

Porting: Qt

JAM812

Kizito Kasengulu

November 29-30, 2012

BlackBerry 10 development

Native SDK

C/C++

Cascades



WebWorks SDK

HTML5

Android SDK

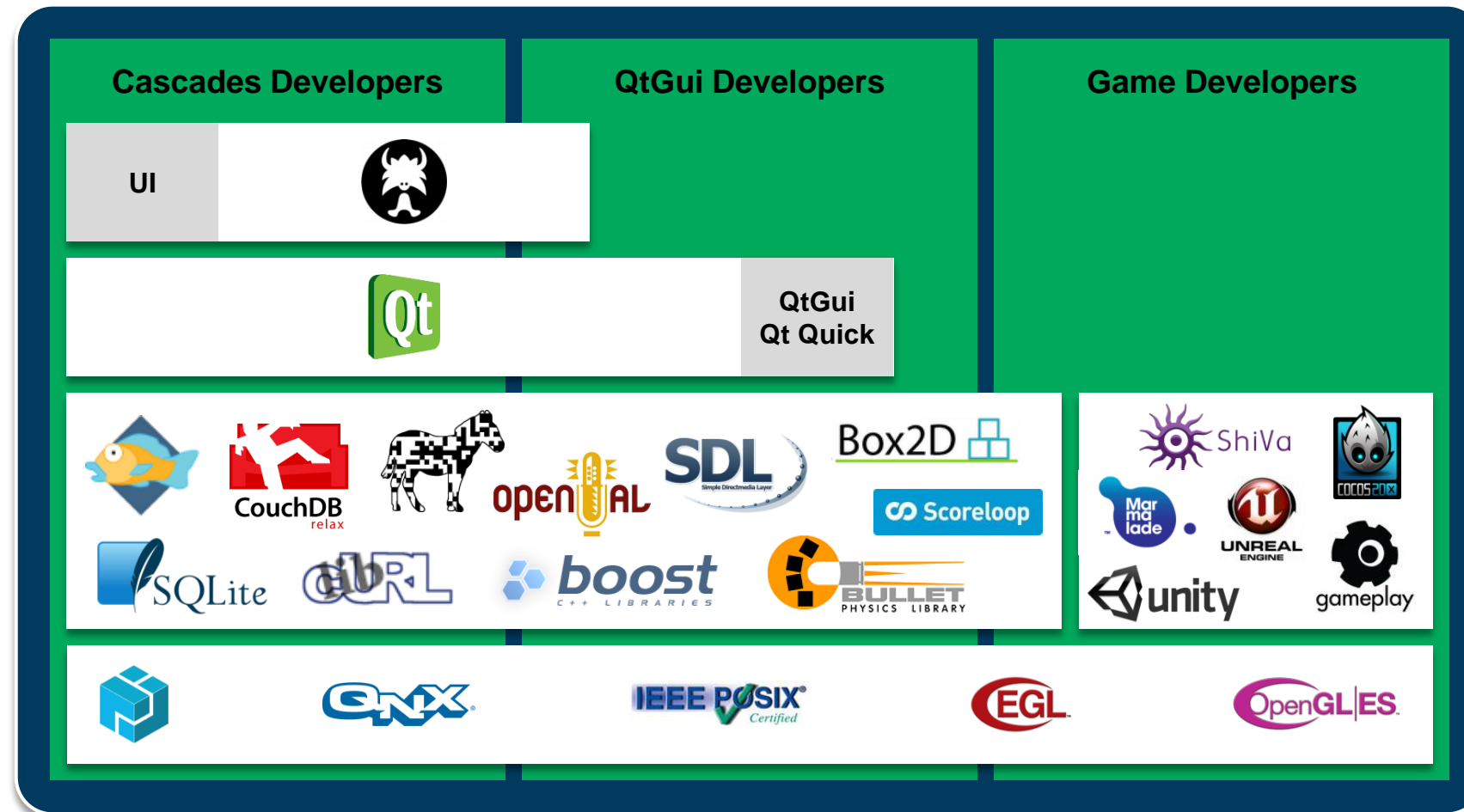
Java

AIR SDK

AS3

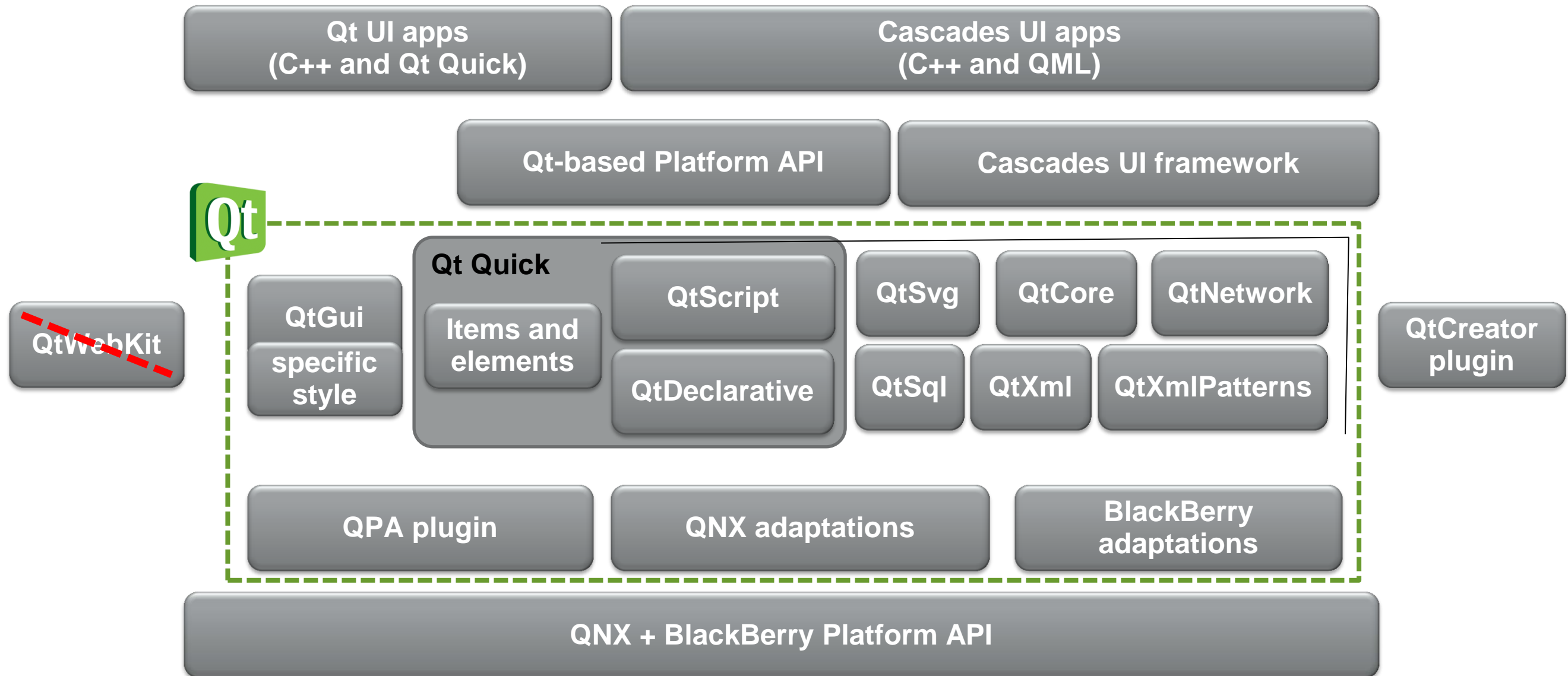
Flex

