

KDDockWidgets

A framework for custom tailored docking systems

A graphic consisting of several blue triangles of different sizes and shades, arranged in a cluster in the bottom left corner.

Sérgio Martins
sergio.martins@kdab.com

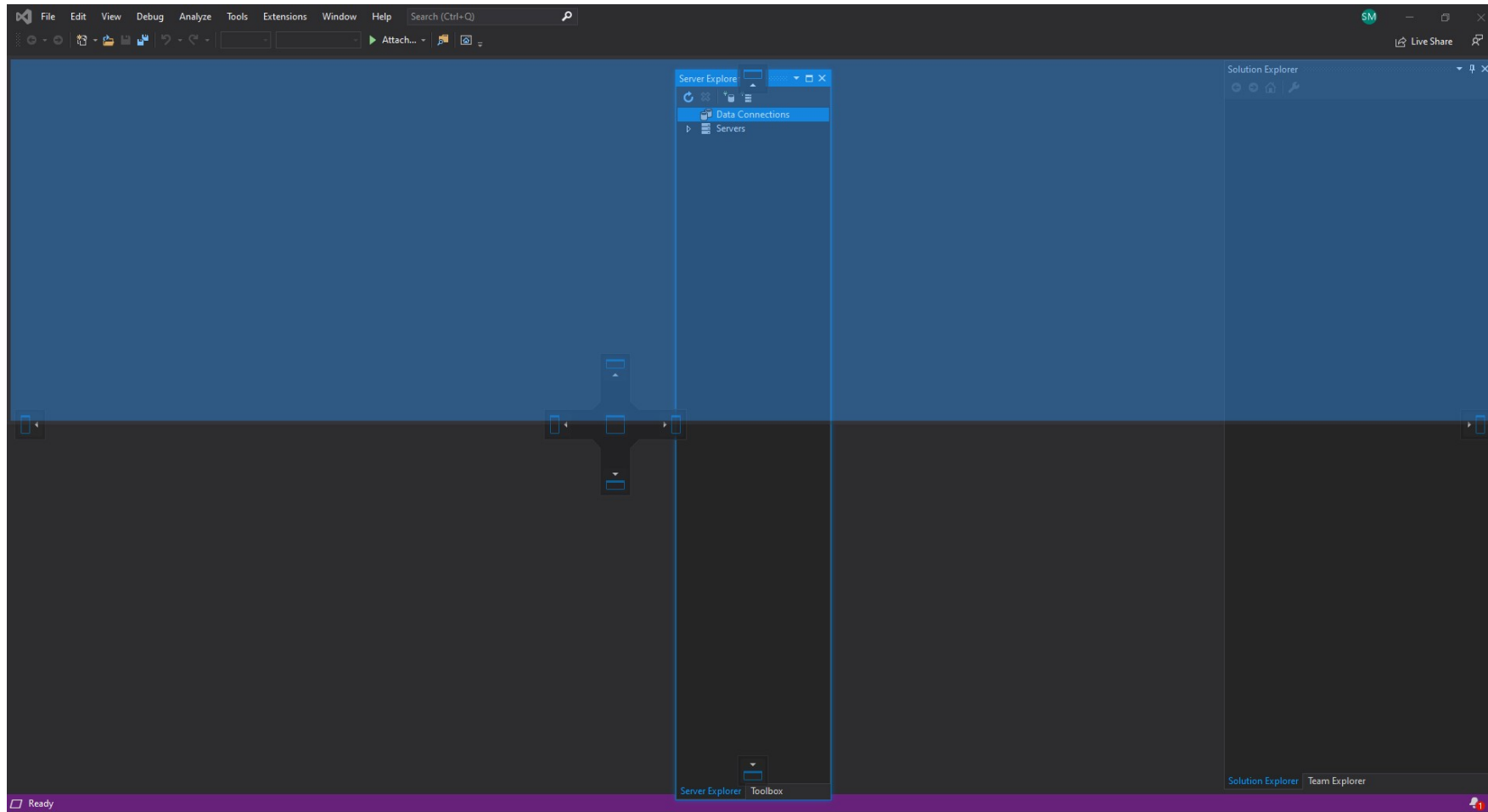
Dock Widgets ?

Dock Widgets ?

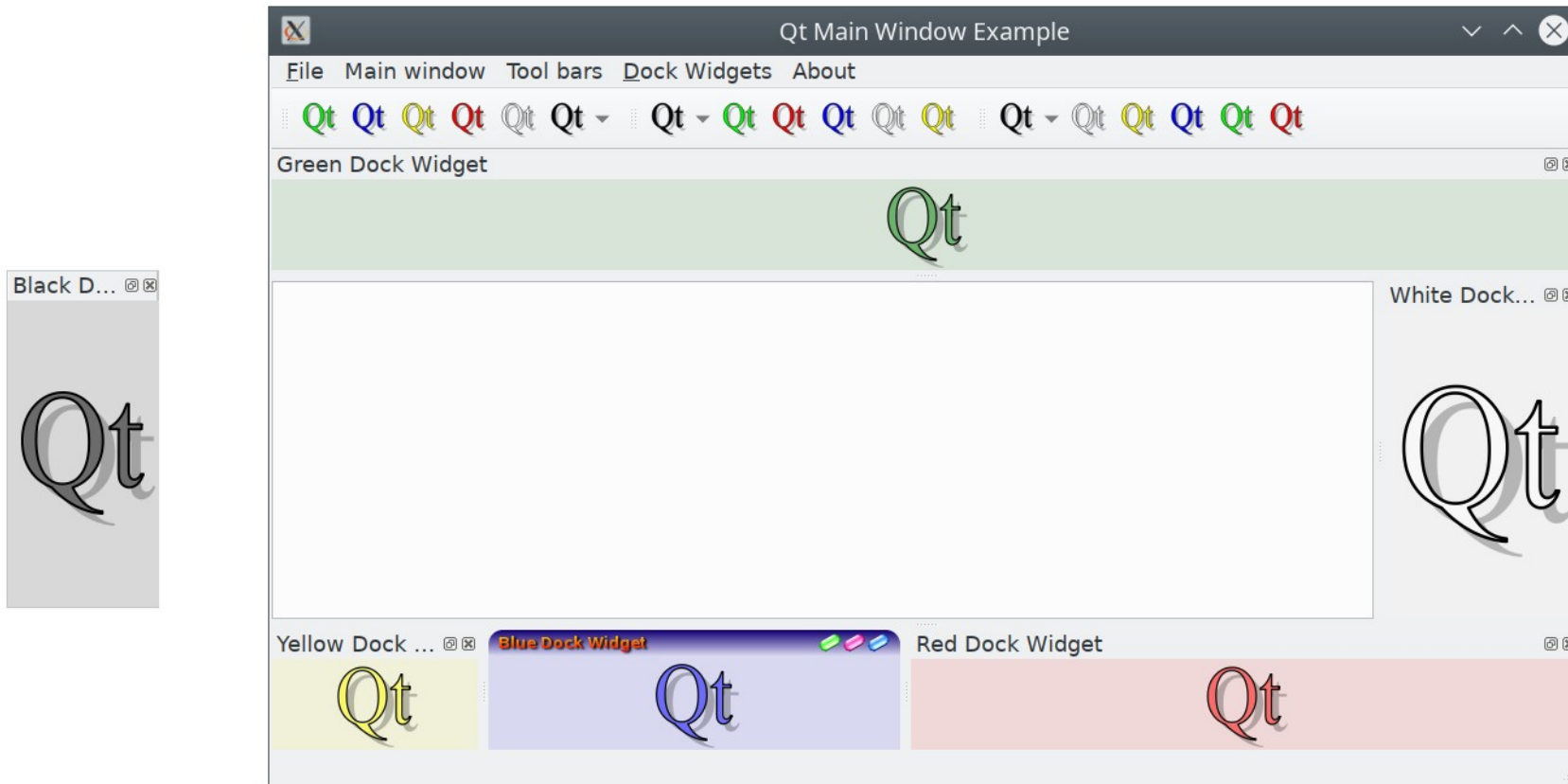
- Docking panes, utility windows, palette windows, floating palettes
- Implemented in Qt with QDockWidget + QMainWindow
- Putting windows inside windows



