



Пожалуйста, подождите. Jenkins готовится к работе

Эта страница будет перезагружена когда Jenkins будет готов.

Автоматизация тестирования Jenkins CI

Страница сгенерирована



Содержание

1. Задачи CI в рамках автоматизации тестирования

2. Разворачивание Jenkins

- **2.1. Установка Java**

- **2.2. Установка war** контейнера

3. Работа с Jenkins

- **3.1. Plugins**

- **3.2. Slaves**

- **3.3. Jobs**

- **3.3.1. Общие настройки проекта**

- **3.3.2. Структура задач**

- **3.3.3. Процесс получения результата**

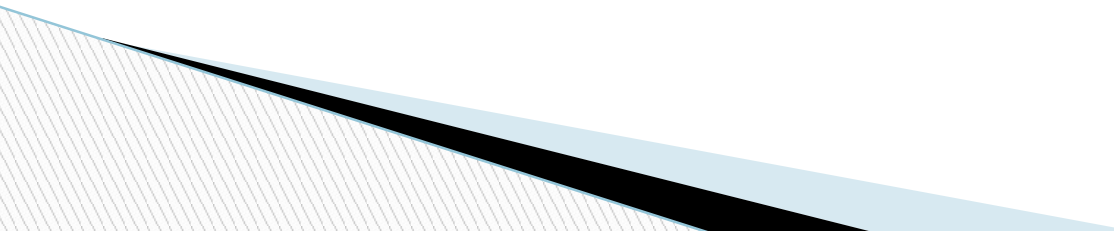
- **3.3.4. Настройка View**

- **3.3.5. Работа с артефактами**

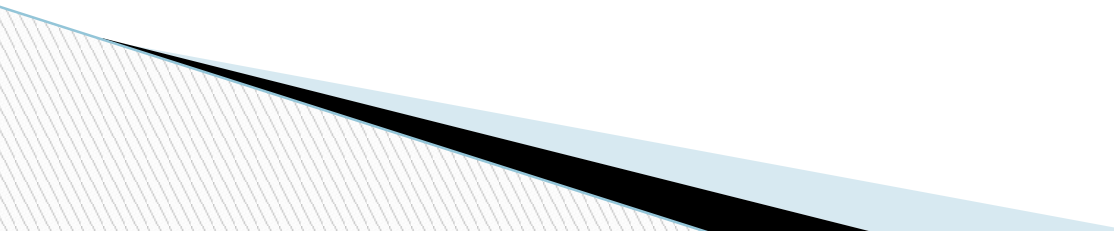
Предмет лекции

Предметом лекции является изучение настройки и работы с **Jenkins CI** в рамках процесса автоматизации тестирования.

Результатом является приобретение необходимых навыков для самостоятельной установки окружения для запуска тестов при помощи **Jenkins**



Достоинства

- **1.** Open source
 - **2.** Простой в установке
 - **3.** Простой в управлении
 - **4.** Гибкий
 - **5.** Управление с использованием файловой системы
- 

