目录

[一、内容回顾： 1](#_Toc402449240)

[二、memcache基本概念： 2](#_Toc402449241)

[三、memcache的安装： 4](#_Toc402449242)

[1、下载软件，并解压，拷贝到指定的位置，一般便于管理，和服务器的环境在同一个目录下 4](#_Toc402449243)

[2、以管理员的方式进入到cmd窗口，并进入到memcache所在的目录 4](#_Toc402449244)

[（3）安装完成后，要启动服务， 5](#_Toc402449245)

[（4）安装时要注意： 5](#_Toc402449246)

[四、如何连接memcache服务器进行使用。 6](#_Toc402449247)

[五、使用telnet来进行连接到memcache服务器（类似于黑窗口） 6](#_Toc402449248)

[六、通过php来连接memcache服务器。 9](#_Toc402449249)

[七、完成一个使用memcache存储数据的一个案例： 12](#_Toc402449250)

[八、通过使用memcahe客户端源码来连接操作memcache服务器。 14](#_Toc402449251)

[九、把session文件可以存储到memcache里面 14](#_Toc402449252)

[十、memcache的分布式存储，在网站的架构中可以架构多台memcache服务器，数据是分布存储的。 16](#_Toc402449253)

[十一、memcache的细节讨论: 17](#_Toc402449254)

[1、生命周期的问题： 17](#_Toc402449255)

[2、memcache中善于存储哪些数据 17](#_Toc402449256)

[3、memcache的安全性。 18](#_Toc402449257)

# 一、内容回顾：

（1）缓存的配置：在httpd.conf配置文件中打开那个模块：expires

配置方式：

ExpiresActive on

ExpiresByType image/jpeg " access plus 20 days"

在有些情况下，需要对某些资源不进行缓存可以配置：（前提：要加载header模块）

<filesMatch "\.(gif)$">

header set cache-control "no-store,must-revalidate"

</filesMatch>

（2）配置压缩：

开启deflate模块，在虚拟主机里面进行配置（不要写在.htaccess文件中）

<ifmodule mod\_deflate.c>

DeflateCompressionLevel 6

AddOutputFilterByType DEFLATE text/plain

AddOutputFilterByType DEFLATE text/html

AddOutputFilterByType DEFLATE text/xml

</ifmodule>

（3）真静态的案例：ob缓存 模板替换，都是结合使用的。

要达到前台静态化，后台无需静态化。

（4）伪静态：rewrite机制。通过机制完成url的转换。

配置rewrite机制：

首先要加载rewrite的扩展。

rewriteengine on

rewritecond 条件

rewritecond 条件

rewriterule规则

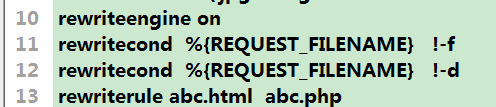
比如完成一个案例：

当我们在地址栏输入[www.abc.com/abc.html](http://www.abc.com/abc.html)

如果abc.html文件存在则直接访问abc.html文件。

如果abc.html文件不存在则跳转到abc.php

具体的配置内容：



不是一个目录

不是一个文件

（5）在项目中如何选择真静态还是伪静态。

1. 网站实时性要求高,不要使用静态化（真静态，伪静态均不适宜）。
2. 如果网站访问量较小，没有必要使用静态化技术
3. 如果数据项目不多，但是访问频率极大，建议使用真静态,比如新浪新闻频道。
4. 如果数据项目海量，使用真静态会生成海量的html静态页面，建议使用伪静态。