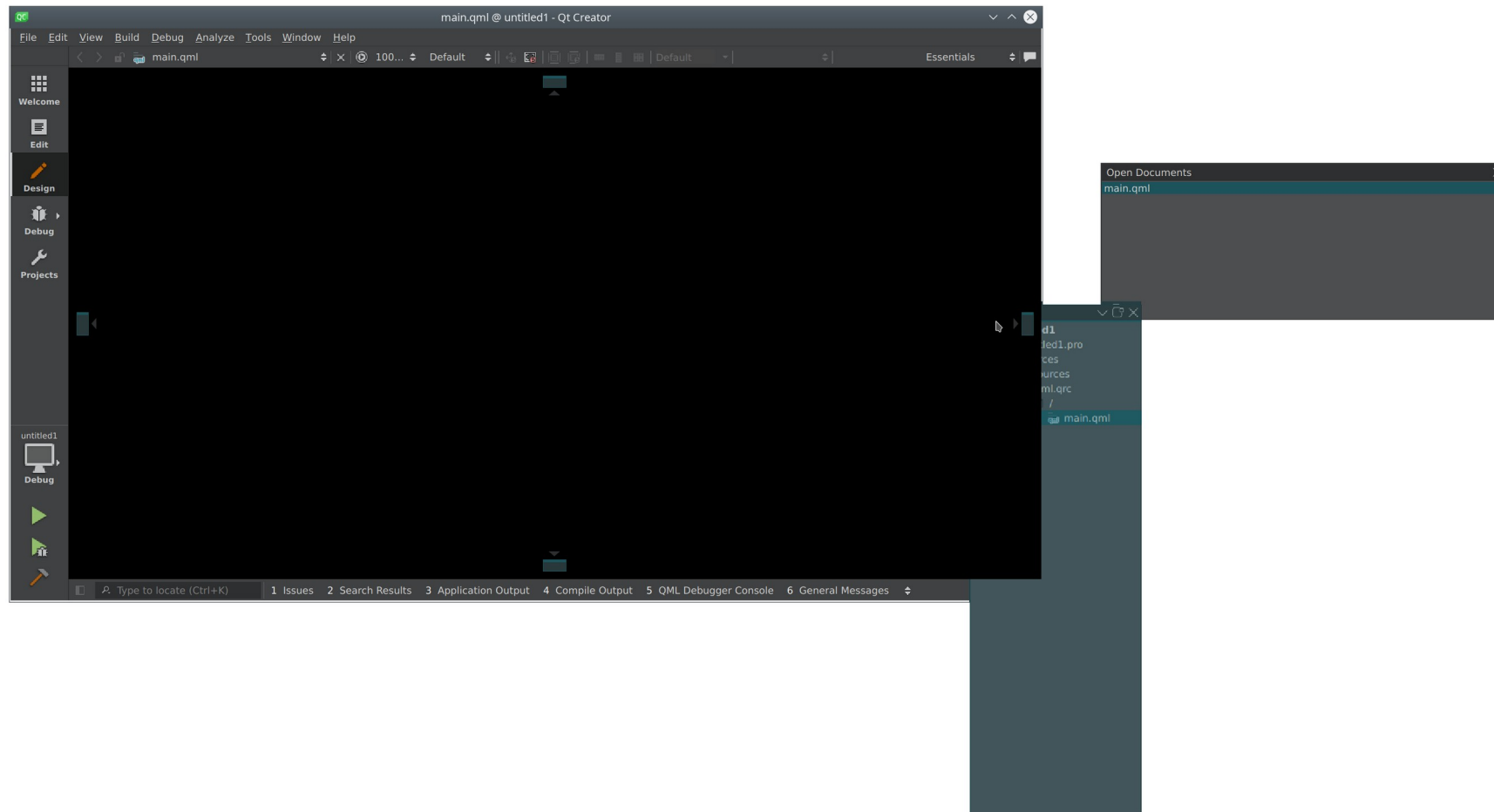
Visual Studio



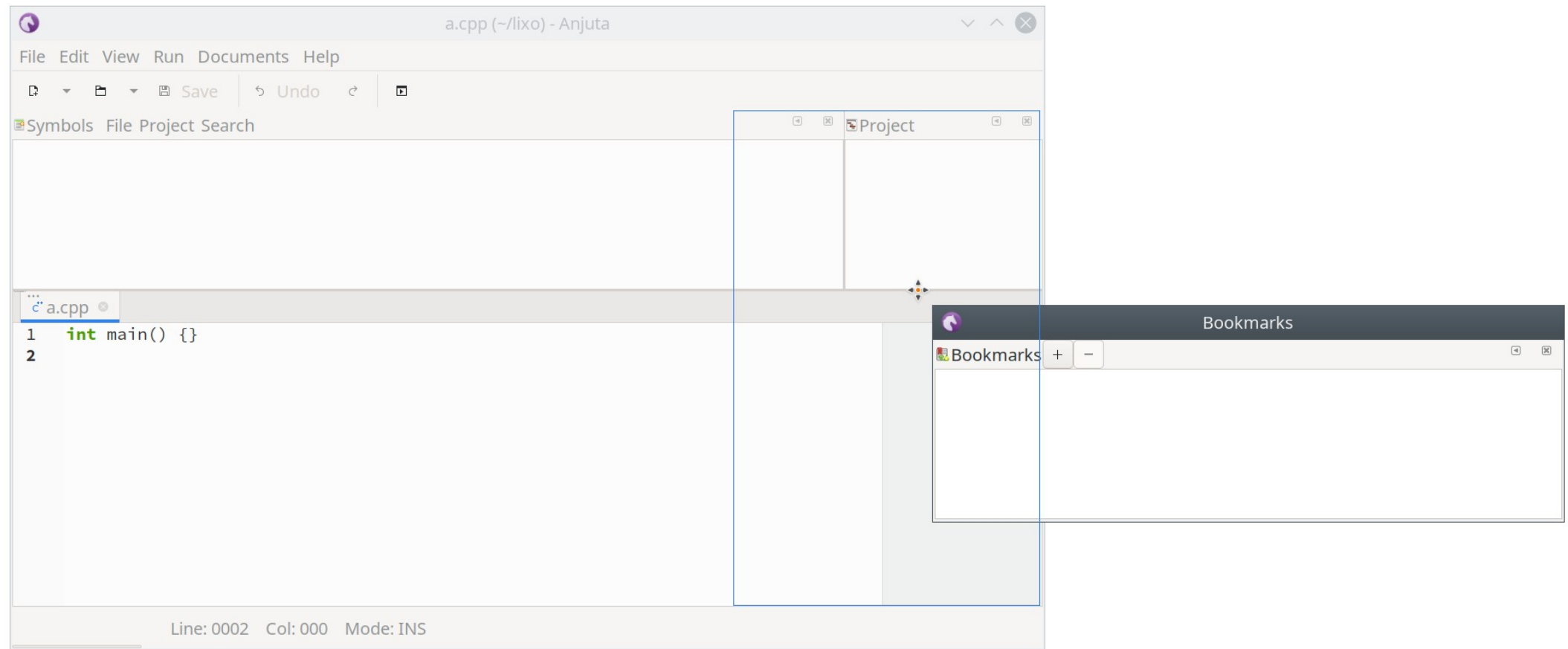
