

Maven как средство сборки проекта

Докладчик:

Раскосов Александр

email:

oleksandr_raskosov@epam.com

skype:

ralexander13

- 
1. Обзор существующих framework'ов для сборки проекта
 2. Обзор механизма работы maven
 3. Пример1. Использование COS (Cloud Oriented Storage) как maven репозитория
 4. Пример2. Maven плагин для оптимизации JavaScript

ANT

- Большое количество стандартных заданий (> 150)
- Большая гибкость при разработке скрипта для сборки

Ivy

- Менеджер зависимостей для Ant
- По умолчанию использует maven2 репозиторий

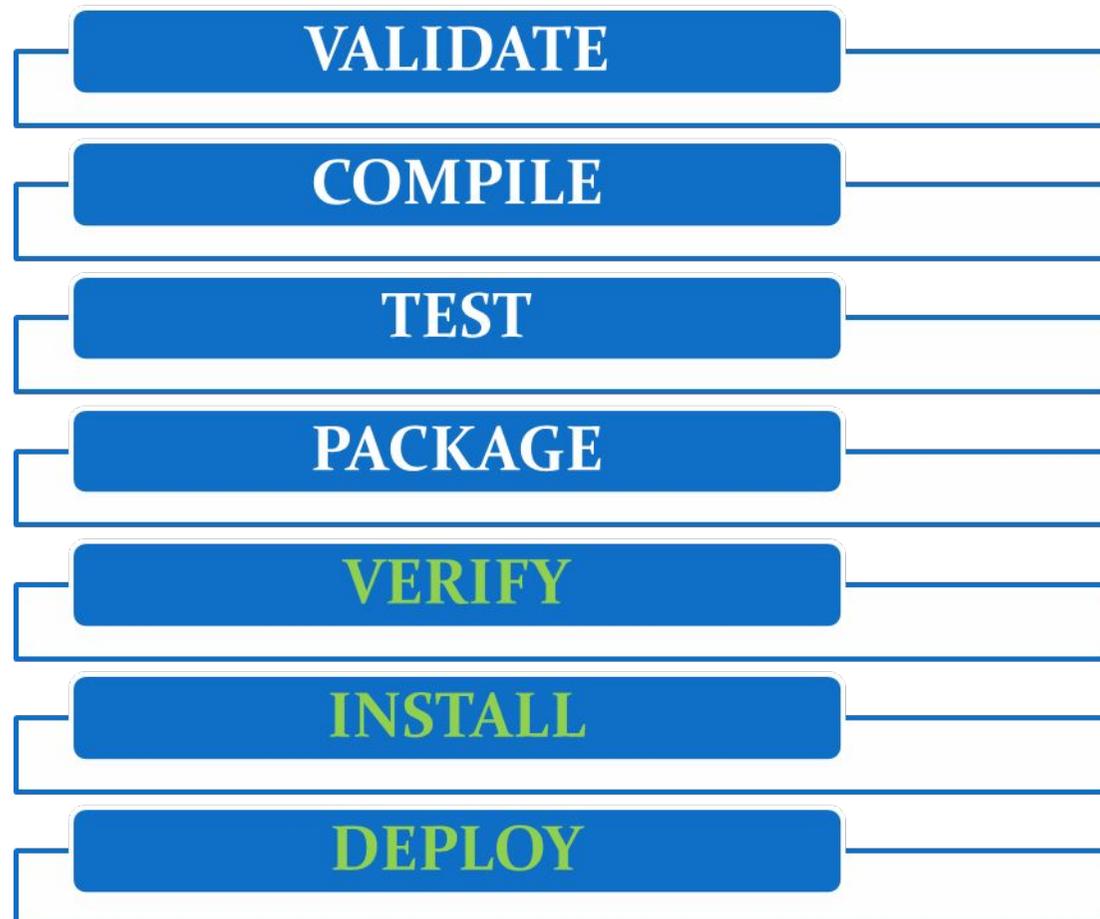
Maven

- Описание структуры проекта, а не только механизма сборки (независимость от IDE)
- Автономный репозиторий сторонних библиотек (> 253 000)
- Лёгкое подключение модулей
- Большой набор плагинов (> 50)
- Поддержка Ant

Статистика использования

- Ant - ?
- Ivy - ?
- Maven – ? + Раскосов Александр

Жизненный цикл maven





Пример 1

**Использование COS (Cloud Oriented Storage)
как maven репозитория**

Использование COS как maven репозитория

При сохранении в репозиторий:

1. Поддержка только FTP и SSH
2. Сторонняя библиотека только для Amazon S3

Использование COS как maven репозитория

```
public class AtmosWagon extends AbstractWagon {  
    public void put(File source, String destination) {...}  
  
    public boolean getIfNewer(  
        String resourceName, File destination,  
        long timestamp) {...}  
  
    public void get(String resourceName, File destination) {...}  
}
```

Использование COS как maven репозитория

```
../resources/META-INF/plexus/components.xml
<component-set>
  <components>
    <component>
      <role>org.apache.maven.wagon.Wagon</role>
      <role-hint>atmos</role-hint>
      <implementation>
        com.maestro.maven.wagon.atmos.AtmosWagon
      </implementation>
      <instantiation-strategy>per-lookup</instantiation-strategy>
    </component>
  </components>
</component-set>
```

Использование COS как maven репозитория

При чтении из репозитория:

1. Необходимость конфигурации maven для использования сертификатов для работы с репозиторием по https



Пример 2

Maven плагин для оптимизации JavaScript

Maven плагин для компиляции и оптимизации JavaScript

Стандартные механизмы:

1. Скачиваемое Java приложение
2. Работа через web приложение

Maven плагин для компиляции и оптимизации JavaScript

```
/**
 * @goal run
 * @phase compile
 */
public class GoogleCompilerMojo extends AbstractMojo {
    /**
     * @parameter expression="false"
     */
    private boolean skip;

    public void execute()
        throws MojoExecutionException, MojoFailureException {...}
}
```

Maven плагин для компиляции и оптимизации JavaScript

```
../pom.xml
```

```
...
```

```
<packaging>maven-plugin</packaging>
```

```
...
```



Вопросы???