# 日志框架

## 起源

小张；开发一个大型系统；

1、System.out.println("")；将关键数据打印在控制台；去掉？写在一个文件？

2、框架来记录系统的一些运行时信息；日志框架 ； zhanglogging.jar；

3、高大上的几个功能？异步模式？自动归档？xxxx？ zhanglogging-good.jar？

4、将以前框架卸下来？换上新的框架，重新修改之前相关的API；zhanglogging-prefect.jar；

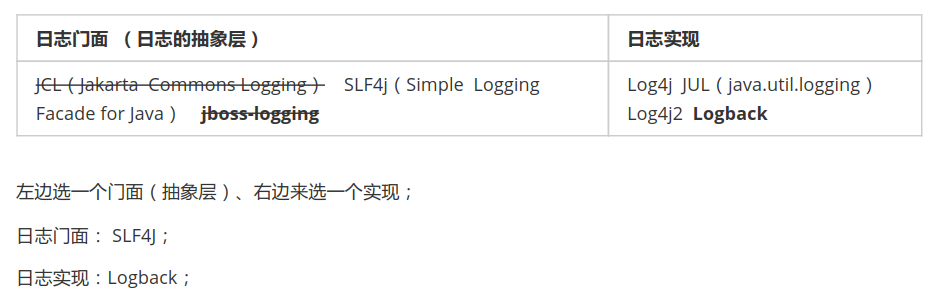
5、JDBC---数据库驱动；

写了一个统一的接口层；日志门面（日志的一个抽象层）；logging-abstract.jar；

给项目中导入具体的日志实现就行了；我们之前的日志框架都是实现的抽象层；

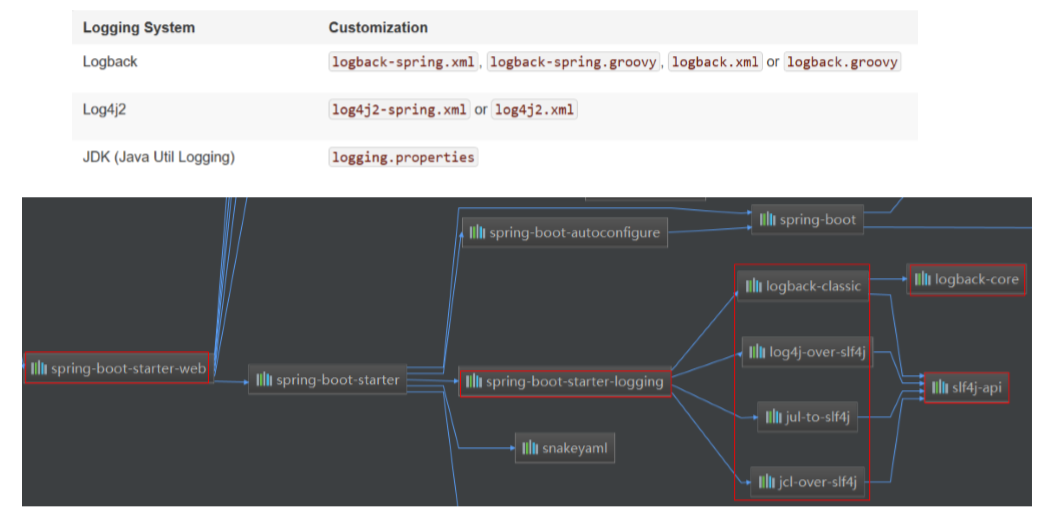
## 市面上的日志框架；

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



## Spring和spring boot

市场上存在非常多的日志框架。JUL（java.util.logging），JCL（Apache Commons Logging），Log4j，Log4j2，Logback、SLF4j、jboss-logging等。 Spring Boot在框架内容部使用JCL，spring-boot-starter-logging采用了 slf4j+logback的形式，Spring Boot也能自动适配（jul、log4j2、logback） 并 简化配置