BlackBerry 10 NDK

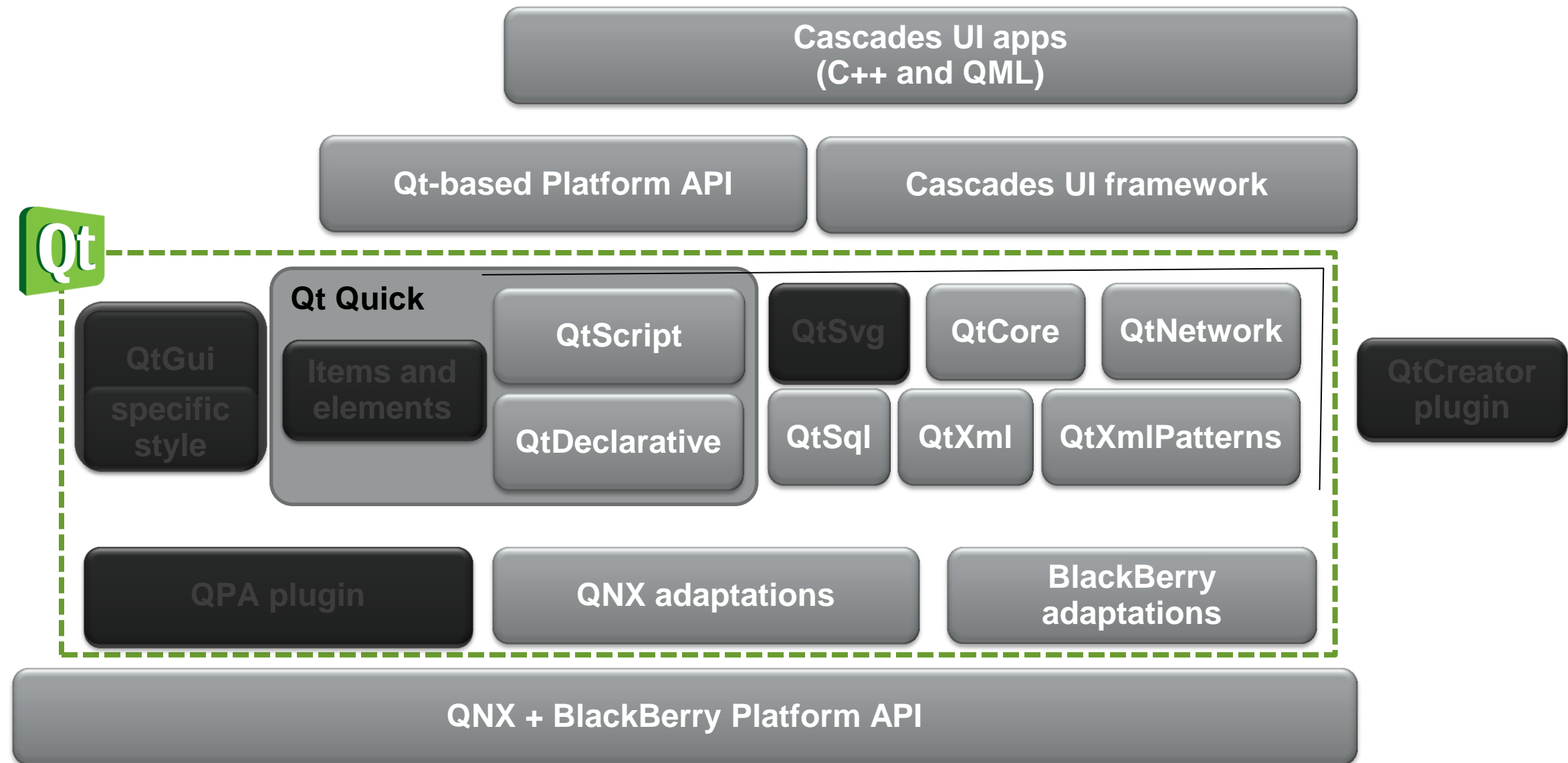


More under <http://developer.blackberry.com/native/beta/documentation/>

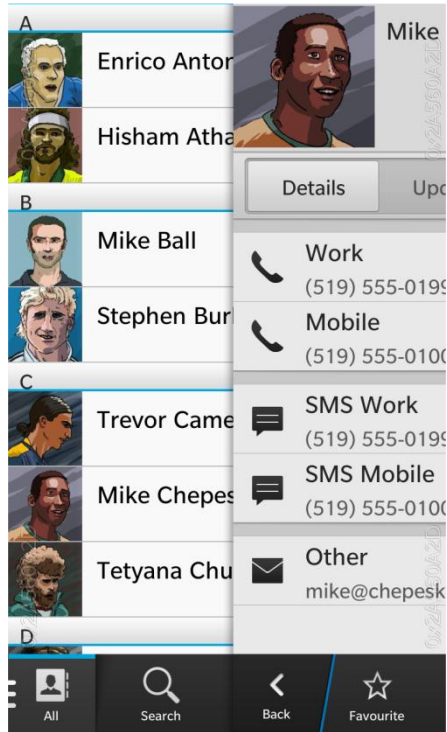
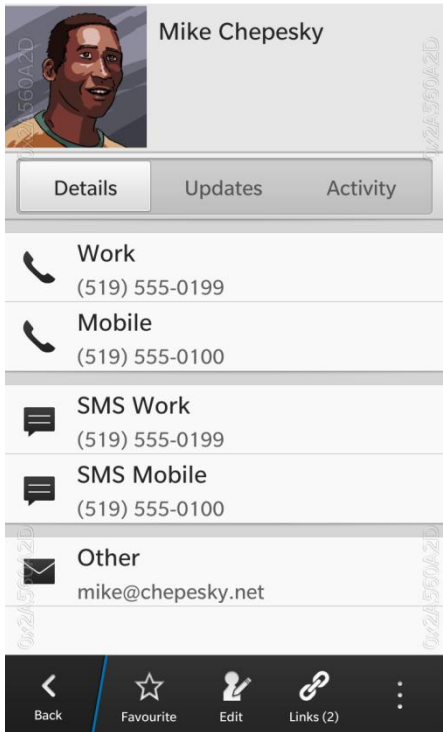
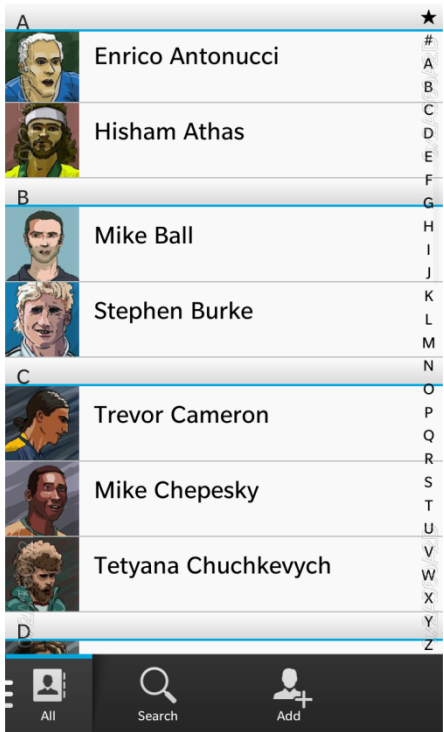
Qt on BlackBerry 10



