

第13章 LAMP环境配置

讲师：武永亮

课程目标

- 了解Web编程的整体架构
- 掌握LAMP的环境配置
- 了解LAMP应用案例

课程内容



Linux下的脚本语言

- 常用的脚本语言
 - ✓ Bash —— 系统必备
 - ✓ PHP —— 简明单纯
 - ✓ Perl —— 凝练晦涩
 - ✓ Python —— 优雅明晰
 - ✓ Ruby —— 精巧灵动
- 脚本语言的主要用途
 - ✓ 系统管理的自动化
 - ✓ 动态Web编程

CentOS 下的脚本语言

- 脚本语言的安装

- ✓ CentOS 官方仓库提供了 Perl/Python/PHP/Ruby
- ✓ 可以使用 yum 安装

- 脚本语言的模块管理工具

- ✓ PHP : pear、pecl
- ✓ Perl : cpan
- ✓ Python : easy_install或pip
- ✓ Ruby : gem

关系数据库系统

动态网站数据库

- 动态Web站点并非一定要有数据库支持，但大多数应用需要数据库支持
- 动态网站常用的开源数据库
 - ✓ 关系型数据库（ Relational database ）
 - MySQL : <http://www.mysql.com>
 - PostgreSQL : <http://www.postgresql.org/>
 - SQLite : <http://sqlite.org/>
 - ✓ 面向文档的数据库（ Document-oriented database ）
 - mongoDB : <http://www.mongodb.org/>
 - CouchDB : <http://couchdb.apache.org/>

MySQL/MariaDB数据库简介

- MySQL是一个单进程多线程、支持多用户、基于客户机/服务器（ Client/Server简称C/S ）的关系数据库管理系统。
 - ✓ 由一个服务器守护程序mysqld和很多不同的客户程序和库组成
 - ✓ 支持FreeBSD、Linux、MAC、Windows等多种操作系统平台
- MySQL由瑞典MySQL AB公司开发。
 - ✓ 2008年1月MySQL被美国的SUN公司收购。
 - ✓ 2009年4月SUN公司又被美国的甲骨文（ Oracle ）公司收购。

MySQL数据库特点

- 可以同时处理几乎不限数量的用户
- 可以处理拥有上千万条记录的大型数据
- 简单有效的用户特权系统
- 支持常见的SQL语句规范
- 可移植行高，安装简单，小巧
- 良好的运行效率，有丰富信息的网络支持
- 相对其他大型数据库而言调试、管理，优化简单
- 提供多种存储引擎支持，如（ MyISAM、InnoDB等 ）。MySQL5.5默认使用高效的事务引擎InnoDB
- 支持复制功能（ Replication ）功能，为高可用的MySQL系统提供了可靠方案

MySQL的版本

- 根据MySQL的开发情况，可以将MySQL分为
 - ✓ Alpha、 Beta、 Gamma
 - ✓ 和Generally Available (GA)
- MySQL官方为Linux下的每一种GA版本提供了
 - ✓ RPM包、二进制包和源码包
 - ✓ 为RedHat系列发型版提供了YUM仓库
 - <http://repo.mysql.com/yum/>
 - ✓ 为Debian系列发型版提供了APT仓库
 - <http://repo.mysql.com/apt/>

MySQL的表类型和存储引擎

- MySQL的表类型由存储引擎（ Storage Engines ）决定，针对不同的存储引擎可以指定相应不同的配置
- MySQL 的表主要支持六种类型
 - ✓ 事务安全型(transaction-safe)：InnoDB和BDB
 - ✓ 非事务安全型(non-transaction-safe)：MYISAM、HEAP、ISAM、MERGE
- 显示当前数据库支持的存储引擎：
 - ✓ show engines;
- MySQL 5.5/5.6 的默认存储引擎是InnoDB

MySQL的存储引擎比较

特点	Myisam	InnoDB	BDB	Memory	Archive
批量插入的速度	高	低	高	高	非常高
事务安全		支持	支持		
全文索引	支持				
锁机制	表锁	行锁	页锁	表锁	行锁
存储限制	没有	64TB	没有	有	没有
B树索引	支持	支持	支持	支持	
哈希索引		支持		支持	
集群索引		支持			
数据缓存		支持		支持	
索引缓存	支持	支持		支持	
数据可压缩	支持				支持
空间使用	低	高	低	N/A	非常低
内存使用	低	高	低	中等	低
支持外键		支持			