QDockWidget



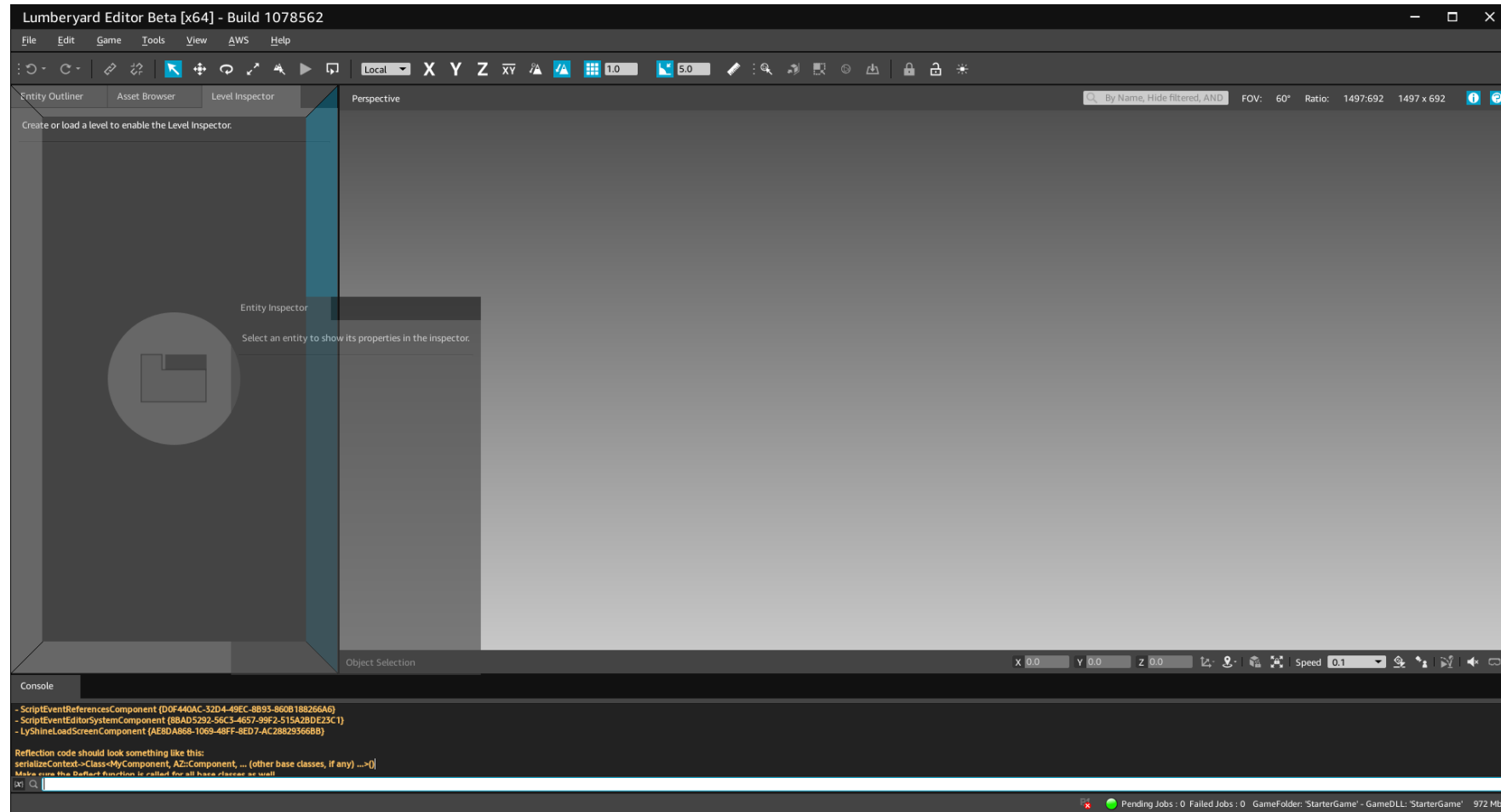
QtCreator 4.13 – QtQuick designer



Anjuta IDE (GTK)



