

Загрузка классов Java

Как можно идентифицировать класс в Java ?

Как можно идентифицировать класс в Java ?

- Полное имя класса

Как можно идентифицировать класс в Java ?

- Полное имя класса

`classloaders.Person`

Как можно идентифицировать класс в Java ?

- Полное имя класса

`classloaders.Person`

Можно ли иметь несколько разных классов с одинаковым названием ?

Как можно идентифицировать класс в Java ?

- Полное имя класса

`classloaders.Person`

Можно ли иметь несколько классов с одинаковым идентификатором ?

Ответ — Да, если эти классы загружены разными `ClassLoader`

- Java — язык с отложенной (ленивой) загрузкой классов
- Класс в Java может загружаться как из скомпилированного байт кода, так и может быть также получен из других источников (загружен по сети, сгенерирован самим приложением)

- Java — язык с отложенной (ленивой) загрузкой классов
- Класс в Java может загружаться как из скомпилированного байт кода, так и может быть получен из других источников (загружен по сети, сгенерирован самим приложением)

Так как же это реализовано в Java ?



- Согласно спецификации Java SE для того, чтобы получить работающий в JVM код, необходимо выполнить три этапа:

- Согласно спецификации Java SE для того, чтобы получить работающий в JVM код, необходимо выполнить три этапа:

- 1) загрузка байт-кода из ресурсов и создание экземпляра класса Class

- Согласно спецификации Java SE для того, чтобы получить работающий в JVM код, необходимо выполнить три этапа:

1) загрузка байт-кода из ресурсов и создание экземпляра класса Class

2) связывание (или линковка)

- **Verification**, происходит проверка корректности полученного байт-кода.

- **Preparation**, выделение оперативной памяти под статические поля и инициализация их значениями по умолчанию (при этом явная инициализация, если она есть, происходит уже на этапе инициализации).

- **Resolution**, разрешение символьных ссылок типов, полей и методов.

- Согласно спецификации Java SE для того, чтобы получить работающий в JVM код, необходимо выполнить три этапа:

1) загрузка байт-кода из ресурсов и создание экземпляра класса Class

2) связывание (или линковка)