在一个大型网站中，静态化技术是综合使用的，这个需要大家经验的积累，多做项目。（sina网站：新闻频道，真静态，基金频道：伪静态，管理后台：不使用静态技术

# 二、memcache基本概念：

磁盘缓存（页面静态化）

内存缓存（memcache/redis）

（1）mysql 是c/s架构，客户端 /服务器。

memcache也是**c/s**架构的。

（2）mysql的数据是存储到哪里的。是存储到磁盘里面的。

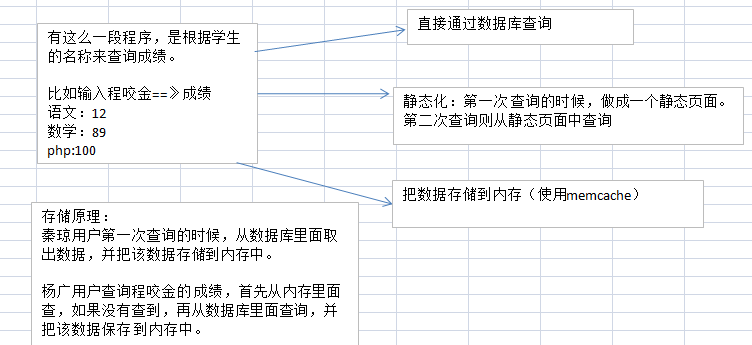
memcahe它的数据是存储到**内存**里面的。

计算机访问数据的时候，是在内存里面取出数据快呢？还是在硬盘里面快呢？是在内存里面读取数据的速度快。

（3）mysql里面存储数据的格式，是由数据库，数据库下面的表，表里面的记录。

memcache里面存储数据的格式是：**键值对**,类似于关联数组。

原理图示：



对memcahce的了解

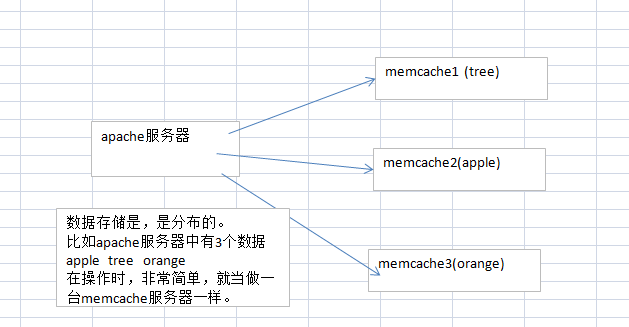
**memcached基本概念**

1. **Memcached是danga的一个项目，最早是LiveJournal 服务的，最初为了加速 LiveJournal 访问速度而开发的，后来被很多大型的网站采用。** 官方网站: [www.danga.com](http://www.danga.com/) 和 memcached.org
2. **Memcached是一个高性能的分布式的内存对象缓存系统，目前全世界不少人使用这个缓存项目来构建自己大负载的网站，来分担数据库的压力，通过在内存里维护一个统一的巨大的hash表，它能够用来存储各种格式的数据，包括图像、视频、文件以及数据库检索的结果等。简单的说就是将数据调用到内存中，然后从内存中读取，从而大大提高读取速度。(注: 摘自百度全科)**

**Memcached是一个高性能的分布式的内存对象缓存系统**

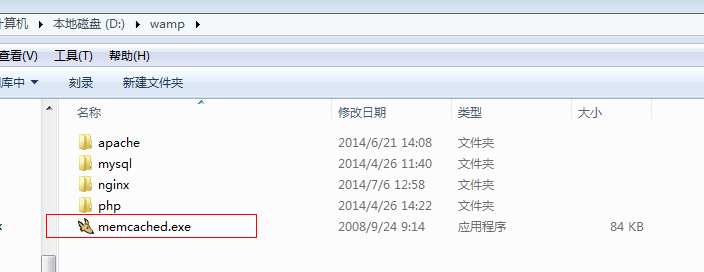
**要理解分布式的概念：**

**我们在实际的项目中可以使用多台memcache服务器，共同工作，存储的数据是分布式的。**



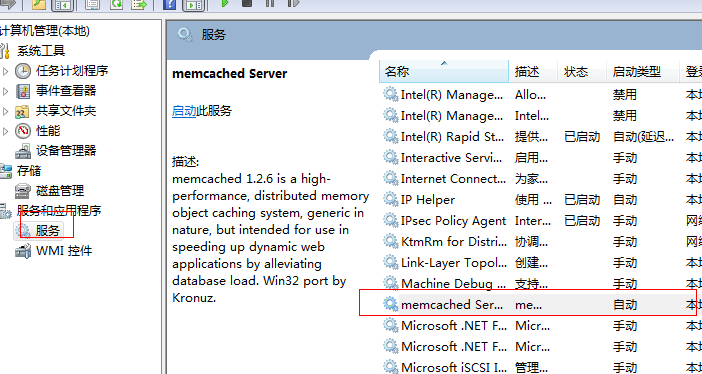
# 三、memcache的安装：

## 1、下载软件，并解压，拷贝到指定的位置，一般便于管理，和服务器的环境在同一个目录下



## 2、以管理员的方式进入到cmd窗口，并进入到memcache所在的目录

memcache.exe –d install 安装：安装完成后，可以通过计算机-》服务=》查看是否安装成功。



## （3）安装完成后，要启动服务，

memcached.exe –d start

启动后，可以通过netstat –an命令查看memcahce的端口是否监听状态，memcache的端口号是，11211.



