

Инструменты разработки приложений и взаимодействие с устройствами

Марк Заславский, mark.zaslavskiy@fruct.org

План

- Какие есть инструменты?
- Как настроить среду?
- Как начать разработку?
- Как отлаживать локально/удаленно/на устройстве?
- Как сделать свое первое приложение на **QML/C++** для SailFishOS?



SAILFISH OS

Основные языки разработки

- **QML(+JS)** - декларативный язык для создания пользовательских интерфейсов и описания бизнес-логики.
- Qt/QML/C++
- Python

Python

- Есть возможность вести разработку, подробнее по ссылке <https://sailfishos.org/develop/tutorials/creating-application-in-python/>
- Идея: добавляем специально обернутый python-код в QML приложение

Использование JS-наработок - описание

- QML совместим с JavaScript.
- Возможны следующие варианты:
 - Встроить код напрямую в QML
 - Подключить внешний *.js файл

import "my_cool.js" as super_backend

- <http://doc.qt.io/qt-5/qtqml-javascript-topic.html>

Использование JS-наработок - инструменты

- Импорт .js/.qml внутри других .js

```
Qt.include("utils.js")  
.import "view.qml" as view
```

- Property для .js

```
// counter.js  
property var counter = 0;  
function incCounter(){ ...}
```

- .js библиотеки. Загружаются один раз и работают по аналогии с разделяемыми библиотеками. Обязаны не менять QML-состояния

```
// util_libs.js  
.pragma library
```

Портирование существующих приложений на JS

- Среда выполнения (доступные API и глобальные переменные) отличаются
- Нет доступа к DOM API
- Нет доступа к CSS
- <http://doc.qt.io/qt-5/qtqml-javascript-hostenvironment.html>
- <http://doc.qt.io/qt-5/qml-qtquick-canvas.html#tips-for-porting-existing-html5-canvas-applications>

Инструменты разработчика

1. Образ с SDK:
 - a. Lubuntu 64 с установленным sdk
 - b. Находиться на выданных участникам флеш-накопителях
2. Sailfish OS SDK
 - a. QtCreator + Sailfish build tools
 - b. Ссылка для скачивания https://sailfishos.org/wiki/Application_SDK_Installation

Подробнее про SDK

- a. Sailfish OS SDK (QtCreator+)
 - i. Mer Build Engine - отдельная виртуальная машина (qemu) для сборки.
 - ii. Виртуальная машина (VirtualBox) с образом Sailfish OS.
 - iii. Libsailfishapp для OS-specific функций.
 - iv. Готовые приложения доставляются на устройство в виде **RPM**.

Комплекты

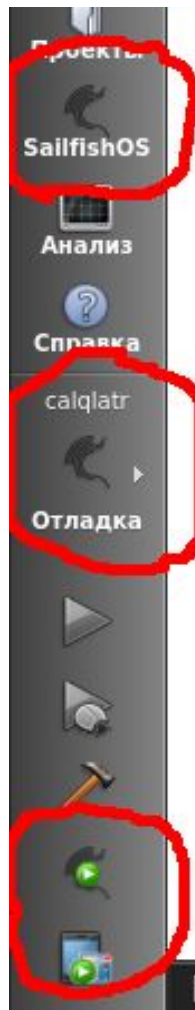
Комплект - набор инструментов отладки и компиляции.

