# 广泛的爬行

Scrapy默认值针对特定网站的爬网进行了优化。这些站点通常由单个Scrapy蜘蛛处理，尽管这不是必需的或不需要的（例如，有一些通用蜘蛛可以处理抛出它们的任何给定站点）。

除了这种“集中爬行”之外，还有另一种常见类型的爬网，它覆盖了大量（可能无限制）的域数，并且仅受时间或其他任意约束的限制，而不是在域被爬行到完成时停止或当没有更多的请求要执行时。这些被称为“广泛爬行”，是搜索引擎使用的典型爬虫。

这些是广泛爬行中常见的一些常见属性：

- 他们抓取许多域（通常是无限制的）而不是特定的一组站点
- 他们不一定要抓取域完成，因为这样做是不切实际的（或不可能的），而是限制爬行的时间或页数爬行
- 它们在逻辑上更简单（与具有许多提取规则的非常复杂的蜘蛛相反），因为数据通常在单独的阶段中进行后处理
- 他们同时抓取多个域，这使得他们可以通过不受任何特定站点约束限制来实现更快的爬网速度（每个站点都被慢慢爬行以尊重礼貌，但许多站点并行爬行）

如上所述，Scrapy默认设置针对集中爬行进行了优化，而不是针对广泛爬网进行了优化。但是，由于其异步架构，Scrapy非常适合执行快速广泛爬网。本页概述了使用Scrapy进行广泛爬网时需要记住的一些事项，以及Scrapy设置调整的具体建议，以实现有效的广泛爬行。

## 增加并发性

并发是并行处理的请求数。存在全局限制和每个域限制。

Scrapy中的默认全局并发限制不适合并行爬网许多不同的域，因此您需要增加它。增加多少将取决于您的爬虫可用的CPU数量。一个很好的起点是`100`，但最好的方法是通过做一些试验并确定您的Scrapy流程获得CPU限制的并发性。为获得最佳性能，您应该选择CPU使用率为80-90%的并发性。

要增加全局并发使用：

```
CONCURRENT_REQUESTS = 100
```

## 增加Twisted IO线程池最大大小

目前，Scrapy使用线程池以阻塞方式进行DNS解析。使用更高的并发级别，爬网可能会很慢，甚至无法达到DNS解析器超时。增加处理DNS查询的线程数的可能解决方案。将更快地处理DNS队列，从而加速建立连接和整体爬行。

要增加最大线程池大小，请使用：

```
REACTOR_THREADPOOL_MAXSIZE = 20
```

## 设置自己的

如果您有多个爬网进程和单个中央DNS，它可能会像DNS服务器上的DoS攻击一样，导致整个网络速度变慢甚至阻塞您的计算机。为了避免这种情况，请设置您自己的具有本地缓存的DNS服务器，以及一些大型DNS（如OpenDNS或Verizon）的上游。

## 降低日志级别

在进行广泛爬网时，您通常只对您获得的爬网率和发现的任何错误感兴趣。使用`INFO`日志级别时，Scrapy会报告这些统计信息 。为了节省CPU（和日志存储要求），`DEBUG`在生产中执行大型广泛爬网时，不应使用日志级别。`DEBUG`在开发（宽）爬虫时使用级别可能没问题。

要设置日志级别，请使用：

```
LOG_LEVEL = 'INFO'
```

## 禁用

除非您*确实*需要，否则禁用cookie 。在进行广泛爬网（搜索引擎抓取工具忽略它们）时通常不需要Cookie，并且它们通过节省一些CPU周期并减少Scrapy爬网程序的内存占用来提高性能。

要禁用cookie，请使用：

```
COOKIES_ENABLED = False
```

## 禁用重试

重试失败的HTTP请求可能会大大减慢爬网速度，特别是当站点导致响应非常慢（或失败）时，从而导致超时错误多次重试，不必要地阻止爬网程序容量重用于其他域。

要禁用重试，请使用：

```
RETRY_ENABLED = False
```

## 减少下载超时

除非您从非常慢的连接中爬行（对于广泛爬网不应该这样），否则请减少下载超时，以便快速丢弃卡住的请求并释放处理下一个请求的容量。

要减少下载超时，请使用：

```
DOWNLOAD_TIMEOUT = 15
```

## 禁用重定向

考虑禁用重定向，除非您有兴趣关注它们。在进行广泛爬网时，通常会在以后抓取时重新访问网站时保存重定向并解决它们。这也有助于保持每个爬网批处理的请求数不变，否则重定向循环可能导致爬网程序在任何特定域上专用太多资源。

要禁用重定向，请使用：

```
REDIRECT_ENABLED = False
```

## 启用“Ajax可抓取页面”的抓取

有些页面（根据2013年的经验数据，高达1%）宣称自己是ajax可抓取的。这意味着它们提供通常只能通过AJAX提供的纯HTML内容版本。页面可以通过两种方式指示：

1. 通过`#!`在URL中使用- 这是默认方式;
2. 通过使用特殊元标记 - 这种方式用于“主”，“索引”网站页面。

Scrapy自动处理（1）; 处理（2）启用 AjaxCrawlMiddleware：

```
AJAXCRAWL_ENABLED = True
```

在进行广泛爬网时，抓取大量“索引”网页是很常见的; AjaxCrawlMiddleware有助于正确抓取它们。默认情况下它处于关闭状态，因为它具有一些性能开销，并且使其能够进行聚焦爬行没有多大意义。