Amazon Lumberyard (Game Engine)



Introducing KDDockWidgets

KDAB R&D

- Clazy, Heaptrack, GammaRay, KDTToolBox, Hotspot
- Kuesa, Qt3D features, Qt bugfixing
- <https://github.com/KDAB>
- <https://www.kdab.com/category/blogs/>
- KDDockWidgets



History

- Started in 2018
- Got to 90% of KDDW done in a week

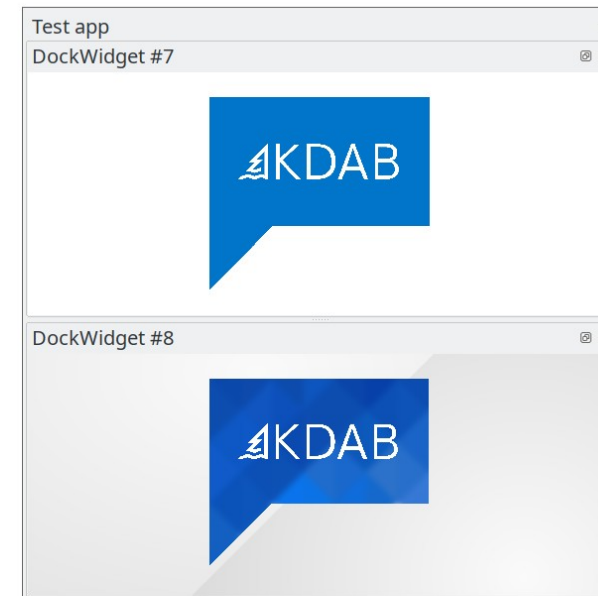
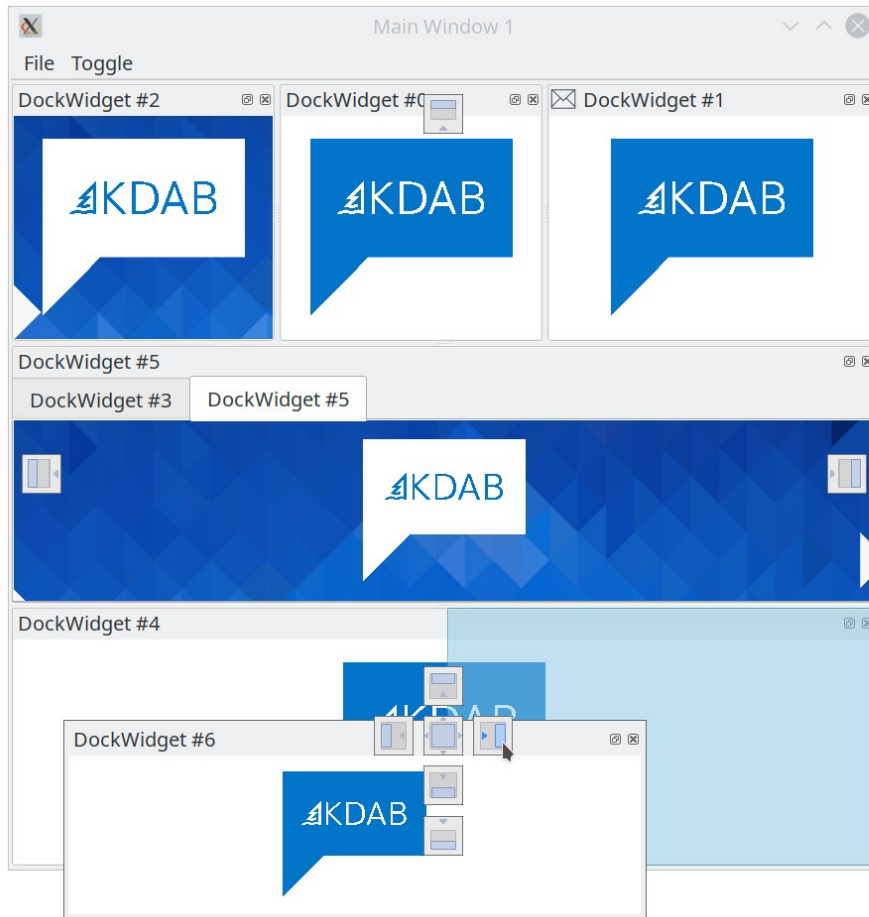
History

- Started in 2018
- Got to 90% of KDDW done in a week
- But the remaining 10% took 1 year

