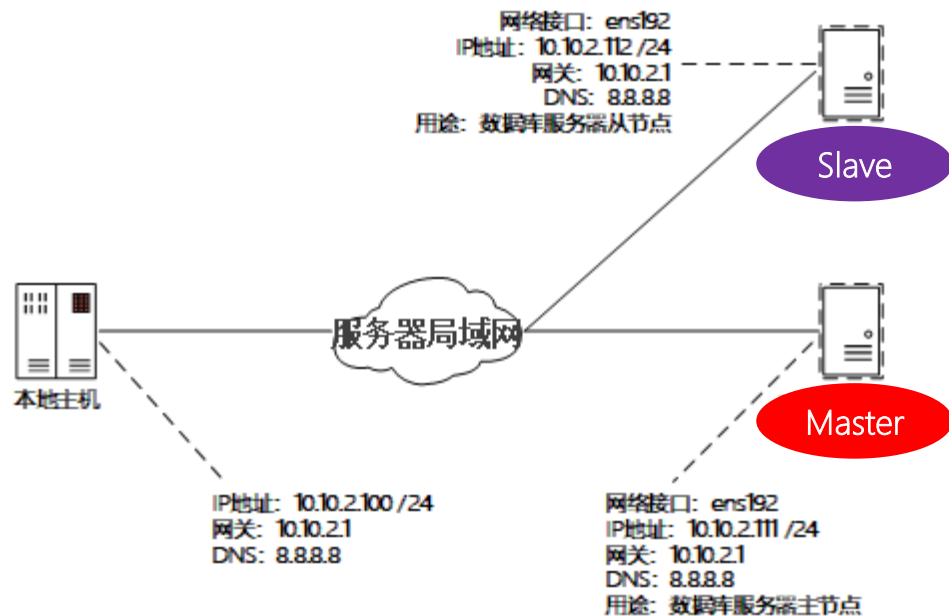


表 5-4-1 服务器规划表

序号	虚拟机名称	业务名称	作用
1	VM-Project-05-Task-02-10.10.2.111	数据库服务器-1	作为 MariaDB 集群主节点
2	VM-Project-05-Task-03-10.10.2.112	数据库服务器-2	作为 MariaDB 集群从节点





操作视频 / 现场演示



- ✓ 任务5：通过主从集群实现MariaDB的高可用
  - 任务目标
    - 完成MariaDB主从双机系统的部署
    - 对MariaDB主从双机系统进行测试



# 在MariaDB-Master上操作







## 命令指南 / 操作引导

1. # Create YUM
2. touch /etc/yum.repos.d/MariaDB.repo
3. # Write MariaDB YUM source configuration information to a file
4. echo [mariadb] > /etc/yum.repos.d/MariaDB.repo
5. echo name = MariaDB >> /etc/yum.repos.d/MariaDB.repo
6. echo baseurl = https://mirrors.aliyun.com/mariadb/yum/10.4/centos8-amd64 >> /etc/yum.repos.d/MariaDB.repo
7. echo module\_hotfixes=1 >> /etc/yum.repos.d/MariaDB.repo
8. echo gpgkey= https://mirrors.aliyun.com/mariadb/yum/RPM-GPG-KEY-MariaDB >> /etc/yum.repos.d/MariaDB.repo
9. echo gpgcheck=1 >> /etc/yum.repos.d/MariaDB.repo
10. # Install MariaDB
11. yum install MariaDB-server -y
12. # Start MariaDB
13. systemctl start mariadb
14. # View MariaDB INFO
15. systemctl status mariadb | head -10
16. # Configure the mariadb service as A boot-on
17. systemctl enable mariadb
18. # Verify that the mariadb service is boot-on
19. systemctl is-enabled mariadb
20. systemctl stop firewalld
21. setenforce 0





## 命令指南 / 操作引导

1. # Create and authorize accounts for synchronization
2. `mysql -e "CREATE USER 'replication_user'@'%' IDENTIFIED BY 'centos@mariadb#123';"`
3. `mysql -e "GRANT REPLICATION SLAVE ON *.* TO 'replication_user'@'%';"`
4. `mysql -e "show master status;"`
  
5. `echo "Please record file value and Position value, Press 【Enter】 to continue"`
6. `read -n 1`
  
7. `echo "Please execute Script 2 on server-2 and press 【Enter】 to continue after execution"`
8. `read -n 1`
  
  
  
  
  
  
  
  
  
  
9. # Add data to the database server primary node
10. `mysql -e "create database fourthdb;"`
11. `mysql -e "use fourthdb;create table test_table(id int(11),name varchar(20),sex enum('0','1','2'),primary key (id));"`
12. `mysql -e "use fourthdb;insert into test_table ( id, name,sex ) VALUES ( 1, 'name1','0' );"`
13. `mysql -e "use fourthdb;select * from test_table;"`



# 在MariaDB-Slave上操作





## 命令指南 / 操作引导

1. # Create a YUM source file
2. touch /etc/yum.repos.d/MariaDB.repo
3. # Write MariaDB YUM source configuration information to a file
4. echo [mariadb] > /etc/yum.repos.d/MariaDB.repo
5. echo name = MariaDB >> /etc/yum.repos.d/MariaDB.repo
6. echo baseurl = https://mirrors.aliyun.com/mariadb/yum/10.4/centos8-amd64 >> /etc/yum.repos.d/MariaDB.repo
7. echo module\_hotfixes=1 >> /etc/yum.repos.d/MariaDB.repo
8. echo gpgkey= https://mirrors.aliyun.com/mariadb/yum/RPM-GPG-KEY-MariaDB >> /etc/yum.repos.d/MariaDB.repo
9. echo gpgcheck=1 >> /etc/yum.repos.d/MariaDB.repo
10. # Install MariaDB
11. yum install MariaDB-server -y
12. # Start mariadb
13. systemctl start mariadb
14. # Look MariaDB
15. systemctl status mariadb | head -10
16. # Configure the mariadb service as A boot-on
17. systemctl enable mariadb
18. # Verify that the mariadb service is boot-on
19. systemctl is-enabled mariadb
20. systemctl stop firewalld
21. setenforce 0





## 命令指南 / 操作引导

1. # Configure the first database server as the primary node
2. echo [mariadb] >> /etc/my.cnf
3. echo log-bin >> /etc/my.cnf
4. echo server\_id=2 >> /etc/my.cnf
5. # Restart mariadb
6. systemctl restart mariadb
7. read -p "Please enter the server-1 IP address:" server1\_ip
8. read -p "Please enter the File value:" log\_file
9. read -p "Please enter the Position value:" log\_position
  
10. # Set the option to connect the primary server from the server
11. mysql -e "CHANGE MASTER TO
12. MASTER\_HOST='\$server1\_ip',
13. MASTER\_USER='replication\_user',
14. MASTER\_PASSWORD='centos@mariadb#123',
15. MASTER\_PORT=3306,
16. MASTER\_LOG\_FILE='\$log\_file',
17. MASTER\_LOG\_POS=\$log\_position,
18. MASTER\_CONNECT\_RETRY=10;"
  
19. # Start replication
20. mysql -e "start slave;"
21. # View synchronization status by looking at the node
22. mysql -e "show slave status \G"
23. echo "Please execute Script 1 on server-1 and press Enter to continue when you are finished"
24. read -n 1
  
25. # Verify that the server is synchronized from the database server
26. mysql -e "show databases;"
27. mysql -e "use fourthdb;show tables;"
28. mysql -e "use fourthdb;select \* from test\_table;"



# 验证MariaDB的主从集群可用性