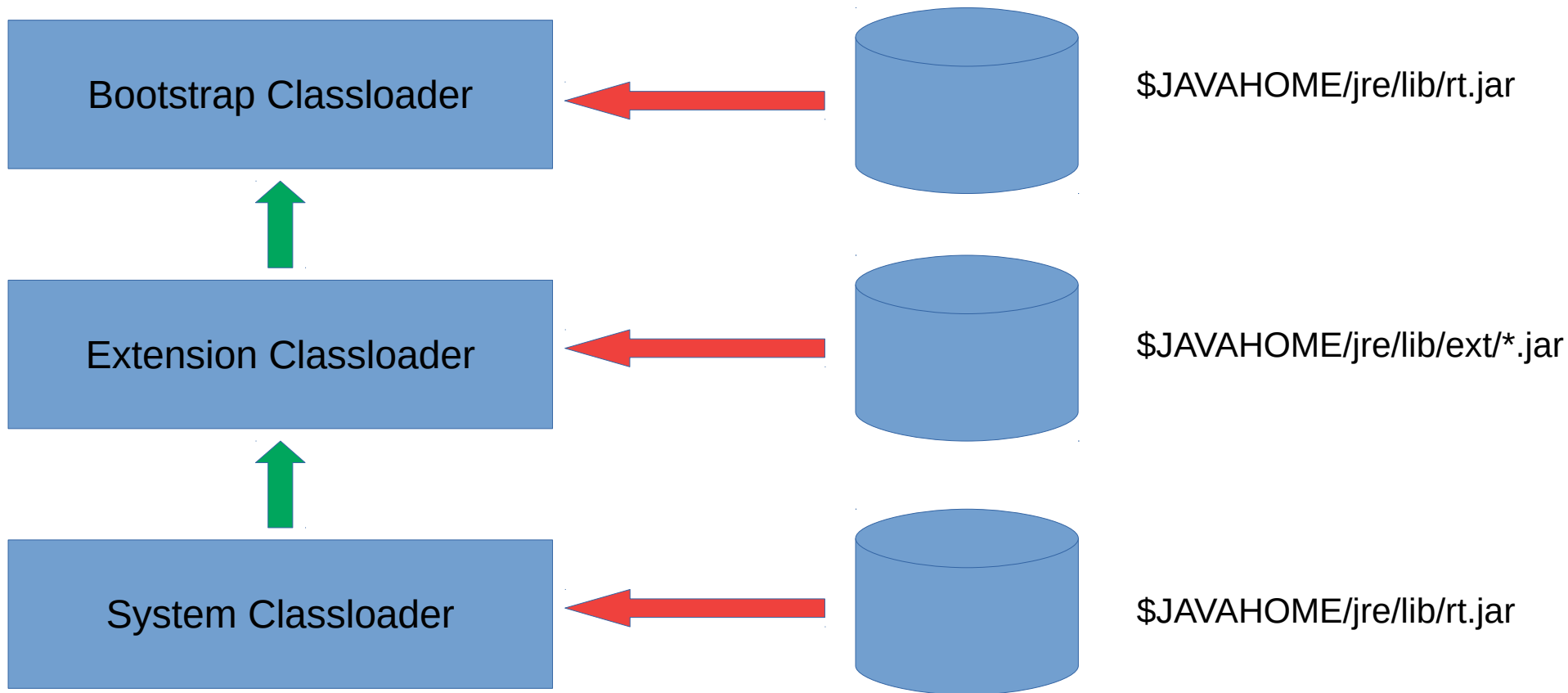
- **Verification**, происходит проверка корректности полученного байт-кода.

- **Preparation**, выделение оперативной памяти под статические поля и инициализация их значениями по умолчанию (при этом явная инициализация, если она есть, происходит уже на этапе инициализации).

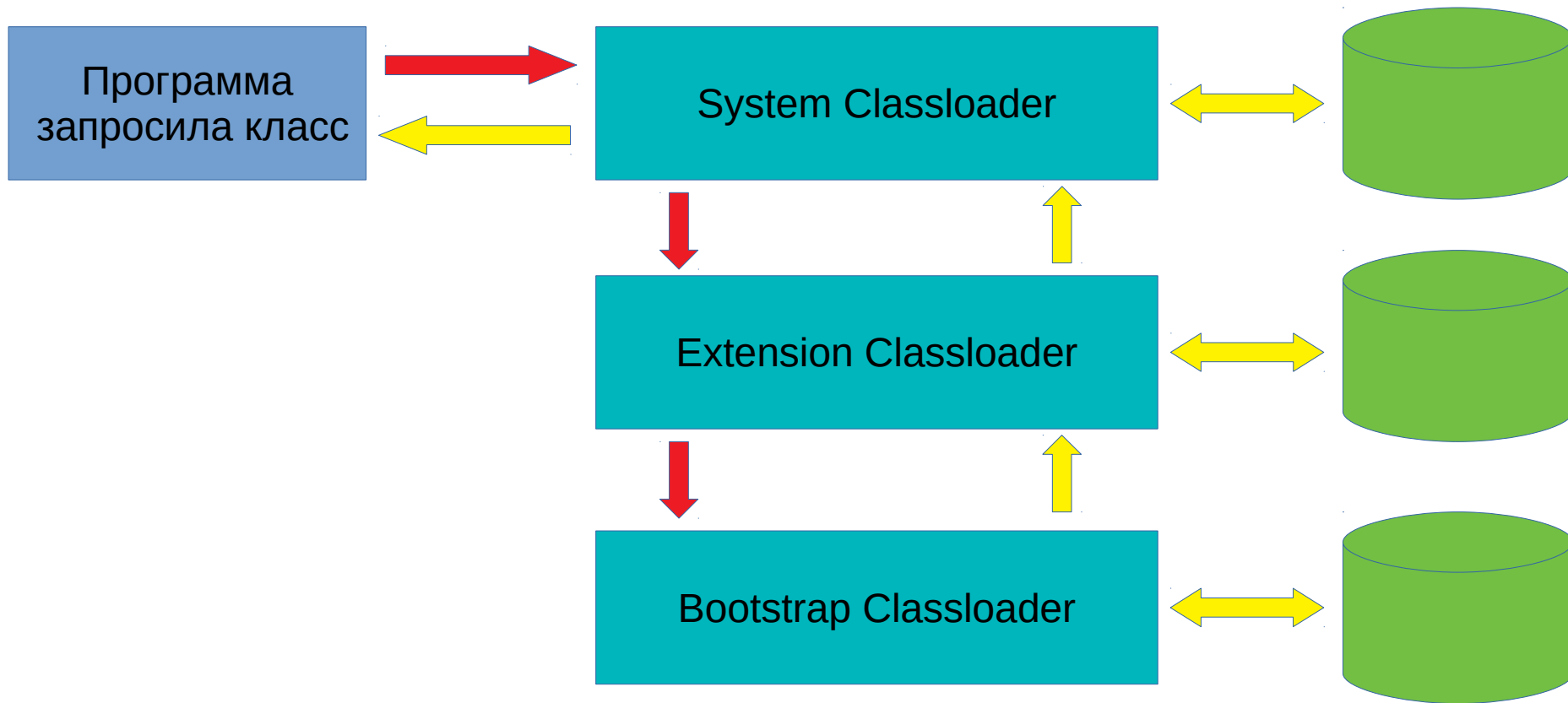
- **Resolution**, разрешение символьных ссылок типов, полей и методов.