Недостатки

- ▣ **1.** Вероятные баги при обновлении до новой версии

Задачи CI в рамках автоматизации тестирования

- Управление планами сборок
- Запуск
- Хранение результатов

2. Разворачивание Jenkins

2.1. Установка Java

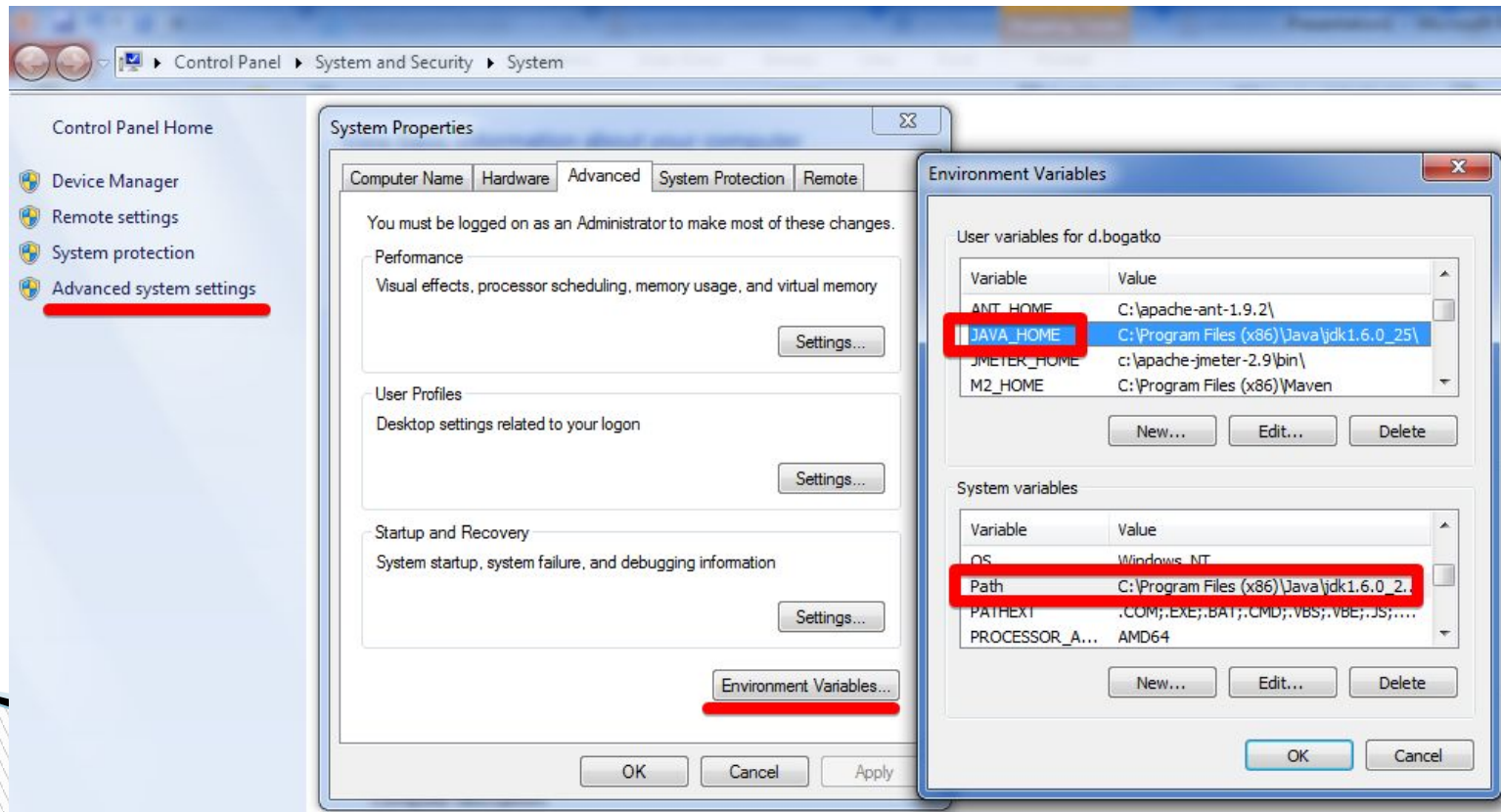
1. Установка JAVA

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Добавление переменных окружения:

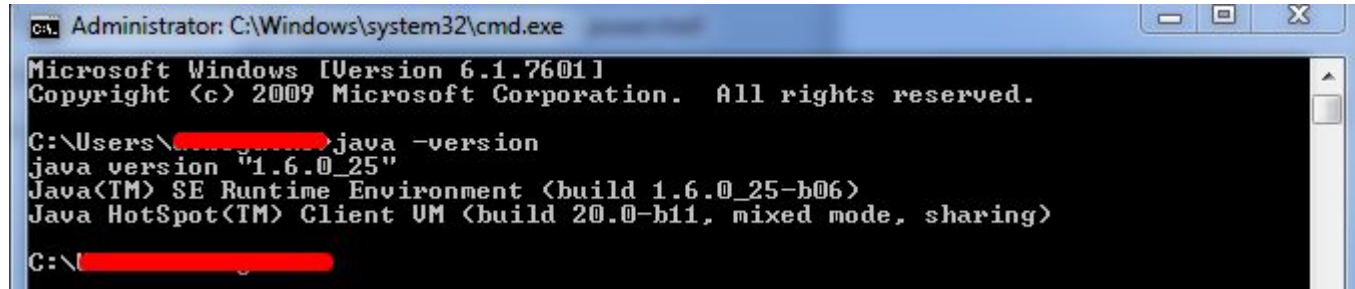
JAVA_HOME: C:\Program Files (x86)\Java\jdk1.6.0_25\