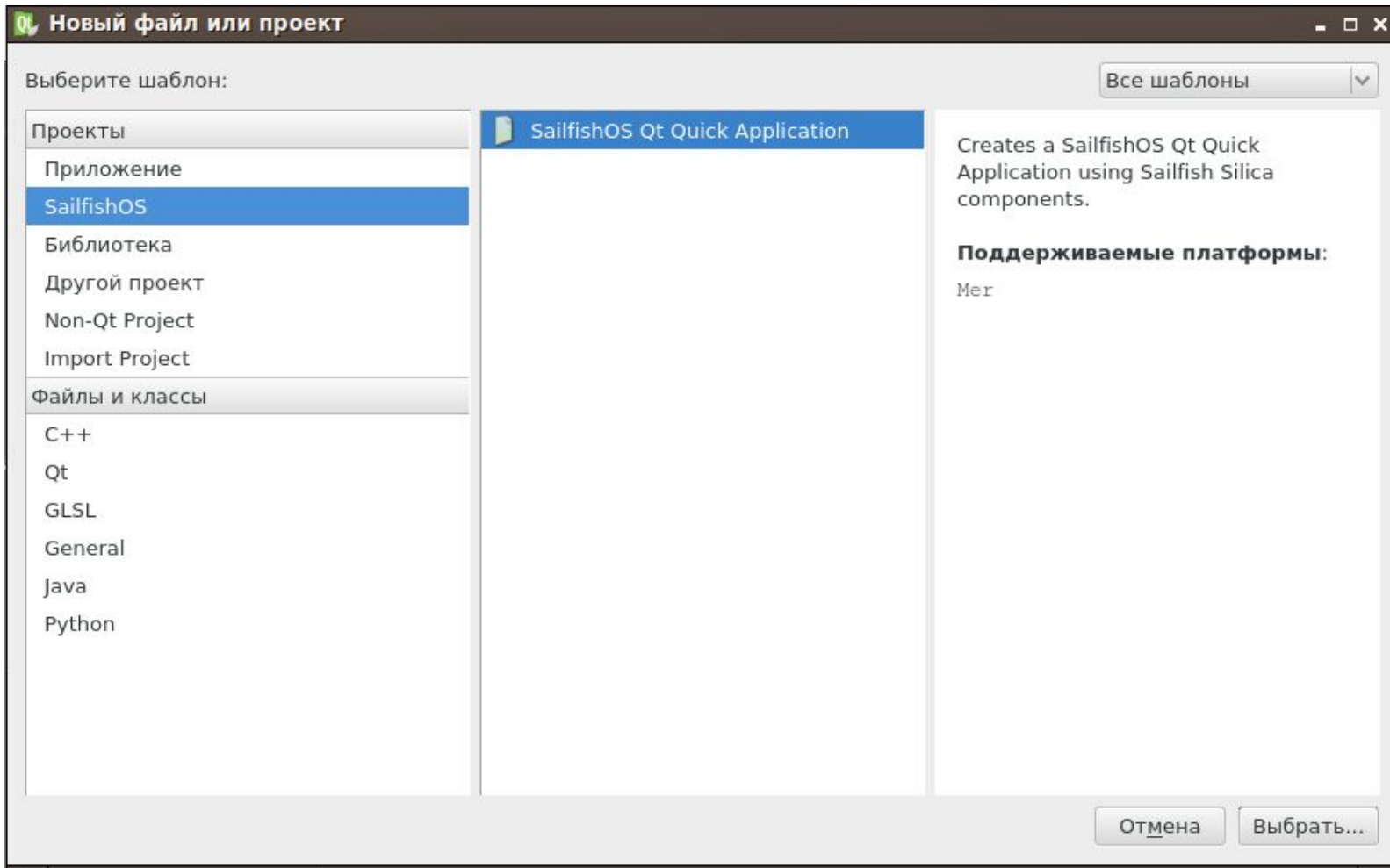
Доступны по умолчанию:

- ARM - “железное” устройство
- I386 - эмулятор, входящий в состав SDK

Демо - создаем проект и смотрим, что внутри

- Запускаем SDK
- Отличия от QtCreator
- Смотрим где живут примеры и учебники
- Создаем проект





Файл helloworld.cpp

```
#ifdef QT_QML_DEBUG  
#include <QtQuick>  
#endif
```

```
#include <sailfishapp.h>
```

```
int main(int argc, char *argv[])  
{  
    return SailfishApp::main(argc, argv);  
}
```