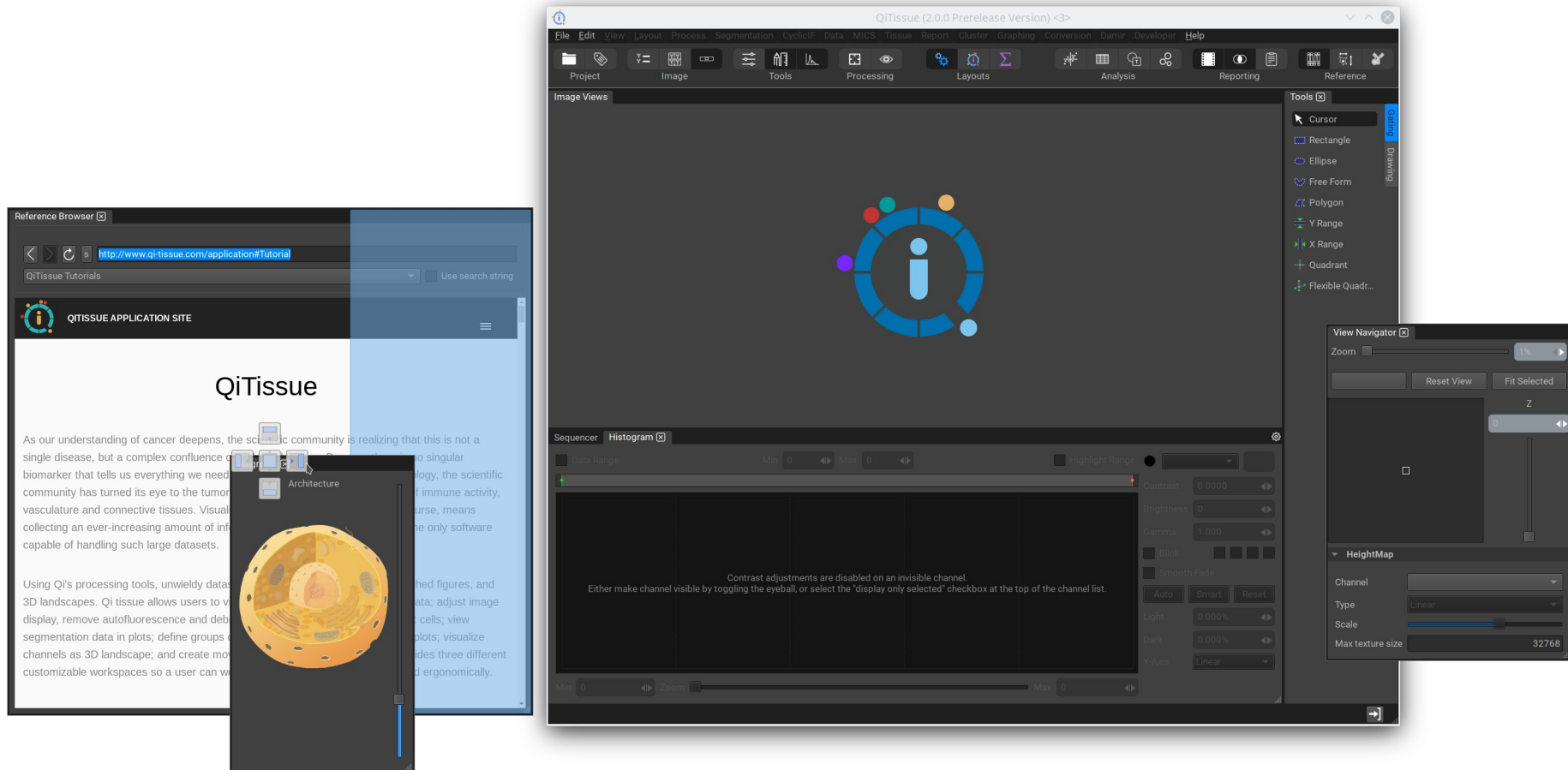
History

- Started in 2018
- Got to 90% of KDDW done in a week
- But the remaining 10% took 1 year
- Pretty stable, being used in production

KDDockWidgets – example app



KDDockWidgets in production: QiTissue



Introducing KDDockWidgets

- <https://github.com/KDAB/KDDockWidgets>
- GPLv3
- Contact info@kdab.com for commercial/custom license

What's missing in QDockWidget and why can't we fix it ?

QDockWidget

- Buggy
- Spaghetti code base prone to regressions (mixes GUI with logic)
 - 6000 LOC shoved into 2 source files
- Different bugs on different platforms

QTBUG-57466
[REG 5.6->5.7.0] QDockWi...

QTBUG-55639
[REG: 5.5.1->5.6.1] QDock...

QTBUG-15897
[REG] QMainWindow crash...

QTBUG-6744
[REG] When a QDockWidg...

QTBUG-42818
[REG 5.3.2->5.4] XCB: QDo...

QTBUG-86362
[Reg 5.14 to 5.15.0]Tabbe...

QTBUG-70239
[REG 5.6 - 5.9] QMainWindow...

QTBUG-47977
[REG: Qt 5.4.1-> Qt 5.4.2] F...

QTBUG-68800
[Reg 5.9->5.10] QDockWid...

QTBUG-44813
[REG 5.3.2 ->5.4] floating ...

QTBUG-46882
REG [5.4.2-5.5.0] Floating ...

QTBUG-11780
[REG] Active QDockWidge...

QTBUG-66454
[REG 5.6.1->5.10.0] QDock...

QTBUG-70571
[Reg 5.9.6->5.11.1]QDock...

QTBUG-65796
[Reg 5.7.1 -> 5.8] Setting a ...

QTBUG-69922
[REG 5.10.0->5.11.1] Can't ...

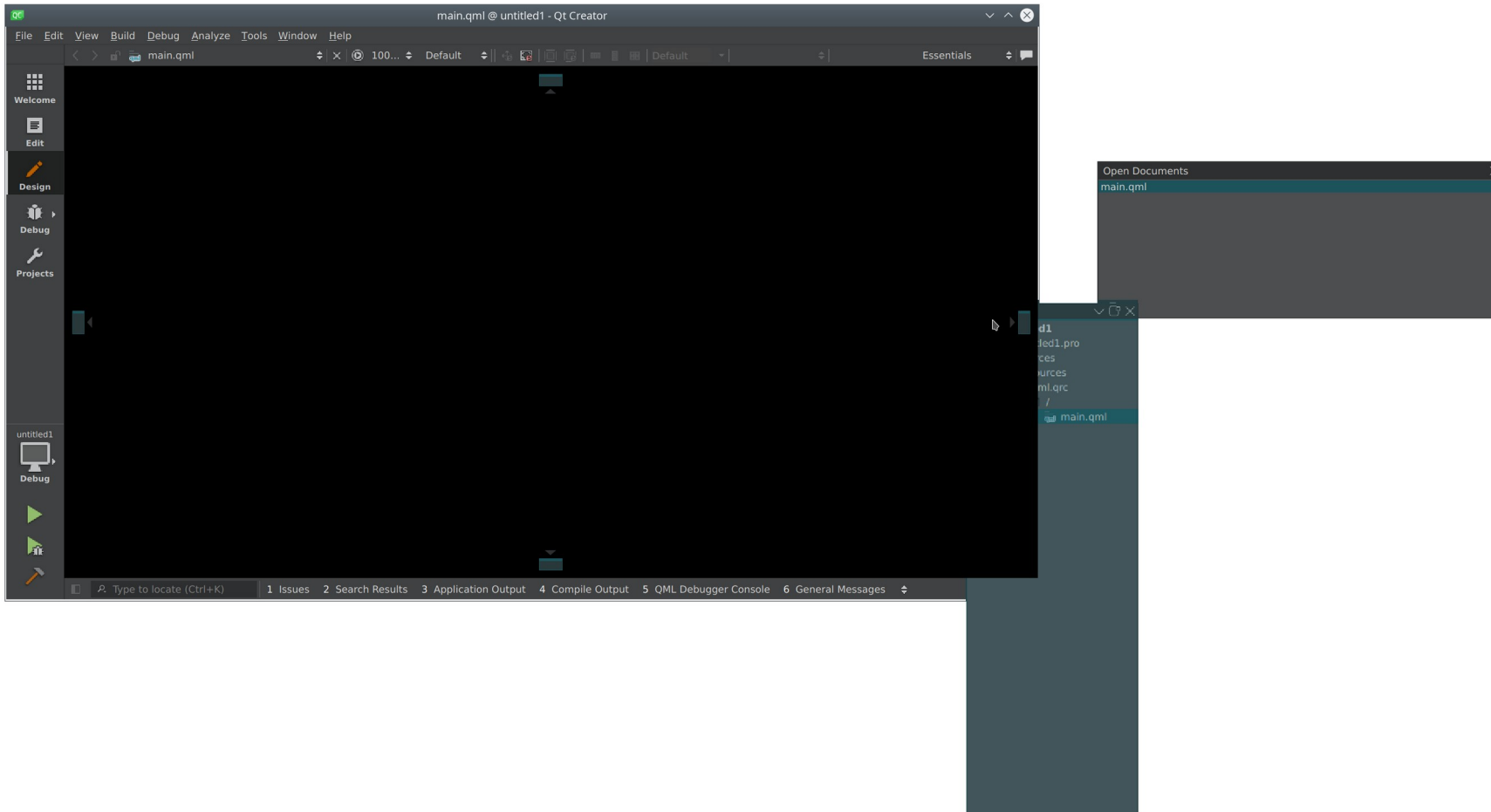
QTBUG-73805
[Reg 5.5->5.6] QDockWidg...

