# Назначение CMakeTool

Программа CMakeTool предназначена для удобного конфигурирования проектов, использующих фреймворк DAVA.

В программе учитываются такие особенности конфигурирования, как специфический cmake под Windows 10, использование UnityBuild и др.

Все особенности прописаны в соответствующем [конфигурационном файле](#_Конфигурация_CMakeTool).

# Общие сведенияC:\Users\k_lyashkevich\Pictures\cmakeTool.jpg

Рисунок 1: элементы управления CMakeTool

1. При запуске утилиты пользователю требуется в секции 2 выбрать платформу и опции сборки;
2. после этого в секции 4 будет написано, какие поля необходимо заполнить, чтобы можно было сконфигурировать проект;
3. когда все необходимые поля в секции 1 заполнены – в секции 4 появляется командная строка, которая будет использована для вызова cmake;
4. по нажатию кнопки run cmake (7) будет запущена сгенерированная командная строка, вывод от которой пойдет в текстовое поле (5).

Более подробно все опции описаны в разделе «[Элементы управления](#_Элементы_управления)».

Программа формирует список платформ и опций из [конфигурационного файла](#_Конфигурация_CMakeTool). Там же прописано, какие параметры командной строки при этом будут использованы.

# Конфигурация CMakeTool

Конфигурация программы описана в формате JSON и лежит в папке с тулзой (под MAC OS X в папке Resources/Data).

По скольку под MAC OS X и Windows разные параметры сборки и разные комплекты – используются разные конфигурационные файлы, config\_windows.json и config\_mac.json соответственно.

Конфигурационный файл описывает список платформ и глобальных опций.

### Платформы

Раздел платформ имеет ключ «platforms» и реализует массив с объектами, соответствующими каждой отдельной платформе.

Свойства платформы:

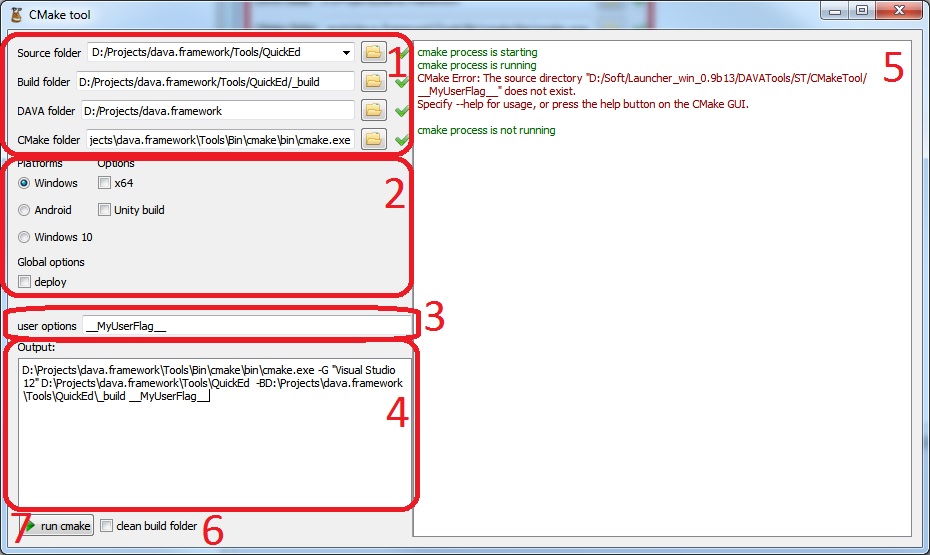
* *name -* имя платформы, отображаемое в тулзе;
* *value* – текст, который будет вставлен в командную строку (поле 4 на рисунке 1);
  + значение *value* форматируется через аргументы, которые идут в порядке %1, %2, %3 и т.д. Значения аргумента берутся из полей *defaults* и *options* (ниже)
  + так же в формировании командной строки из value участвуют глобальные переменные:
    - *SOURCE\_PATH –* Sourc Folder из секции 1
    - *DAVA\_FRAMEWORK\_PATH –* DAVA folder из секции 1
    - *CMAKE\_PATH* – Cmake folder из секции 1
* *defaults* – массив строчек – значения аргументов из *value* по умолчанию; расположены в том же порядке, что и аргументы *value*;
* *options* – массив объектов – значения аргументов из value, которые будут отображены в графическом виде; Формат объекта из options:
  + *name* – отображаемое в тулзе имя, например “x64” на рисунке 1;
  + *type* – тип виджета. Поддерживаемые значения «checkbox» и «radiobutton»;
  + *substring number* – какому по номеру аргументу относится данная опция;. К примеру для аргумента %2 substring number будет равен 2;
  + *value* – строка, которая будет подставляться в аргумент, когда соответствующий виджет отмечен (is checked);

### Глобальные опции

Раздел глобальных опций имеет ключ «global options» и реализует массив объектов с ключами «name» и « value». Все объекты отображаются в виде флажков (checkboxes), name – отображаемое в программе имя флажка (например deploy на рисунке 1), а value – текст, который будет добавлен к команде в поле 4 если флажок отмечен.

# Элементы управления

Элементы управления разбиты по секциям и представлены на рисунке:



1. Поля ввода, нужные для создания командной строки (в секции 4):
   1. Source folder – путь к папке, для которой требуется сгенерировать проект;
      1. при запуске cmake нажатием кнопки *run cmake* (7) конфигурация запоминается и заносится в всплывающий список;
      2. соответствует переменной *SOURCE\_PATH* в конфигурационном файле;
   2. Build folder – путь к папке, где будет создан проект после конфигурации cmake;
      1. если такой папки нет – она создается по нажатию кнопки *run cmake* (7);
      2. это поле ввода не участвует в конфигурационном файле
   3. DAVA folder – путь к папке, где находится DAVA framework;
      1. нужен не для всех платформ – зависит от конфигурационного файла;
      2. соответствует переменной *DAVA\_FRAMEWORK\_PATH* в конфигурационном файле;
   4. CMake folder – путь к исполняемому файлу CMake;
      1. под MAC находится внутри app файла, при выборе app в файловом диалоге автоматически достраивает путь;
      2. соответствует переменной *CMAKE\_PATH* в конфигурационном файле;
2. контент, определяемым конфигурационным файлом: для какой платформы генерировать проект и с какими опциями; Все пункты описаны в разделе «[Конфигурация](#_Конфигурация_CMakeTool)»;
3. дополнительные опции конфигурации: сюда можно написать все, что угодно. По нажатию *run cmake* (7) это значение будет запомнено для выбранной sourceFolder;
4. текст командной строки; формируется из выбранной платформы, опций, глобальных опций и дополнительных опций (пункты 2 и 3); По нажатию *run cmake* (7) оправляется операционной системе;
5. Вывод от процесса cmake; Зеленым текстом отмечается статус процесса, черным – вывод (std output), красным – поток ошибок (std error);
   1. так же сюда пишется дополнительная информация: получилось или нет очистить папку с проектом, если выделен п. 6 – *clean build folder*, а так же получилось ли создать папку *build folder* из секции 1, если этой папки нет в файловой системе;
6. флажок *clean build folder* – перед запуском командной строки папка *Build folder* из секции 1 будет очищена;
7. кнопка run cmake – создает задачу очистить *Build folder* (1), если был отмечен флажок *clean build folder* (6) и запустить командную строку;