MyISAM 的特点

- 数据存储方式简单，使用 B+ Tree 进行索引
- 使用三个文件定义一个表：.MYI .MYD .frm
- 少碎片、支持大文件、能够进行索引压缩
- 二进制层次的文件可以移植 (Linux 、 Windows)
- 访问速度快，是所有MySQL文件引擎中速度最快的
- 不支持一些数据库特性，比如 事务、外键约束等
- 使用表级锁（ Table level lock ），性能稍差，更适合读取多的操作
- 表数据容量有限，一般建议单表数据量介于 50w–200w

InnoDB 的特点

- 使用表空间（ Table Space ）的方式进行数据存储 (ibdata1, ib_logfile0)
- 支持 事务、外键约束等数据库特性
- 使用行级锁（ Rows level lock ），读写性能都非常优秀
- 能够承载大数据量的存储和访问
- 拥有自己独立的缓冲池，能够缓存数据和索引
- 在关闭自动提交的情况下，与MyISAM引擎速度差异不大

存储引擎的选择

- 如果应用不需要事务，处理的只是基本的CRUD操作，那么MyISAM是不二选择
 - ✓ MyISAM不支持事务、也不支持外键，但其访问速度快
- 一般来说，如果需要事务支持，并且有较高的并发读写频率，InnoDB是不错的选择
 - ✓ InnoDB存储引擎提供了具有提交、回滚和崩溃恢复能力的事务安全。
 - ✓ 比起MyISAM存储引擎，InnoDB写的处理效率差一些并且会占用更多的磁盘空间以保留数据和索引。

MariaDB

- 由原来 MySQL 的作者 Michael Widenius 创办的公司所开发的免费开源的数据库服务
- 是采用 Maria 存储引擎的 MySQL 分支版本
- 与 MySQL 相比较，MariaDB 更强的地方在于，二者支持不同的引擎。通常可以通过 `show engines` 命令来查看两种数据库服务器支持的不同的引擎。
- CentOS 7 已默认提供了 MariaDB 而非 MySQL

安装MariaDB服务

- 安装

- ✓ # yum install mariadb mariadb-server

- 启动

- ✓ # systemctl start mariadb

- ✓ # systemctl enable mariadb

- 设置MySQL的root用户口令

- ✓ # yum install pwgen

- ✓ # pwgen -1 20

- ✓ Aed7ahBuu7ru2Wooyohg

- ✓ # mysqladmin -u root password 'Aed7ahBuu7ru2Wooyohg'

MariaDB服务概览

- 软件包： mariadb-server
- 服务类型： 由Systemd启动的守护进程
- 配置单元： /usr/lib/systemd/system/mariadb.service
- 端口： 3306
- 配置： /etc/my.cnf
- 相关软件包： mariadb、 php-mysql、 perl-DBD-mysql

MariaDB/MySQL的配置文件

- 配置文件为/etc/my.cnf
- MariaDB/MySQL的详细配置参数的解释请参考MariaDB/MySQL手册
- 建议DBA从头编制适合特定应用的配置文件

MariaDB/MySQL的配置原则

- 针对不同的服务器硬件进行合理配置（如CPU核数、内存大小等）
- 针对是否是独立的服务器进行合理配置（若Mysql服务器还同时运行其他服务，就该适当削减其资源占用）
- 针对 MyISAM 或 InnoDB 不同引擎进行不同定制性配置
- 针对不同的应用情况进行合理配置，尽量使应用本身达到最合理的情况

MySQL常用的公共配置选项

选项	缺省值	推荐值	说明
<code>max_connections</code>	100	1024	MySQL服务器同时处理的数据库连接的最大数量
<code>query_cache_size</code>	0（不打开）	16M	查询缓存区的最大长度，按照当前需求，一倍一倍增加，本选项比较重要
<code>sort_buffer_size</code>	512K	16M	每个线程的排序缓存大小，一般按照内存可以设置为2M以上，推荐是16M，该选项对排序order by，group by起作用
<code>record_buffer</code>	128K	16M	每个进行一个顺序扫描的线程为其扫描的每张表分配这个大小的一个缓冲区，可以设置为2M以上
<code>table_cache</code>	64	512	为所有线程打开表的数量。增加该值能增加mysqld要求的文件描述符的数量。MySQL对每个唯一打开的表需要2个文件描述符。