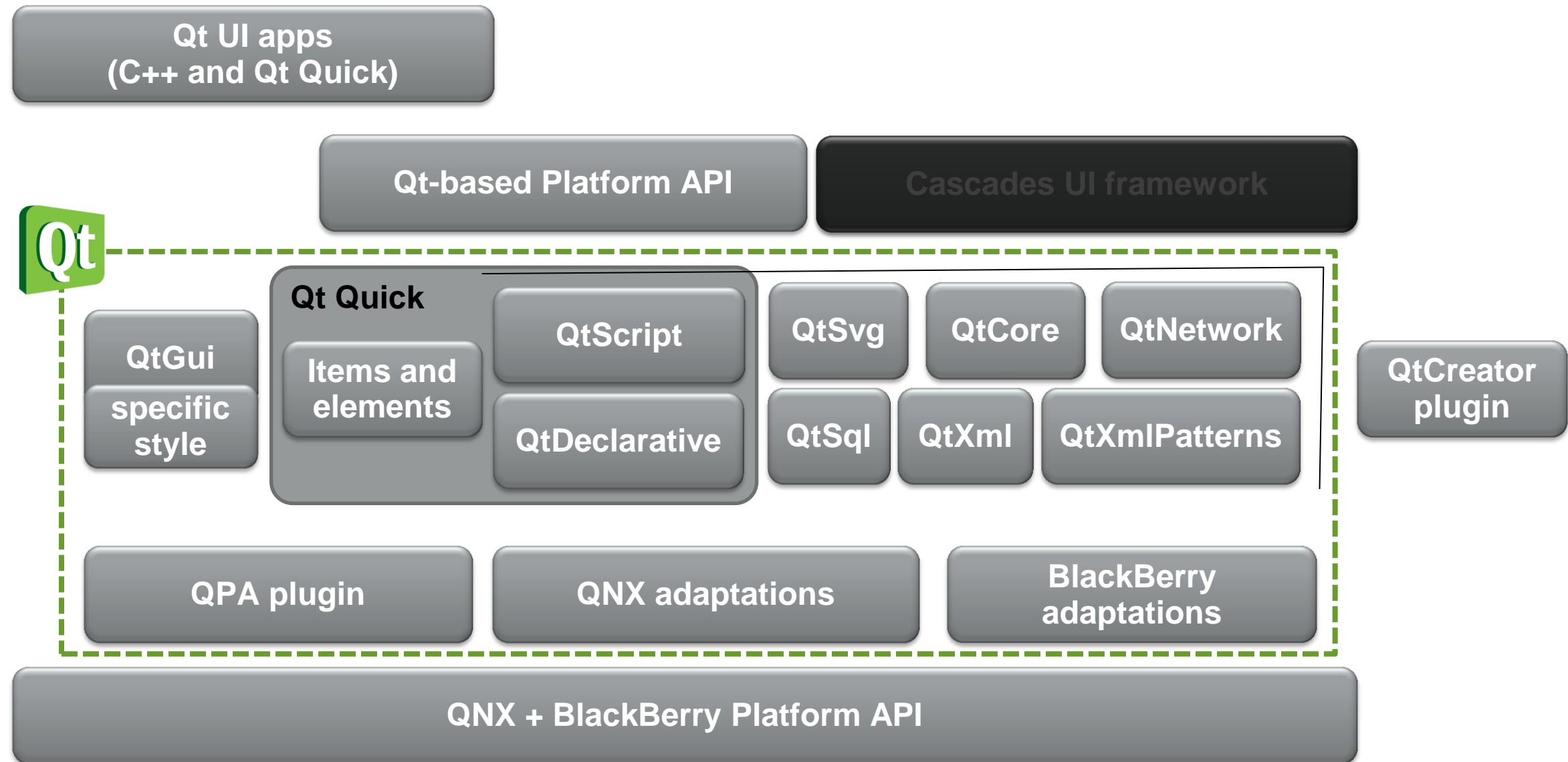
BlackBerry Cascades apps



BlackBerry Cascades apps



Pure Qt apps

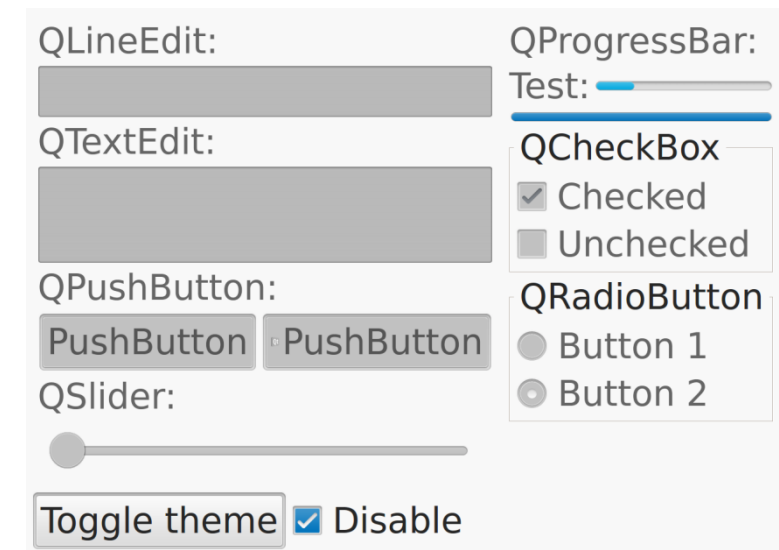
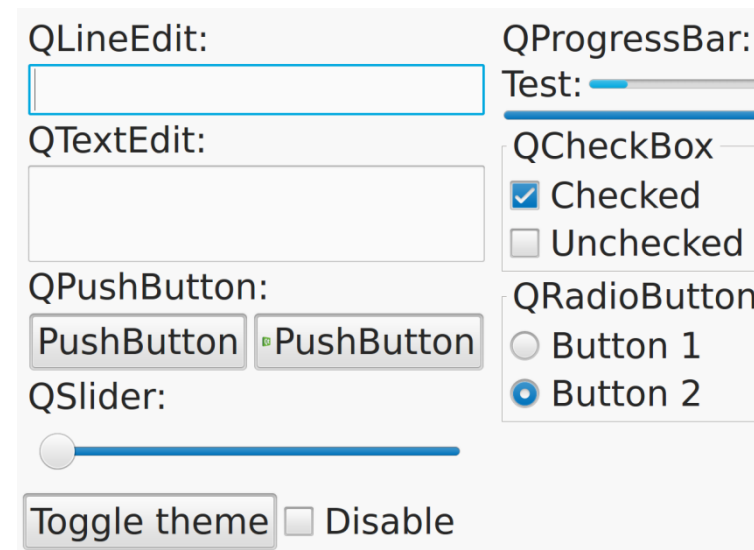


Pure Qt apps

- Qt Quick custom UI



- Qt C++ widgets
- BlackBerry 10 style
(prototype developed by KDAB)



- QtCore
- QtDeclarative*
- QtGui*
- QtMultimedia*
- QtNetwork
- QtOpenGL*
- QtScript
- QtScriptTools*
- QtSql
- QtSvg*
- QtTest
- QtXml
- QtXmlPatterns
- QtMultimediaKit*
- QtSensors
- QtLocation → QtLocationSubset
- QtConnectivity → QtNfcSubset
- QCollator → QtCollator

* Contains APIs not compatible with Cascades UI

Qt Creator for Qt Development on BlackBerry 10

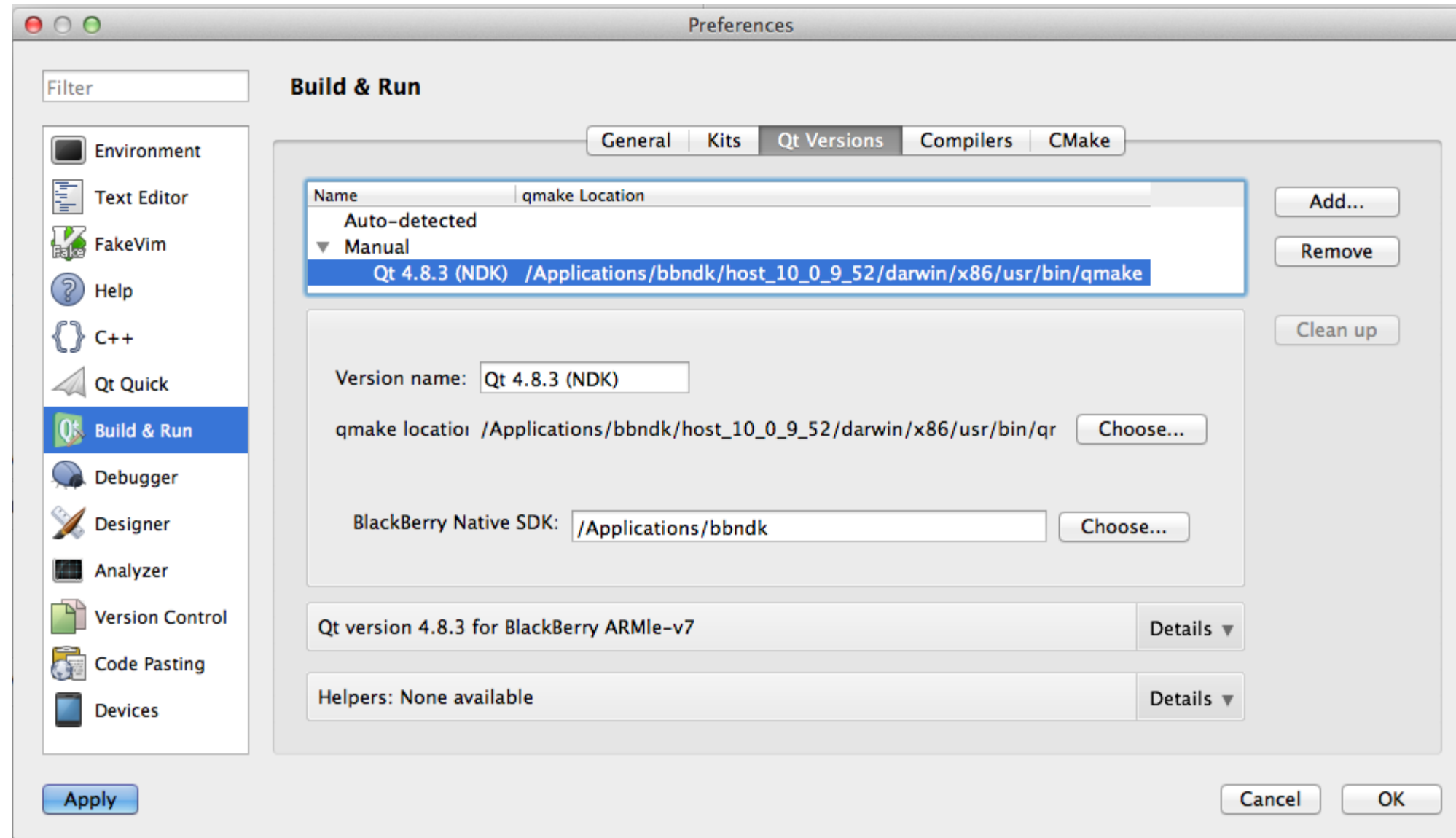
Qt Creator for BlackBerry 10:

- Set up the tool chain.
- Set up packaging and deployment.
- Fix the look-and-feel issues.
- Address platform specific runtime issues.
- Add platform specific features (device integration).
- Submit to App-World.

More under <http://qt-project.org/wiki/Qt-Creator-with-BlackBerry-10>

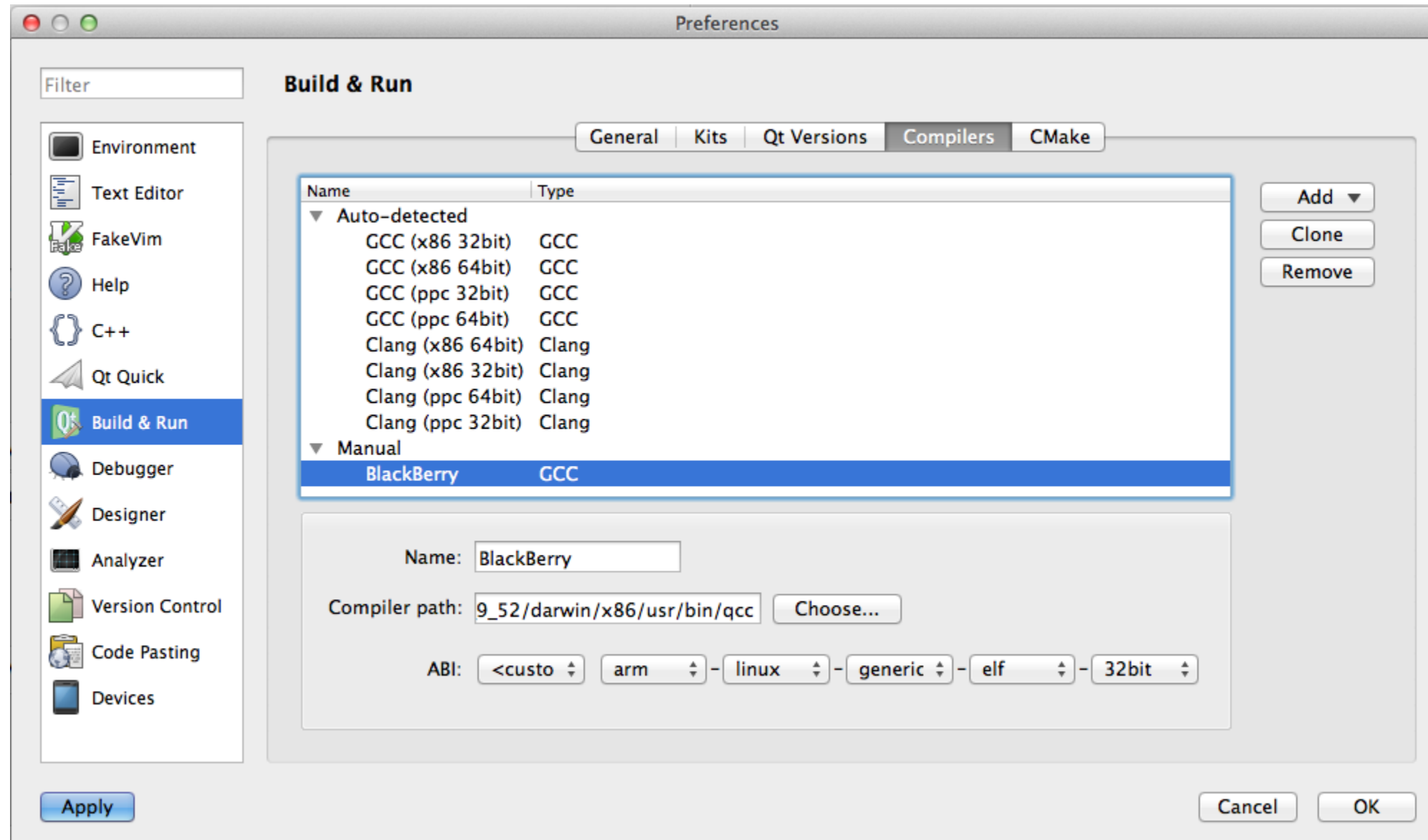
Qt Creator: step by step

- Set up the tool chain: choose the Qt version



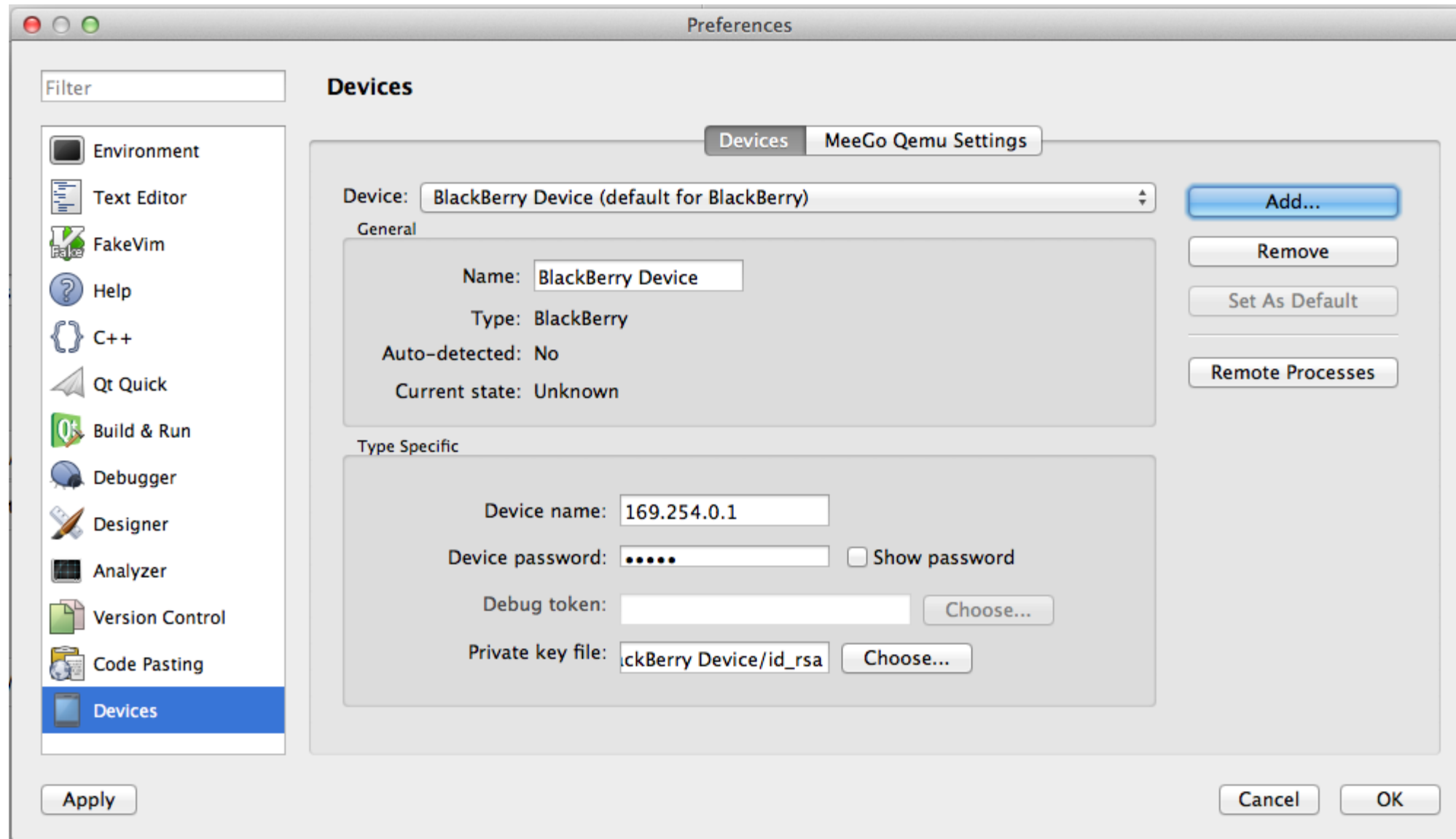
Qt Creator: step by step

- Set up the tool chain: choose the compiler



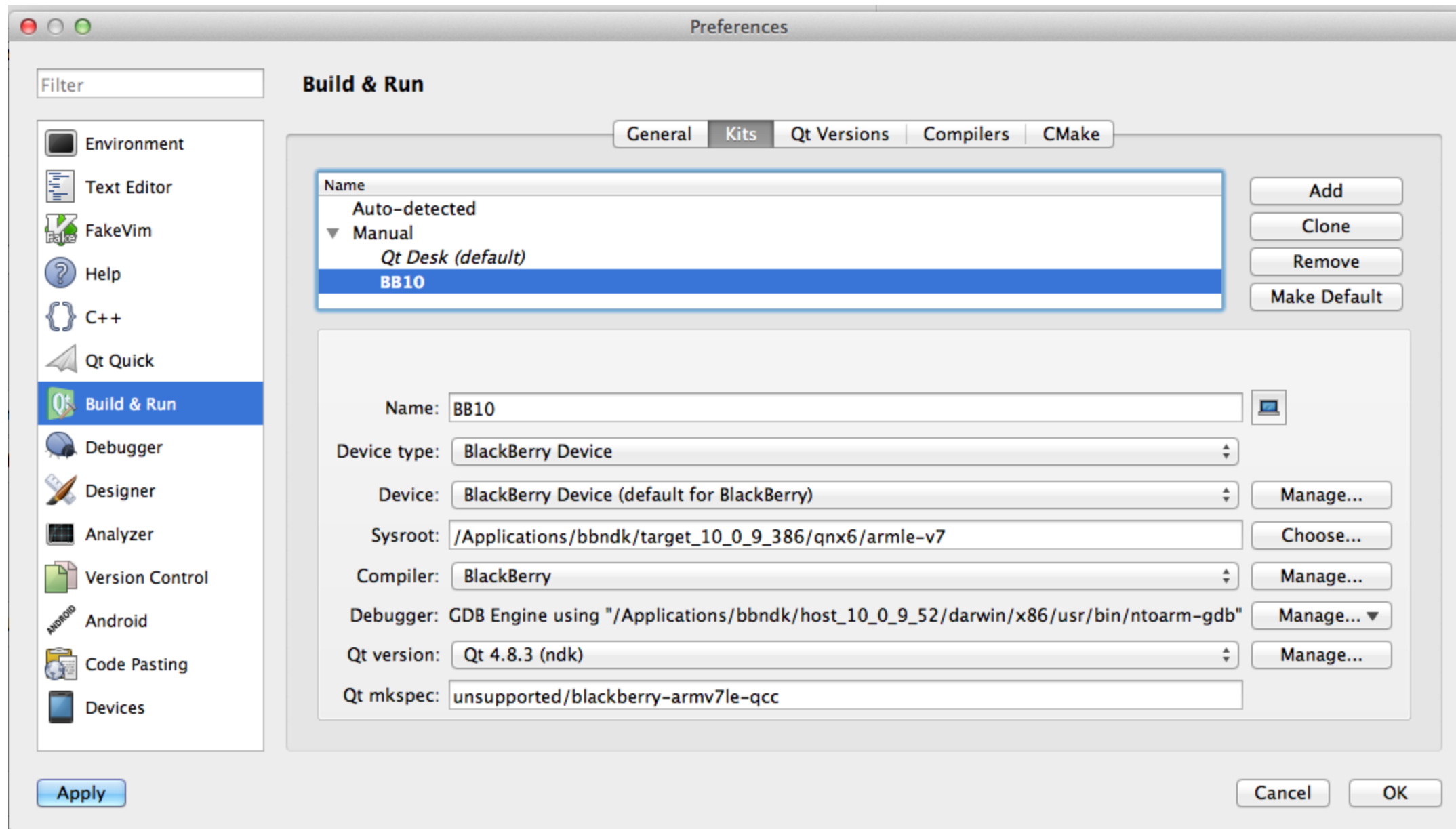
Qt Creator: step by step

- Set up the tool chain: add new BlackBerry device



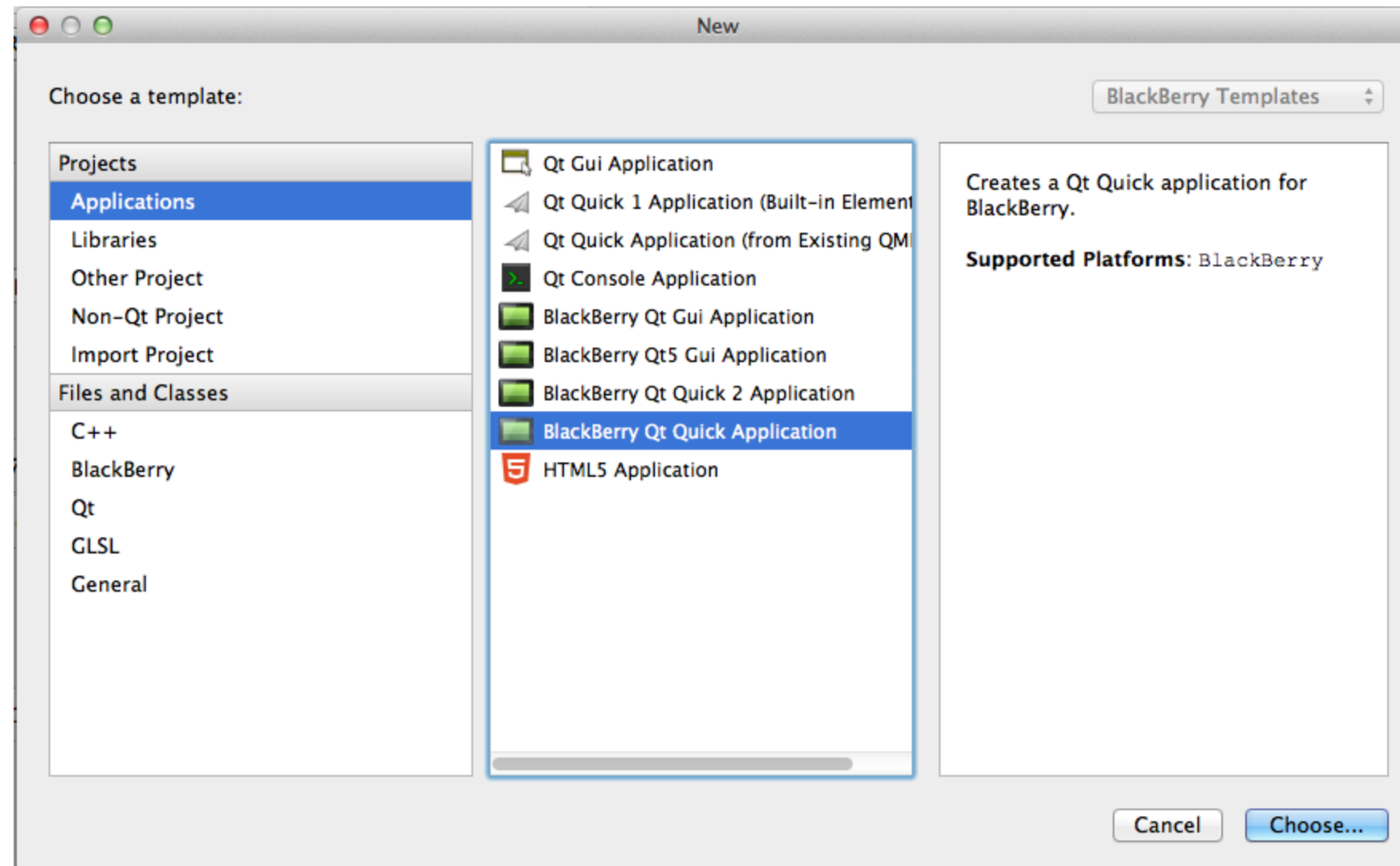
Qt Creator: step by step