QTBUG-70276
Reg[5.9->5.11] font-weigh...

QDockWidget – but we needed new features

- Lots of pressure from customers for new features
- Hacks on top of QDockWidget
 - Event filter, fake mouse events, private API
 - Most of it broke with Qt version updates
 - Very stressful and working on weekends

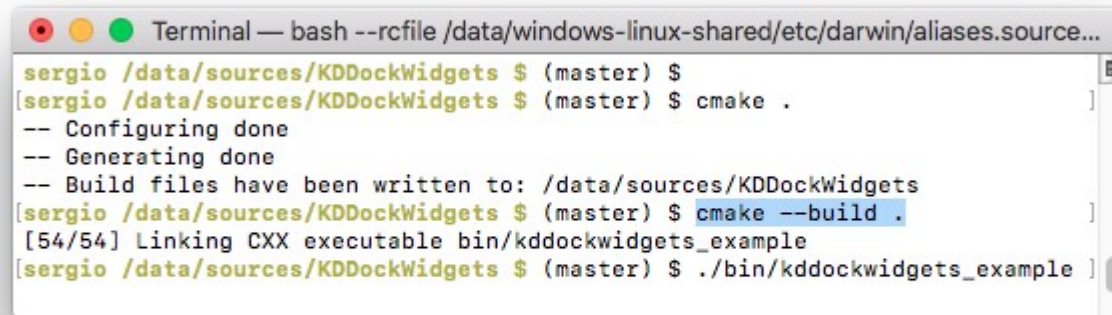
QtCreator using 3rdparty docking instead of QDockWidget



Enter KDDockWidgets (KDDW for short)