Path: C:\Program Files (x86)\Java\jdk1.6.0_25\bin



2.1. Установка Java

cmd.exe java -version



```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\>java -version
java version "1.6.0_25"
Java(TM) SE Runtime Environment (build 1.6.0_25-b06)
Java HotSpot(TM) Client VM (build 20.0-b11, mixed mode, sharing)

C:\>
```

2.1. Распаковка war контейнера

Для начала необходимо скачать war файл jenkins с официального сайта:
<http://jenkins-ci.org/>

java -jar jenkins.war

Распаковка будет производится в C:\Jenkins

При необходимости установки в отличную дирректорию, следует добавить переменную окружения JENKINS_HOME

Jenkins будет доступен по порту адресу localhost:8080

При необходимости смены порта команда распаковки будет следующей:

java -jar jenkins.war --httpPort=9999


3. Работа с Jenkins


3.1. Plugins


← → ↻


localhost:8080/manage


Jenkins ▶

 [История сборки](#)

 [Связи проектов](#)

 [Проверить хэш файла](#)

 [Настроить Jenkins](#)


 [Credentials](#)


Очередь сборки.


Очередь сборки пуста


Состояние сборщиков


#	Статус.
1	В ожидании


 Незащищенный Jenkins позволяет каждому, у кого есть доступ к серверу, чтобы избежать злоупотреблений.

 Secrets in your \$JENKINS_HOME has been re-keyed successfully. [Please](#)

[Конфигурирование системы](#)
Конфигурировать глобальные настройки и пути.

[Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.

[Перечитать настройки из файла](#)
Отменить все загруженные данные в памяти и перечитать все из файла.

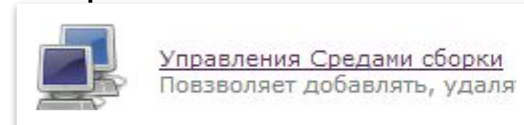
[Управление плагинами](#)
Добавить, удалить, выключить или включить плагины, расширяющие функционал Jenkins.

3. Работа с Jenkins

3.2. Slaves

Подключение других машин к вашему Jenkins

- Переходим в настройки --> управление средами сборки и:



- Создаем новый узел



- Конфигурируем подключение «Launch slave agents via Java Web Smart»

Использование	Utilize this slave as much as possible
Способ запуска	Launch slave agents via Java Web Start
Доступность	Keep this slave on-line as much as possible

Node Properties

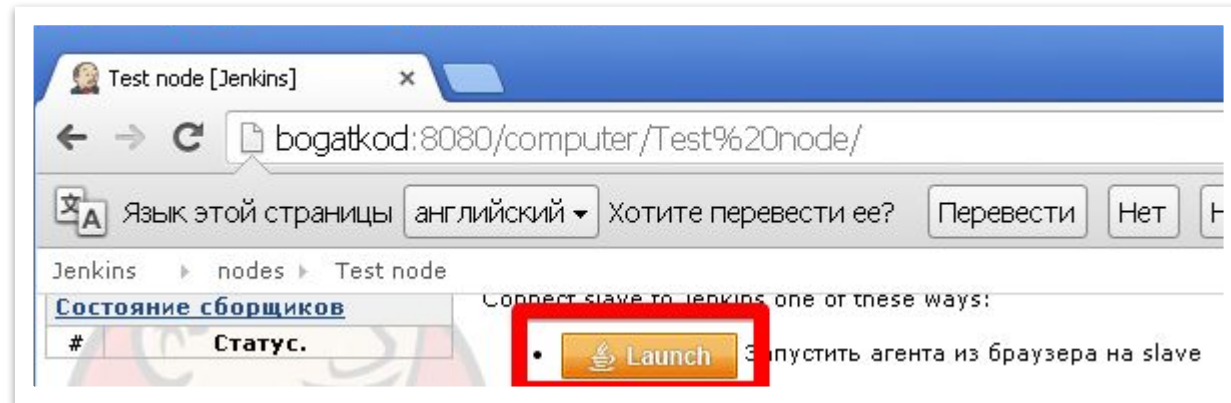
☐ Environment variables

☐ Tool Locations

Save

Подключение slave-agent.jnlp

- скачиваем slave-agent.jnlp:



- подключаем ноду к Jenkins: **javaws**
slave-agent.jnlp



3.3. Jobs

3.3.1. Создание новой задачи

← → ↻

Jenkins

Jenkins > All >

- [Создать Job](#)
- [Люди](#)
- [История сборок](#)
- [Связи проектов](#)
- [Проверить хэш файла](#)
- [Настроить Jenkins](#)
- [Credentials](#)

Очередь сборок.

