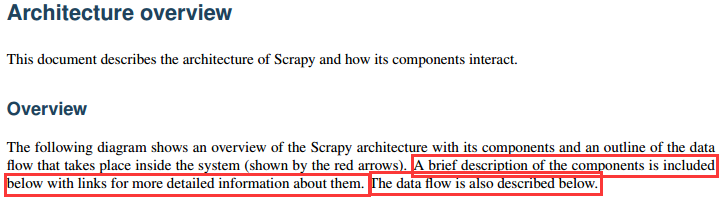
Scrapy框架的原理分析

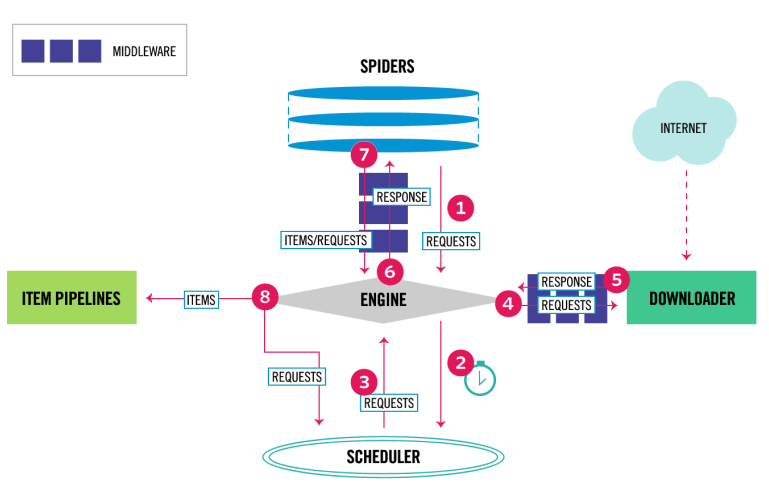
# Scrapy1.4.0 参考文档的第六章

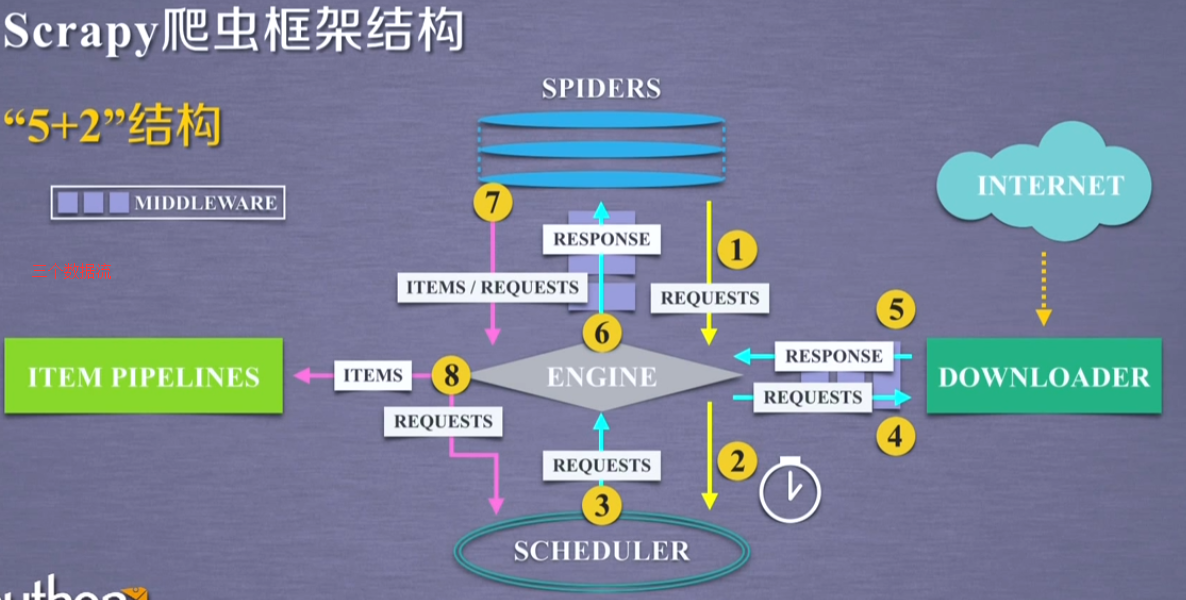
**Scrapy1.4.0 参考文档的第六章**对应的就是Scrapy框架的组件介绍及原理分析，名称为 **Chapter6 Extending Scrapy。**