## （4）安装时要注意：

安装可能失败的原因：

a 如果你是用win7,win8系统，他对安全性要求高，因此，需要大家使用管理员的身份来安装和启动. 具体是 程序开始===>所有程序==》附件==》cmd(单击右键，选择以管理员的身份来执行)

b存放memcached.exe 目录不要有中文或者特殊字符

c 安装成功，但是启动会报告一个错误信息，提示缺少xx.dll ，你可以从别的机器拷贝该dll文件，然后放入到system32下即可.

d如果上面三个方法都不可以，你可以直接这样启动mem

cmd>memcached.exe -p 端口 【这种方式不能关闭窗口】

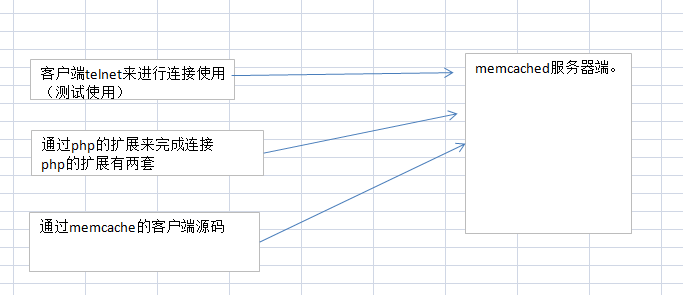
（5）在启动时指定的一些基本命令；

**-p 监听的端口   
-l <ip地址>绑定地址（默认：所有都允许，无论内外网或者本机ip,有安全隐患，若设置为127.0.0.1就只能本机访问）  
-d start 启动memcached服务   
-d restart 重起memcached服务   
-d stop|shutdown 关闭正在运行的memcached服务   
-d install 安装memcached服务   
-d uninstall 卸载memcached服务**

**-u 以的身份运行 (仅在以root运行的时候有效)   
-m 最大内存使用，单位MB。默认64MB**

**-M 内存耗尽时返回错误，而不是删除项   
-c 最大同时连接数，默认是1024   
-f 块大小增长因子，默认是1.25   
-n 最小分配空间，key+value+flags默认是48   
-h 显示帮助**

# 四、如何连接memcache服务器进行使用。



# 五、使用telnet来进行连接到memcache服务器（类似于黑窗口）

连接方式：telnet ip地址 端口号



增删改查操作

1、添加数据

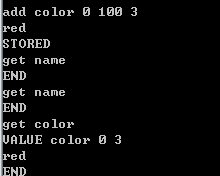
add 键的名称 0 缓存的时间 数据的长度 (回车后添加内容) 0表示不压缩：

比如：add name 0 60 5 表示向memcache里面添加的数据键为name 缓存的周期是60秒，数据的长度是5.

