



Multi-Platform Applications with Qt

LET'S TALK ABOUT QT

MỤC LỤC:

- I. What is Qt ?
- II. Qt Qwidget
- III. Signal and Slot
- IV. QT 6 AND QT ROADMAP
- V. Multi-Screen Approaches with Qt

WHAT IS QT ?

- QT LÀ MỘT PLATFORM PHÁT TRIỂN ỨNG DỤNG ĐA NỀN TẢNG DÀNH CHO MÁY TÍNH ĐỂ BÀN, NHÚNG VÀ THIẾT BỊ DI ĐỘNG.
- CÁC NỀN TẢNG ĐƯỢC HỖ TRỢ BAO GỒM

Desktop

windows

Linux

OS x

Solaris

EnterpriseUNIX

Embedded

Windows Embedded
(Standard/Compact 7)

Embedded Linux

INTEGRITY

QNX

VxWorks

Mobile

Android (5.2, beta 5.1)

iOS (5.2, alpha 5.1)

Win8 on ARM (WinRT) (5.2?)

BlackBerry 10

Jolla Sailfish

WHAT IS QT ?

- Qt **không Phải** là một ngôn ngữ lập trình của riêng nó. Nó là một framework được viết bằng C++. Một bộ tiền xử lý MOC, sử dụng ngôn ngữ C++ với các tính năng như signals and slots.
- Bản thân Framework và các ứng dụng/ thư viện sử dụng nó có thể được biên dịch bởi bất kỳ trình biên dịch C++ tuân thủ tiêu chuẩn như Clang, GCC, ICC, MinGW và MSVC.
- Cmake trên Qt5: Khi tạo ra 1 phần mềm lớn và phức tạp. Cmake có thể phù hợp hơn Qmake
- IDE : Qt đi kèm với Môi trường phát triển tích hợp(IDE) của riêng mình được đặt tên là Qt creator. Ngoài ra các nhà phát triển trên windows còn có sử dụng Qt Visual studio tools

Qt/C++ còn hỗ trợ

QtCore

- Data types, containers
- Threads, Processes,
- IPC File I/O
- String handling

- QtNetwork

TCP/UDP, HTTP, FTP, SSL

- QSql
- QtWebkit
- Qt Serial Port (new in 5.1)
- Etc.

Qt App

Ui Level

Qt Quick

Qt Quicks (C++)

Logic

Javascript/C++

C++

Extension/Interfacing

C++

Xây dựng hệ thống

Mặc dù bất kỳ hệ thống xây dựng nào cũng có thể được sử dụng với Qt, Qt mang đến qmake của riêng mình. Nó là một giao diện người dùng đa nền tảng cho các hệ thống xây dựng gốc nền tảng, như GNU Make, Visual Studio và Xcode.

CMake cũng là một giải pháp thay thế phổ biến để xây dựng các dự án Qt, hỗ trợ Qt 4 đã được tích hợp từ nhiều năm trước và Qt 5 đã hỗ trợ sớm.

Một nền tảng mới xuất hiện gần đây: Qt Build Suite hay còn gọi là Qbs. Qbs là một hệ thống xây dựng dựa trên QML cũng cung cấp hỗ trợ cho Javascript. Hệ thống xây dựng này không chỉ cung cấp khả năng xây dựng mà còn đóng gói như cmake.

QT APPLICATION

PROJECT DEFINITION FILE (*.pro)

-Khi tạo một project Qt bất kỳ, sẽ có một file có đuôi là *.pro trong project.

-File này chứa những thông tin về các module mà project sẽ sử dụng.

FORMS (mainwindow.ui)

Chứa biểu mẫu, tiện ích hoặc giao diện người dùng được tạo bằng bộ phát triển phần mềm giao diện người dùng đồ họa Qt.

main.cpp

Sử dụng công cụ QQuickView để hiển thị giao diện người dùng QML của bạn

C++ CLASS (mainwindow.h mainwindow.cpp)

Chứa các thao tác, biến, hằng số và chức năng được sử dụng bởi các tệp khác trong project

The background is a light gray gradient. In the top-left and bottom-right corners, there are several realistic water droplets of various sizes, rendered with highlights and shadows to give them a 3D appearance. A faint, large, circular, textured pattern is centered in the upper half of the image, resembling a lens flare or a subtle watermark.

LAYOUT

LAYOUT

```
graph LR; LAYOUT --> QHBoxLayout; LAYOUT --> QVBoxLayout; LAYOUT --> QFormLayout; LAYOUT --> QGridLayout; QHBoxLayout --> HDesc; QVBoxLayout --> VDesc; QFormLayout --> FDesc; QGridLayout --> GDesc;
```

`QHBoxLayout`

Lớp `QHBoxLayout` sắp xếp các widget theo chiều ngang

`QVBoxLayout`

Lớp `QVBoxLayout` sẽ sắp xếp các widget theo hàng dọc. Để thêm widget vào layout thì ta dùng phương thức `addWidget()`.

`QFormLayout`

`QFormLayout` là layout trợ giúp chúng ta trong việc tạo các form (biểu mẫu). Layout này chia thiết kế làm hai cột, một bên là tên các trường, một bên là các trường để điền, có thể là các text box hay spin box.