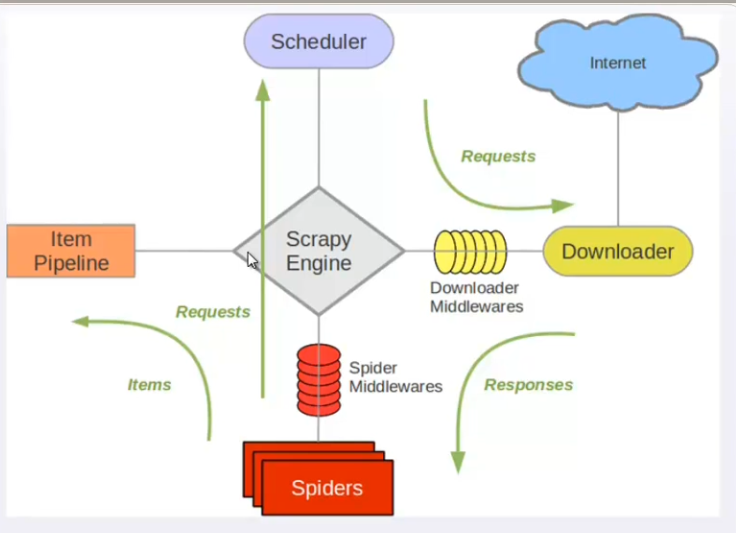


# Scrapy框架结构及工作原理

## Scrapy框架的结构及数据流：







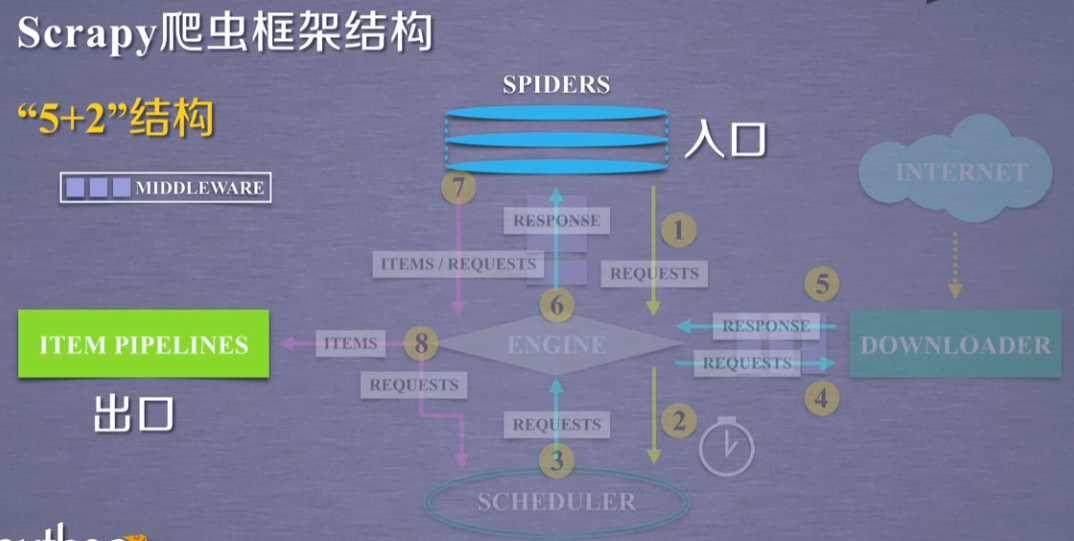
### “5+2”结构

**5个单元和2个中间件**

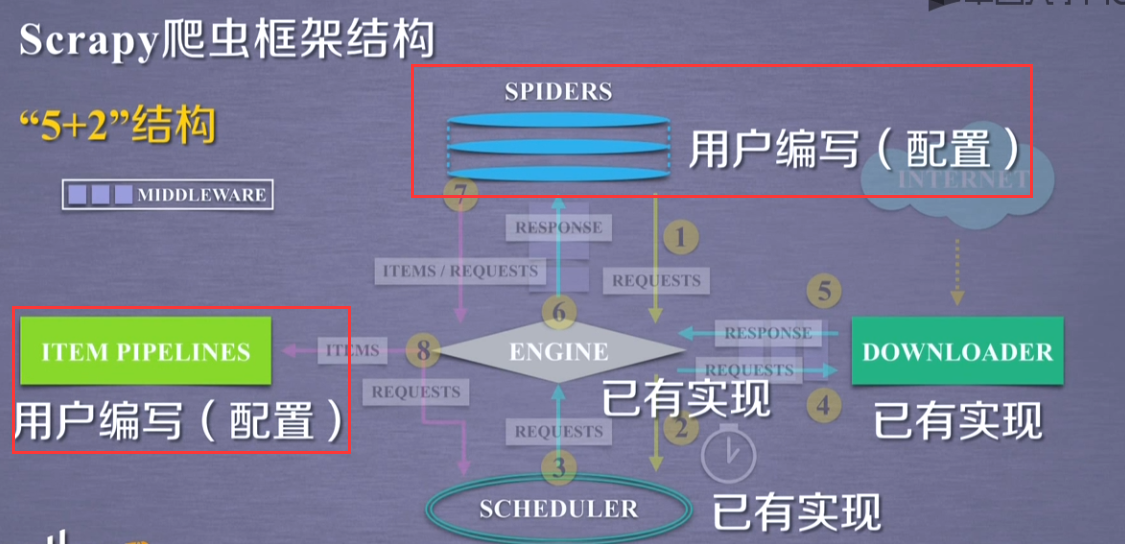
### 三个主要的数据流

**入口：Spiders**

**出口：Item pipelines**



### 用户需要编写（配置）的是？

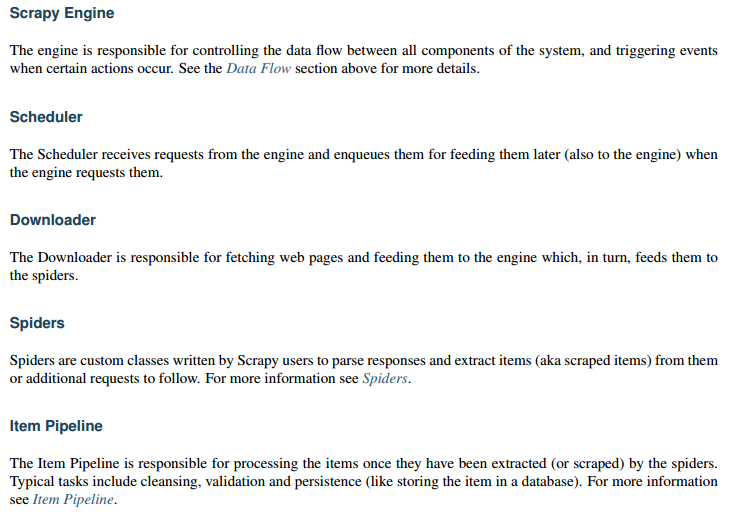


## 模块功能介绍：

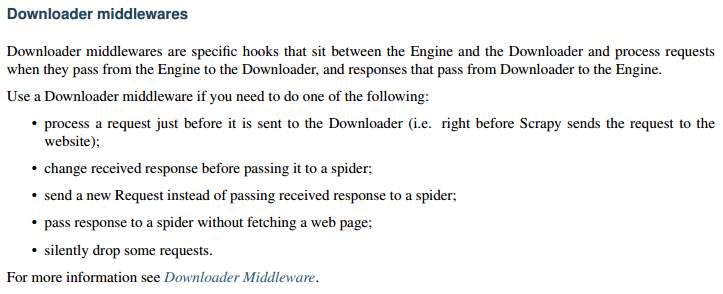


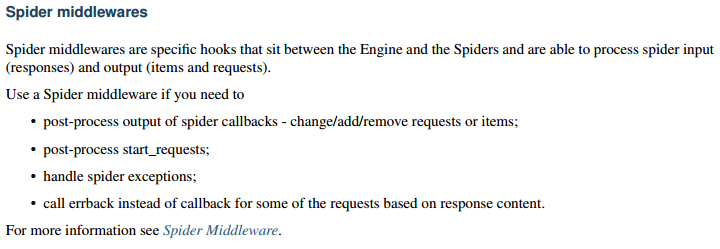


**Components**：5+2

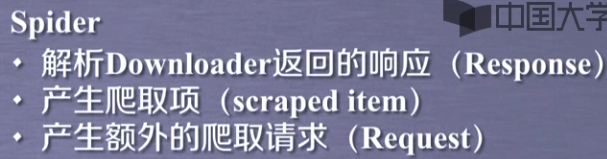