**数据存储格式，类似于关联数组**

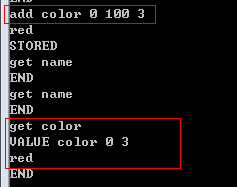
**第一列 第二列**

**键名（一般是字符串） 具体的数据内容（除了资源）**



2、获取数据

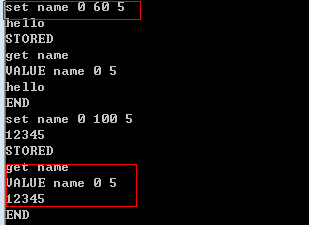
get 键名



3、修改数据

set set 指令也可以替换数据，如果键存在则替换，如果不存在，则等于添加。

replace 要求键名必须存在，如果不存在，则提示一个错误。

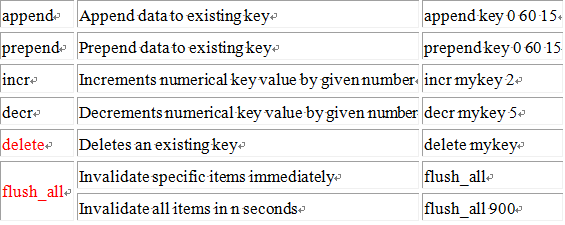


4、删除数据

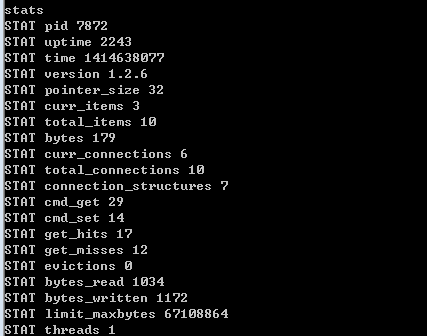
基本语法：delete 键名称

flush\_all 清空所有的数据

5、其他的一些命令



为了测试memcache的效率，获取的查询的成功率,也就是memcache 的效率，get\_hits/cmd\_get



有17次查询能够查询到数据

总共执行查询的次数

memcache服务器启动了多长时间

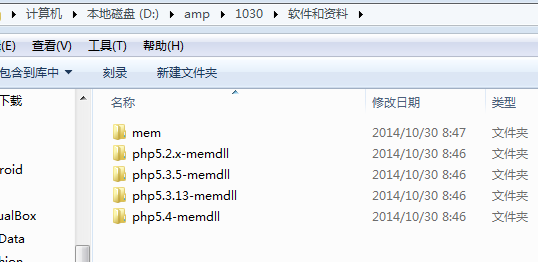
从服务器启动，到现在总共设置了多少个键

当前内存中有3个键值

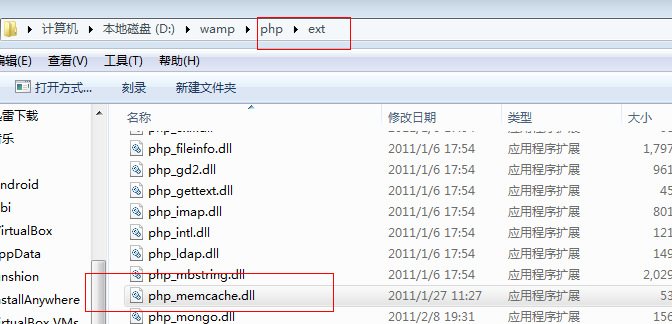
# 六、通过php来连接memcache服务器。

（1）从网上下载memcache php的扩展，该扩展要和php的版本一致。

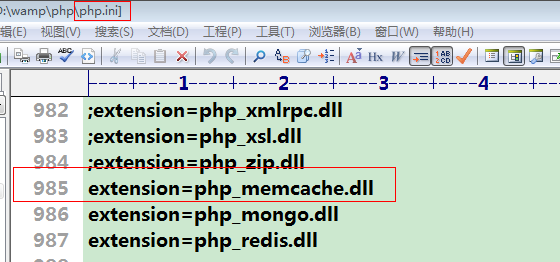