Очередь сборок пуста

Состояние сборщиков

Имя Job'a

- ☐ **Создать задачу со свободной конфигурацией**
Это основная функция Jenkins. Jenkins будет с
- ☐ **Создать проект maven2/3**
Создать проект maven2/3. Jenkins будет испол
- ☐ **Мультиконфигурационный проект**
Подходит для проектов, требующих большое к
- ☐ **Отслеживать внешнюю задачу**
Этот тип задачи позволяет вам сохранить резул
существующей системой автоматизации. Смотри
- ☐ **Копия существующего Job'a**
Копировать из

3.3.2. Структура задач

Общие настройки проекта

Проект имя	<input type="text" value="Test job"/>
Описание	<div></div>
	[Raw HTML] Предпросмотр
<input type="checkbox"/> Удалять устаревшие сборки	
GitHub project	<input type="text"/>
<input type="checkbox"/> Это параметризованная сборка	
<input type="checkbox"/> Приостановить сборки (Новые сборки не будут выполняться пока опция включена)	
<input type="checkbox"/> Выполнять процесс сборки по необходимости	
<input type="checkbox"/> Ограничить количество узлов, которые могут собирать этот проект	

3.3.2. Структура задач

Управление исходным кодом

Управление исходным кодом

- ☐ CVS
- ☐ CVS Projectset
- ☐ Git
- ☐ Subversion
- ☒ Нет

Триггеры сборки

Триггеры сборки

- ☐ Запустить сборку после завершения другой
- ☐ Build when a change is pushed to GitHub
- ☐ Опрашивать SCM об изменениях
- ☐ Собирать периодически

Pre-build Steps, Build Steps and Post-build steps

Add pre-build step ▼

- Android: Run tests in Testdroid Cloud
- Build a Visual Studio project or solution using MSBuild
- Copy artifacts from another project
- Create Android build files
- Execute NAnt build
- Install Android package
- Install Android project prerequisites
- Invoke Rake
- Invoke TestLink
- Progress JIRA issues by workflow action
- Run Android monkey tester
- Run unit tests with MSTest
- Scriptler script
- Uninstall Android package
- Вызвать Ant
- Вызвать цели Maven верхнего уровня
- Выполнить команду Windows
- Выполнить команду shell

3.3.3. Процесс получения результата

1

Сборка

 **Выполнить команду Windows**

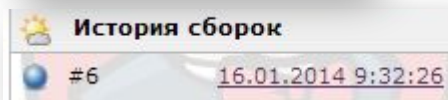
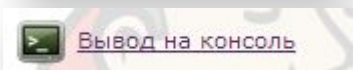
Команда

[See the list of available environment variables](#)

