

TeamCity



Алексей
Метляков



Алексей Метляков

DevOps Engineer

OpenWay



Алексей Метляков

План занятия

1. [TeamCity](#)
2. [Сущности TeamCity](#)
3. [Дополнения](#)
4. [Итоги](#)
5. [Домашнее задание](#)

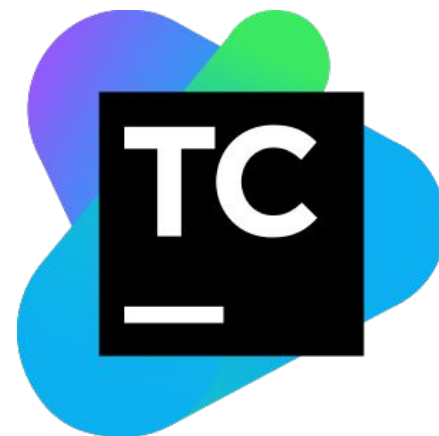


TeamCity

Что такое TeamCity?

TeamCity - ППО для управления CI\CD процессами

- Существует в следующих исполнениях:
 - **Professional** - бесплатная версия с запуском до 3-х агентов и 100 конфигураций сборок
 - **Build Agent License** - добавляет +1 агент и +10 сборок к Professional
 - **Enterprise** - набор платных лицензий с разными пакетами агентов и сборок
 - Программа поддержки [OpenSource](#)
- Использует встроенный синтаксис **Kotlin DSL**

