

用PHP做负载均衡指南 PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022__E7_94_A8P_HP_E5_81_9A_E8_c103_645661.htm 过去当运行一个大的web应用时候意味着需要运行一个大型的web服务器。因为你的应用吸引了大量的用户，你将不得不在你的服务器里增加更多的内存和处理器。今天，“大型服务器”模式已经过去，取而代之的是大量的小服务器，使用各种各样的负载均衡技术。“更多小服务器”的优势超过过去的“大型服务器”模式体现在两个方面：1. 如果服务器宕机，那么负载均衡系统将停止请求到宕机的服务器，转而分发负载到其他正常运行的服务器上。2. 扩展你的服务器更加容易。你要做的仅仅是加入新的服务器到负载均衡系统。不需要中断你的应用运行。所以，把握住这个机会。当然，代价就是这要求你的应用开发时增加一点复杂度。这就是本文要覆盖的内容。这时你可能对自己说：“但是我怎么知道我正在使用负载均衡呢？”。最诚实的回答是，如果你正在问这个问题，那么答案是你多半没有在使用负载均衡系统并且你的系统不需要考虑这个问题。大多数情况，当应用成长足够大的规模时，负载均衡就需要明确提出和设置了。然而，我也偶尔看见虚拟主机公司为客户的应用做这个负载均衡，或者像下面描述的那样要自己来做。注意，我一直提“web应用”而不是website,这是想区分“web应用”是那些复杂的站点往往涉及服务器端编程和数据库，而不是website那样只显示简单的静态内容。1. PHP文件第一个问题是，如果你有大量的小型服务器，你怎么把你的php文件上传到所有的服务器上?有如下的方法供你参考：

分别上传所有的文件到每一个服务器，这种方法带来的问题是：想像一下你有20个服务器，那么上传过程中这将很容易导致错误，并且更新时极有可能导致不同服务器上有不同版本的文件。使用‘rsync’（或类似的软件），这样的工具能同步本地目录和多个远程主机目录上的文件。使用版本控制软件（如subversion），这是我最喜欢的方法。用它可以在每一个服务器上运行svn update命令同步。这种方法也使切换服务器得代码到过去的某一个版本更加容易。使用一个文件服务器（你可能发现NFS非常适合做这件事情），这种方式是使用一个文件服务器来存放你的web应用。当然，如果你的文件服务器宕机，那么多所有你的站点将不能使用。这时，你就需要花费更多的开支来恢复它。选择哪种方式依赖于你的需求和你掌握的技能。如果你使用版本控制系统，那么你可能得计划一个方法如果同时执行一个更新命令更新所有服务器上的代码。然而，如果使用文件服务器，你就要实现一些失败恢复机制，防止万一服务器宕机导致请求失败。

2. 文件上传

当只有一台服务器时，文件上传不是一个问题。但是当我们有多台服务器时，那么上传的文件应该怎么存放呢？上传文件的问题和跨服务器php文件存储是类似的。下面是几种可能的方案：

把文件存储到数据库中。大多数数据允许存储二进制数据。当你请求文件下载时，访问数据把二进制数据和相应的文件名和类型输出给用户。在使用这种方案前应该考虑数据库怎样存储你的文件。该方法的问题在于如果数据库服务器宕机将使文件不可用。

在一个文件服务器上存储上传的文件，与前面的介绍一样，你要安装一个文件服务器让所有web

服务器共享，把所有上传的文件上传到这里，上传后所有的web服务器就都可以使用它。但是，如果文件服务器宕机，那么可能发生图像文件下载中断。设计你自己的上传机制传输文件到服务器到每一个服务器。这个方法没有单个文件服务器或者数据库方案的缺陷，但是将增加你代码的复杂度。例如，如果上传到多个服务器过程中，服务器宕机，你要怎么处理？用数据库存储上传文件但是设计一个文件缓存机制是一个不错的方案。当服务器接收一个文件下载请求时，首先检查缓存系统中是否有该文件，如果发现那么从缓存系统下载，否则从数据库读取并把它缓存到文件系统中。

3. 会话(Sessions) 如果你熟悉php的session处理，你将可能知道默认情况下，它存储session数据在服务器的临时文件里。而且，这个文件仅仅在你请求处理的那个服务器上，但是接下来的请求可能被另外一个服务器处理，这将在另一个服务器上生成新的session。这导致session频繁地不被识别，如登录用户总是要求重新登录。我推荐的方案是，要么重新php内建的session处理机制存储session数据到数据库，或者实现你自己的机制保证发送一个用户的请求到同一台服务器。

4. 配置(Configuration) 尽管这个话题不是和php特别相关，我感觉还是有必要提及。当运行集群服务器时，用某种方法保持服务器之间的配置文件同步是一个好主意。如果配置文件不一致，可能导致一些非常奇怪的断断续续的行为导致很难排查这些问题。我推荐使用版本控制系统单独管理他们。这样你可以为不同的项目安装存储不同的php配置文件，也可以保持所有服务器配置文件同步。

5. 日志(Logging) 像配置问题一样，logging不是仅仅和php相关。但是对于保持服务器健康运行

它仍然是非常重要的。没有正确的logging系统，你怎么知道如果PHP代码开始产生错误(在系统正式运行时，你总是关闭display_errors 设置，不是吗?) 有几种方法你可以实现logging：

1. 在每一个服务器上记录日志。这是最简单的方法。每一个机器仅仅记录一个文件。好处是简单，可能只要很少的配置。但是，随着服务器数量的增多，监控每台服务器上的日志文件将变得非常困难。
2. 记录日志到一个共享 这种方法每一个服务器仍然有这个日志文件，但是他们通过共享机制被存储在一个中央文件服务器上，这将使监控日志变得更简单。该方案的问题在于，如果文件服务器不可用将导致一个简单的日志不能写入问题最终导致整个应用崩溃。
3. 记录日志到logging服务器 你可以使用一个logging软件，如syslog 来把所有的日志写到一个中央服务器。尽管这个方法要求更多的配置，但是他也提供了最健壮的方案。

百考试题温馨提示：本内容来源于网络，仅代表作者个人观点，与本站立场无关，仅供您学习交流使用。其中可能有部分文章经过多次转载而造成文章内容缺失、错误或文章作者不详等问题，请您谅解。如有侵犯您的权利，请联系我们，本站会立即予以处理。

编辑特别推荐: #0000ff>linuxln命令详解 #0000ff>nginx关于服务静态文件的配置 #0000ff>使用expect实现ssh自动交互 100Test
下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com