常用的 MyISAM 配置选项

选项	缺省值	推荐值	说明
key_buffer_size	8M	256M	用来存放索引区块的缓存值， 建议128M以上， 不要大于内存的30%
read_buffer_size	128K	16M	用来做MyISAM表全表扫描的缓冲大小. 为从数据表顺序读取数据的读操作保留的缓存区的长度
myisam_sort_buffer_size	16M	128M	设置, 恢复, 修改表的时候使用的缓冲大小， 值不要设的太大

常用的 InnoDB 配置选项

选项	缺省值	推荐值	说明
<code>innodb_buffer_pool_size</code>	32M	1G	InnoDB使用一个缓冲池来保存索引和原始数据，这里你设置越大，你在存取表里面数据时所需要的磁盘I/O越少，一般是内存的一半，不超过2G，否则系统会崩溃，这个参数非常重要
<code>innodb_additional_mem_pool_size</code>	2M	128M	InnoDB用来保存 metadata 信息，如果内存是4G，最好本值超过200M
<code>innodb_flush_log_at_trx_commit</code>	1	0	0 代表日志只大约每秒写入日志文件并且日志文件刷新到磁盘； 1 为执行完没执行一条SQL马上commit； 2 代表日志写入日志文件在每次提交后，但是日志文件只有大约每秒才会刷新到磁盘上。对速度影响比较大，同时也关系数据完整性
<code>innodb_log_file_size</code>	8M	256M	在日志组中每个日志文件的大小，一般是 <code>innodb_buffer_pool_size</code> 的25%，官方推荐是 <code>innodb_buffer_pool_size</code> 的 40-50%，设置大一点来避免在日志文件覆写上不必要的缓冲池刷新行为
<code>innodb_log_buffer_size</code>	128K	8M	用来缓冲日志数据的缓冲区的大小。推荐是8M，官方推荐该值小于16M，最好是 1M-8M 之间

配置MySQL (/etc/my.cnf)

```
[mysqld]
# GENERAL
datadir = /var/lib/mysql
socket = /var/lib/mysql/mysql.sock
pid_file = /var/lib/mysql/mysql.pid
user = mysql
port = 3306
storage_engine = InnoDB
# INNODB
innodb_buffer_pool_size = <value>
innodb_log_file_size = <value>
innodb_file_per_table = 1
innodb_flush_method = O_DIRECT
# MyISAM
key_buffer_size = <value>
# LOGGING
log_error = /var/lib/mysql/mysql-error.log
log_slow_queries = /var/lib/mysql/mysql-slow.log
# OTHER
tmp_table_size = 32M
max_heap_table_size = 32M
query_cache_type = 0
query_cache_size = 0
max_connections = <value>
thread_cache_size = <value>
table_cache_size = <value>
open_files_limit = 65535

[client]
socket = /var/lib/mysql/mysql.sock
port = 3306
```


MySQL参考资源

● 书籍

✓ 高性能MySQL (第3版)

- <http://www.highperfmysql.com/>

✓ Effective MySQL 系列 (3册 , 有中译本)

