Scrapyd 简介与应用场景介绍

Scrapyd 是由著名的爬虫框架 Scrapy 团队开发的，用于部署 Scrapy 项目并让使用者能够通过网络请求向爬虫发出指令的应用程序。

目前，Scrapyd 的最新版本为 1.2.0 版，对应文档可以点击👉这儿跳转。

Scrapyd 可以并行运行多个进程，并启动尽可能多的进程来处理负载，它提供了一个 API 服务来上传新的项目版本和爬虫。

它还可以管理多个项目，每个项目可以有多个版本（但只有最新的版本是可使用的）。

Scrapy 是什么?

开源的爬虫框架数量不少，但优秀的爬虫框架屈指可数，而 Scrapy 正是其中的佼佼者。因为它便捷的功能、丰富的组件、强大的异步处理能力以及良好的技术生态，让它在爬虫界所向披靡。无论你是萌新、初入职场的爬虫小白还是纵横职场的爬虫老司机，Scrapy 是你职业道路上的**必经之路**。

在实际工作中，爬虫都需要部署么?

是否部署取决于项目的需求，跟职业、非职业关联并不大，并不存在网上传言的"职业爬虫工程师的爬虫项目肯定是要部署的，直接运行的都是菜鸡"这样的情况。

通过不同的任务理解爬虫的部署需求

假设现在有这么一个任务:

> 公司要求获取某体育赛事网站上所有足球联赛及球队的信息数据，如名称、队标、市值以及球队荣誉与成立年份等属性，并存入数据库中为后续的数据分析和计算做准备。

需求整理： 世界范围的足球球队数量比较庞大，有赛事记录的球队已有十多万支，并且公司要求将数据存入数据库中，考虑后续会使用。

需求分析： 由于球队基础信息变更频率比较低，所以只需要每隔十几天启动 1 次爬虫以更新数据即可；并且由于更新周期较长，目标网站在此周期内有可能会更改页面结构，影响爬虫工作；部署本身就需要耗费时间，并且部署过程中还有可能产生其他的问题。

部署建议： 每次使用时只需在电脑上启动爬虫即可，这样既节省了爬虫部署的时间，又可以避免因为周期较长而导致爬虫代码修改后二次部署可能导致的其他问题。综合考虑，此次爬虫并**不需要部署**。

再来看另外一个任务:

> 公司要求每 10 分钟爬取一次指定的体育资讯网站上的资讯文章及图集，所需要爬取的内容如封面图、文章标题和文章内容、文章来源、作者信息以及读者评论点赞数据等等，并将数据存入数据库。

需求整理： 大型体育资讯网站、内容分散、爬取频率高、定时启动。

需求分析： 大型体育资讯网站的资讯量总体数量庞大，每日更新文章次数非常频繁，尤其是赛事期间。通常资讯文章的列表页与内容页是分开的，甚至作者信息和评论信息也是存放在不同的网络资源地址，所以每一篇文章所需的请求可能会有 2-5次。而且资讯类文章有时效性要求，一般为当天新闻，甚至半小时内的新闻资讯。

部署建议： 高频率的爬取意味着多次启动，以 10 分钟爬取 1 次为例，人力是难以兼顾的；至于定时任务的要求，人力是不可能满足需求的；最好的选择是将爬虫部署到服务器上并为它设置定时调度，按需求每 10 分钟调度 1 次，同时 Scrapyd 提供了爬虫日志记录，解决了运行信息保存和爬虫纠错的问题。

当然，实际的爬虫需求可能会更复杂，但是从上面的两个例子来看，已经可以说明在实际的工作中爬虫部署的选择是根据具体的任务需求来决定的。

有时候甚至连 Scrapy 框架都不需要使用，Requests 库就可以应对需求。

但是为了保持代码风格统一以及项目的后续维护管理，我们在工作中会尽量沿用统一的手法来解决问题。

为什么部署爬虫要选择 Scrapyd?

因为 Scrapyd 是 Scrapy 框架官方指定的、唯一的爬虫部署管理平台。Scrapyd 为工程师们准备了完整的爬虫打包工具 Scrapyd-client ，使得爬虫打包和部署变得轻而易举。并且 Scrapyd 作为爬虫部署管理平台，拥有爬虫调度、日志记录以及状态监视等功能，都是其他的部署平台难以具备的。

选择 Scrapyd 的理由：

方便灵活的 API： Scrapyd 为我们提供了几个基础而又强大的 API ，通过这些 API 我们可以及时了解爬虫的运行信息和状态，并且对状态进行更改，比如启动爬虫、取消爬虫任务以及添加新的爬虫项目等。

异步： Scrapy 框架和 Scrapyd 都是基于 Twisted 编写的，所以它们都具备异步的优势，我们通过 API 调用爬虫的请求是不会被阻塞的，哪怕你在 `10` 秒中内给它发送 `1000` 个请求。

操作简单： Scrapy 爬虫项目的打包与部署只需要进行 1 次配置，再使用 1 行命令即可完成。

易更新： 项目代码更新后，无需再为打包和部署编写配置，使用第一次部署时的配置即可，同样是使用 1 行命令即可完成新项目的打包和部署操作。

方便调用： 爬虫部署在 Scrapyd 后，可以在其他项目或程序中通过网络请求调度 Scrapyd 上的爬虫，比如 Java 项目 、C# 项目和 PHP 项目。