`QGridLayout`

Lớp `QGridLayout` thiết kế các widget theo dạng lưới (hay bảng).

The background of the slide is a light gray gradient. In the top-left and bottom-right corners, there are several realistic-looking water droplets of various sizes, some overlapping. A faint, circular, embossed-style logo is centered in the upper half of the slide, behind the text. The logo appears to be a stylized globe or a similar abstract design.

QT QUICK/QML AND QT WIDGETS

Qt cung cấp 2 giải pháp thiết kế giao diện người dùng là Qt Quick và QWidgets

- **Qt quick** là một ngôn ngữ kịch bản được gọi là QML cho phép sử dụng JavaScript. Qt Quick cũng là một lựa chọn tốt để phát triển nhanh chóng một cho các thiết bị di động.

Qt Quick/QML

Mobile

Embedded

Touch Screens

- **Qt widget** là mô hình giao diện người dùng hướng đến máy tính để bàn truyền thống (Windows, Mac và linux)

Qt Widgets

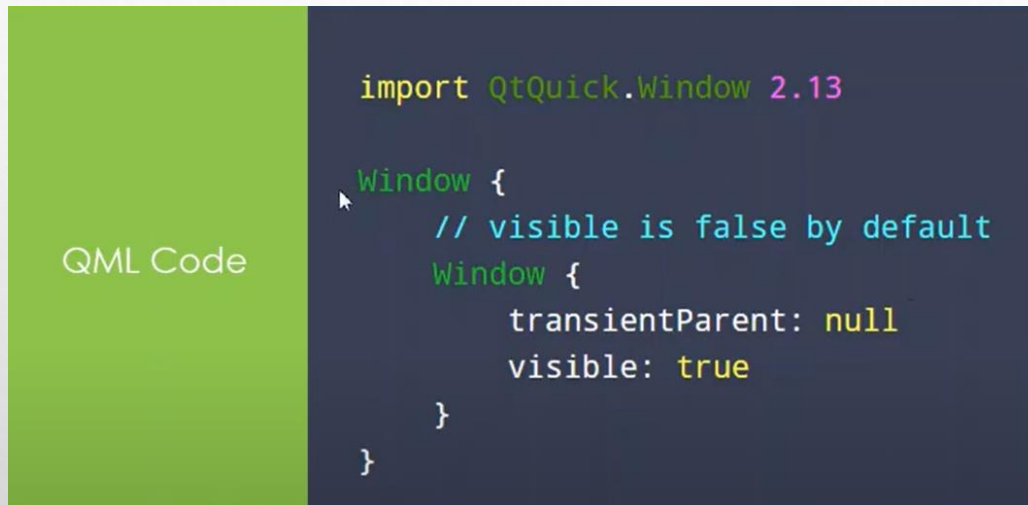
Windows

Linux

Mac

QT QUICK

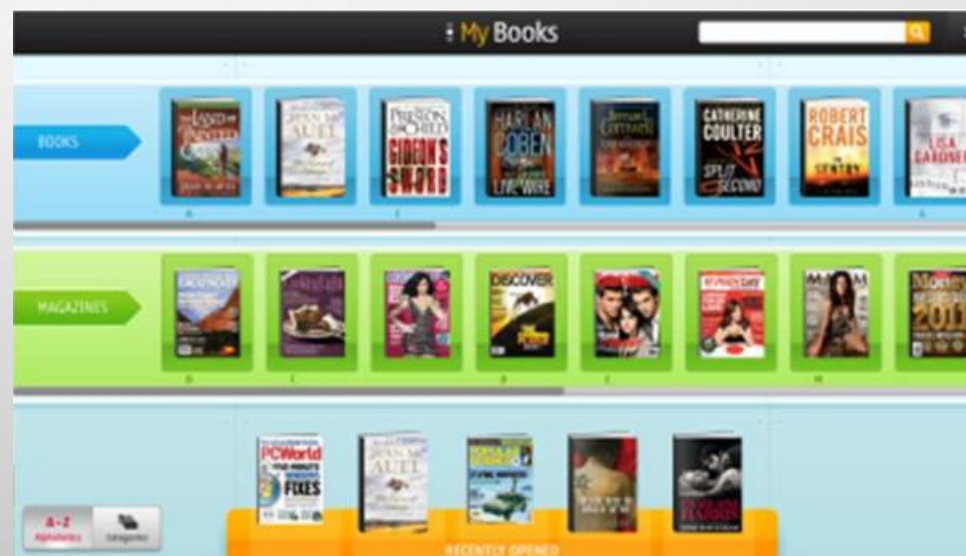
- GUI được viết bằng QtQuick được viết bằng QML, QML là một ngôn ngữ mô tả đối tượng khai báo tích hợp Javascript cho lập trình thủ tục. QtQuick cung cấp các mô-đun cần thiết để phát triển GUI với QML. Chỉ có thể viết toàn bộ ứng dụng trong QML, nhưng thường chỉ có GUI được viết bằng QML và backend của ứng dụng được triển khai bằng C++.



```
import QtQuick.Window 2.13

Window {
    // visible is false by default
    Window {
        transientParent: null
        visible: true
    }
}
```

Đây là một ví dụ về QML, ta có thể đặt các phần nhỏ lại với nhau và có thể ghép lại nguyên mẫu lại với nhau.