- <http://effectivemysql.com/>

● 配置

✓ 在线MySQL配置向导

- <https://tools.percona.com/wizard>

Web服务器

- Apache
- Tomcat
- Nginx

Apache的动态网站技术

- CGI
- Apache的第三方脚本语言模块
- FastCGI

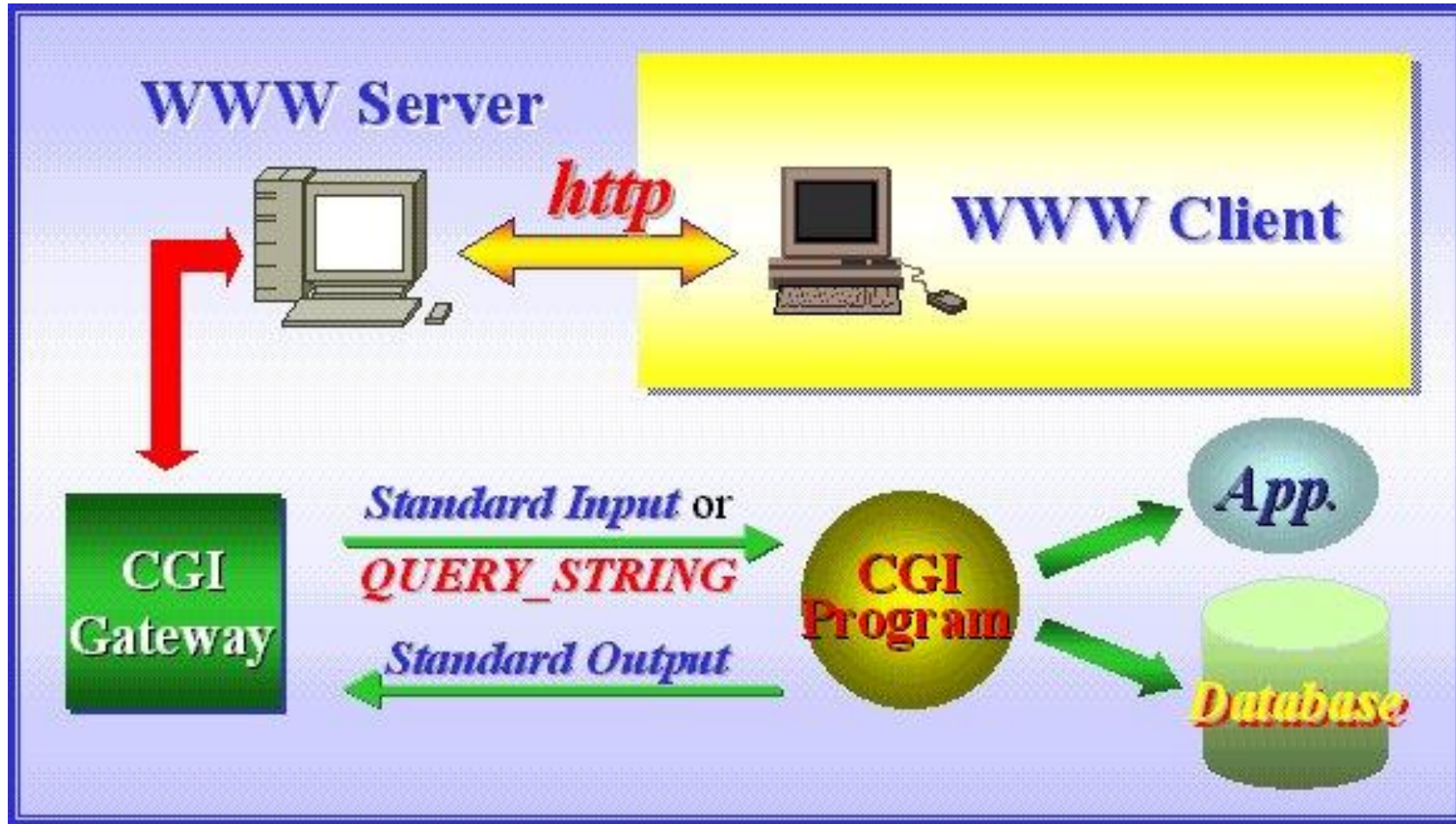
CGI 简介

- CGI (Common Gateway Interface , 通用网关接口) 是一个连接外部应用程序到 HTTP 服务器的标准
- CGI 定义了 Web 服务器与外部内容生成程序 (通常称为 CGI 脚本或 CGI 程序) 之间交互的方法 , 即 : 一种基于浏览器的输入、在 Web 服务器上运行的程序方法 , 从而实现动态 Web 的功能

CGI 程序

- CGI 程序可以用任何一种语言编写
 - ✓ 只要这种语言具有标准输入、输出和环境变量。
 - ✓ 例如：perl、python、ruby、php、bash、C 等
- CGI 程序通常是挂平台的
 - ✓ 可以运行在类 UNIX 和 Windows 等众多平台的服务器上
 - ✓ 实现同一功能的程序在不同平台上可能会有细微差异

CGI 的工作原理



CGI的优缺点

- 优点

- ✓ 安全性好
- ✓ 用C语言写的CGI程序,编译后的运行速度比脚本运行速度要快

- 缺点

- ✓ 需要开独立进程（ fork-and-execute 模式 ）来处理用户请求，密集请求的情况下容易崩溃
- ✓ 维护成本比脚本语言高
- ✓ 通常CGI程序使用Perl编写，其语法相对复杂