Файлы в составе проекта 1

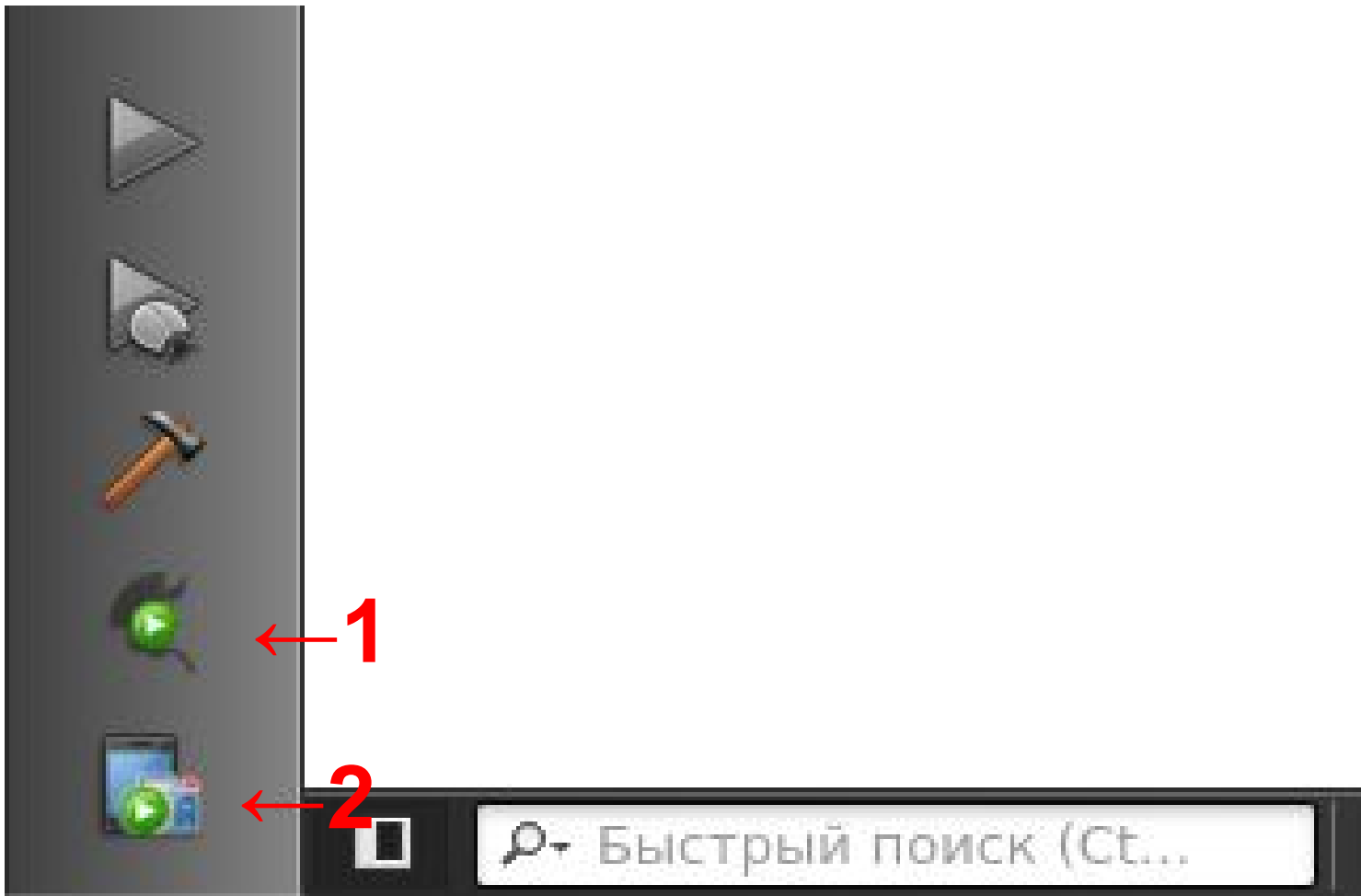
- Pro - основной файл проекта для Qt, настройки и указание на файлы исходного кода
- Rpm
 - Changes.in - история изменений.
 - Spec - генерируемый (!) файл с настройками для сборки rpm.
 - Yaml - исходник для генерации Spec. Содержит информацию о зависимостях сборки и выполнения.