QT WIDGETS

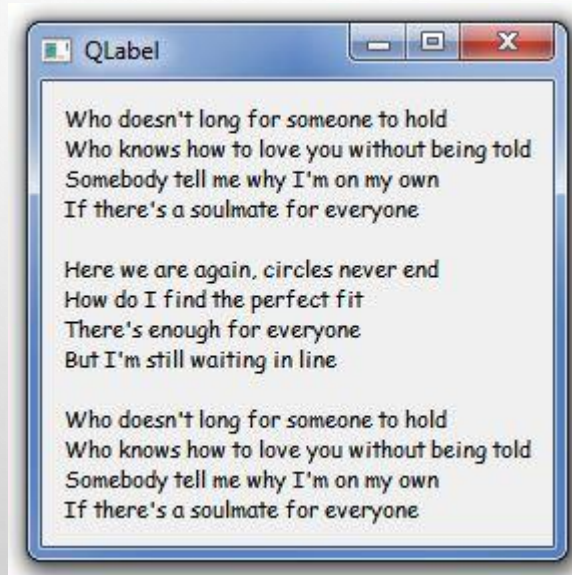
Qt đi kèm với một loạt các widget tiêu chuẩn mà người dùng các ứng dụng hiện đại đã mong đợi.

Chúng ta sẽ nói về một số widget cơ bản trong Qt5 gồm:

QLabel, QSlider, QComboBox, QSpinBox, QLineEdit, và QMainWindow.

QLabel

QLabel được dùng để hiển thị văn bản và hình ảnh. Nhưng không thể tương tác với người dùng.



QSlider

QSlider là widget hiển thị một thanh trượt có thể kéo qua kéo lại. Thường dùng để thiết lập giá trị cho một công việc nào đó.



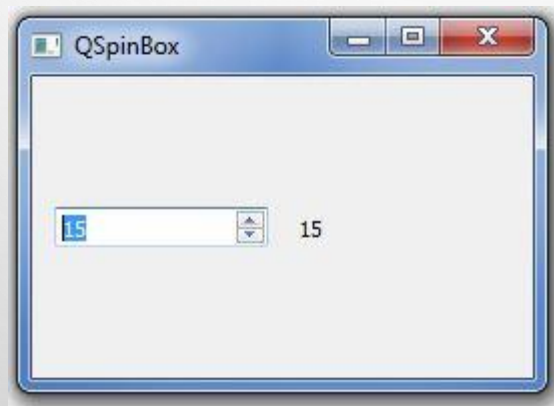
QComboBox

Lớp QComboBox hiển thị một danh sách các item mà user có thể chọn lựa.



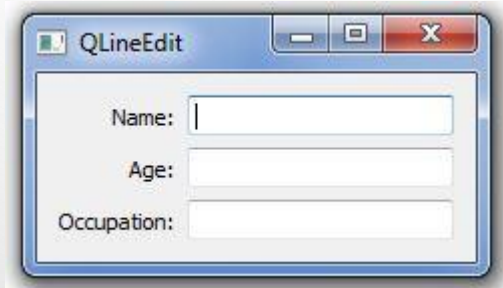
QSpinBox

QSpinBox là widget chuyên dùng để làm việc với số nguyên và các giá trị rời rạc. Ví dụ dưới đây sẽ hiển thị một spinbox chứa các số trong tập từ 0→99. Số nào được chọn thì hiển thị lên label.



QLineEdit

QLineEdit là widget cho phép gõ một dòng văn bản



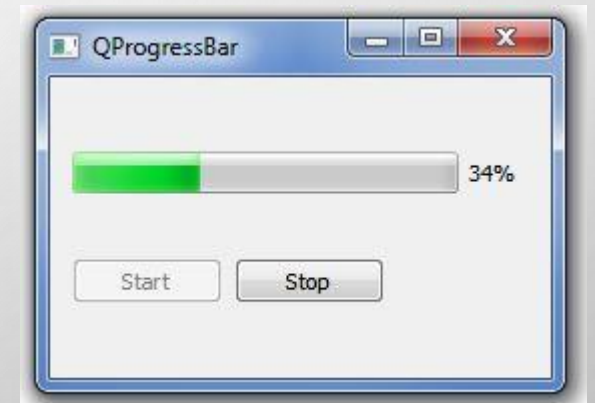
Ngoài ra còn có một số widget hay dùng trong Qt 5 gồm có:
QCheckBox, QListWidget, QProgressBar, QPixmap, QSplitter, và QTableWidget.

QCheckBox:

QListWidget: QListWidget dùng để hiển thị một danh sách các item,

QProgressBar: dùng để theo dõi tiến trình hoàn thành của một công việc nào đó.

Qpixmap: là lớp dùng để làm việc với ảnh.



The background is a light gray gradient. In the top-left and bottom-right corners, there are several realistic water droplets of various sizes, rendered with soft shadows and highlights to give them a three-dimensional appearance. The text "SIGNAL VÀ SLOT" is centered in the middle of the image.

