

ОБУЧЕНИЕ ШКОЛЬНИКОВ КРОССПЛАТФОРМЕННОЙ РАЗРАБОТКЕ НА ОСНОВЕ QT И QT QUICK

Кирилл Чувилин

Начальник отдела развития и поддержки разработчиков

Заместитель директора по стратегическому развитию



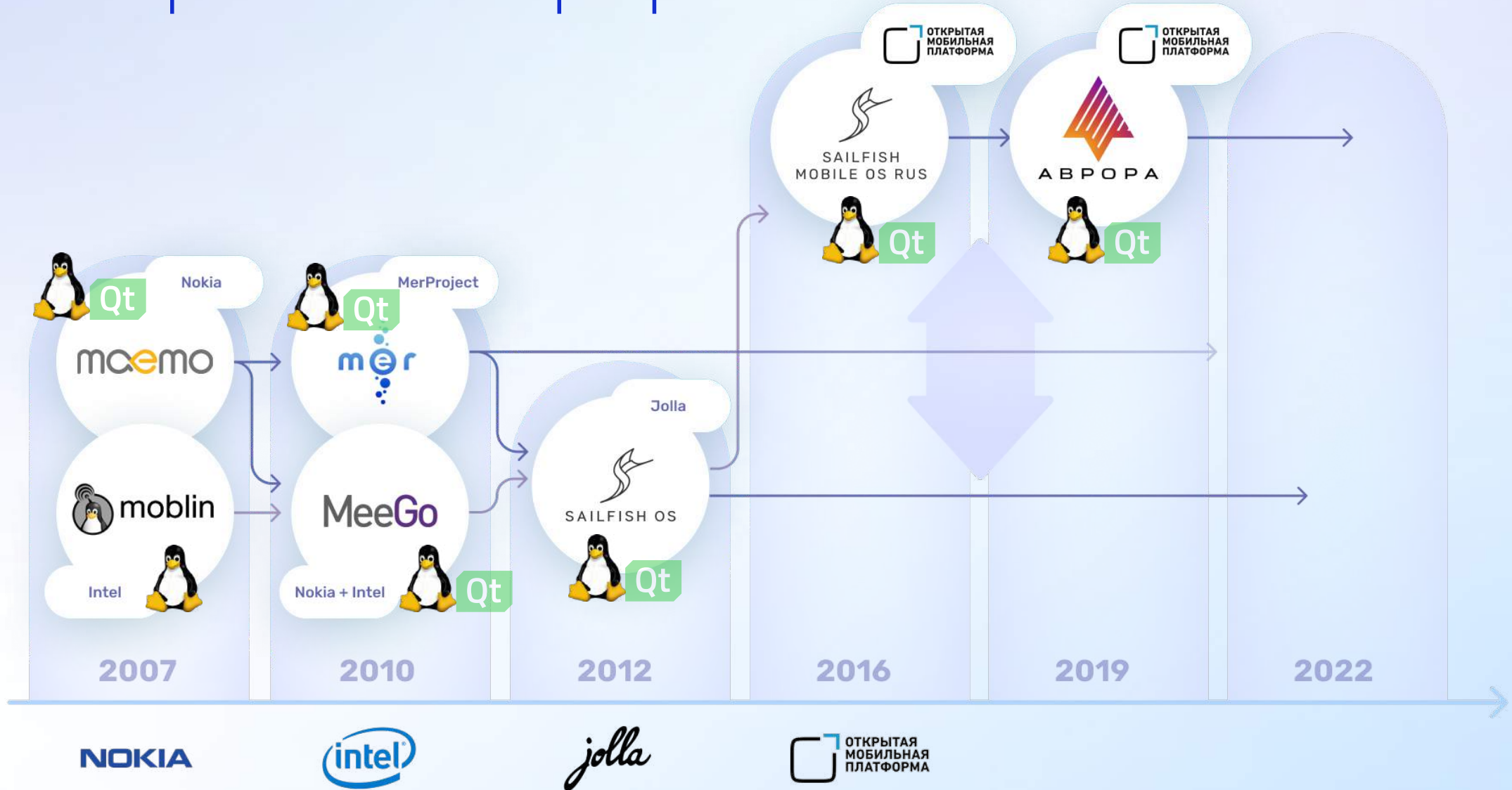
Агенда

- 1 Предпосылки
- 2 Технологии Qt и Qt Quick
- 3 Темы для школьников
- 4 Если возможностей Qt Quick не хватило