- Set up the tool chain: Add new “Kit” for BlackBerry 10



Qt Creator: step by step

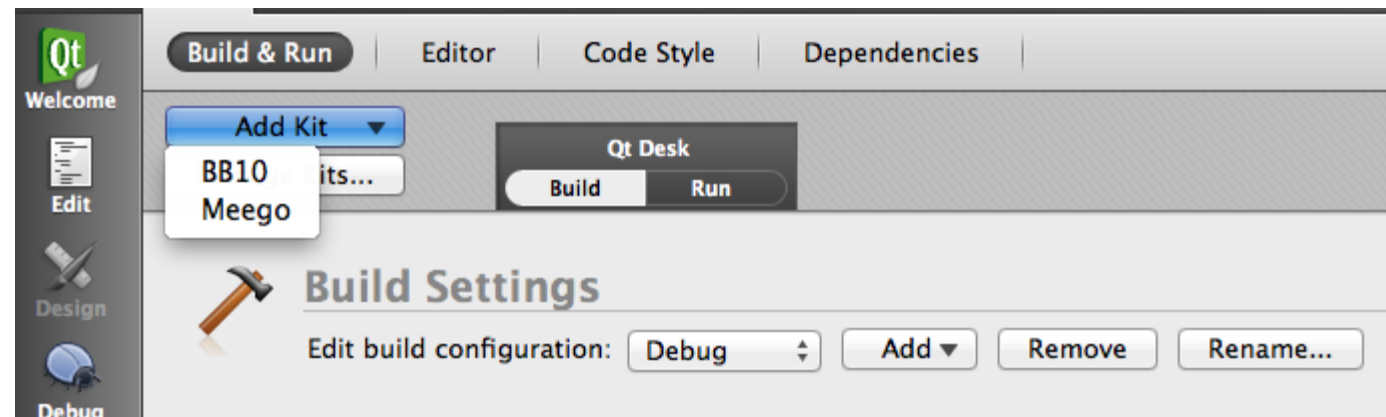
- Packaging and deployment: New BlackBerry 10 Qt Project



Packing and deployment files are automatically generated by Qt Creator when you create the BlackBerry Qt Quick project

Qt Creator: step by step

- Packaging and deployment: Existing Qt Project
 - Add the BlackBerry 10 Kit to your project:



- Add a new BAR file descriptor to you project via “New->Files and Classes->Application descriptor”

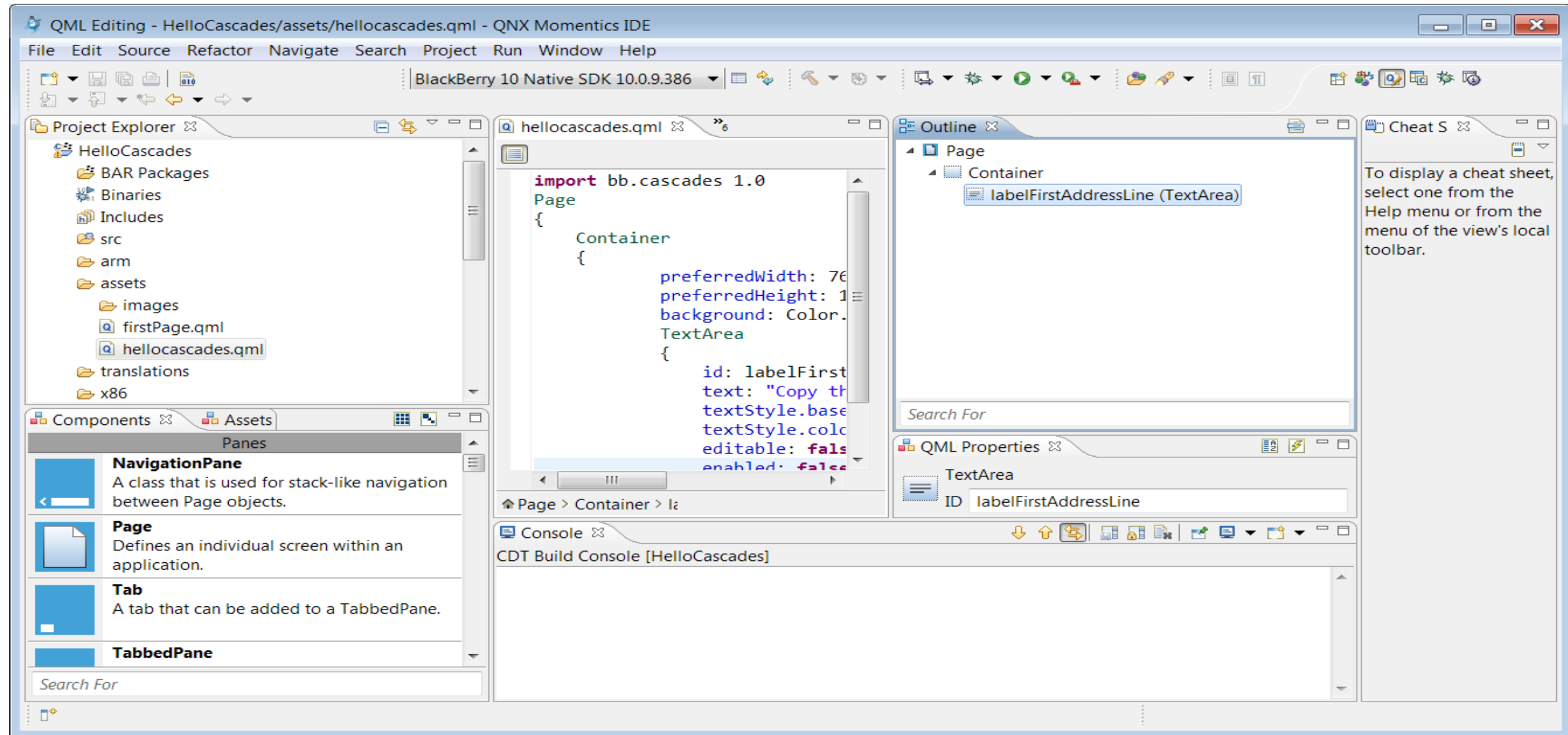
You need to add packaging and deployment files as shown above.

- Fix the look-and-feel issues:
 - Compensate for change in screen resolution, windows sizes, ...
- Address platform specific runtime issues.:
 - Rotation handling, touch gestures, ...
- Add platform specific features (device integration):
 - Use RIM's Qt-based APIs (Sensors,...)
- Submit to App-World.

More under <http://qt-project.org/wiki/Qt-Creator-with-BlackBerry-10>

QNX Momentics IDE for Cascades development on BlackBerry 10

QNX Momentics IDE



- The QNX Momentics IDE is 100% ready for Cascades development

Qt Quick/QtGui or Cascades

Pure Qt and Cascades are supported

We support both, you chose what is best for you.

However, using Cascades is the easiest path to authentic BlackBerry 10 look and feel and, seamless integration with the platform.

