# php-fpm解读-进程管理的三种模式

> 《我是程序媛》系列——php-fpm进程管理，感谢大表哥亲情赞助时间，读了php-fpm源码。

php-fpm进程管理一共有三种模式：ondemand、static、dynamic，我们可以在同一个fpm的master配置三种模式，看下图1。php-fpm的工作模式和nginx类似，都是一个master，多个worker模型。每个worker都在accept本pool内的监听套接字（linux已不存在惊群现象）。

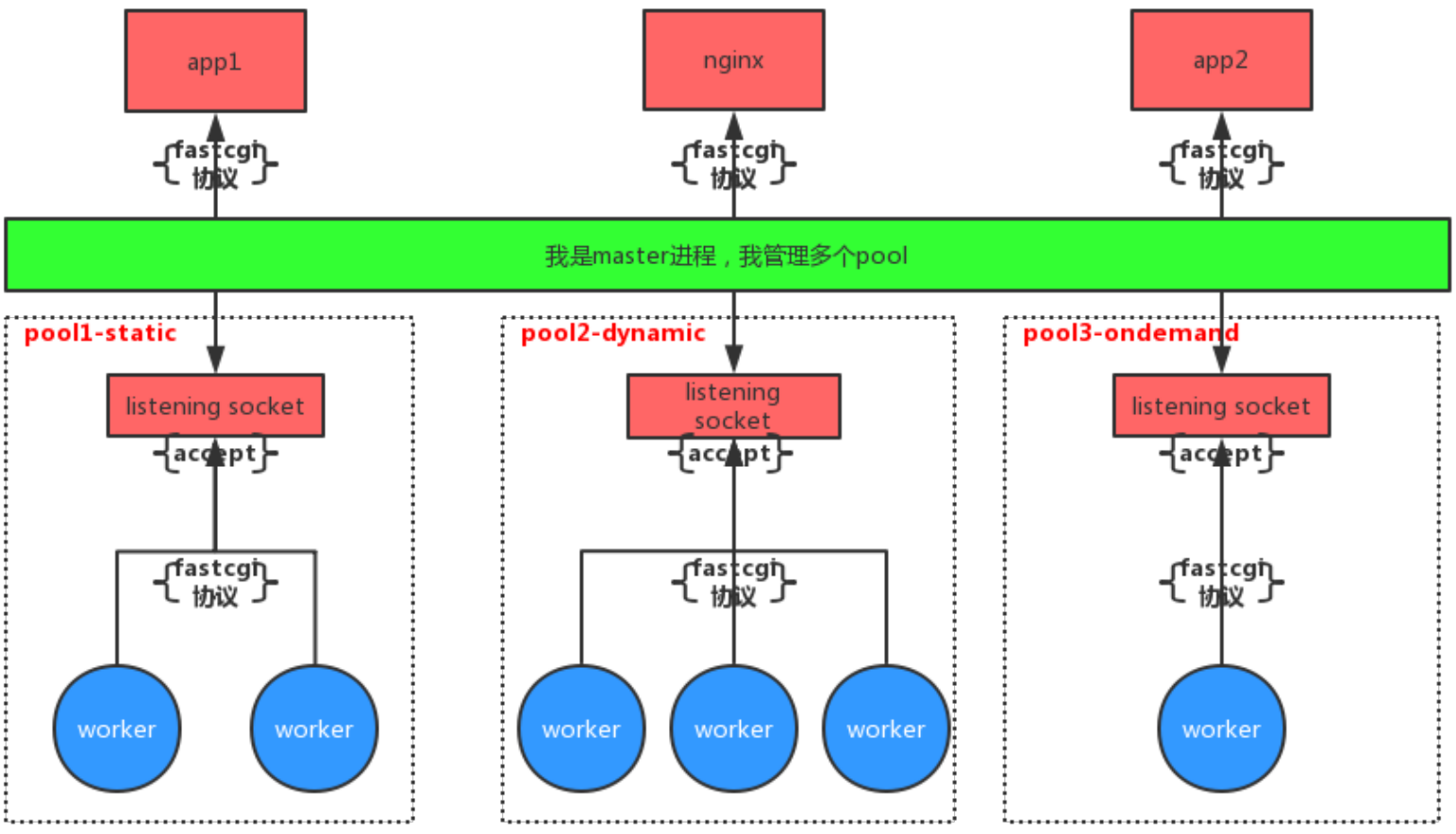


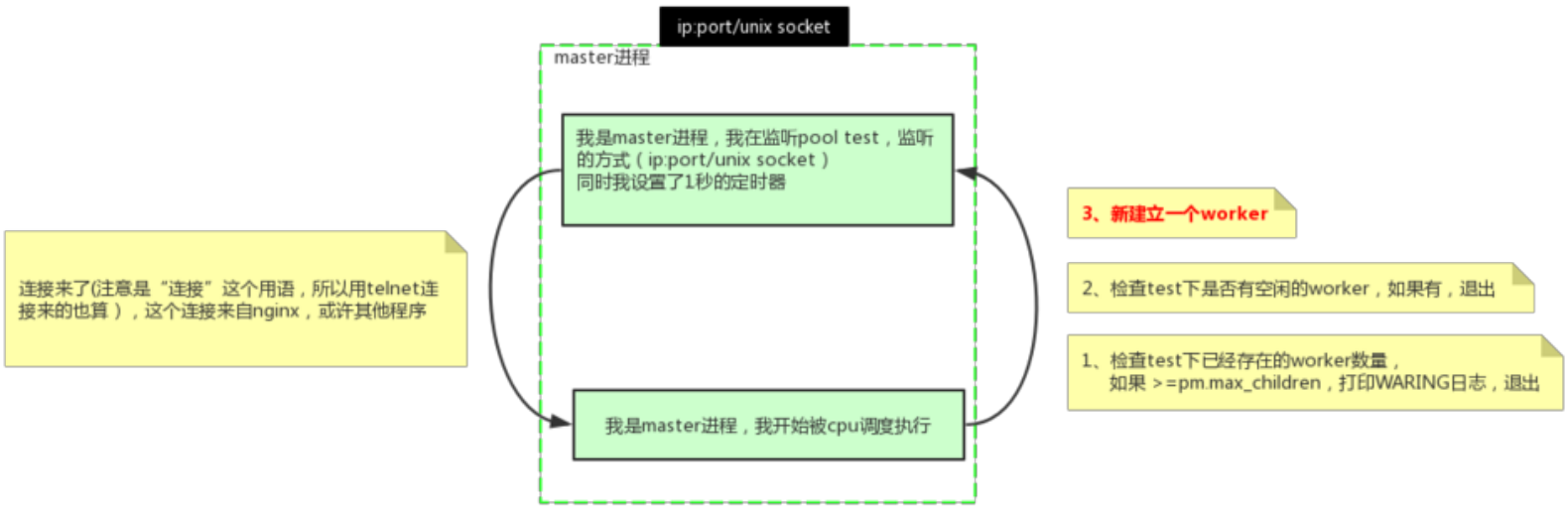

图1

## ondemand

> 在php-fpm启动的时候，不会给这个pool启动任何一个worker，是按需启动，当有连接过来才会启动。

配置文件（我的配置文件地址为：/usr/local/php/etc/php-fpm.conf）

## 原理




ondemand原理图

1. 从上图可以看出，新建worker的触发条件是连接的到来，而不是实际的请求（例如，只进行连接比如telnet，不发请求数据也会新建worker）

2. worker的数量受限于pm.max_children配置，同时受限全局配置process.max（准确的说，三种模式都受限于全局配置）

3.1秒定时器作用

找到空闲worker，如果空闲时间超过pm.process_idle_timeout大小，关闭。这个机制可能会关闭所有的worker。

## 配置项要求

1. pm.max_children> 0

2. pm.process_idle_timeout> 0，如果不设置，默认10s

## 优缺点

优点：按流量需求创建，不浪费系统资源（在硬件如此便宜的时代，这个优点略显鸡肋）

缺点：由于php-fpm是短连接的，所以每次请求都会先建立连接，建立连接的过程必然会触发上图的执行步骤，所以，在大流量的系统上master进程会变得繁忙，占用系统cpu资源，不适合大流量环境的部署

## dynamic

> 在php-fpm启动时，会初始启动一些worker，在运行过程中动态调整worker数量，worker的数量受限于pm.max_children配置，同时受限全局配置process.max

## 原理

1. 1秒定时器作用