Предпосылки

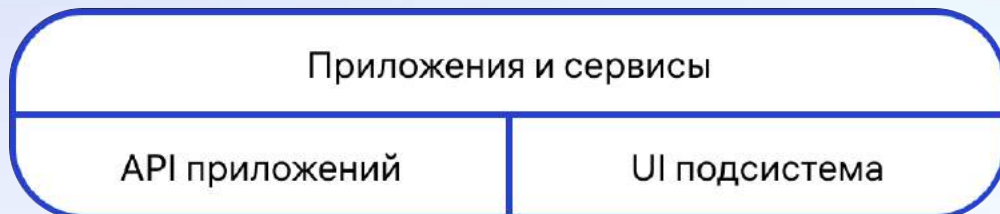


История ОС Аврора

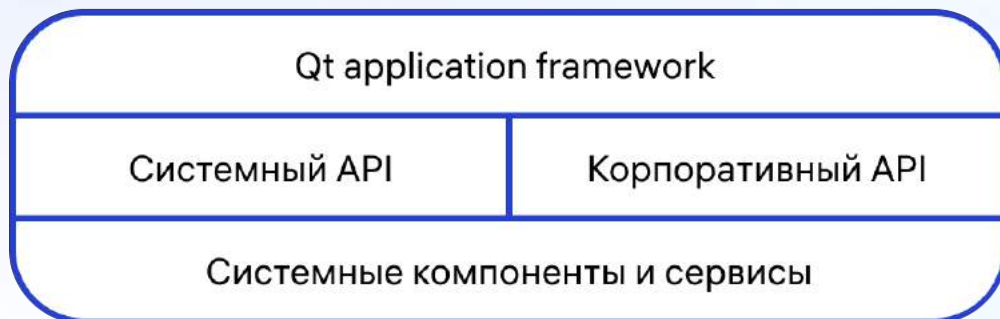


Архитектура ОС Аврора

App/UI



Middleware



Kernel




- Ядро Linux
<https://kernel.org/>
- POSIX
<https://www.iso.org/standard/73314.html>
- Фреймворк Qt
<https://www.qt.io/>
- 1600+ пакетов open-source и закрытых