3) инициализация полученного объекта

Типы и иерархия загрузчиков классов



Процесс загрузки классов (модель делегирования)



Что будет если ни один из загрузчиков не
найдет нужный класс ?

Что будет если ни один из загрузчиков не
найдет нужный класс ?

`java.lang.NoClassDefFoundError`

Caused by: `java.lang.ClassNotFoundException`

Что будет если ни один из загрузчиков не
найдет нужный класс ?

`java.lang.NoClassDefFoundError`

Caused by: `java.lang.ClassNotFoundException`

А кто знает, почему может возникнуть такое
исключение ?

`java.lang.NoSuchMethodError`

Каждый загрузчик, за исключением базового, является потомком абстрактного класса **java.lang.ClassLoader**

Каждый загрузчик, за исключением базового, является потомком абстрактного класса **java.lang.ClassLoader**

Например :

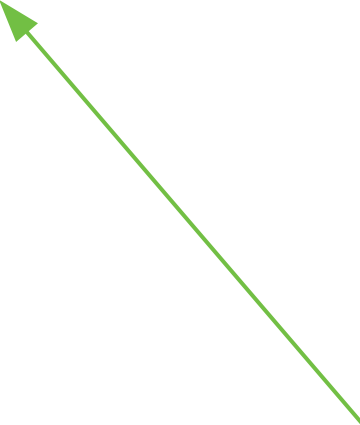
- **sun.misc.Launcher\$ExtClassLoader** - реализацией загрузчика расширений
- **sun.misc.Launcher\$AppClassLoader** - реализация системного загрузчика
- Базовый загрузчик является нативным и его реализация включена в JVM.

Текущий загрузчик классов

```
public class A {  
    public void doSmoth() {  
        B b = new B();  
        b.doSmothElse();  
    }  
}
```

Текущий загрузчик классов

```
public class A {  
    public void doSmoth() {  
        B b = new B();  
        b.doSmothElse();  
    }  
}
```



Эквивалентно

```
A.class.getClassLoader().loadClass("B");
```



Спасибо.

Денис Павлов

GitHub : <https://github.com/DenisPavlov/>