# 正向代理与反向代理

**正向代理的概念**

正向代理(forward)是一个位于客户端【用户A】和原始服务器(origin server)【服务器B】之间的服务器【代理服务器Z】，为了从原始服务器取得内容，用户A向代理服务器Z发送一个请求并指定目标(服务器B)，然后代理服务器Z向服务器B转交请求并将获得的内容返回给客户端。

正向代理,也就是传说中的代理,他的工作原理就像一个跳板,  
简单的说,  
我是一个用户,我访问不了某网站,但是我能访问一个代理服务器  
这个代理服务器呢,他能访问那个我不能访问的网站  
于是我先连上代理服务器,告诉他我需要那个无法访问网站的内容  
代理服务器去取回来,然后返回给我

从网站的角度,只在代理服务器来取内容的时候有一次记录  
有时候并不知道是用户的请求,也隐藏了用户的资料,这取决于代理告不告诉网站

结论就是 正向代理 是一个位于客户端和原始服务器(origin server)之间的服务器，为了从原始服务器取得内容，客户端向代理发送一个请求并指定目标(原始服务器)，然后代理向原始服务器转交请求并将获得的内容返回给客户端。客户端必须要进行一些特别的设置才能使用正向代理。

**反向代理的概念**

客户端向反向代理的命名空间(name-space)中的内容发送普通请求，接着反向代理将判断向何处(原始服务器)转交请求，并将获得的内容返回给客户端

继续举例:  
例用户访问 http://bigc.at/readme  
但bigc.at上并不存在readme页面  
他是偷偷从另外一台服务器上取回来,然后作为自己的内容吐给用户

但用户并不知情  
这很正常,用户一般都很笨

这里所提到的 bigc.at 这个域名对应的服务器就设置了反向代理功能

结论就是 反向代理正好相反，对于客户端而言它就像是原始服务器，并且客户端不需要进行任何特别的设置。客户端向反向代理 的命名空间(name-space)中的内容发送普通请求，接着反向代理将判断向何处(原始服务器)转交请求，并将获得的内容返回给客户端，就像这些内容 原本就是它自己的一样。

**两者区别**

从**用途**上来讲：

正向代理的典型用途是为在防火墙内的局域网客户端提供访问Internet的途径。正向代理还可以使用缓冲特性减少网络使用率。反向代理的典型用途是将 防火墙后面的服务器提供给Internet用户访问。反向代理还可以为后端的多台服务器提供负载平衡，或为后端较慢的服务器提供缓冲服务。

另外，反向代理还可以启用高级URL策略和管理技术，从而使处于不同web服务器系统的web页面同时存在于同一个URL空间下。

从**安全性**来讲：

正向代理允许客户端通过它访问任意网站并且隐藏客户端自身，因此你必须采取安全措施以确保仅为经过授权的客户端提供服务。

反向代理对外都是透明的，访问者并不知道自己访问的是一个代理。

# url重写与反向代理的区别

## url重写

[URL](http://baike.baidu.com/view/1496.htm)重写就是首先获得一个进入的URL请求然后把它重新写成网站可以处理的另一个URL的过程。举个例子来说，如果通过[浏览器](http://baike.baidu.com/view/7718.htm)进来的URL是“UserProfile.aspx?ID=1”那么它可以被重写成 “UserProfile/1.aspx”，这样的URL，这样的网址可以更好的被网站所阅读。

先通过Rewrite对该地址进行重写，然后载入重写后的地址，进行解析

URL Rewrite技术。该技术的核心方程是在服务器端，主要通过服务器的重写技术，一种基于正则表达式的重定向技术。

    原理如：index.asp?id=124这个地址，服务器事先定义，/可以替换，重写为?id=则，这个地址就变成index.asp/124。这个定义的意思是说，当服务器www.auyou.com接受到一个http请求，请求访问其上的index.asp/124这个文件时，先通过Rewrite对该地址进行重写，然后载入重写后的地址，进行解析。这样当你访问http://www.auyou.com/index.asp/123时，并没有直接到达这个文件，而是这个地址，在服务器那里被换掉了，而这个换掉的过程，对客户端而言是隐蔽的，不可知的。

这里说的，只是原理，简单的替换，事实上，URL Rewrite还可以做许多更复杂的代换。如喜悦国际村的，由showthreads.php?postid=12345转换为12345.html。具体可以参照apache手册中的说明。

    为什么要应用URL Rewrite技术。

    从原理上可知，这种技术并不能实际提高访问速度，减少负载，与生成静态页面有着本质上的区别，但为什么有那么多网站热衷于此呢？说到这里，我们不得不提到搜索引擎。搜索引擎后台的蜘蛛引擎可以动态抓取互联网上的页面，经分析后存入数据库。但动态页面，一般却是不被蜘蛛引擎考虑的，只有很少部分，可能会被自动收录进去。而蜘蛛引擎识别静态页面和动态页面是从一点基本点着手，即静态页面一般不包含参数。即?id=...&post=..之类的。而动态页面多有这个东西。所以蜘蛛引擎在收录过程中，?后的部分基本是不考虑的。这样造成的结果就是showthreads.php?id=123456由于?后部分被忽略，就变成了showthreads.php。当然这个页面并不具备任何实际内容的。即，URL Rewrite技术多为搜索引擎而来。

    具体的URL Rewrite方法，参数，可参阅apache手册中URL Rewrite部分。如果是windows的操作系统，可能要装个插件才能使用(需要安装 ISAPI\_Rewrite 插件)。

## 区别

url重写是修改传入的url，然后载入重写后的地址，进行重定向。

url重定向也是要代理服务器处理，只是它并不转发该url到其它机器，还是传到目标机器。

可以理解为是在目标服务器上对url进行处理（不会更改域名，重写url的参数部分）

而反向代理将外部请求转发到内部的某个机器上，并不重写url，而是决定如何转发请求（如根据当前负荷情况）