检查空闲worker数量，按照一定策略动态调整worker数量，增加或减少。增加时，worker最大数量<=max_children · <=全局process.max；减少时，只有idle >pm.max_spare_servers时才会关闭一个空闲worker。

idle > pm.max_spare_servers，关闭启动时间最长的一个worker，结束本次处理

idle >= pm.max_children，打印WARNING日志，结束本次处理

idle < pm.max_children，计算一个num值，然后启动num个worker，结束本次处理

## 配置项要求

1. pm.min_spare_servers/pm.max_spare_servers有效范围(0,pm.max_children]

2. pm.max_children> 0

3. pm.min_spare_servers<=pm.max_spare_servers

4. pm.start_servers有效范围[pm.min_spare_servers,pm.max_spare_servers]如果没有配置，默认pm.min_spare_servers + (pm.max_spare_servers - pm.min_spare_servers) / 2

## 优缺点

优点：动态扩容，不浪费系统资源，master进程设置的1秒定时器对系统的影响忽略不计；

缺点：如果所有worker都在工作，新的请求到来只能等待master在1秒定时器

内再新建一个worker，这时可能最长等待1s；

## static

> php-fpm启动采用固定大小数量的worker，在运行期间也不会扩容，虽然也有1秒的定时器，仅限于统计一些状态信息，例如空闲worker个数，活动worker个数，网络连接队列长度等信息。

## 原理

## 配置项要求

1、pm.max_children> 0 必须配置，且只有这一个参数生效

## 优缺点

如果配置成static，只需要考虑max_children的数量，数量取决于cpu的个数和应用的响应时间，我司配置的是50。

我司不考虑动态的增加减少那么十几个或者几十个worker，我们的内存没有紧张到这个程度，所以，我们一步到位，把worker数配置到支持最大流量，（哈哈，50也是随便定的，足矣足矣呢）

最后我们再介绍下worker的工作流程

fastcgi与php-fpm的关系一句话解读：fastcgi只是通信应用协议，php-fpm就是实现了fastcig协议，并嵌入了一个 PHP 解释器。

完

大表哥说fpm必须让大象背上，设计能力有限，凑合看吧