2



3

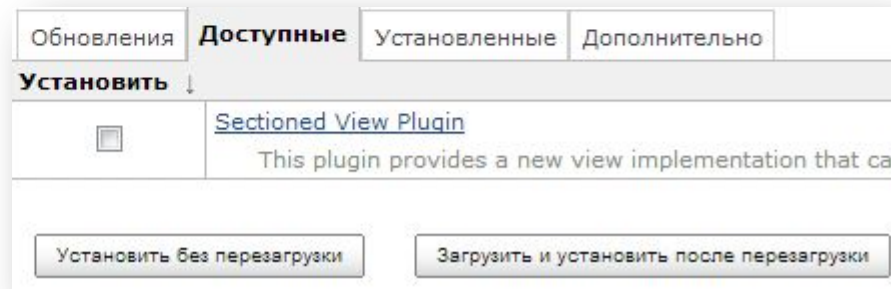


4

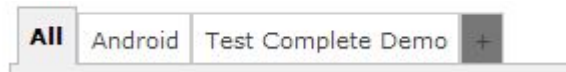


3.3.4. Работа с View

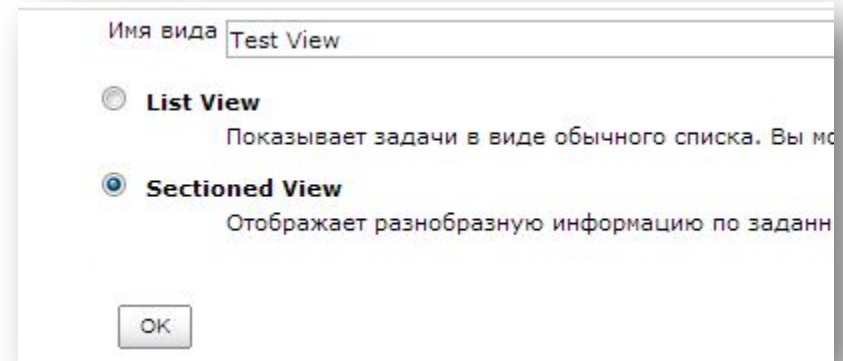
- Для работы с View удобно использовать Plugin: Sectioned View Plugin



Для создания вкладки достаточно нажать “+”



и выбрать необходимый тип



3.3.4. Работа с View

- Sectioned View Plugin предоставляет след.
ВОЗМОЖНОСТИ

Sections

Job Graphs Section

Name

List View Section

Name

Test Result Section

Name

Text Section

Name

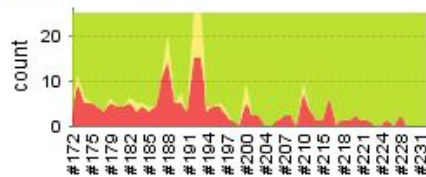
3.3.4. Работа с View

S	W	Console	Name	Крайний успех	Number of b
			Profiles	1 день 1 час - #232	12
			Registration	1 день 0 часов - #317	16
			Registration 2	21 часов - #290	4 94
			Smoke	1 день 0 часов - #475	3 78
			UnitNotifications	5 часа 31 минут - #173	1 31

[Profiles](#)

Test Result Trend

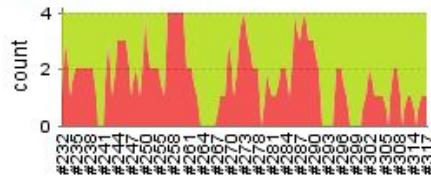
Latest:(Нет ошибок)



[Registration](#)

Test Result Trend

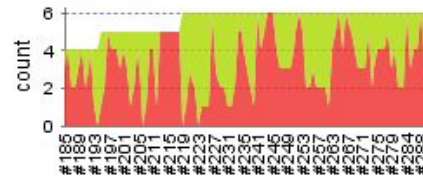
Latest:(1 ошибка / ±0)



[Registration 2](#)

Test Result Trend

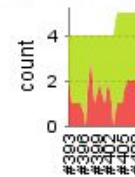
Latest:(2 ошибок / -2)



[Smoke](#)