**2个中间件**：

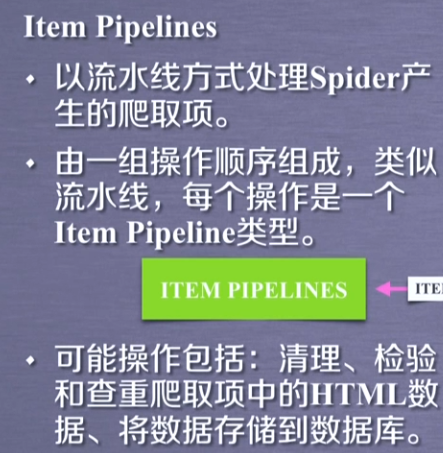




### Spider



### Item Pipelines

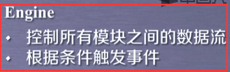




### 中间件：Spider Middleware



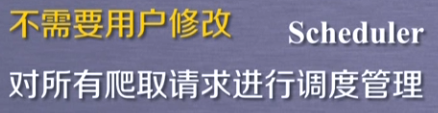
### Engine（核心模块）



### Downloader

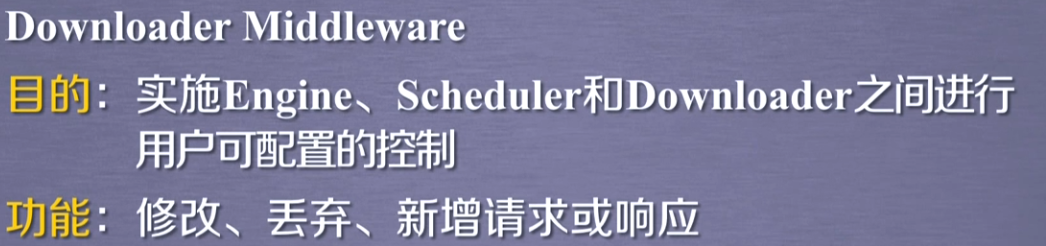


### Scheduler



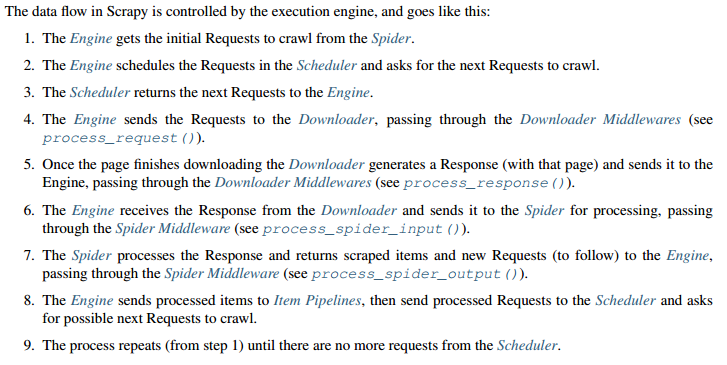
### 中间件：Downloader Middleware

**可以对Request和Response做一些预处理**。

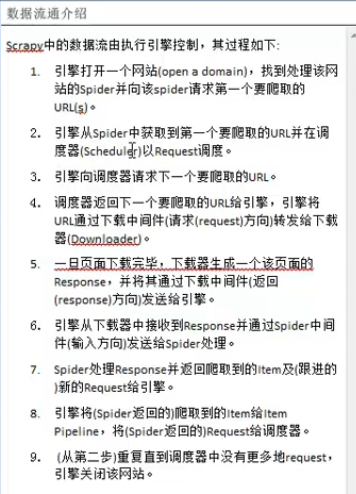