要下载php版本对应的memcache的扩展。



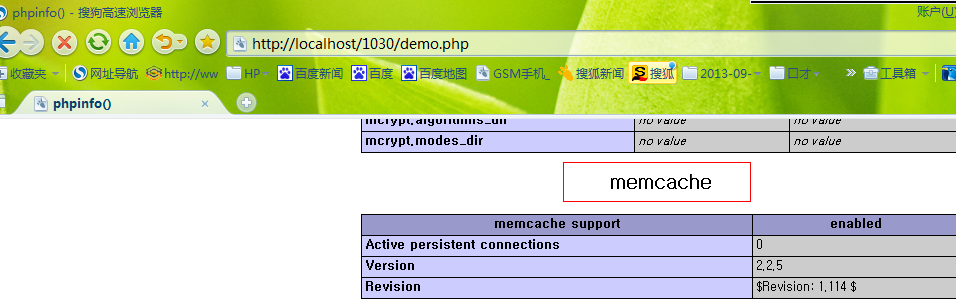
（2）把对应的memcache扩展拷贝到php安装目录的ext下面。



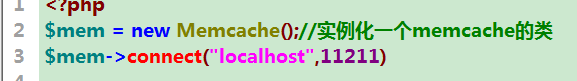
（3）打开php.ini文件加载该扩展



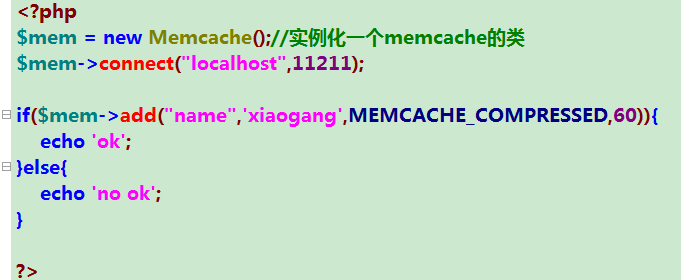
（4）重启apache,通过运行phpinfo()函数来进行测试。



（5）基本案例：



（6）基本操作



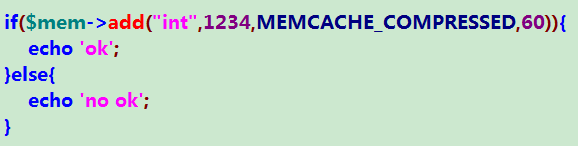
在使用add函数添加数据时，键名不能相同。

取出数据

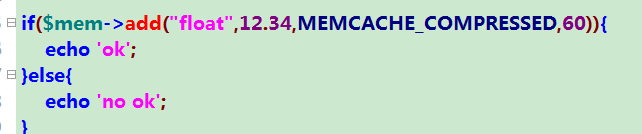
$mem->get(键名);

讨论测试存储的数据类型。

**存储整数：**

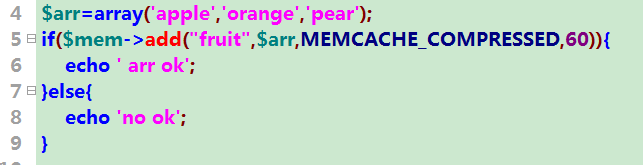


**存储的小数：**



**存储数组**

存储数组，以序列化的方式来存储的。



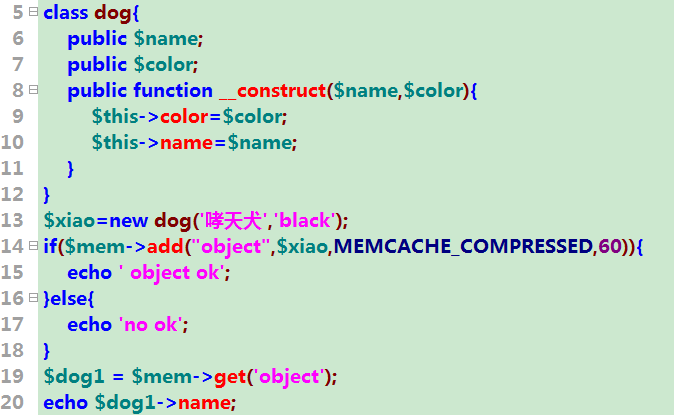
什么是序列化：就是在保存数据时，不但保存数据本身，还保存数据类型。

（1）利于在网络间进行数据传输

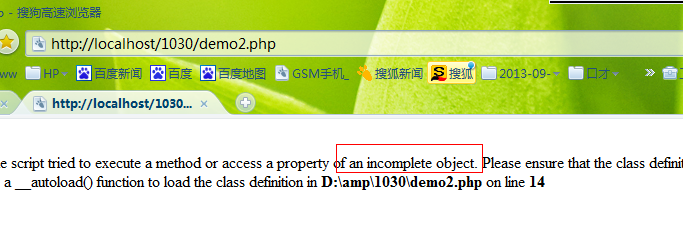
（2）便于重新读取并恢复数据

相关的函数：serialize();unserialize();