02-49-59 8
ОБУЧЕНИЕ РАЗРАБОТКЕ НА QT ДЛЯ МОБИЛЬНЫХ УСТРОЙСТВ

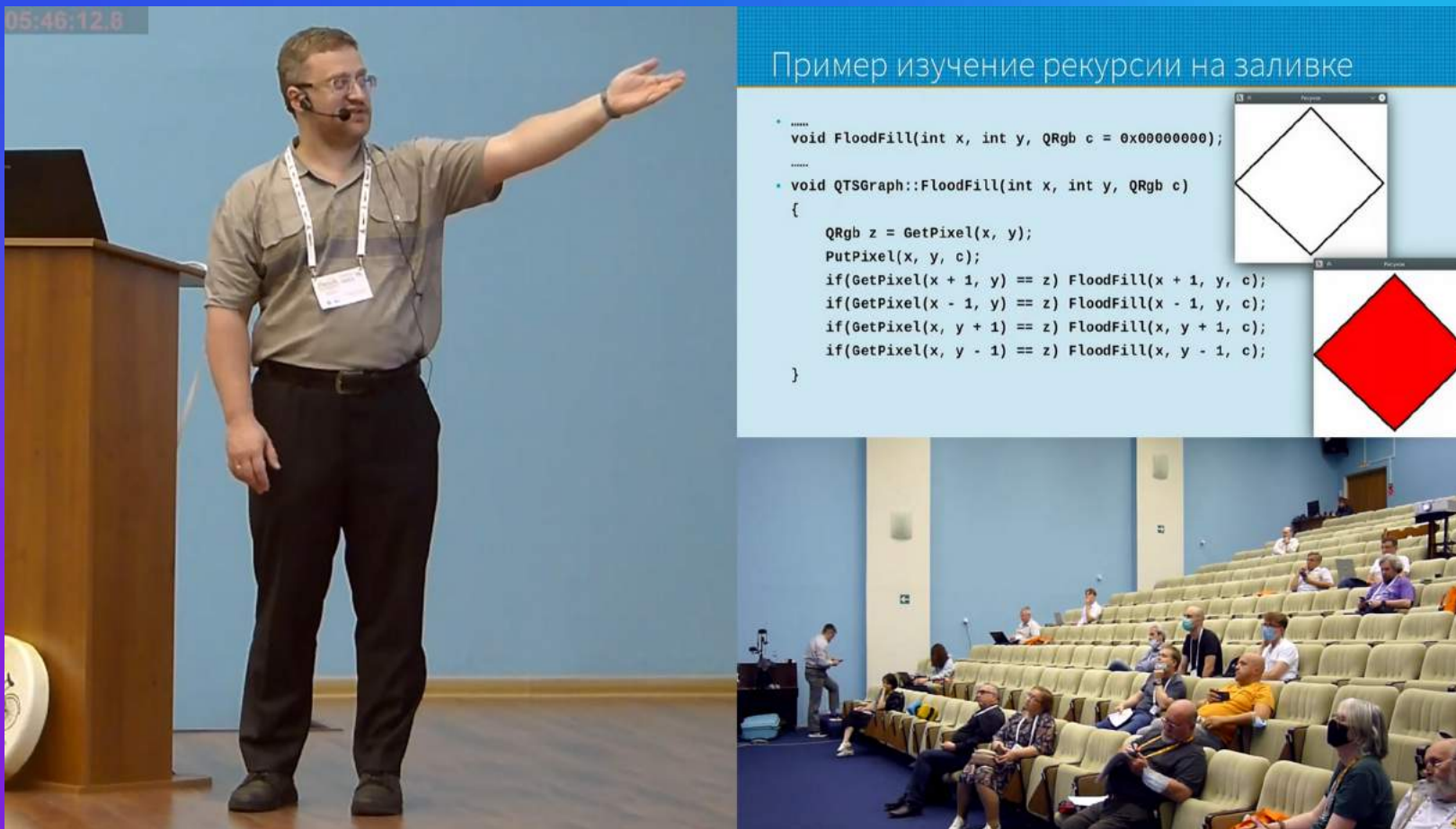
РАБОТА С ВУЗАМИ

1. ПетрГУ, 2016-2017
 - » Составлен семестровый курс разработки для Sailfish Mobile OS RUS
2. Университет Иннополис, весна 2017
 - » Проведены недельных элективных модуля разработки для Sailfish Mobile OS RUS
3. Запись первой онлайн-версии Введение в разработку приложений на Sailfish OS
4. ОмГУ, СибАДИ, весна 2017
 - » Введены семестровые курсы разработки для Sailfish Mobile OS RUS
5. Университет Иннополис и МФТИ, 2018-2019
 - » Семестровый курс по мобильной разработке на Qt (OS Аврора + Android)
6. Тиражирование курса
 - » ННГУ, ЯрГУ, СХФУ, КФУ...
 - » Мы открыты для сотрудничества!



