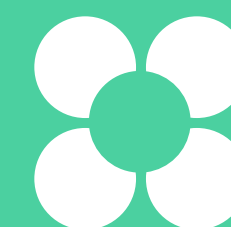


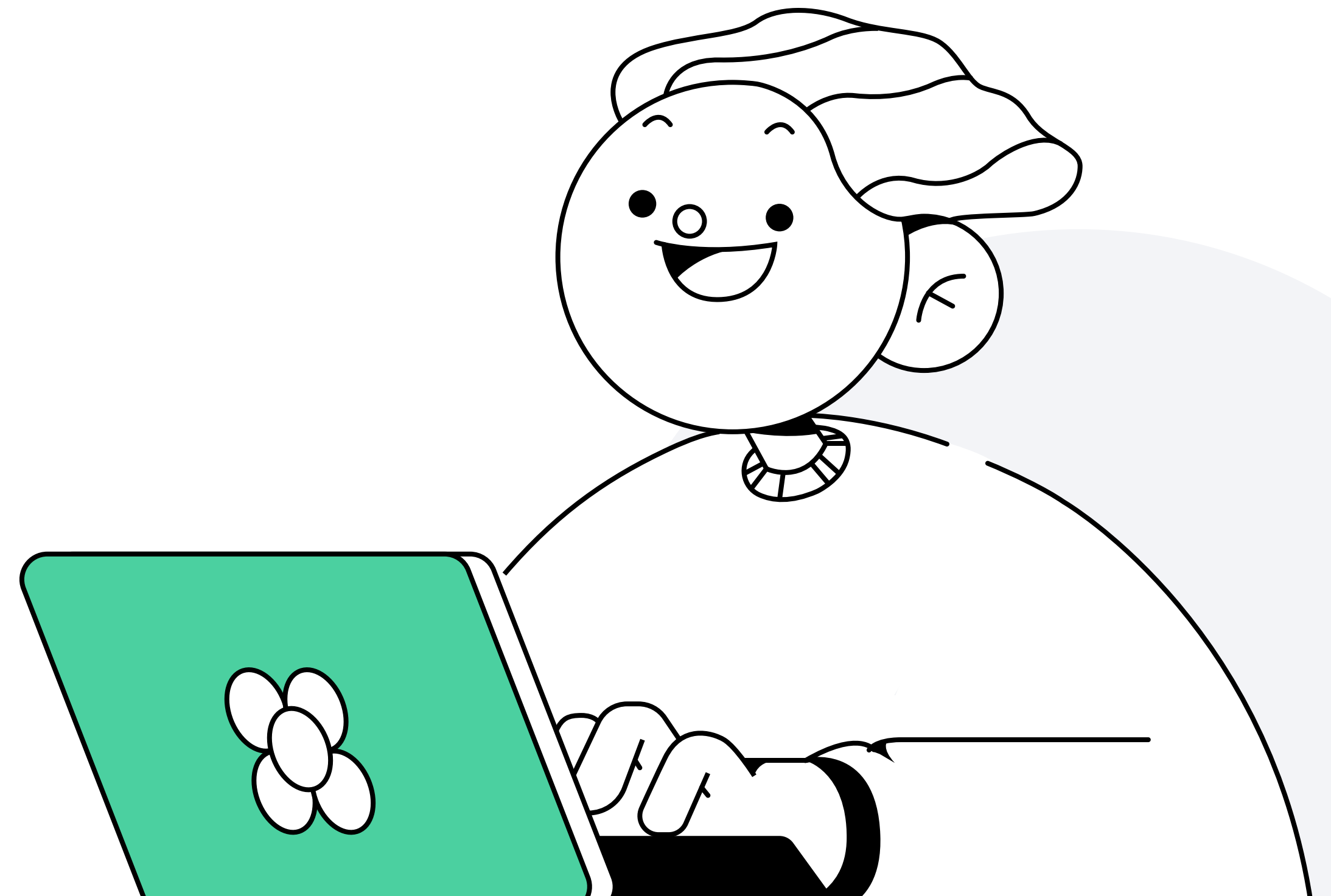
# TeamCity

Алексей Метляков  
Tech lead DevOps engineer в OpenWay



# План занятия

- 1 TeamCity
- 2 Сущности TeamCity
- 3 Настройки и дополнения



# TeamCity

Алексей Метляков

Tech lead DevOps engineer в OpenWay