Key points:

- Qt Creator for BlackBerry 10 development
- Minimal adaptations
- Accessibility to RIM's none-UI Qt-based APIs
- Customization of the UI to the BlackBerry 10 style

Best practices:

- Use QtMultimediaKit from QtMobility
- Avoid using Qt Widget on BlackBerry 10 except for basic widgets, use Qt Quick instead

Key points:

- BlackBerry 10 native look and feel
- Advanced UI
- Optimized for BlackBerry hardware
- Accessibility to RIM's Qt-based APIs
- UI and UX guidelines
- BlackBerry 10 certification

Best practices:

- Use MediaPlayer APIs from RIM
- Use QML for the UI and C++ for the business logic
- Keep UI activities in own thread

Symbian Qt Quick -> Cascades

Category	Symbian QML Elements of Qt Quick	Cascades Components	Note
UIObject	QML ApplicationWindow Element	Abstract Pane	
Controls	QML BusyIndicator Element	Activity Indicator	
Controls-Buttons	QML Button Element	Button, ImageButton	
Controls	QML ButtonColumn Element	Container	Buttons in Container with Vertical StackLayout
Controls	QML ButtonRow Element	Container	Buttons in Container with Vertical StackLayout
Controls-ToggleButton	QML Checkable Element	Checkbox	
Controls-ToggleButton	QML CheckableGroup Element	Checkbox	

For More Information ...

- “Qt on Blackberry and QNX” group on qt-project.org
- <https://github.com/blackberry/Qt>
- Wiki: <http://qt-project.org/wiki/BlackBerry>
- <http://qt-project.org/wiki/QNX>
- https://developer.blackberry.com/cascades/reference/qt_index.html
- <http://developer.blackberry.com/native/beta/documentation/porting.html>
- RIM’s blog: *Writing Qt apps for the BlackBerry 10 platform*

THANK YOU

JAM812

Kizito Kasengulu

November 29-30, 2012

kkasengulu@rim.com
@kizkasengulu

Customer presentation

Porting from **Native QML** to **BlackBerry 10 Cascades**

Chanan G
Senior Software Engineer
Hua Lumpong (HLP)

Porting from Native QML to BlackBerry 10 Cascades

Copy/Paste QML/**JavaScript** files

Copy/Paste image resources

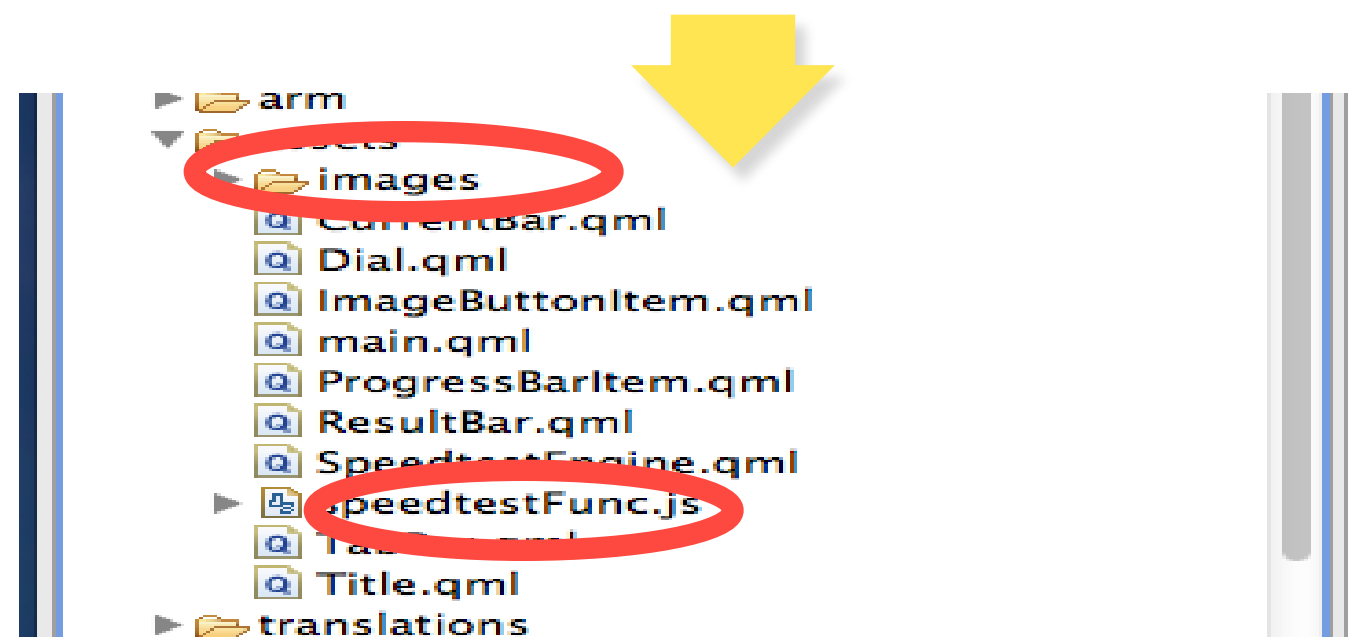
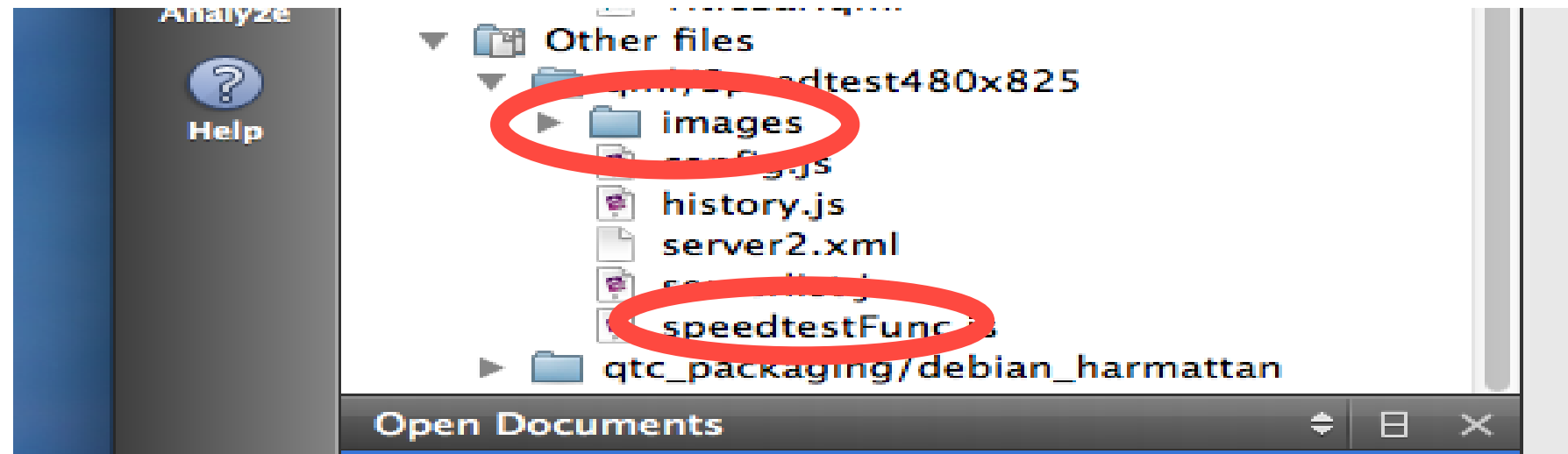
Change path referencing to URL referencing

Change view group to **Container**

Adjust layout and position

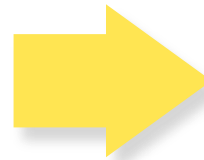
Adjust units

Steps in porting from Native QML to BlackBerry 10 Cascades



Steps in porting from Native QML to BlackBerry 10 Cascades

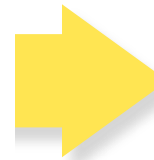
```
1 import QtQuick 1.0
2
3 Item {
4     width: 480
5     height: 133
6
7     Image {
8         id: imgBackground
9         anchors.fill: parent
10        source: "images/bg_current.png"
11
12        Image {
13            id: image1
14            x: 127
15            y: 28
16            width: 25
17            height: 30
18            source: "images/arrow_download.png"
19        }
20
21        Image {
22            id: image2
23            x: 328
24            y: 21
25            width: 40
26            height: 44
27            source: "images/arrow_upload.png"
28        }
29    }
30 }
31
32 }
33
```



```
3 Container {
4     preferredWidth: 480
5     preferredHeight: 133
6     layout: AbsoluteLayout{}
7
8     ImageView {
9         id: imgBackground
10        preferredWidth: 480
11        preferredHeight: 133
12        layoutProperties: AbsoluteLayoutProperties {
13            positionX: 0
14            positionY: 0
15        }
16        imageSource: "asset:///images/bg_current.png"
17    }
18
19    ImageView {
20        id: image1
21        layoutProperties: AbsoluteLayoutProperties {
22            positionX: 127
23            positionY: 28
24        }
25        preferredWidth: 25
26        preferredHeight: 30
27        imageSource: "asset:///images/arrow_download.png"
28    }
29
30    ImageView {
31        id: image2
32        layoutProperties: AbsoluteLayoutProperties {
33            positionX: 328
34            positionY: 21
35        }
36        preferredWidth: 40
37        preferredHeight: 44
38        imageSource: "asset:///images/arrow_upload.png"
39    }
40 }
```