Обучение разработке на Qt для мобильных устройств.
Что нужно студентам и профессионалам
Кирилл Чувилин, OSEUDUCONF-2021





Машинная графика в школе на свободных библиотеках
Артём Проскурнёв, OSEDUCONF-2021





27-28 октября Кружковое движение НТИ
и компания «Открытая мобильная платформа»
провели обучение для методистов и педагогов

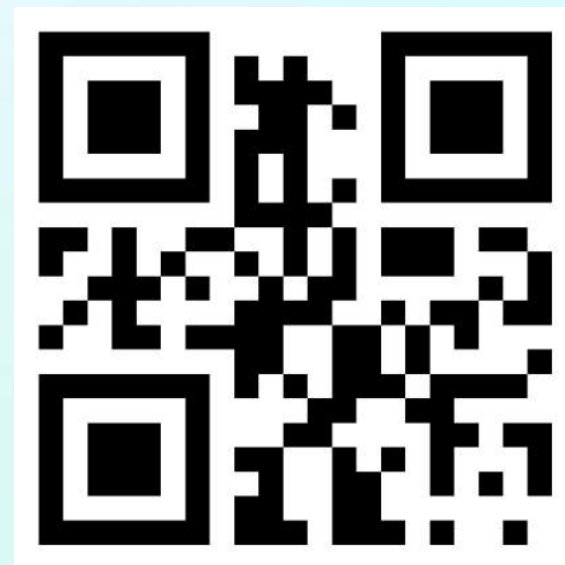


Технологии Qt и Qt Quick



Кроссплатформенный фреймворк Qt

- Среда разработки ПО для различных типов устройств
 - ПК (Linux, Windows, macOS)
 - Мобильные (Аврора и др. Linux, Android, iOS)
 - Встраиваемые
- Сообщество разработчиков более 1 500 000 человек
- Можно бесплатно использовать в коммерческом ПО



Qt API

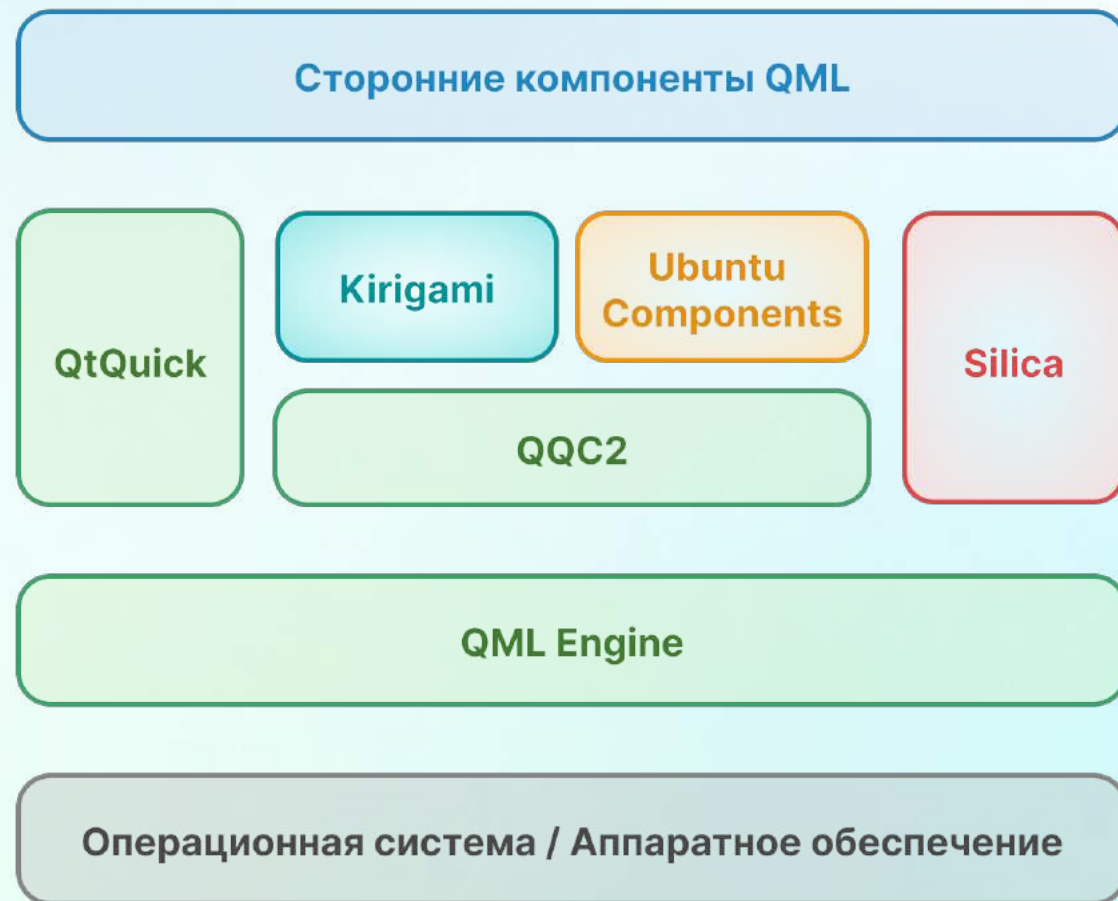
- Общее
 - Модели и представления
 - Многопоточность
 - Файловая система, БД
 - Мультимедиа
 - Сеть, WebView
 - ...
- Для мобильных
 - Элементы UI, сенсорный экран
 - Геолокация и картография
 - Датчики, NFC