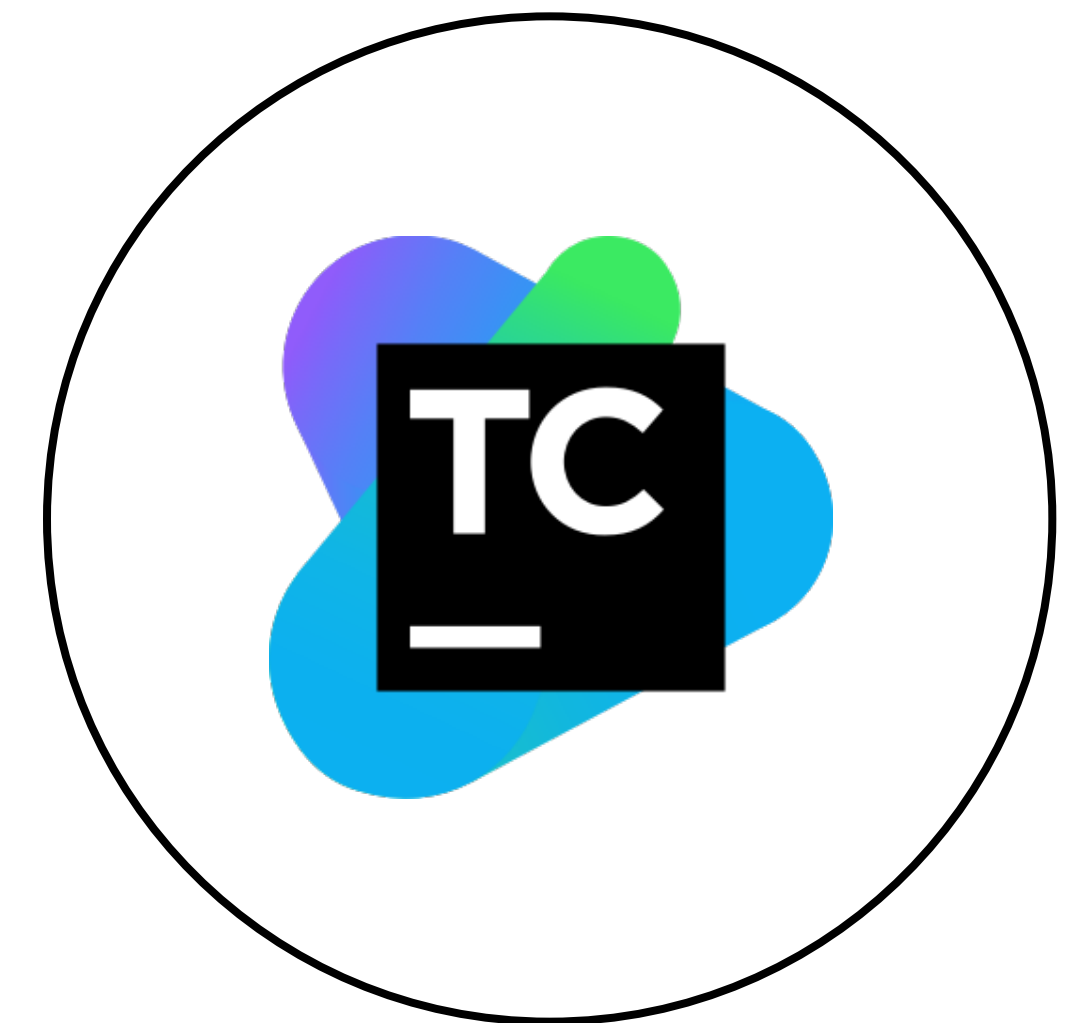
# Что такое TeamCity

**TeamCity** – ППО для управления **CI\CD**-процессами

✓ Существует в следующих исполнениях:

- **Professional** – бесплатная версия с запуском до 3 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**