解决CGI的低效率

- 为了适应密集请求（高负载）型的Web服务器
- 解决CGI的低执行效率的方法
 - ✓ 使用Apache的第三方脚本语言模块
 - 模块当 Apache 运行后就常驻内存
 - 不会像 CGI 那样每次都要花费时间去 fork 一次
 - ✓ 使用FastCGI技术
 - 是一种常驻（Long-Live）型的 CGI
 - 类似于系统守护进程
 - 可以一直执行着为来自服务器的请求提供服务
 - 只要激活后，不会每次都要花费时间去 fork 一次

FastCGI的优点



- 稳定性：FastCGI 是以独立的进程池运行来 CGI
 - ✓ 单独一个进程死掉,系统可以很轻易的丢弃,然后重新分配新的进程来运行之
- 安全性：FastCGI 和宿主服务器完全独立
 - ✓ 即使 FastCGI 僵死也不会导致服务器宕机
- 扩展性：FastCGI是一个中立的技术标准
 - ✓ 可以支持任何语言写的处理程序，如：PHP、Perl、Python、Ruby、Java、C/C++ 等

FastCGI的优点（续）

● 高性能

- ✓ FastCGI 将动态逻辑的处理从 Web 服务器中分离出来
 - 大负荷的 I/O 处理还是留给宿主服务器
 - 宿主服务器可以一心一意作 I/O 处理
 - 大量的图片等静态 I/O 处理完全不需要逻辑程序的参与
- ✓ 可以让 Web 服务器运行多个 FastCGI 应用程序的副本来提高性能
- ✓ FastCGI 可以很有效地利用内存来作缓存来提高性能

Apache的脚本语言模块

- PHP : mod_php
✓ <http://www.php.net/>
- Perl : mod_perl
✓ <http://perl.apache.org/>
- Python : mod_python
✓ <http://www.modpython.org/>
- Ruby : mod_passenger
✓ <http://www.modrails.com/>