SIGNAL VÀ SLOT

Cơ chế hoạt động của Signal và Slot

Trong lập trình GUI thì có một thứ rất quan trọng đó là sự kiện (event), khi một sự kiện nào đó xảy ra thì sẽ có các đối tượng xử lý sự kiện đó. Chẳng hạn như khi click vào nút X trên góc cửa sổ thì thoát chương trình. Qt xử lý sự kiện bằng cách tạo ra Signal và Slot.

```
// connect Signal and slots for math buttons

connect(ui->Add, SIGNAL(released()), this,
        SLOT(MathButtonPress()));
connect(ui->Subtract, SIGNAL(released()), this,
        SLOT(MathButtonPress()));
connect(ui->Multiply, SIGNAL(released()), this,
        SLOT(MathButtonPress()));
connect(ui->Divide, SIGNAL(released()), this,
        SLOT(MathButtonPress()));
```

Signal: tiếng Việt có nghĩa là *tín hiệu*. Trong Qt, khi một sự kiện nào đó xảy ra, một signal sẽ được phát đi giống như đài truyền hình phát sóng vậy, thực ra nó chỉ là một phương thức của một lớp nhưng không có phần thân hàm { }. Các lớp Widget có sẵn trong Qt có rất nhiều signal được định nghĩa sẵn, và chúng ta cũng có thể viết các signal riêng cho các lớp của chúng ta. Signal không có kiểu trả về, kiểu trả về của signal luôn luôn là void.

Slot: Slot chẳng qua cũng là một phương thức bình thường của một lớp, các phương thức này sẽ được gọi khi có một signal nào đó được phát đi. Cũng giống như signal, các lớp Widget trong Qt cũng có sẵn rất nhiều slot và chúng ta cũng có thể viết slot cho lớp của riêng chúng ta.

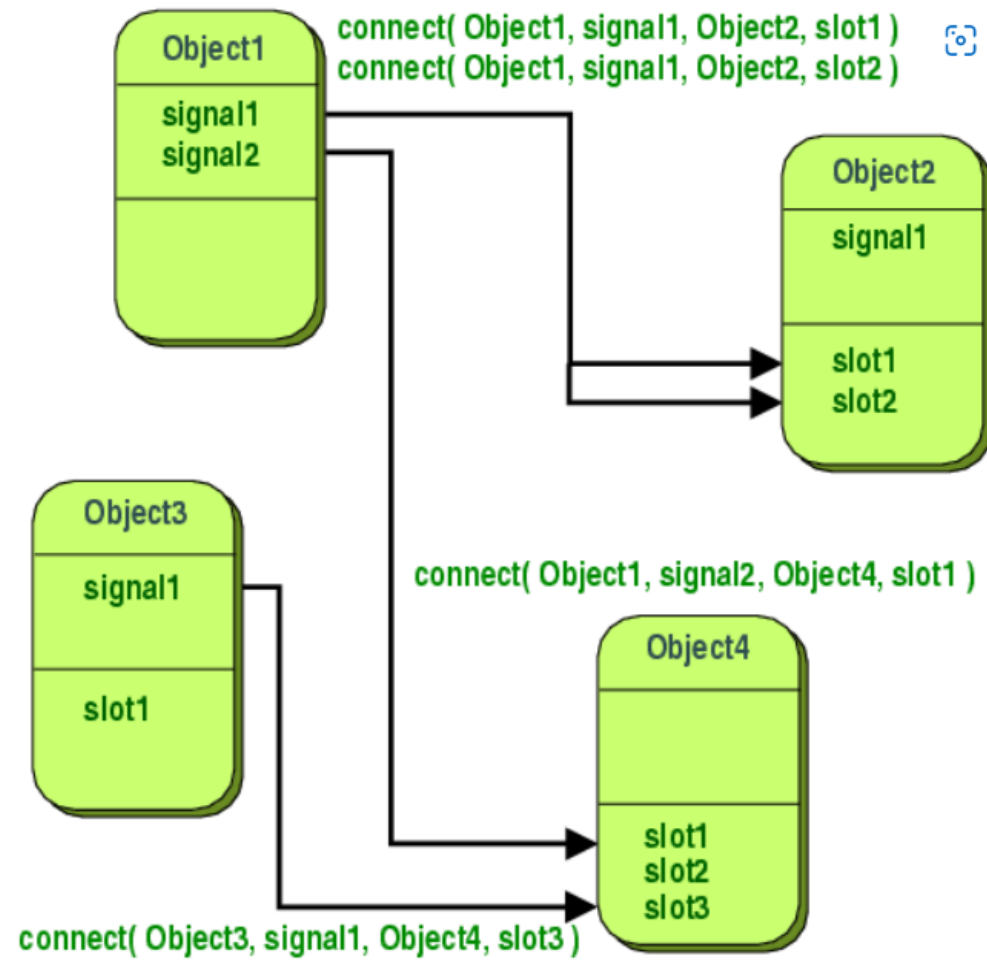
Connect: Signal và slot được kết nối qua từng đối tượng Tức là chúng ta chỉ có kết nối đối tượng này với đối tượng kia chứ không kết nối lớp này với lớp kia, giả sử chúng ta có đối tượng object1, object2 thuộc lớp A và object3 thuộc lớp B thì chúng ta chỉ có thể kết nối object1->object2, object1-> object3 hoặc object3->object2 chứ không kết nối lớp A đến lớp B.

