# Scrapy 框架

Scrapy是用Python实现一个为了爬取网站数据、提取结构性数据而编写的应用框架，用途非常广泛。

框架的优势在于，用户只需要定制开发几个模块就可以轻松的实现一个爬虫，用来抓取网页内容以及各种图片，非常之方便。

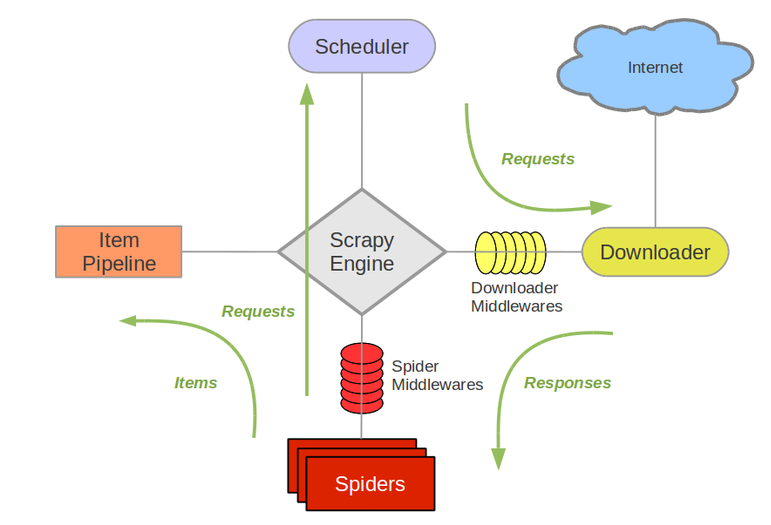
安装：通过pip 安装 Scrapy 框架pip install Scrapy

使用cmd安装scrapy的时候，出现error: Unable to find vcvarsall.bat 错误:安装vs;

error: Microsoft Visual C++ 14.0 is required错误:安装Visual C++

**具体Scrapy安装流程参考：**[**http://doc.scrapy.org/en/latest/intro/install.html#intro-install-platform-notes**](http://doc.scrapy.org/en/latest/intro/install.html#intro-install-platform-notes)**里面有各个平台的安装方法**

**架构图：**



**Scrapy Engine(引擎)**: 负责Spider、ItemPipeline、Downloader、Scheduler中间的通讯，信号、数据传递等。

**Scheduler(调度器)**: 它负责接受引擎发送过来的Request请求，并按照一定的方式进行整理排列，入队，当引擎需要时，交还给引擎。

**Downloader（下载器）**：负责下载Scrapy Engine(引擎)发送的所有Requests请求，并将其获取到的Responses交还给Scrapy Engine(引擎)，由引擎交给Spider来处理，

**Spider（爬虫）**：它负责处理所有Responses,从中分析提取数据，获取Item字段需要的数据，并将需要跟进的URL提交给引擎，再次进入Scheduler(调度器)，

**Item Pipeline(管道)**：它负责处理Spider中获取到的Item，并进行进行后期处理（详细分析、过滤、存储等）的地方.

**Downloader Middlewares（下载8）**：你可以当作是一个可以自定义扩展下载功能的组件。

**Spider Middlewares（Spider中间件）**：你可以理解为是一个可以自定扩展和操作引擎和Spider中间通信的功能组件（比如进入Spider的Responses;和从Spider出去的Requests）

### 制作 Scrapy 爬虫 一共需要4步：

新建项目 (scrapy startproject xxx)：新建一个新的爬虫项目

明确目标 （编写items.py）：明确你想要抓取的目标

制作爬虫 （spiders/xxspider.py）：制作爬虫开始爬取网页

存储内容 （pipelines.py）：设计管道存储爬取内容

## 新建项目(scrapy start project)

在开始爬取之前，必须创建一个新的Scrapy项目。进入自定义的项目目录中，运行下列命令：

scrapy startproject mySpider

其中， mySpider 为项目名称，可以看到将会创建一个mySpider 文件夹.

下面来简单介绍一下各个主要文件的作用：

scrapy.cfg ：项目的配置文件

mySpider/ ：项目的Python模块，将会从这里引用代码

mySpider/items.py ：项目的目标文件

mySpider/pipelines.py ：项目的管道文件

mySpider/settings.py ：项目的设置文件

mySpider/spiders/ ：存储爬虫代码目录