## 数据流通介绍

### 文档英文介绍：

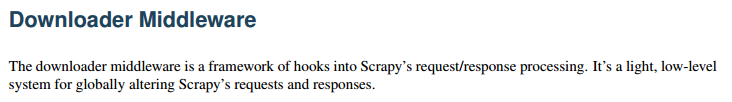


### 中文描述



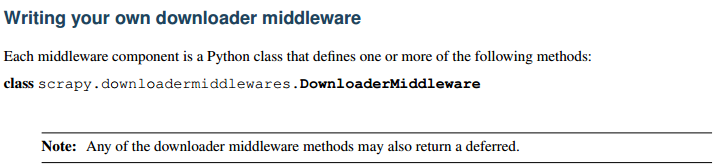
# Downloader Middleware

### 简单介绍



### 自定义Downloader Middleware

需要实现**三个方法**。



process\_request(request, spider)

process\_response(request, response, spider)

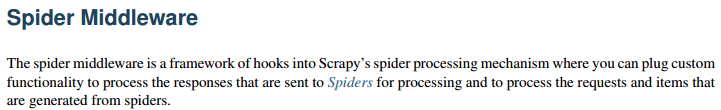
process\_exception(request, exception, spider)

### 内置的Built-in downloader middleware reference

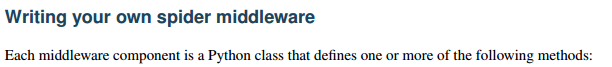