Файлы в составе проекта 2

- Translations - файлы локализации
- Desktop - указание на миниатюру изображения, команду для запуска, имя в системе
- qml/cover qml/pages qml/*qml

Демо - запускаем проект в эмуляторе 1

1. Запускаем Mer
2. Удостоверяемся, что Mer успешно заработал (иконка изменила значок)
3. Заходим на страницу Sailfish
 - a. Страница на самом деле WebView, в котором страницы веб-сервера, развернутого на Mer. Поэтому может работать медленно.
 - b. Targets - управление комплектами
 - c. Updates - обновление SDK
 - d. Harbour - настройки публикации и упаковки
4. Установка зависимостей в Target/Manage



MerSDK

Targets

Updates

Harbour Tools

Manage Targets

Each "Target" lets you build apps for specific hardware running a specific OS version.

SailfishOS-armv7hl

manage

delete

SailfishOS-i486

manage

delete

Add New Target

Pre-configured targets

select a target

Local target name (optional)

add

Toolchain

select a toolchain

target name

target url

add


Sailfish
SDK Control Center

Демо - запускаем проект в эмуляторе 2

1. Запускаем эмулятор
2. Способы сборки/запуска
 - a. Deploy as RPM package
 - b. Deploy By Building An RPM package - сборка RPM **без** деплоя
 - c. Deploy By Copying Binaries

Устройство:

SailfishOS Emulator (по умолчанию для Mer Device) ▾

Проект: **calqlatr**

Запуск: **calqlatr (on Mer Device)**

Комплект	Сборка	Установка
MerSDK-SailfishOS-armv7hl	Выпуск	Deploy As RPM Package
MerSDK-SailfishOS-i486	Отладка	Deploy By Building An RPM Package Deploy By Copying Binaries

Демо - компилируем и запускаем на устройстве

1. Подключение устройства
 - a. Инструменты-Параметры-Устройства
 - b. Add: Mer device, Physical device
 - c. Открываем настройки “Режима разработчика” на устройстве
 - d. Копируем настройки в окно настройки нового устройства
 - e. Нажимаем Test connection
2. Запуск
 - a. Выбираем arm комплект
 - b. Выбираем устройство в верхнем select
 - c. Выбираем тип запуска

Устройства

Устройства

Устройство: Jolla (ARM)

Общее

Название: Jolla (ARM)

Тип: Mer Device

Автоопределённое: Нет

Текущее состояние: Unknown

Специальное

Host name: 192.168.43.163

Username: nemo

SSH port: 22

Timeout: 10s

Free ports: 10000-10100

Authentication: SSH Key

Private SSH Key: /home/vood/SailfishOS/vmshare/ssh/private_keys/Jolla_(ARM)/
nemo

Демо - отладка приложения в эмуляторе

- Сознательно допускаем ошибку
- Пишем в console.log
- <http://doc.qt.io/qt-5/qtquick-debugging.html>

Демонстрация приложения на хакатоне

- Проблема: без специальных средств показать на устройстве сложно и не удобно.
- Решение: <https://together.jolla.com/question/43626/implementednot-packaged-manage-jolla-from-desktop/>
 - Работает только через usb-подключение
 - Достоверное известно, что из клиентов поддерживает как минимум **krdc**

ССЫЛКИ

- a. <https://sailfishos.org/develop/sdk-overview/develop-installation-article/>
- b. <https://sailfishos.org/develop/sdk-overview/develop-firstapp-article/>
- c. <https://sailfishos.org/develop/docs/>

Спасибо за внимание!

Вопросы?