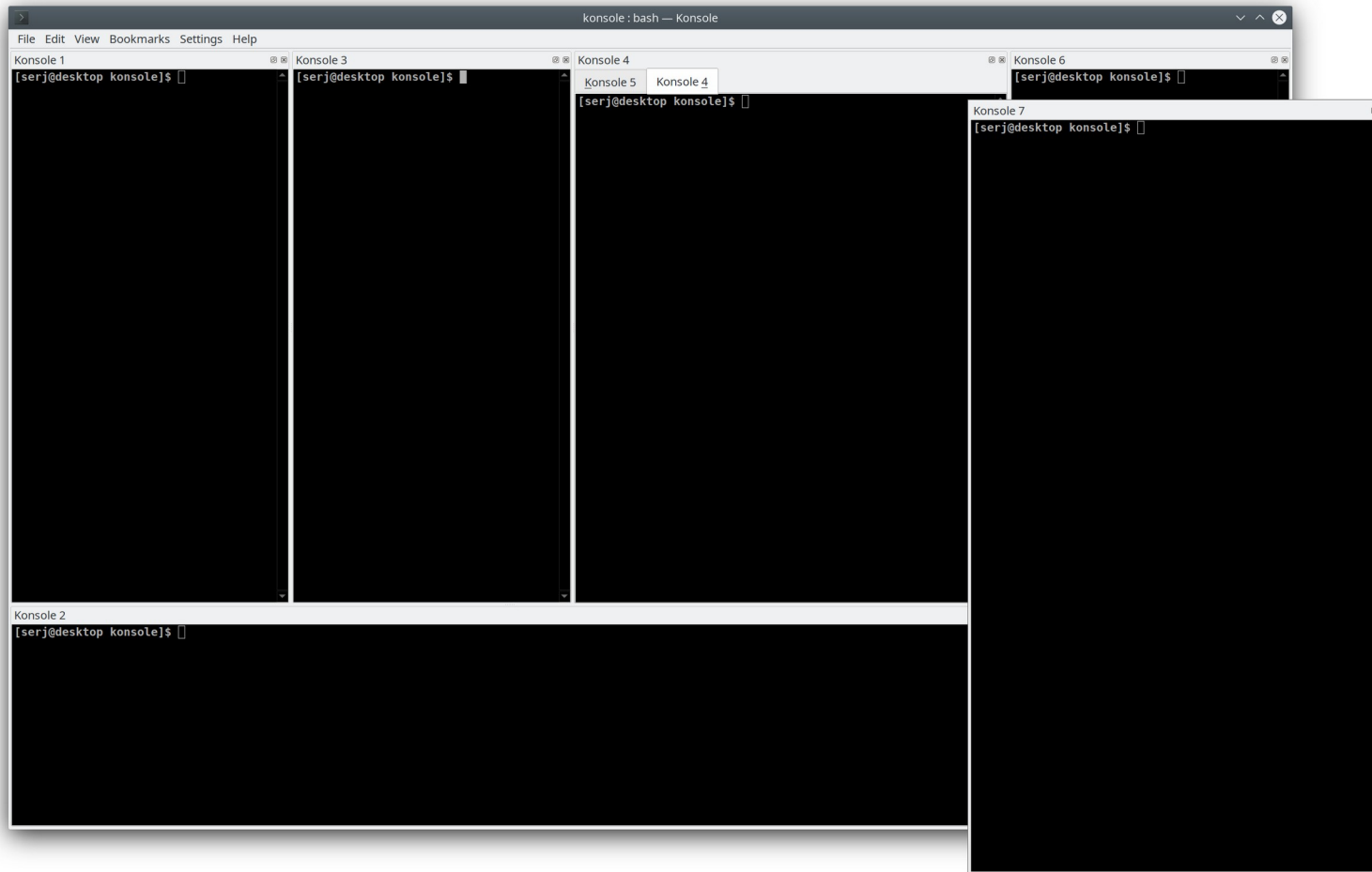
Building

- `git clone https://github.com/KDAB/KDDockWidgets.git`



```
Terminal — bash --rcfile /data/windows-linux-shared/etc/darwin/aliases.source...
sergio /data/sources/KDDockWidgets $ (master) $
[sergio /data/sources/KDDockWidgets $ (master) $ cmake .
-- Configuring done
-- Generating done
-- Build files have been written to: /data/sources/KDDockWidgets
[sergio /data/sources/KDDockWidgets $ (master) $ cmake --build .
[54/54] Linking CXX executable bin/kddockwidgets_example
[sergio /data/sources/KDDockWidgets $ (master) $ ./bin/kddockwidgets_example ]
```

Ported konsole in 10 minutes

[Show demo](#)

Minimal Hello Dock example

```
12 cmake_minimum_required(VERSION 3.7)
13 project(kddockwidgets_minimal_example)
14
15 set(CMAKE_AUTOMOC ON)
16 set(CMAKE_INCLUDE_CURRENT_DIRS ON)
17
18 find_package(KDDockWidgets REQUIRED) # // HERE
19
20 qt5_add_resources(RESOURCES_EXAMPLE_SRC ${CMAKE_CURRENT_SOURCE_DIR}/../dockwidgets/resources_example.qrc)
21
22 add_executable(kddockwidgets_minimal_example
23     main.cpp
24     ../dockwidgets/MyWidget.cpp
25     ${RESOURCES_EXAMPLE_SRC}
26 )
27
28 target_link_libraries(kddockwidgets_minimal_example PRIVATE KDAB::kddockwidget) # // HERE
```


Minimal Hello Dock example

```
14  #include <kddockwidgets/DockWidget.h>
15  #include <kddockwidgets/MainWindow.h>
```

```
30      // # 1. Create our main window
31
32      KDDockWidgets::MainWindow mainWindow(QStringLiteral("MyMainWindow"));
33      mainWindow.setWindowTitle("Main Window");
34      mainWindow.resize(1200, 1200);
35      mainWindow.show();
```

Minimal Hello Dock example

```
37 // # 2. Create a dock widget, it needs a unique name
38 auto dock1 = new KDDockWidgets::DockWidget(QStringLiteral("MyDock1"));
39 auto widget1 = new MyWidget();
40 dock1->setWidget(widget1);
```

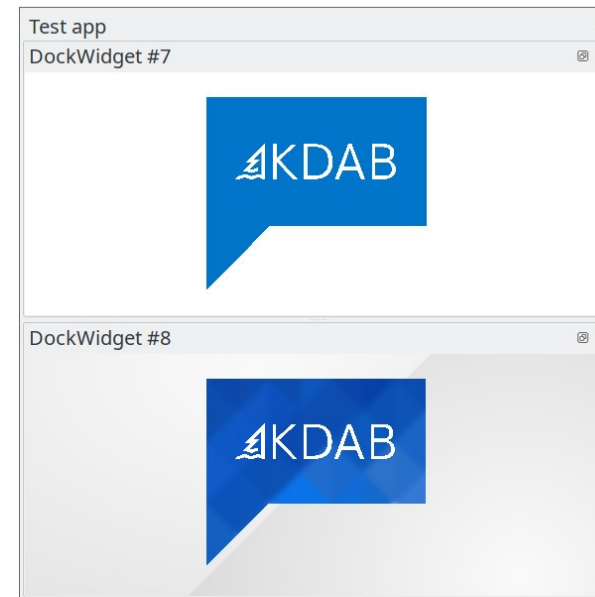
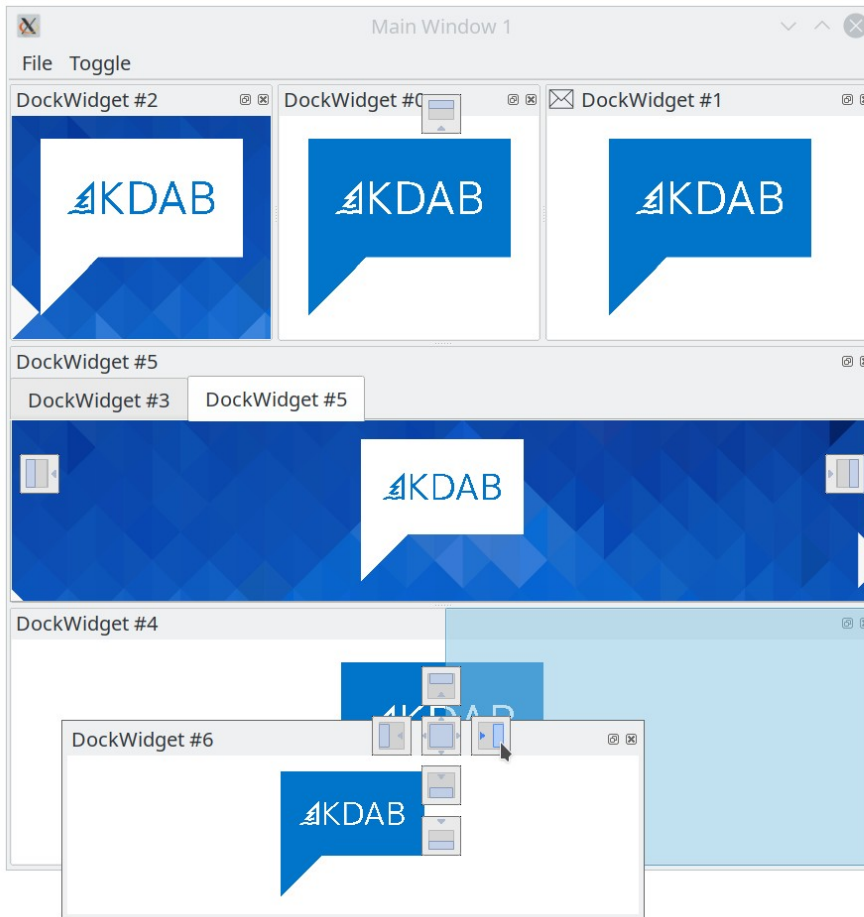
```
58 // 3. Add them to the main window
59 mainWindow.addDockWidget(dock1, KDDockWidgets::Location_OnLeft);
60 mainWindow.addDockWidget(dock2, KDDockWidgets::Location_OnTop);
61
62 // 4. Add dock3 to the right of dock2
63 mainWindow.addDockWidget(dock3, KDDockWidgets::Location_OnRight, dock2);
64
65 // 5. dock4 will be its own top level (floating window)
66 dock4->show();
```