Qt Quick

Технология быстрой разработки

- QML — язык разметки
 - Декларативный
 - Поддерживает вставки на «JavaScript»
 - Доступны свойства объектов C++
- QML Engine — интерпретатор QML
- QtQuick — общие компоненты QML, поставляемые Qt
`import QtQuick 2.0`



Элементы компонента QML

- **id**
Идентификатор для обращения к объекту
- **Properties**
Свойства заданных типов, обладающие названиями и значениями
- **Methods**
Исполняемый код на «JavaScript»
- **Signals**
Уведомления от объекта QML
- **Signal Handlers**
Выражения или функции, инициируемые сигналами
- **Nested objects**
Вложенные объекты



QML: объекты, свойства, привязки

```
import QtQuick 2.6
import Sailfish.Silica 1.0

Page {
    Column {
        id: column
        width: parent.width
        spacing: Theme.paddingLarge

        PageHeader { title: qsTr("Page UI") }

        Label {
            x: Theme.horizontalPageMargin
            text: qsTr("Hello world!")
            color: Theme.highlightColor
            font.pixelSize: Theme.fontSizeHuge
        }
    }
}
```



Примеры форматированного текста

```
Text {  
  y: 0; width: parent.width  
  text: "<b>Hello</b> <i>World!</i>"  
  color: "red"  
  font.pointSize: 48  
}  
Text {  
  y: 200; width: parent.width  
  text: "<b>Hello</b> <i>World!</i>"  
  color: "green"  
  font { pointSize: 48; underline: true }  
  textFormat: Text.RichText  
}  
Text {  
  y: 400; width: parent.width  
  text: "<b>Hello</b> <i>World!</i>"  
  color: "blue"  
  font { pointSize: 32; bold: true }  
  textFormat: Text.PlainText  
}
```

Hello World!

Hello World!

Hello *World!*

Пример image

```
Image {  
    width: parent.width; height: parent.height  
    source: "avrova.svg"  
}
```

- **fillMode** : **enumeration**
 - **Image.Stretch** — масштабировать по элементу
 - **Image.PreserveAspectRatio** — масштабировать без обрезки
 - **Image.PreserveAspectCrop** — масштабировать с обрезкой
 - **Image.Tile** — дублировать горизонтально и вертикально
 - **Image.TileHorizontally** — растянуть вертикально, дублировать
 - **Image.TileVertically** — растянуть горизонтально, дублировать
 - **Image.Pad** — не масштабировать



Пример нажимаемой кнопки

```
Rectangle {  
    color: mouseArea.pressed ? "green" : "blue"  
  
    Text {  
        anchors.centerIn: parent  
        text: "Click Me"  
        color: "red"  
        font { bold: true; pixelSize: 48 }  
    }  
    MouseArea {  
        id: mouseArea  
        anchors.fill: parent  
        onClicked: console.log("clicked")  
    }  
}
```