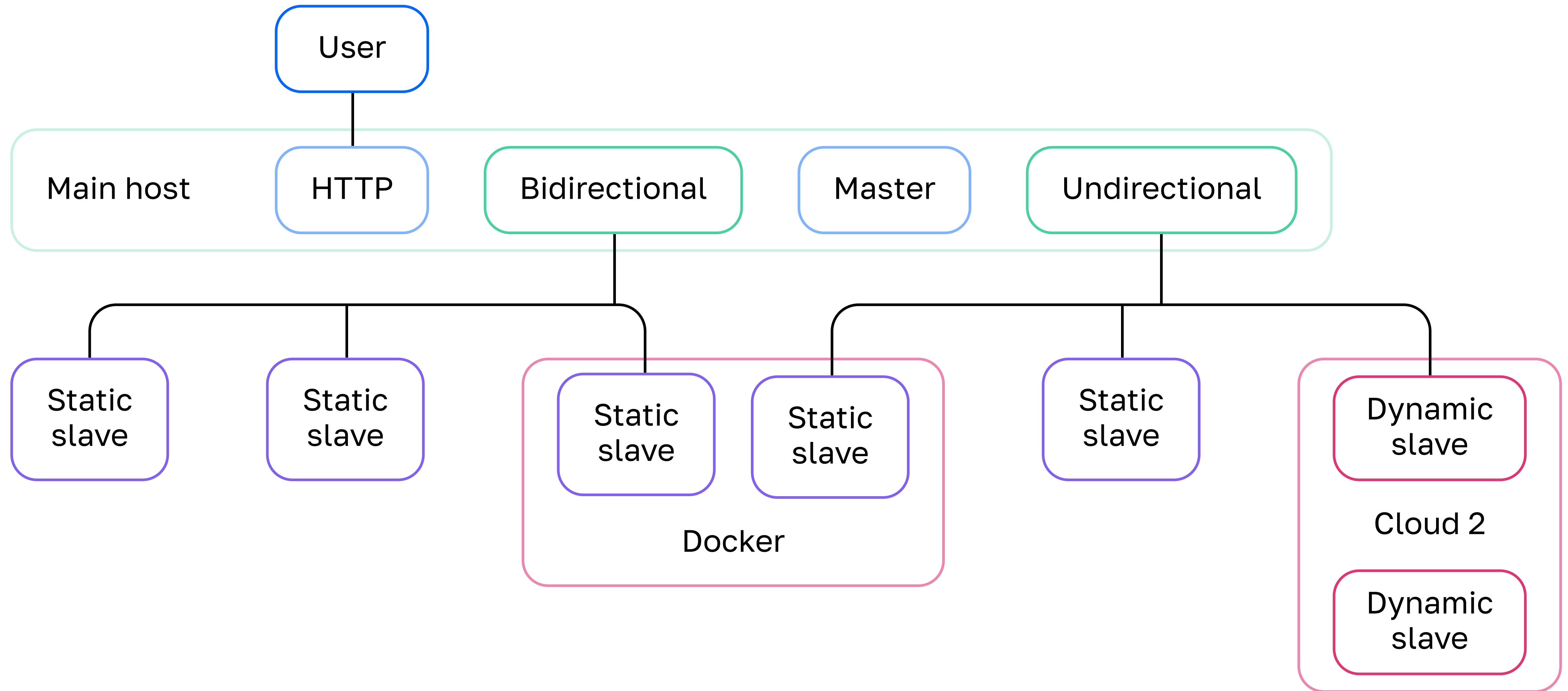


# Что такое TeamCity

Некоторые основные возможности системы:

- создавать конвейер **автоматизации**
- создавать пользователей и **управлять** ими
- синхронизироваться с **AD**
- ограничивать права пользователей на разных вложенностях сущностей
- устанавливать **плагины**
- настраивать **Custom Tool**
- использовать динамические и статические окружения