**有很多：**CookiesMiddleware、DefaultHeadersMiddleware等等。

# Spider Middleware

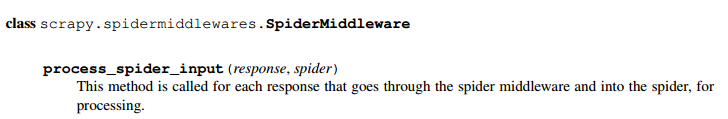
### 简单介绍

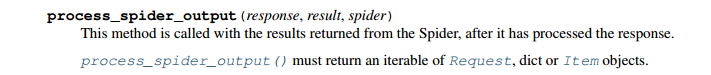


### 自定义Spider Middleware



需要实现四个方法：







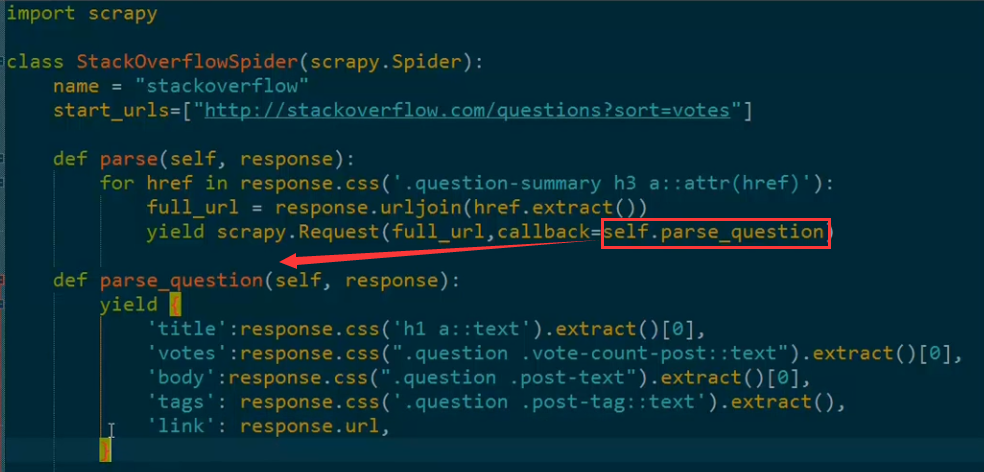


### 内置的Built-in spider middleware reference

也有很多，如DepthMiddleware、HttpErrorMiddleware等等。

# Scrapy框架原理

## 实例：



## 原理讲解：



执行命令 ： scrapy runspider **somefile.py** –o xx.csv