Пример модели xml

```
import QtQuick.XmlListModel 2.0
Page {
    XmlListModel {
        id: xmlListModel
        source: "books.xml"
        query: "/catalog/book"
        XmlRole { name: "title"; query: "title/string()" }
        XmlRole { name: "year"; query: "year/number()" }
        XmlRole { name: "author"; query: "author/string()" }
    }
    ListView {
        anchors.fill: parent
        model: xmlListModel
        delegate: Column {
            Text { text: title + " (" + year + ")" }
            Text { text: author }
        }
    }
}
```

Qt 5 Cadaques (2014)
Juergen Bocklage-Ryannel
C++ GUI Programming with Qt 4 (2006)
Jasmin Blanchette
Programming with Qt (2002)
Matthias Kalle Dalheimer

Пример проигрывателя аудио

```
Audio {
    id: player
    source: StandardPaths.music + "/Rainforest.mp3"
}
ProgressBar {
    label: "%1 – %2".arg(player.metaData.author)
        .arg(player.metaData.title)
    value: player.position
    maximumValue: player.duration
}
Row {
    IconButton {
        icon.source: "image://theme/icon-m-play"
        onClicked: player.play()
    }
    IconButton {
        icon.source: "image://theme/icon-m-pause"
        onClicked: player.pause()
    }
}
```



Мини резюме

- Qt предлагает API для написания полноценных приложений
- Qt Quick предоставляет низкий порог вхождения
- Есть положительный опыт обучения студентов
- Есть запрос на обучение школьников

Темы для школьников



Темы для школьников по Qt Quick

- 1 Основы Qt Quick
- 2 Расширенные возможности Qt Quick
- 3 Организация UI приложений
- 4 Использование системных API

Основы Qt Quick

- 1 Знакомство с фреймворком Qt и технологией Qt Quick
 - 2 Инструменты разработки
 - 3 Структура проекта
 - 4 Знакомство с языком QML
 - 5 Основные визуальные типы
 - 6 Позиционирование элементов
 - 7 Обработка событий
 - 8 Определения новых свойств объектов
 - 9 Взаимодействие с пользователем
- Что на выходе
 - Навыки работы с инструментами
 - Синтаксис и типы данных
 - Отрисовка графики
 - Простые игры

Расширенные возможности Qt Quick

- 1 Подключение JS-библиотек
 - 2 Трансформации и состояния
 - 3 Анимации
 - 4 Работа с компонентами
 - 5 Модели
 - 6 Представления
 - 7 Многопоточность
- Что на выходе
 - Понимание моделей-представлений
 - Понимание концепции библиотек
 - Возможность использования «тяжёлой» логики
 - Возможность взаимодействия с RestAPI

Организация UI приложений

- 1 Библиотеки системных компонентов
 - 2 Навигация в приложениях
 - 3 Работа с текстом
 - 4 Элементы управления
 - 5 Меню
 - 6 Масштабирование приложений
 - 7 Стилизация приложений
 - 8 Локализация приложений
- Что на выходе
 - Богатые UI-приложения
 - Обработка ввода пользователя

Использование системных API

- 1 Работа с файлами
 - 2 Мультимедиа
 - 3 Камера
 - 4 Работа с HTTP
 - 5 Web-сокеты
 - 6 Датчики
 - 7 Геолокация
 - 8 Работа с базами данных
 - 9 Взаимодействие приложений
 - 10 Системные QML-плагины
- Что на выходе
 - Полнофункциональные приложения

Что по нагрузке?