Te

Latest:(1 оши

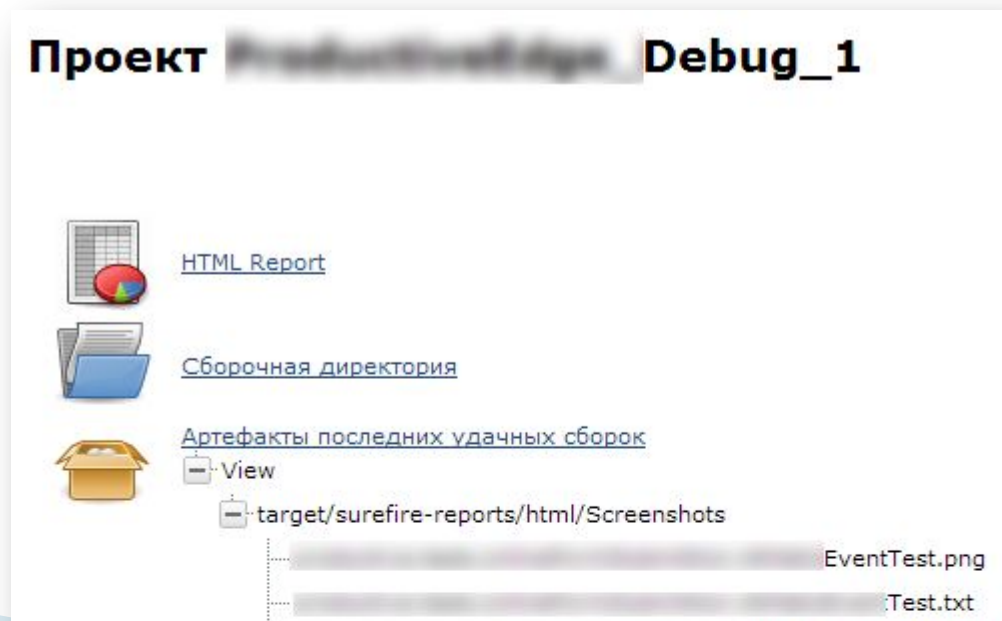


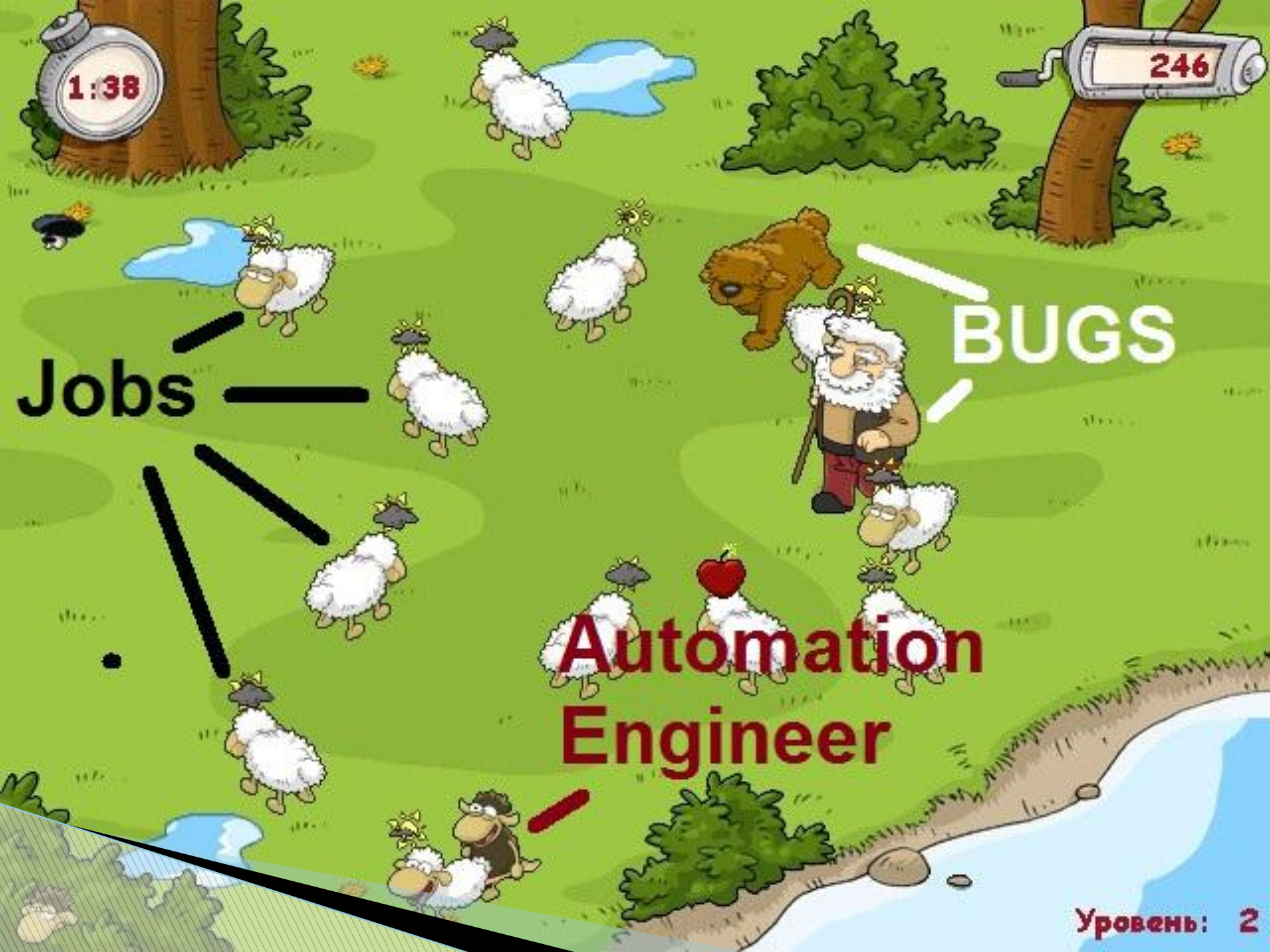
3.3.5. Работа с артефактами

В настройках Job



По завершении билда





1:38

246

BUGS

Automation
Engineer

Jobs

Уровень: 2