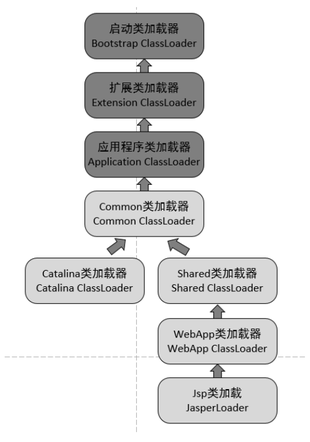
类加载

生命周期

- ❶ 加载
  - 作用
    - 通过类的全限定名来获取类的二进制字节流
    - 将字节流所代表的静态存储结构转化为方法区的运行时数据结构
    - 生成 java.lang.Class 对象作为访问入口
  - 字节流来源
    - 从 ZIP 压缩包中获取（JAR、EAR、WAR）
    - 从网络中获取（Web Applet）
    - 运行时计算生成（动态代理）
    - 由其他文件生成（JSP）
    - 从数据库中读取
    - 从加密文件中获取
- ❷ 连接
  - 验证
    - 作用 — 确保 Class 字节流中的信息符合虚拟机约束要求
    - 阶段
      - 文件格式验证
      - 元数据验证
      - 字节码验证
      - 符号引用验证
  - 准备
    - 作用 — 为类变量（static）分配内存并设置初始值（数据类型的零值，ConstantValue 属性除外）
    - 内存分配 — 方法区，Java 堆
  - 解析
    - 作用 — 将常量池内的符号引用替换为直接引用
    - 解析对象
      - 类或接口
      - 字段
      - 类方法
      - 接口方法
      - 方法类型
      - 方法句柄
      - 调用点限定符
- ❸ 初始化
  - 作用 — 初始化类变量和其他资源
  - <clinit>()
    - 静态语句块只能访问在其前面定义的变量，在其后面定义的变量，只能赋值不能访问
    - 父类的 <clinit>() 方法在子类的 <clinit>() 方法之前执行
    - 父类中定义的静态语句块优先于子类的变量赋值操作
    - <clinit>() 方法对类或接口来说不是必需的
    - 执行接口的 <clinit>() 方法不会执行父接口的 <clinit>() 方法；接口的实现类在初始化时不会执行接口的 <clinit>() 方法；仅当接口变量被使用时，接口才会被初始化
    - 类的 <clinit>() 方法在多线程环境中被正确的加锁同步
- ❹ 使用
- ❺ 卸载

类加载器

- 定义 — 实现"通过一个类的全限定名来获取描述该类的二进制字节流"这个动作的代码被称为"类加载器"
- 类与类加载器 — 任意类都由其类加载器与类本身共同确立唯一性
- 应用领域
  - 类层次划分
  - OSGi
  - 程序热部署
  - 代码加密
- 双亲委派模型
  - 目的 — Java 类随类加载器一起具备带优先级的层次关系（保证 Object 类在各种类加载器中是同一个类）
  - JDK 8
    - 层次
      - 启动类加载器 — 加载 lib 目录中或被 -Xbootclasspath 参数所指定的路径中，且能被虚拟机识别（rt.jar、tools.jar）的类库
      - 扩展类加载器 — 加载 ext 目录中或被 java.ext.dirs 系统变量所指定的路径中的所有类库
      - 应用程序类加载器 — 加载用户路径（ClassPath）上所有的类库
      - 自定义类加载器 — 加载磁盘外的 Class 文件或实现类的隔离、重载
    - 约束 — 除顶层启动类加载器外，其余加载器都应有父加载器（通常使用组合关系复用父加载器代码）
    - 加载流程 — 类加载器收到类加载请求，逐次往上委托给父加载器加载，仅当父加载器无法加载时（找不到类），才由子加载器尝试加载
  - JDK 9
    - 层次
      - 启动类加载器
      - 平台类加载器
      - 应用程序类加载器
      - 自定义类加载器
- 破坏双亲委派模型
  - Tomcat 容器



  - SPI 服务（JNDI、JDBC、JCE、JAXB、JBI）
    - 线程上下文加载器
    - ServiceLoader
  - 代码热替换、模块热部署（OSGi）
  - JDK9 模块化 — 静态模块化，且不支持多版本