	Лекции (ак.ч.)	Практика (ак.ч.)	Самостояте льно (ак.ч.)	Всего (ак.ч.)
Основы Qt Quick	10	20	6	36
Расширенные возможности Qt Quick	8	20	8	36
Организация UI приложений	8	20	8	36
Использование системных API	10	10	16	36
Всего				144

Если возможностей Qt Quick не хватило



Модули Qt 5

Qt Essentials		Qt Add-Ons		
Qt Core	Qt Quick Controls 2	Active Qt	Qt Network Authorization	Qt Speech
Qt GUI	Qt Quick Dialogs	Qt 3D	Qt NFC	Qt SVG
Qt Multimedia	Qt Quick Layouts	Qt Android Extras	Qt Platform Headers	Qt UI Tools
Qt Multimedia Widgets	Qt Quick Test	Qt Bluetooth	Qt Positioning	Qt WebChannel
Qt Network	Qt SQL	Qt Canvas 3D	Qt Print Support	Qt WebEngine
Qt QML	Qt Test	Qt Concurrent	Qt Purchasing	Qt WebSockets
Qt Quick	Qt Widgets	Qt D-Bus	Qt Quick Controls	Qt WebView
		Qt Gamepad	Qt Quick Extras	Qt Windows Extras
		Qt Graphical Effects	Qt Quick Widgets	Qt X11 Extras
		Qt Help	Qt SCXML	Qt XML
		Qt Image Formats	Qt Sensors	Qt XML Patterns
		Qt Location	Qt Serial Bus	Qt Wayland Compositor
		Qt Mac Extras	Qt Serial Port	



Расширение C++ классов

```
class Counter
{

public:
    Counter() { m_value = 0; }
    int value() const { return m_value; }

    void setValue(int value);

private:
    int m_value;
};
```

```
#include <QObject>

class Counter : public QObject
{
    Q_OBJECT
    Q_PROPERTY (int value READ value
                WRITE setValue NOTIFY valueChanged)

public:
    Counter() { m_value = 0; }
    int value() const { return m_value; }

public slots:
    void setValue(int value);

signals:
    void valueChanged(int newValue);

private:
    int m_value;
};
```

Регистрация класса для QML

- Регистрация типа в C++

```
#include <QtQuick>
#include "ClassName.h"
...

int main(int argc, char *argv[])
{
    qmlRegisterType<ClassName>("module.name", 1, 0, "TypeName");
    ...
}
```

- Использование зарегистрированного типа в QML

```
import module.name 1.0
...
TypeName { ... }
...
```

Объявление визуального элемента

```
#include <QQuickItem>

class ClassName : public QQuickItem
{
    Q_OBJECT
    ... // declaration of properties
public:
    explicit ClassName(QQuickItem *parent = nullptr);
    ...

    QSGNode *QQuickItem::updatePaintNode(QSGNode *oldNode,
        QQuickItem::UpdatePaintNodeData *updatePaintNodeData);
signals: ...
public slots: ...
private: ...
private slots: ...
};
```


Пример создания модели

```
#include <QAbstractListModel>

class DemoModel : public QAbstractListModel
{
    Q_OBJECT

public:
    enum DemoRoles {
        NameRole = Qt::UserRole + 1,
    };
    explicit DemoModel(QObject *parent = 0);
    virtual int rowCount(const QModelIndex&) const { return backing.size(); }
    virtual QVariant data(const QModelIndex &index, int role) const;
    QHash<int, QByteArray> roleNames() const;
    Q_INVOKABLE void activate(const int i);

private:
    QVector<QString> backing;
};
```

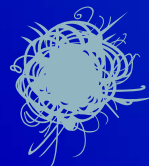
Как получить QML-модули

- Специфичные модули операционных систем
- Студенты
 - Выполняющие задания курса по Qt
 - Проходящие практику
 - Пишущие дипломы
- Сообщество
 - Open source практика
 - Части опубликованных проектов
 - Примеры для ОС Аврора
gitlab.com/omprussia/examples