**存储对象**

****

**在另外一个页面中取出对象时，要有类的声明。否则会报如下的一些错误。**

****

# 七、完成一个使用memcache存储数据的一个案例：

**<?php**

**$mem=new Memcache();**

**$mem->connect("localhost",11211);**

**//先完成数据的取出**

**//如果把取出的data数据给缓存到memcache里面**

**//取出的数据是由sql语句决定的。键由sql语句字符串，值就是取出的数据**

**$sql = "select title from dede\_archives limit 5";**

**$key = md5($sql);**

**echo $key;**

**//先从memcache里面取出数据**

**$data = $mem->get($key);//根据键名称从memcache里面取出数据**

**//判断是否取出数据，如果数据为空，则从数据库里面获取数据。**

**if(!$data){**

**$conn = mysql\_connect("localhost",'root','root');**

**mysql\_query("use itdede");**

**//mysql\_select\_db();**

**mysql\_query("set names utf8");**

**$res = mysql\_query($sql,$conn);**

**$data = array();**

**while($row=mysql\_fetch\_assoc($res)){**

**$data[]=$row;**

**}**

**//从数据库里面获取的数据，添加到memcache里面。**

**$mem->add($key,$data,MEMCACHE\_COMPRESSED,100);**

**}**

**?>**

**<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">**

**<html xmlns="http://www.w3.org/1999/xhtml" lang="zh-CN">**

**<head>**

**<title>新建网页</title>**

**<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />**

**<meta name="description" content="" />**

**<meta name="keywords" content="" />**

**<script type="text/javascript">**

**</script>**

**<style type="text/css">**

**</style>**

**</head>**

**<body>**

**<?php foreach($data as $v){?>**

**<li><?php echo $v['title']?></li>**

**<?php }?>**

**</body>**

**</html>**

## 八、通过使用memcahe客户端源码来连接操作memcache服务器。

**使用范围：在没有php操作memcache扩展的前提下使用，推荐使用php，memcache的扩展。**

**测试前：要关闭扩展。**

**具体的使用步骤：**

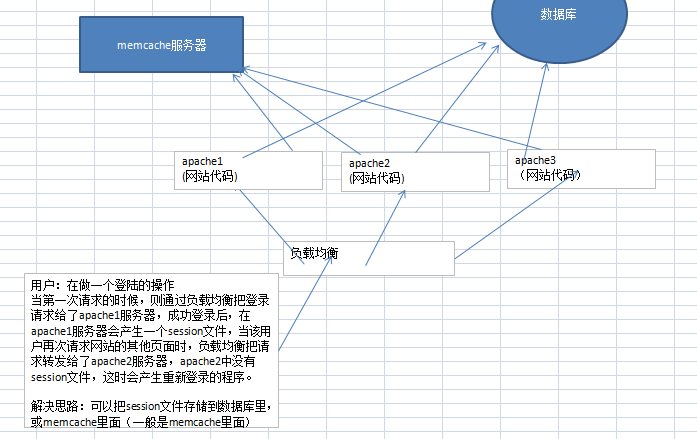
**（1）把源码文件拷贝到项目的指定位置。**

****

**（2）要引入源码文件。**

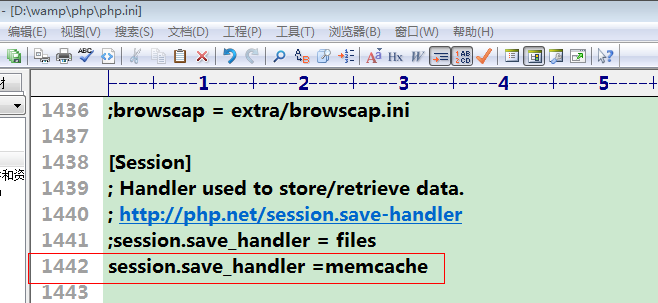
# 九、把session文件可以存储到memcache里面

**使用图示：**

****

**在php.ini文件中，我们可以设置存储方式。**

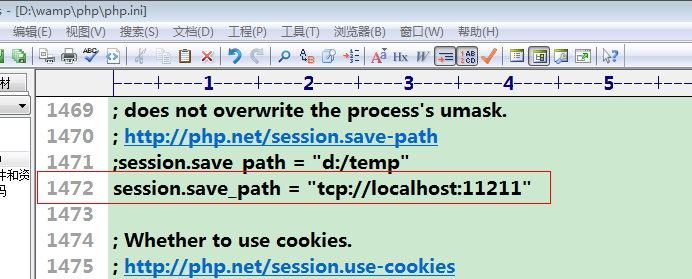
**session.save\_handler = files [user|memcache] 【三种取值.】**