Khi kết nối thì một đối tượng sẽ làm vai trò làm signal, một đối tượng sẽ nhận signal. Đối tượng phát signal tương ứng sẽ thực thi slot của đối tượng đó.

Một đối tượng có thể liên kết đến nhiều slot và một slot có thể liên kết đến nhiều signal

Chú ý: Tham số của signal phải ít hơn hoặc bằng tham số của slot. Khi một signal được phát đi, nó sẽ mang theo dữ liệu là các tham số của nó, và slot nhận signal này sẽ nhận các tham số đó thông qua tham số của nó. Thứ tự các tham số của signal và slot phải giống nhau, chẳng hạn như signal gửi 1 int, sau đó là 1 string thì slot cũng phải nhận 1 int rồi mới tới string.

Một signal cũng có thể kết nối đến một signal khác, tức là như thế sẽ phát ra 2 signal.



QT 6 AND QT ROADMAP

Additions across the whole spectrum

- + **Qt PDF**
- + **Qt Speech**
- + **Qt Location:** update by [Lauri Laanmets](#)
- + Qt for webAssembly.

Desktop

- + Qt6 liên tục update trên các nền tảng MacOS, Win,.. Mới nhất
- + TreeViews,

Mobile platforms

- + update style Material cho Android và Universal cho Windows.
- + cải thiện việc quản lý quyền
- + **API Splash Screen**
- + **cải tiến IME**

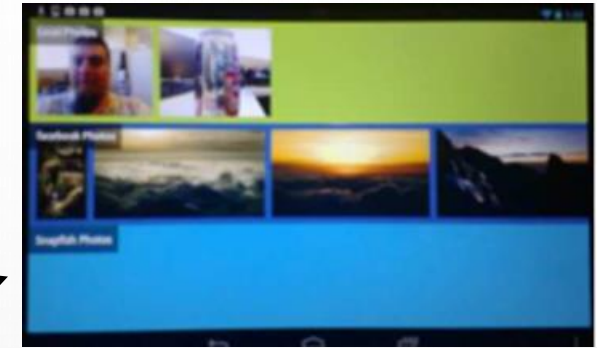
Accelerate the Design –
Develop workflow

The background is a light gray gradient. In the top-left and bottom-right corners, there are several realistic-looking water droplets of various sizes, rendered with soft shadows and highlights. A faint, circular, embossed-like pattern is visible in the upper center of the slide.

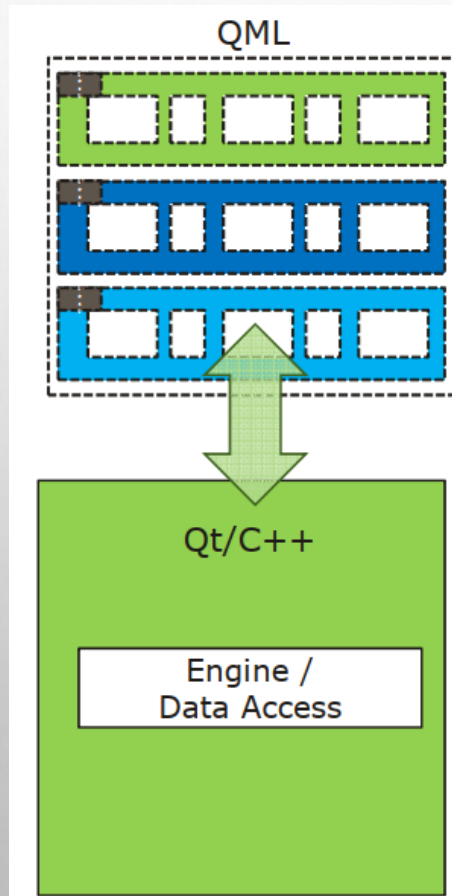
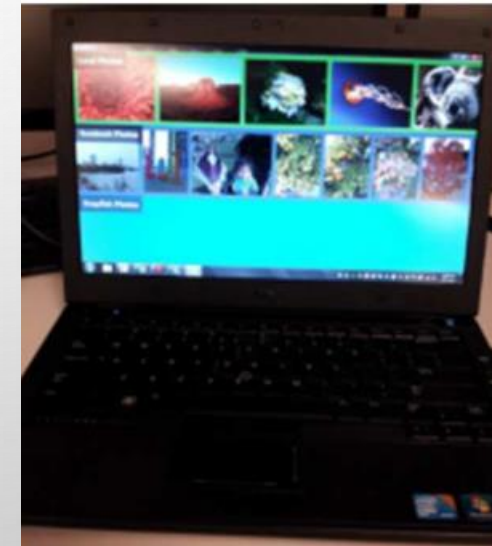
MULTI-SCREEN APPROACHES WITH QT

