# SpringBoot与日志

市面上的日志框架：

JUL、JCL、Jboss-logging、logback、log4j、log4j2、slf4j......

|  |  |
| --- | --- |
| 日志门面（日志抽象层） | 日志实现 |
| JCL、Jboss-logging、**slf4j** | JUL、log4j、log4j2、logback |

## slf4j使用原理

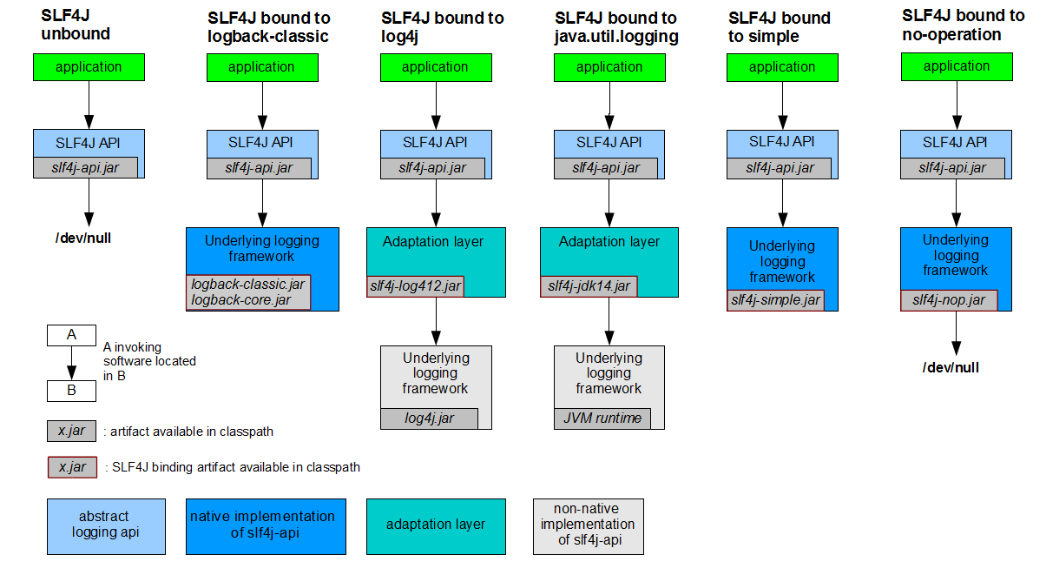
### 如何在系统中使用slf4j

日志记录方法的调用，不应该直接调用接口的实现类，而是调用日志接口抽象层的方法。

官方用户使用手册：<http://www.slf4j.org/manual.html>

|  |
| --- |
| import org.slf4j.Logger; import org.slf4j.LoggerFactory;  public class HelloWorld {   public static void main(String[] args) {     Logger logger = LoggerFactory.getLogger(HelloWorld.class);     logger.info("Hello World");   } } |

系统里要导入slf4j和logback的jar包。



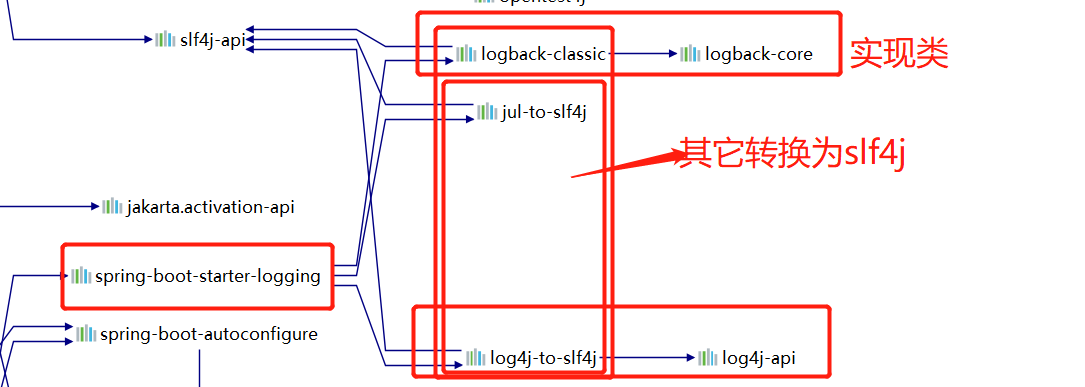
每一个日志的实现框架都有自己的配置文件，使用slf4j之后，**配置文件还是要做成日志实现框架本身的配置文件。**

### 如何系统中的所有日志都统一成slf4j

* 将系统中的其它日志框架先排除出去
* 用中间包来替换原有的日志框架
* 导入slf4j的其它实现

## SpringBoot日志关系

|  |
| --- |
| <**dependency**>  <**groupId**>org.springframework.boot</**groupId**>  <**artifactId**>spring-boot-starter-logging</**artifactId**>  <**version**>2.3.1.RELEASE</**version**>  <**scope**>compile</**scope**> </**dependency**> |



* SpringBoot底层使用的是slf4j+logback方式进行日志记录
* SpringBoot也把其它的日志框架替换成了slf4j
* **如果引入了其它框架，一定要把这个框架的默认日志依赖移除掉**。Springboot能自动适配。

## SpringBoot日志配置

### springboot默认配置

默认级别为info

|  |
| --- |
| Logger **logger** = LoggerFactory.*getLogger*(getClass()); @Test **void** contextLoads() {  *//日志级别，由低到高，可以调整日志的级别，日志就会在该级别及更高的级别生效。* **logger**.trace(**"trace日志。。。"**);  **logger**.debug(**"debug日志。。。"**);  **logger**.info(**"info日志。。。"**);  **logger**.warn(**"warn日志。。。"**);  **logger**.error(**"error日志。。。"**); } |

可以通过配置项来更改级别：**logging.level.com.pcq** = **trace**

### 其它配置：

|  |
| --- |
| *##不指定路径则在当前项目下生成日志文件，也可以指定路径：D:/SpringBoot.log ##logging.file.name=springboot.log  ##在当前磁盘的根路径下创建spring文件夹和log文件夹，使用spring.log作为默认文件* **logging.file.path**=**/spring/log** *##在控制台输出的日志格式* **logging.pattern.console**= *##指定文件中日志输出的格式* **logging.pattern.file**= |

### 指定日志配置文件

在项目目录下配置相应名称的日志文件名即可。

| **Logging System** | **Customization** |  |
| --- | --- | --- |
| Logback | logback-spring.xml, logback-spring.groovy,  logback.xml, or logback.groovy |  |
| Log4j2 | log4j2-spring.xml or log4j2.xml |  |
| JDK (Java Util Logging) | logging.properties |  |

Logback.xml:直接被日志框架识别

Logback-spring.xml：日志框架不直接加载，由springboot解析日志配置，可以使用一些高级功能。例如springProfile的配置，指定在不同的环境中按照不同的日志配置去打印日志。

## 切换日志框架

可以按照slf4j的日志适配图，进行相关的切换。

切换方式：[https://developer.aliyun.com/lesson\_1912\_16330#\_16330](https://developer.aliyun.com/lesson_1912_16330" \l "_16330)