# SLF4j使用

## 使用slf4j

1、如何在系统中使用SLF4j https://www.slf4j.org

以后开发的时候，日志记录方法的调用，不应该来直接调用日志的实现类，而是调用日志抽象层里面的方法；

给系统里面导入slf4j的jar和 logback的实现jar

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

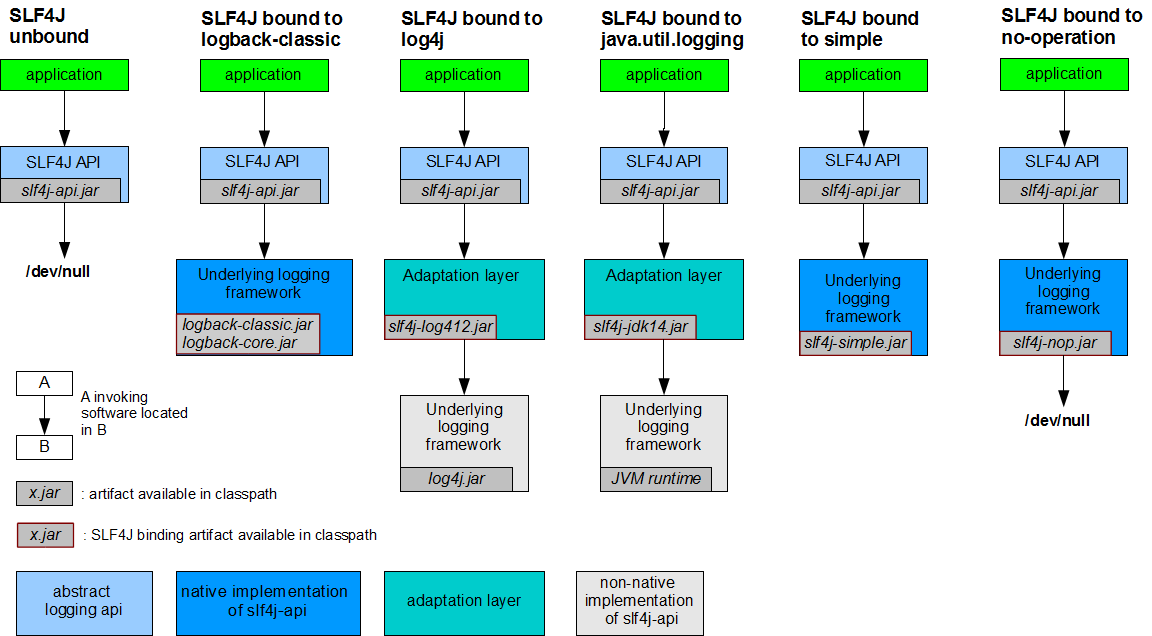
public class HelloWorld {

  public static void main(String[] args) {

    Logger logger = LoggerFactory.getLogger(HelloWorld.class);     logger.info("Hello World");

  }

}

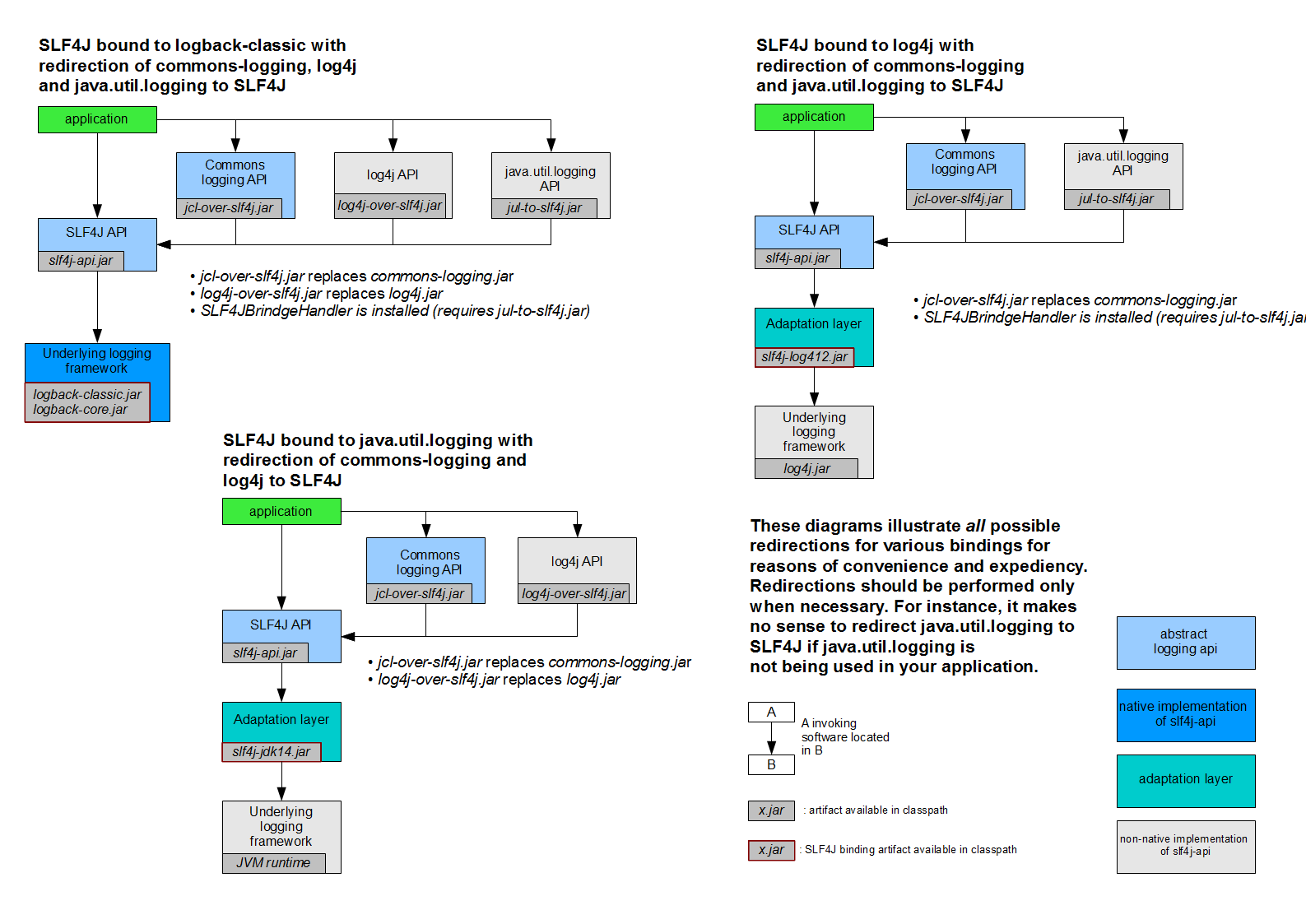


每一个日志的实现框架都有自己的配置文件。使用slf4j以后，配置文件还是做成日志实现框架自己本身的配置文 件；

## 遗留问题

a（slf4j+logback）: Spring（commons-logging）、Hibernate（jboss-logging）、MyBatis、xxxx

统一日志记录，即使是别的框架和我一起统一使用slf4j进行输出？



如何让系统中所有的日志都统一到slf4j；

1、将系统中其他日志框架先排除出去； 2、用中间包来替换原有的日志框架； 3、我们导入slf4j其他的实现