课程内容

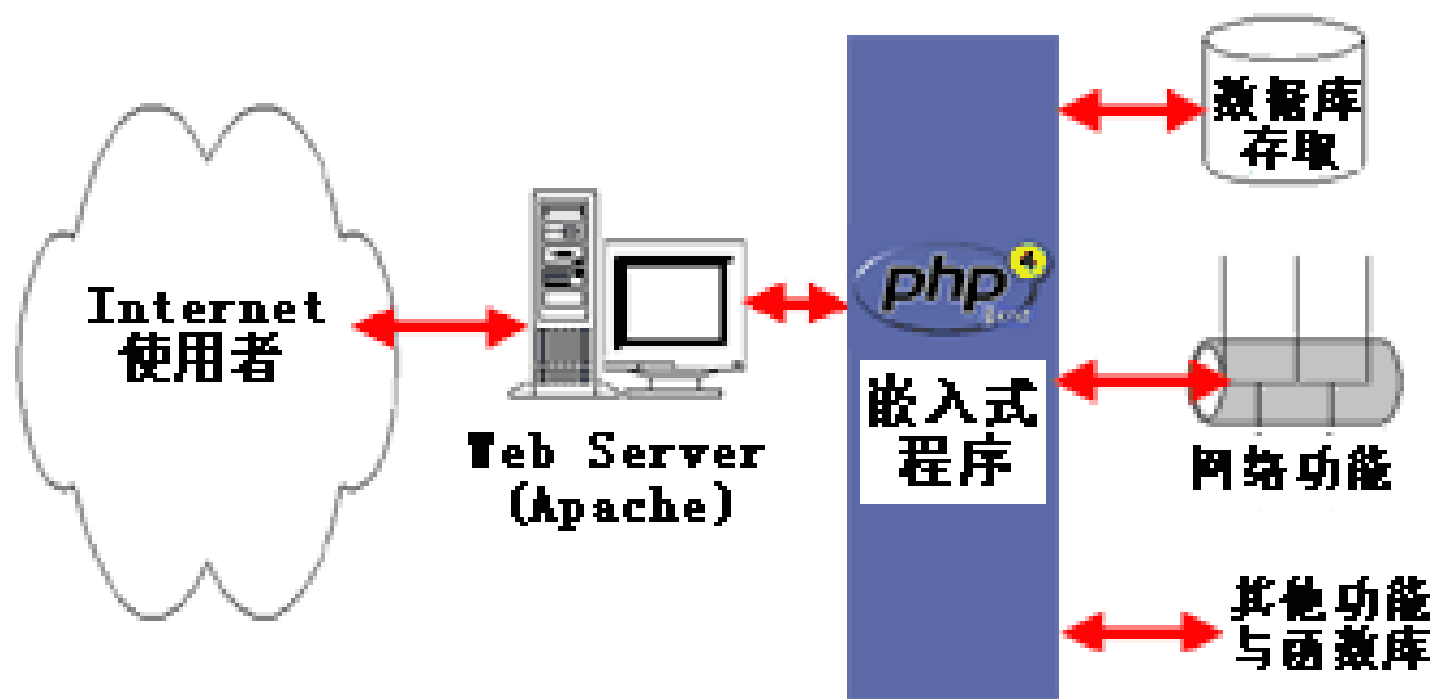


LAMP简介

- LAMP是首字母缩略语
 - ✓ L : Linux 操作系统
 - ✓ A : Apache Web 服务器
 - ✓ M : MySQL 数据库
 - ✓ P : PHP、Perl、Python 或 Ruby 脚本语言
- LAMP的特点
 - ✓ 开放灵活、开发迅速、部署方便
 - ✓ 高可配置、安全可靠、成本低廉等
- 与Java平台和.NET平台鼎足三分
 - ✓ 尤其受中小企业的欢迎

狭义LAMP

- Linux+Apache+MySQL+PHP
- 是最常用的开源平台组合



基于脚本语言的知名Web框架

语言	Web框架	主页
PHP	symfony	http://www.symfony-project.org/
	Zend	http://framework.zend.com/
	CodeIgniter	http://codeigniter.com/
	CakePHP	http://cakephp.org/
Python	Django	http://www.djangoproject.com/
	Pylons	http://pylonshq.com/
	TurboGears	http://www.turbogears.org/
	Grok	http://grok.zope.org/
Ruby	Ruby on Rails	http://www.rubyonrails.org/
Perl	Catalyst	http://www.catalystframework.org/

安装配置LAMP环境

- 安装Apache
- 安装配置PHP
- 安装MySQL（已完成）
- 安装PHP-MySQL支持
- 测试Apache、PHP、MySQL环境

安装Apache

- 安装Apache

```
# yum install httpd
```

- Apache命令

```
systemctl start httpd    //启动apache  
systemctl stop httpd     //停止apache  
systemctl restart httpd  //重启apache  
systemctl enable httpd   //设置apache开机启动
```

安装PHP

- 安装PHP及其相关的软件包

```
# yum install php php-cli php-pear  
# yum install php-pdo php-mysql  
# yum install php-mcrypt php-mbstring  
# yum install php-xml php-pecl-yaml  
# yum install php-gd php-pecl-imagick  
# yum install php-pecl-apc  
# yum install php-pecl-memcached php-pecl-redis
```

查看PHP的配置

- 查看PHP的配置
 - ✓ 查看PHP已加载的模块
 - ✓ # php -m
 - ✓ 显示phpinfo()的信息输出
 - ✓ # php -i
- 查看PHP的配置文件

```
# less /etc/php.ini
# ls /etc/php.d
apc.ini    json.ini  mysql.ini  pdo_sqlite.ini  xmlwriter.ini
curl.ini   mbstring.ini  odbc.ini   phar.ini      xsl.ini
.....
```

配置PHP的主配置文件

● # vi /etc/php.ini

```
# 对于生产平台，应将display_errors设置为 Off
display_errors = Off
# 将log_errors设置为 On
log_errors = On
# 使用 zlib 库压缩输出并设置压缩级别
zlib.output_compression = On
zlib.output_compression_level = 1
# 不暴露PHP被安装在服务器上的事实
expose_php = Off

# 限制一个PHP脚本可能消耗的最大内存量
# 这有助于防止写得不好的脚本消耗服务器上的可用内存。
memory_limit = 256M
```

配置PHP的主配置文件（续）

● # vi /etc/php.ini

为POST方法指定可接受的最大尺寸

post_max_size = 512M

设置可上传文件的最大尺寸

upload_max_filesize = 20M

不能使用URL（如： http:// 或 ftp://）直接打开文件

allow_url_fopen = Off

[Date]