Smart-Phone

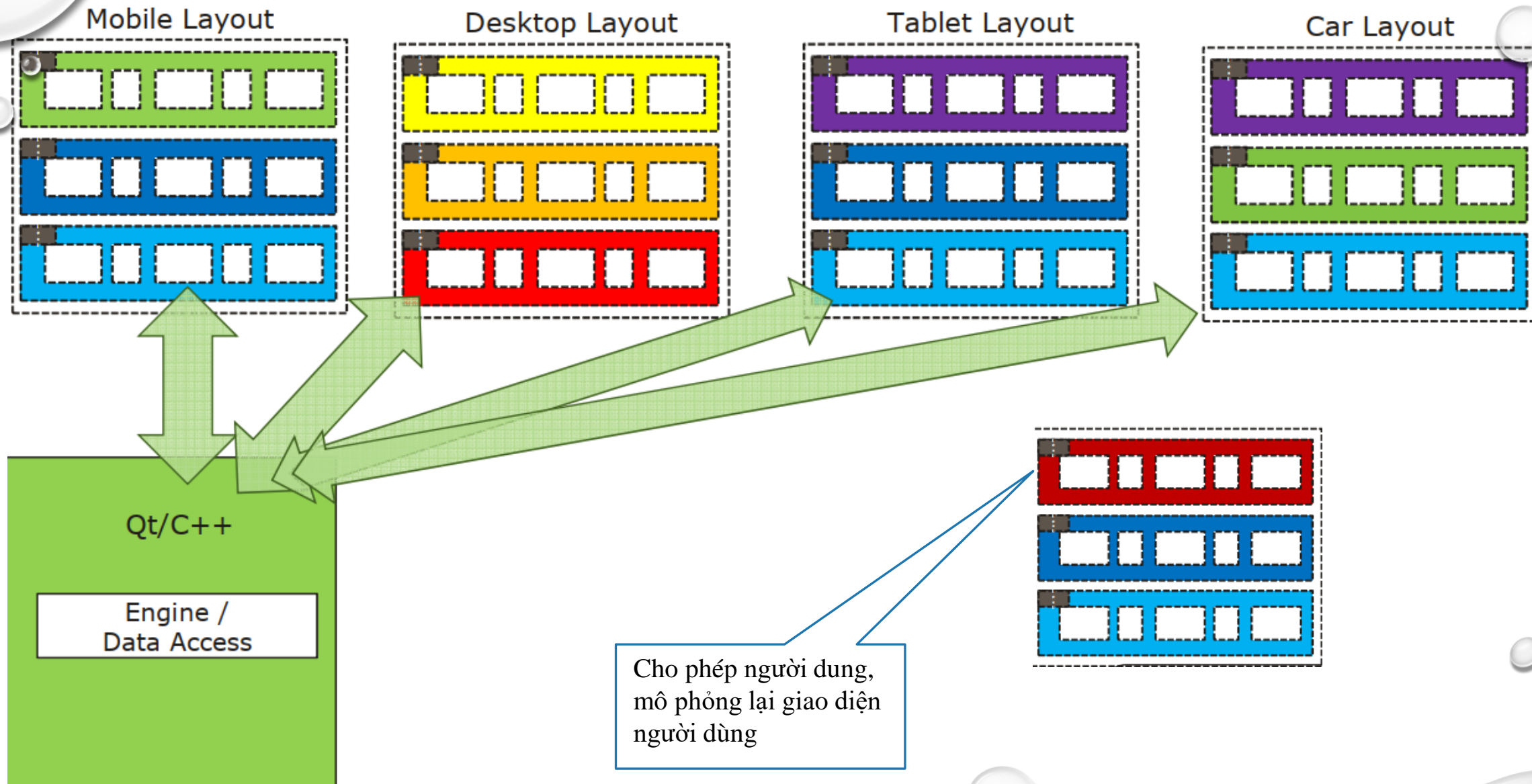
Tablet

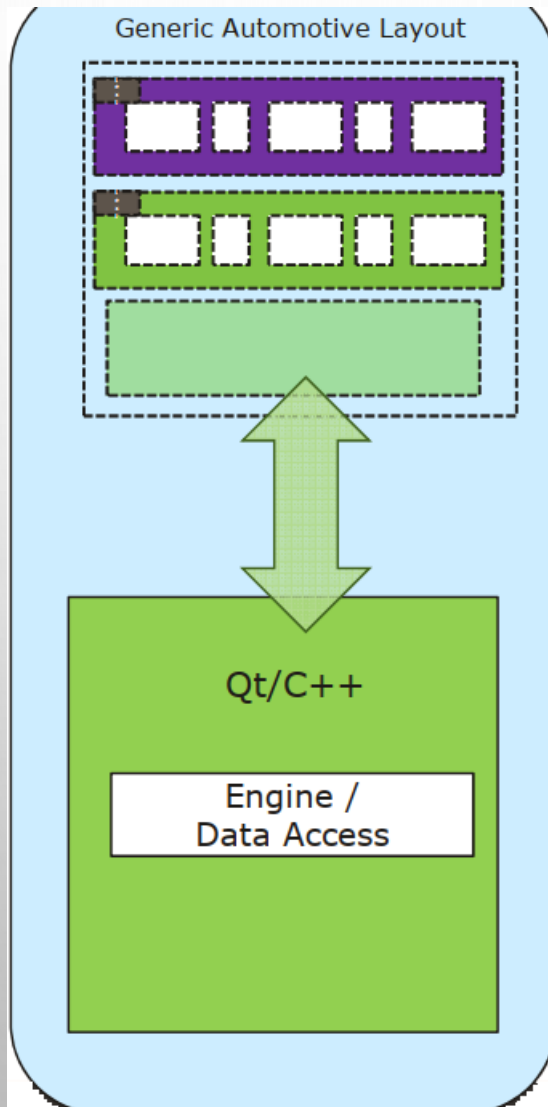


PC



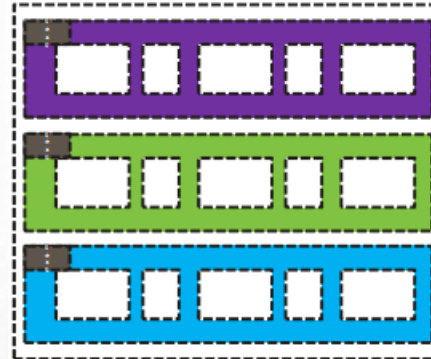
N-Screen Approaches – Different UI Designs