Steps in porting from Native QML to BlackBerry 10 Cascades

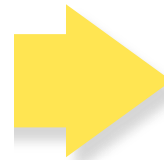
```
1 import QtQuick 1.0
2
3 Item {
4     width: 480
5     height: 133
6
7     Image {
8         id: imgBackground
9         anchors.fill: parent
10        source: "images/bg_current.png"
11
12        Image {
13            id: image1
14            x: 127
15            y: 28
16            width: 25
17            height: 30
18            source: "images/arrow_download.png"
19
20
21        Image {
22            id: image2
23            x: 328
24            y: 21
25            width: 40
26            height: 44
27            source: "images/arrow_upload.png"
28        }
29    }
30 }
31
32 }
33
```



```
3 Container {
4     preferredWidth: 480
5     preferredHeight: 133
6     layout: AbsoluteLayout{}
7
8     ImageView {
9         id: imgBackground
10        preferredWidth: 480
11        preferredHeight: 133
12        layoutProperties: AbsoluteLayoutProperties {
13            positionX: 0
14            positionY: 0
15        }
16        imageSource: "asset:///images/bg_current.png"
17    }
18
19    ImageView {
20        id: image1
21        layoutProperties: AbsoluteLayoutProperties {
22            positionX: 127
23            positionY: 28
24        }
25        preferredWidth: 25
26        preferredHeight: 30
27        imageSource: "asset:///images/arrow_download.png"
28    }
29
30    ImageView {
31        id: image2
32        layoutProperties: AbsoluteLayoutProperties {
33            positionX: 328
34            positionY: 21
35        }
36        preferredWidth: 40
37        preferredHeight: 44
38        imageSource: "asset:///images/arrow_upload.png"
39    }
40 }
```

Steps in porting from Native QML to BlackBerry 10 Cascades

```
1 import QtQuick 1.0
2
3 Item {
4     width: 480
5     height: 133
6
7     Image {
8         id: imgBackground
9         anchors.fill: parent
10        source: "images/bg_current.png"
11
12        Image {
13            id: image1
14            x: 127
15            y: 28
16            width: 25
17            height: 30
18            source: "images/arrow_download.png"
19        }
20
21        Image {
22            id: image2
23            x: 328
24            y: 21
25            width: 40
26            height: 44
27            source: "images/arrow_upload.png"
28        }
29    }
30 }
31
32 }
```



```
3 Container {
4     preferredWidth: 480
5     preferredHeight: 133
6     layout: AbsoluteLayout{}
7
8     ImageView {
9         id: imgBackground
10        preferredWidth: 480
11        preferredHeight: 133
12        layoutProperties: AbsoluteLayoutProperties {
13            positionX: 0
14            positionY: 0
15        }
16        imageSource: "asset:///images/bg_current.png"
17    }
18
19    ImageView {
20        id: image1
21        layoutProperties: AbsoluteLayoutProperties {
22            positionX: 127
23            positionY: 28
24        }
25        preferredWidth: 25
26        preferredHeight: 30
27        imageSource: "asset:///images/arrow_download.png"
28    }
29
30    ImageView {
31        id: image2
32        layoutProperties: AbsoluteLayoutProperties {
33            positionX: 328
34            positionY: 21
35        }
36        preferredWidth: 40
37        preferredHeight: 44
38        imageSource: "asset:///images/arrow_upload.png"
39    }
40 }
```

Steps in porting from Native QML to BlackBerry 10 Cascades

```
CurrentBar {  
    id: currentbar1  
    x: 0  
    y: 571  
    //anchors.bottom: tabbar1.top
```

```
    Text {  
        id: text1  
        x: 196  
        y: 50  
        width: 88  
        height: 33  
        text: + (speedtestEngine.mCurrentSpeed/speedtestEngine.speedFactor).toFixed(2)  
        anchors.verticalCenter: parent.verticalCenter  
        anchors.horizontalCenter: parent.horizontalCenter
```

```
        verticalAlignment: Text.AlignTop  
        font.pixelSize: 27  
    }  
}
```

```
CurrentBar {  
    id: currentBar  
    layoutProperties: AbsoluteLayoutProperties {  
        positionY: 571  
    }  
  
    Label {  
        id: text1  
        layoutProperties: AbsoluteLayoutProperties {  
            positionX: 204  
            positionY: 54  
        }  
        verticalAlignment: VerticalAlignment.Center  
        horizontalAlignment: HorizontalAlignment.Center  
        preferredWidth: 88  
        preferredHeight: 33  
        text: + (speedtestEngine.mCurrentSpeed/speedtestEngine.speedFactor).toFixed(2)  
        textStyle.fontSizeValue: 5.0  
        textStyle.textAlignment: Text.Align.Center  
    }  
}
```

What about the speed in porting?

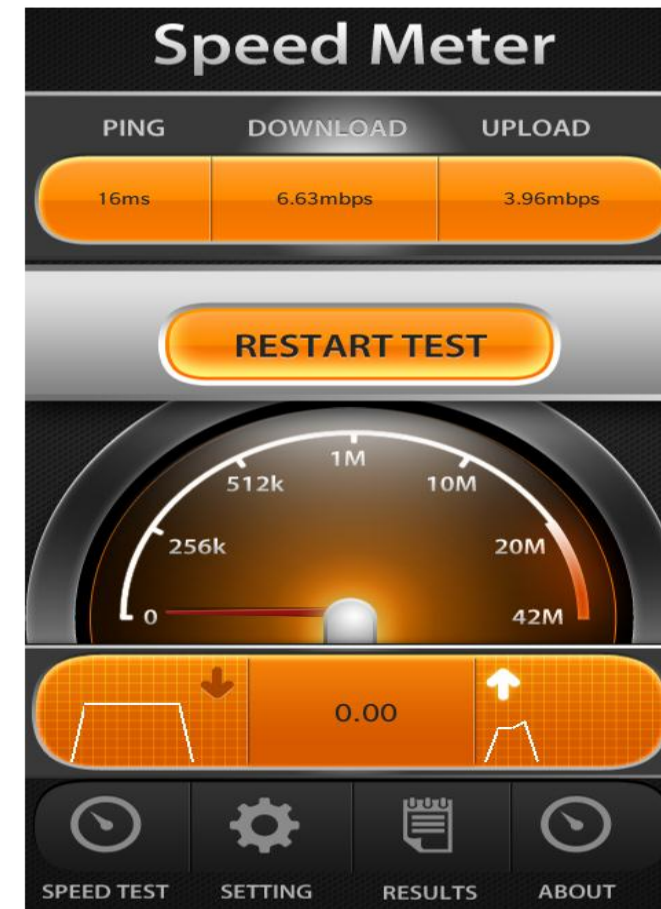
very fast!

It took only one day
for small application

BlackBerry 10 Porting Case Study: Speed Meter

- Application for measuring the speed of the internet link (originally in Meego)
- Port from pure Qt Quick to pure BlackBerry 10 Cascades

[Demo Time]



BlackBerry 10 Porting Case Study: MOLOME

- Application for photo sharing and social network (originally in Symbian)
- Port from Native QtQuick/C++ to BlackBerry 10 Cascades QML/C++

[Demo Time]

Our difficulties **BlackBerry 10 Cascades**

Automated Converter

WYSIWYG Component Editor

Screen Scaler

Simpler Debugger

Thank you

... And a few Quick Notes

- Download the **Mobile Conference Guide** from BlackBerry World™ and start exploring BlackBerry Jam Asia!
- Have more questions? Visit **Ask the Experts** at 6pm tonight by the coding lounge in the Jam Space.
- Check out the winners of the JamHack events in Asia at the **JamHack Finals** and vote for your favourite app at 6pm in Meeting Room 2.
- Don't forget to relax and unwind at the **Appreciation Event** tonight in the Retro Café