为日期函数定义默认时区

date.timezone = Asia/Shanghai

配置PHP的APC模块

● # vi /etc/php.d/apc.ini

```
# 是否启用 APC opcode cache, 若系统中无其他 OPcache可以设置为1
apc.enable_opcode_cache=0
# 启用APC (这是默认配置, 1是启用)
apc.enabled = 1
# 每个共享内存块的大小, (可以使用单位后缀M/G)
apc.shm_size = 64M
# 缓存条目在缓冲区中允许逗留的秒数。0 表示永不超时。建议值为7200~36000
apc.ttl = 7200
# 缓存条目在垃圾回收表中能够存在的秒数
apc.gc_ttl = 3600
# 是否启用脚本更新检查。改变这个指令值要非常小心。
# 默认值 On 表示APC在每次请求脚本时都检查脚本是否被更新,
# 若检查到更新则自动重新编译和缓存编译后的内容。
apc.stat=1
```

配置Apache的php模块

- /etc/httpd/conf.d/php.conf

```
<IfModule prefork.c>
  LoadModule php5_module modules/libphp5.so
</IfModule>
<IfModule !prefork.c>
  LoadModule php5_module modules/libphp5-zts.so
</IfModule>

<FilesMatch \.php$>
  SetHandler application/x-httpd-php
</FilesMatch>

AddType text/html .php

DirectoryIndex index.php

php_value session.save_handler "files"
php_value session.save_path  "/var/lib/php/session"
```

安装PHP-MySQL支持

- 安装PHP-MySQL支持

```
#yum -y install php-mysql
```

- MySQL操作命令

```
systemctl start mariadb #启动MariaDB  
systemctl stop mariadb #停止MariaDB  
systemctl restart mariadb #重启MariaDB  
systemctl enable mariadb #设置开机启动
```


测试PHP和MySQL

- 重新启动Apache
 - ✓ # service httpd restart
- 在/var/www/html目录下编写一个测试脚本
 - ✓ # echo '<?php phpinfo()?>' > /var/www/html/info.php
- 使用浏览器进行测试
 - ✓ # elinks http://ip/info.php

课程内容



phpMyAdmin

- phpMyAdmin 是一个用PHP编写的基于Web的Mysql管理工具
- phpMyAdmin 界面友好，操作简单
- phpMyAdmin的主页
 - ✓ <http://phpmyadmin.sourceforge.net/>
- phpMyAdmin的安装和配置
 - ✓ `# yum install phpMyAdmin`

常用的LAMP应用

- Portal CMS
- LMS/LCMS
- Wiki
- BLOG
- Forum
- Groupware
- WebMail
- BugTrackers
- phpDBadmin
- Web Hosting Control Panel

常用的LAMP应用软件

- Moodle
- Wordpress
- Drupal
- phpBB
-

Moodle 简介

- Moodle为远程教育提供了一种优秀的开源解决方案
 - ✓ 是使用PHP编写的面向对象的模块化动态教学环境
 - ✓ 是由澳大利亚教师Martin Dougiamas基于建构主义教育理论而开发的免费、开源的课程管理系统(Course Management System , CMS)
 - ✓ 具有内容管理、学习管理和课程管理三大功能
 - 包含论坛、测验、资源、投票、问卷、作业、聊天和博客等模块
 - 具有大量功能丰富的第三方插件
 - ✓ 是目前全球范围内应用最广泛的在线教学平台之一

Moodle配置举例

- 下载最新版Moodle
- 配置MySQL服务的InnoDB存储支持
- 配置Apache
- 运行Moodle的安装配置脚本
 - ✓ 或使用浏览器实现交互式安装
- 安排Moodle的cron任务

课程总结



本章思考题

- Linux环境下常用的脚本语言有哪些？各自有何特点？
- 常见的动态网站技术有哪些？与CGI相比FastCGI有哪些特点和优势？
- 什么是LAMP？LAMP的常见应用有哪些？
- 什么是反向代理？Apache的反向代理能代理哪些后端服务？
- 什么是负载均衡？如何分类？Apache使用哪种负载均衡技术和算法？

本章实验

- 学会配置基于Apache动态语言模块的LAMP环境。
- 了解安装和配置常见的LAMP应用软件
 - ✓如：Drupal、Joomla、MediaWiki、Wordpress、phpBB和Moodle等
- 学会配置Tomcat的默认实例和第二实例。
- 学会使用Apache反向代理Tomcat。

THANK YOU!