Ресурсы в интернете

- Документация Qt
doc.qt.io
- Qt QML
doc.qt.io/qt-5/qtqml-index.html
- Прокси для Qt
quterussia.ru/download/
- Для связи
edu@omp.ru
- Для продолжения обсуждений
t.me/omp_edu



habr.com/ru/company/omprussia



stepik.org/users79671378



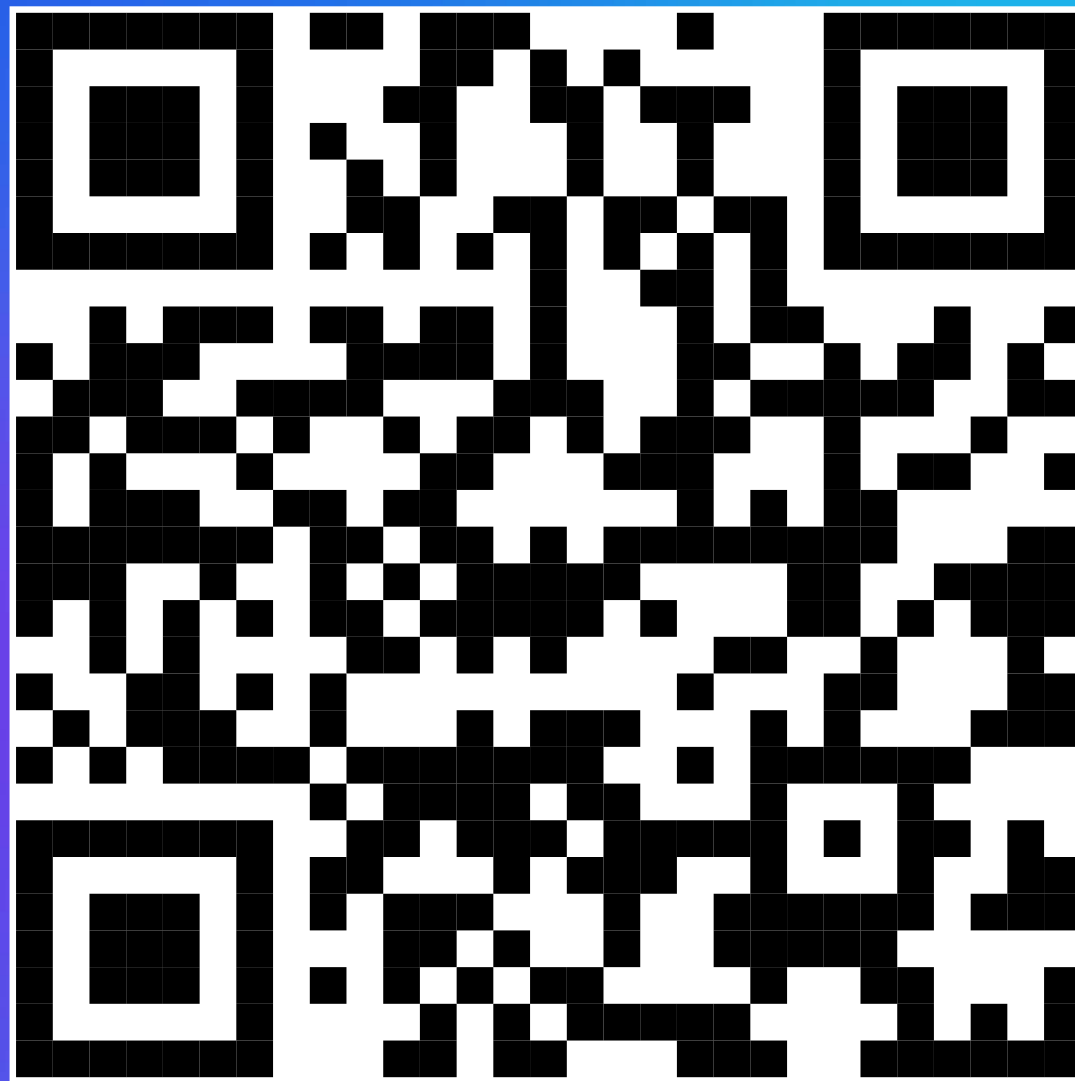
www.youtube.com/c/AuroraOSCommunity



t.me/omp_ru



vk.com/omp_ru



t.me/omp_edu