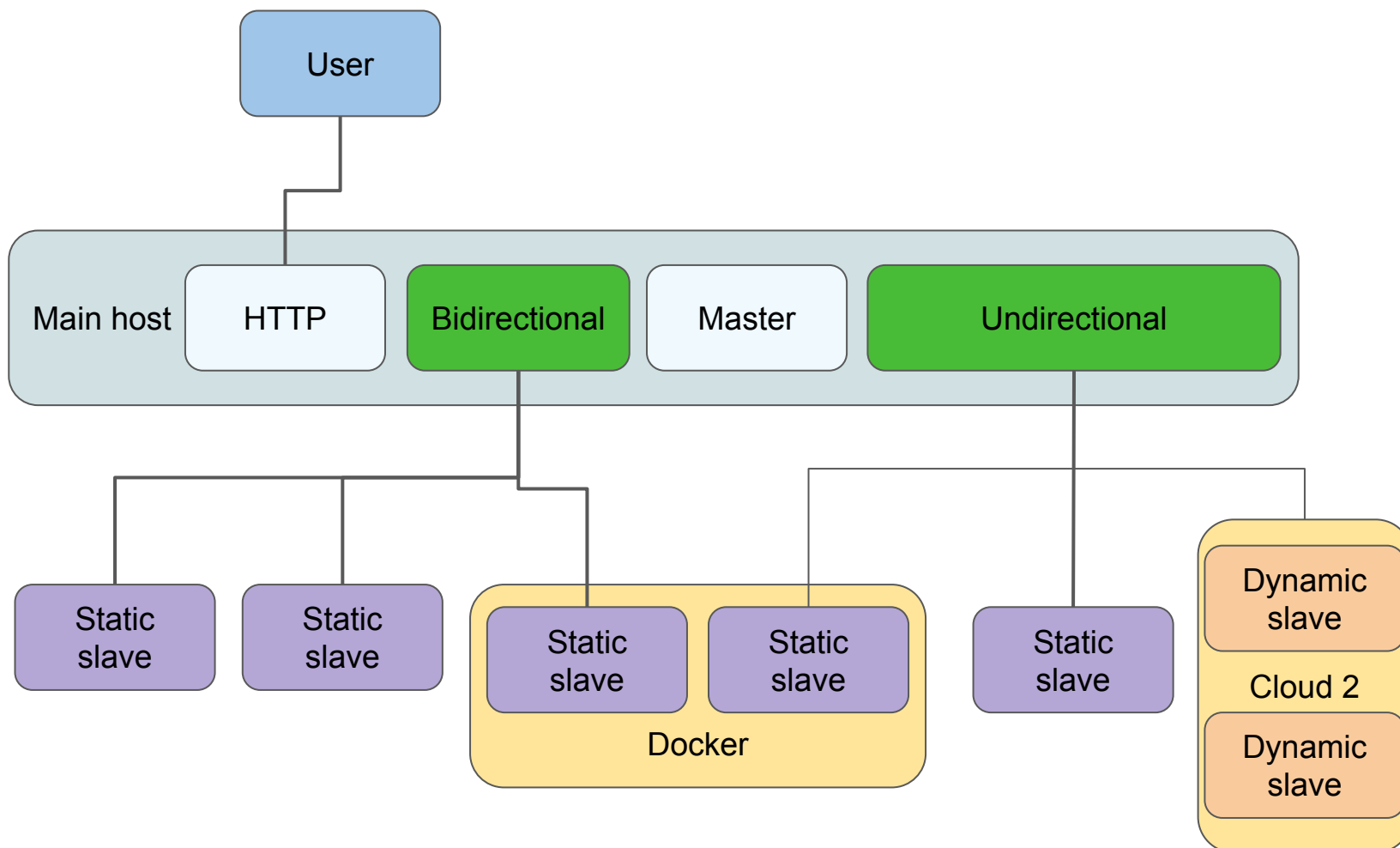


Что такое TeamCity?

Некоторые основные возможности системы:

- Создание конвейера **автоматизации**
- Создание и **управление** пользователями
- Синхронизация с **AD**
- Ограничение прав пользователей на разных вложенностях сущностей
- Установка **плагинов**
- Настройка **Custom Tool**
- Использование динамических и статических окружений

Архитектура TeamCity





Сущности TeamCity



Project

Project - сущность, в которой хранятся
subproject, configuration и\или **VCS roots**



Build Configuration

Build configuration - автоматизация, описание рутины, которую необходимо сделать.

Build Configuration Template

Build Configuration Template - шаблонизация **Build Configuration**.

Шаблон может быть использован в том виде, в котором **создан**, а может быть **переопределён**.

При этом:

- Остальные наследники **родительского** проекта будут использовать первоначальную версию
- Потомки **текущего** проекта - перезаписанную

Kotlin DSL

Kotlin DSL - способ описания **Build Configuration** при помощи языка программирования.

Конфигурация может быть:

- Выгружена в Kotlin DSL
- Сохранена в репозитории

Administration

Основные возможности администрирования:

- Проверка используемого **места** на диске мастера
- Просмотр **истории изменений** конфигураций
- Создание **пользователей** и **групп** для них
- Подключение дополнительных **модулей аутентификации**
- Управление версией **TeamCity**
- Создание вторичных **нод** для мастера
- Получение **диагностической** информации
- Создание **backup**
- Управление **plugin**



Дополнения



Experimental UI

Experimental UI - попытка разработчиков Teamcity сделать интерфейс более удобным.

Добавили новые графические элементы, улучшили просмотр статистики и другое.

Ansible Runner

Ansible Runner- добавляет возможность запускать ansible через синтаксис Kotlin DSL.

Фактически избавляет от необходимости вызова оболочки для вызова плейбука.

Пример синтаксиса можно посмотреть на [официальной странице плагина](#)



Итоги

Итоги

- **TeamCity** - тесно интегрированный с экосистемой **JetBrains** оркестровщик **CI\CD**
- Позволяет создавать **шаблоны конфигурации** для групп проектов
- Имеет возможность **версионирования** конфигураций
- При необходимости, шаблоны можно **переписывать** для конкретных проектов
- Инструмент обладает **гибкостью** в настройке
- Бесплатная версия имеет **ограничение** по количеству агентов и конфигураций

Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задавайте **в чате** мессенджера Slack.
- Задачи можно сдавать **по частям**.
- Зачёт по домашней работе проставляется после того, как **приняты все задачи**.

**Задавайте вопросы и
пишите отзыв о лекции!**

Алексей Метляков