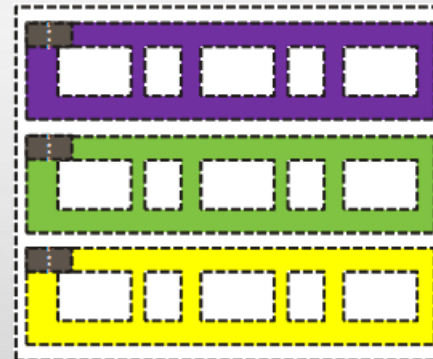


Mass-customization
by Qt customer

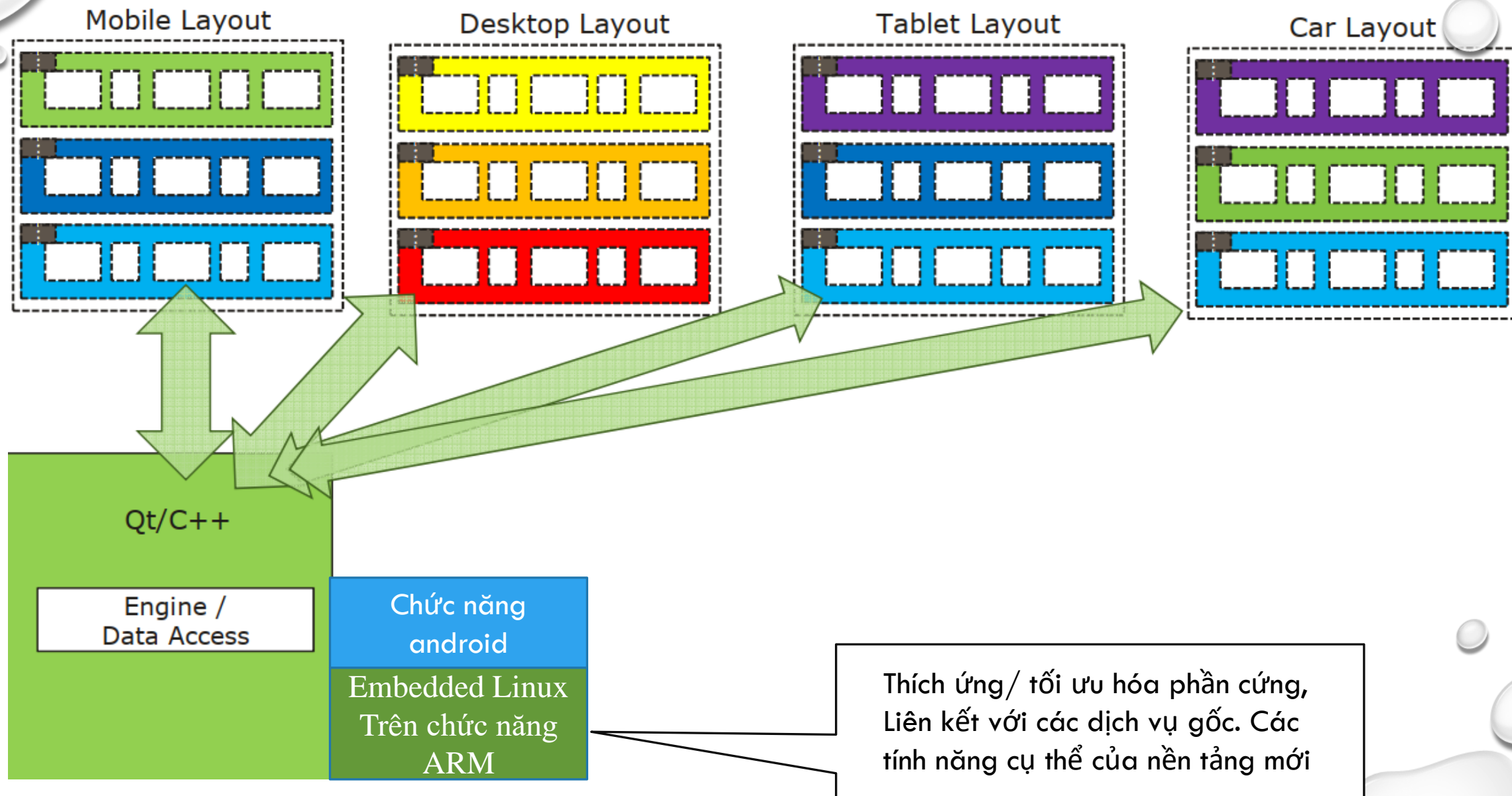
Car MF 1 branded



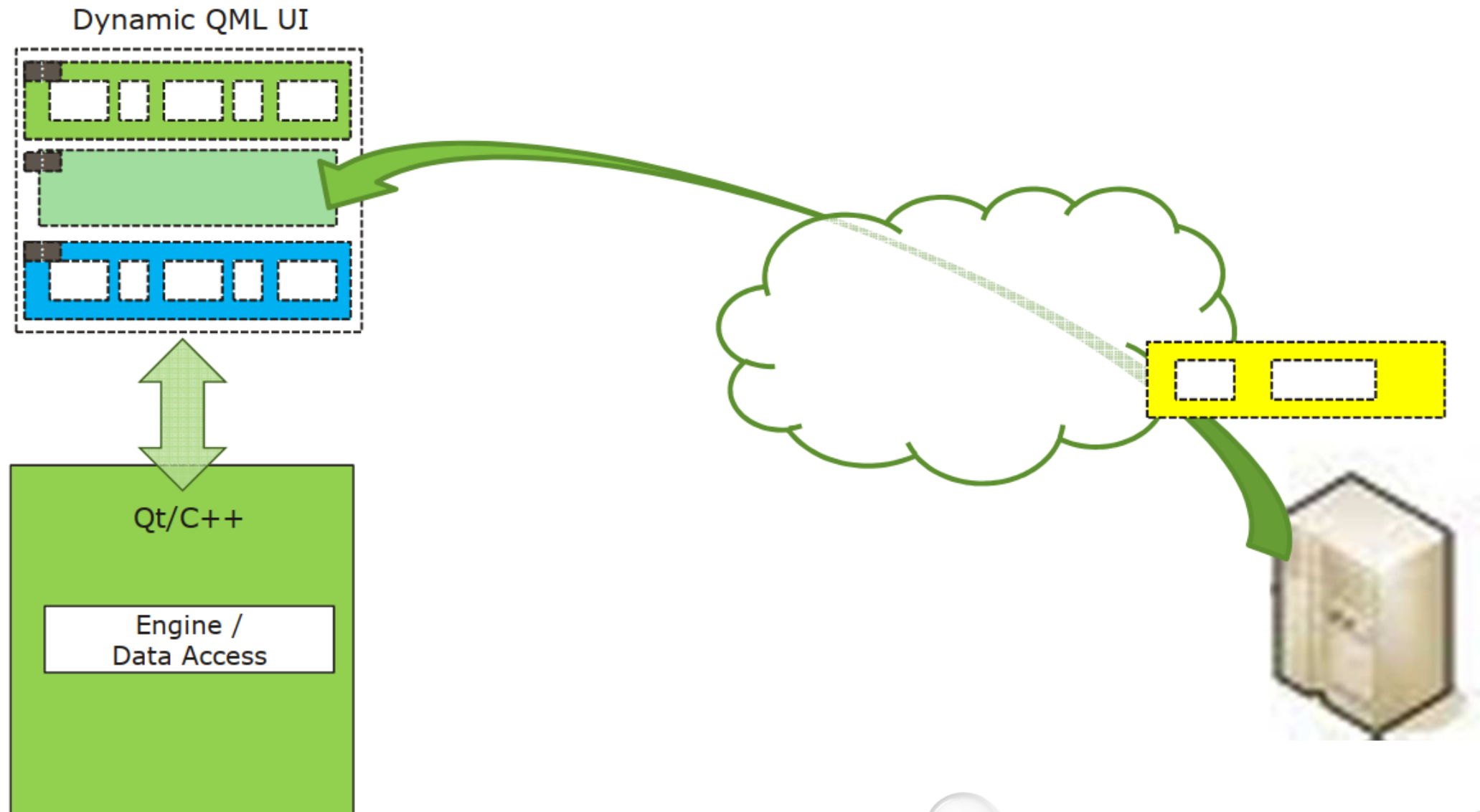
Car MF 2 branded



Platform Specific Functionality



N-Screen Approaches – Partially Remote UI



N-SCREEN APPROACHES – REMOTE UI

Blackberry

Android

iOS

Desktop

<5%

Small screen

Medium screen

Large screen

<5%

Common UI (Quick)

20-50%

Application logic (C++ or JS)

50-80%

Qt

Thank
you!

The image features a central graphic with the words "Thank you!" in a bold, bubbly, 3D-style font. The word "Thank" is rendered in a pink-to-orange gradient, while "you!" is in a blue-to-green gradient. The letters have thick black outlines and are surrounded by several yellow five-pointed stars and colorful, striped ribbons in shades of pink, orange, and blue. The entire graphic is set against a white background decorated with white water droplets of various sizes, some with soft shadows, giving it a clean, fresh, and celebratory feel.