****

**设置存储位置：**

**设置php.ini中session.save\_path=‘tcp://localhost:11211’，** **（可以是用逗号分隔的列表）**

**比如：session.save\_path=‘tcp://localhost:11211,tpc://192.168.1.89:11211’，**

****

**要注意：session文件存储到memcache里面是通过sessionid做为键的。**

**把session文件给出存储到memcache里面后，不影响session的任何操作。**

**要注意：**

**//除了可以在php.ini文件中设置session保存到memcache外，我们还可以直接在文件中设置，其实这样更好，因为**

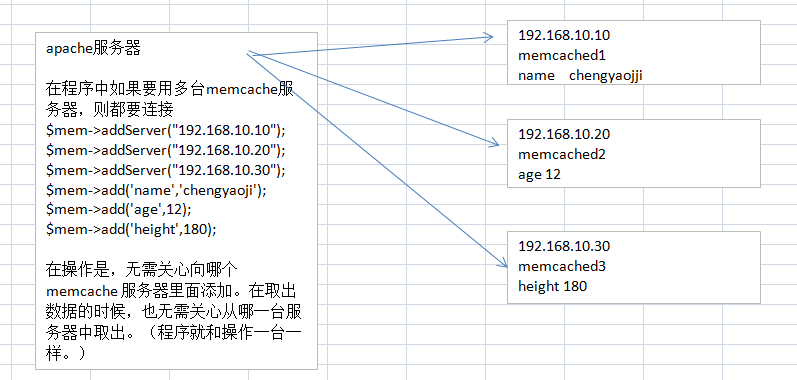
**//不会影响到别人的使用session的方式.**

**ini\_set("session.save\_handler","memcache");**

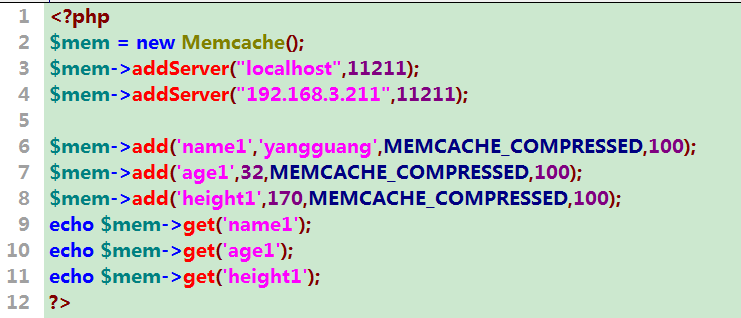
**ini\_set("session.save\_path","tcp://localhost:11211");**

# 十、memcache的分布式存储，在网站的架构中可以架构多台memcache服务器，数据是分布存储的。

**分布原理图示：**

****

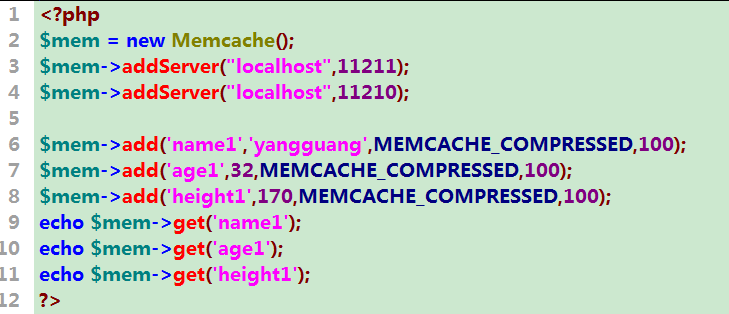
**具体的代码：**

****

**如果只有一台电脑，无需虚拟机也可以模拟两台memcache 服务器。**

**可以通过memcached.exe –p 11210来开启。**

**具体代码：**

****

# 十一、memcache的细节讨论:

## 1、生命周期的问题：

生命周期设置方式有两种：

第一种是：秒数，但是有限制，不能超过 2592000秒（30天）。

第二种是：时间戳，比如我们要设置超过1个月，则如下设置

$mem->add('name1','yangguang',MEMCACHE\_COMPRESSED,time()+3600\*24\*31);

如果设置为0，则表示永不过期。

存储到memcache里面的数据，什么情况下会丢失：

（1）生命周期到了

（2）关闭memcache服务

（3）关机，重启。

## 2、memcache中善于存储哪些数据

（1）更新比较频繁的数据（用户的在线状态），查询比较频繁，数据量不是很大。

（2）安全性不是很高。

如果没有memcache这个产品，可以通过mysql的memory存储引擎。

## 3、memcache的安全性。



**memcache安全性**

**（1）内网（两块网卡，memcache在启动时指定那台访问）**

**memcached –d –u root –l 192.168.1.100 –p 11211.指定只有192.168.1.100服务器才能访问。**

（2）防火墙

在linux中，设置命令。

**iptables -a input -p 协议 -s 可以访问ip -dport 端口 -j ACCEPT**

**例如：**

**iptables –a input –p tcp –s 192.168.1.1 –dport 11211 –j ACCEPT**

**iptables –a input –p tcp –s 192.168.1.1 –dport 11211 –j ACCEPT**

4、**Memcached机制深入了解**

**①基于c/s架构，协议简单**

**②基于libevent的事件处理**

③内置内存存储方式

④基于客户端的分布式

ob 缓存。

模板替换，