# Архитектура TeamCity



# Сущности TeamCity

Алексей Метляков

Tech lead DevOps engineer в OpenWay

# 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**

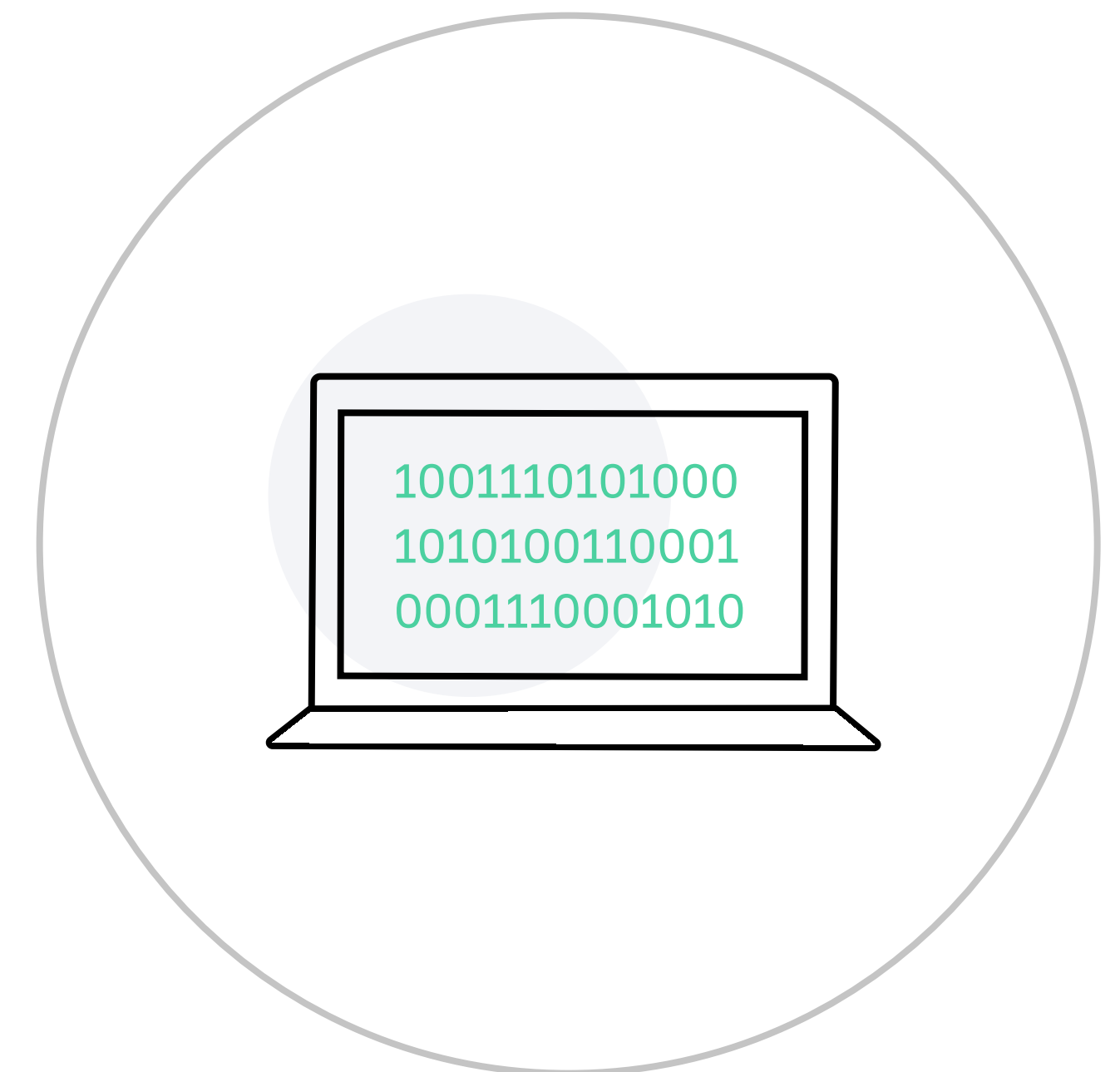


# Build configuration template

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



# Настройки и дополнения

Алексей Метляков

Tech lead DevOps engineer в OpenWay

# Administration

Основные возможности администрирования:

- проверять используемое **место** на диске мастера
- просматривать **истории изменений** конфигураций
- создавать **пользователей** и **группы** для них
- подключать дополнительные **модули аутентификации**
- управлять версией **TeamCity**
- создавать вторичные **ноды** для мастера
- получать **диагностическую** информацию
- создавать **backup**
- управлять **plugin**

# Experimental UI

**Experimental UI** – попытка разработчиков TeamCity сделать интерфейс более удобным, добавив новые графические элементы, улучшив просмотр статистики и многое другое



# Ansible Runner

**Ansible Runner** — добавляет возможность запускать ansible через синтаксис Kotlin DSL.

Фактически избавляет от необходимости вызывать оболочку для вызова плейбука.

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