⌚ No time for showing more code 😊

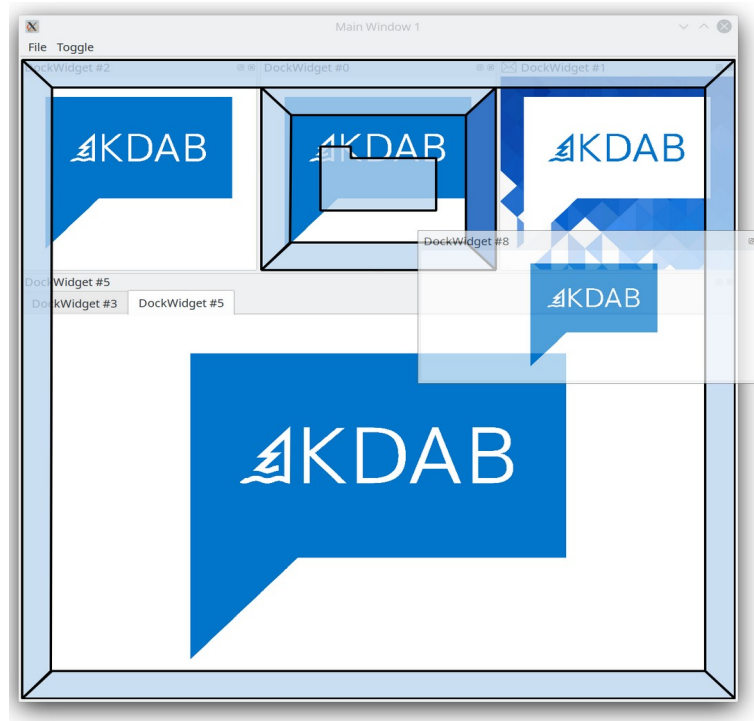
- See doxygen
 - DockWidgetBase.h
 - MainWindowBase.h
 - LayoutSaver.h
 - Config.h
 - FrameworkWidgetFactory.h
 - And everything inside private/ for hardcore usage

Must have features in a docking framework

#1 Docking indicators



#2 Custom docking indicators



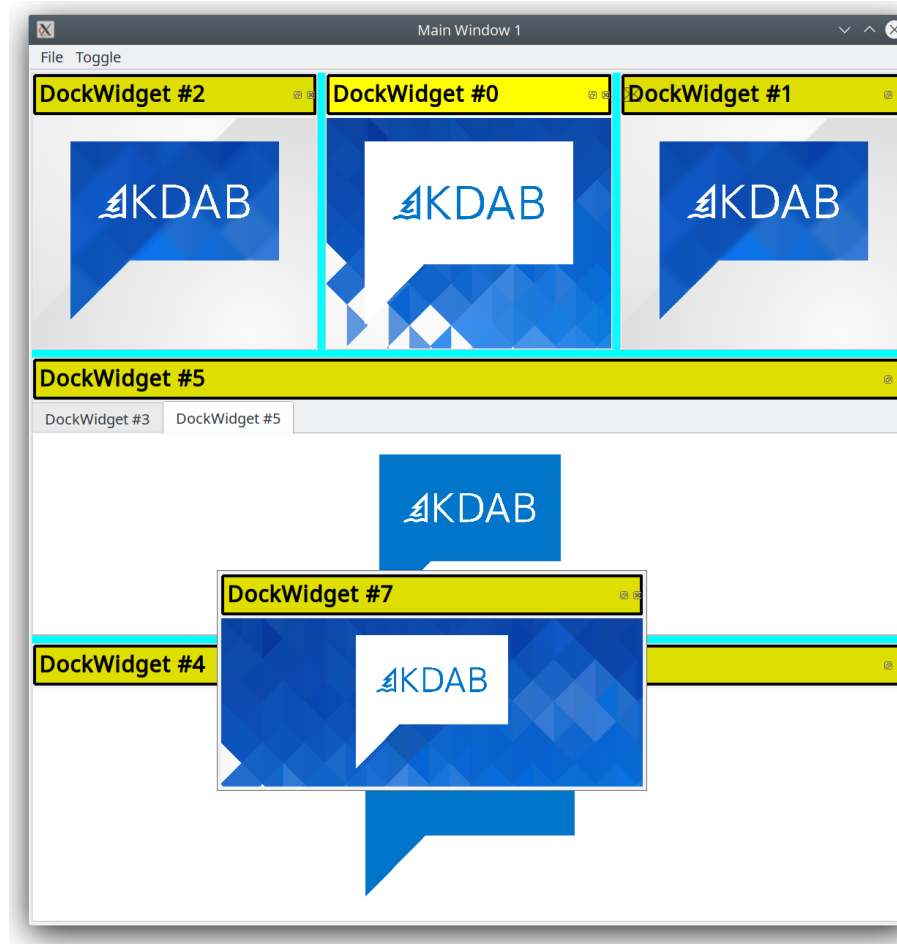
```
22 class DOCKS_EXPORT SegmentedIndicators : public DropIndicatorOverlayInterface
23 {
24     Q_OBJECT
25 public:
26     explicit SegmentedIndicators(DropArea *dropArea);
27     ~SegmentedIndicators() override;
28     void hover_impl(QPoint globalPos) override;
```

#3 Full drop area flexibility

- Dock to central widget (no concept of central widget)
- Dock to another floating window (not only the main window)
- Arbitrary number of rows or columns in any combination of nesting

[Show demo](#)

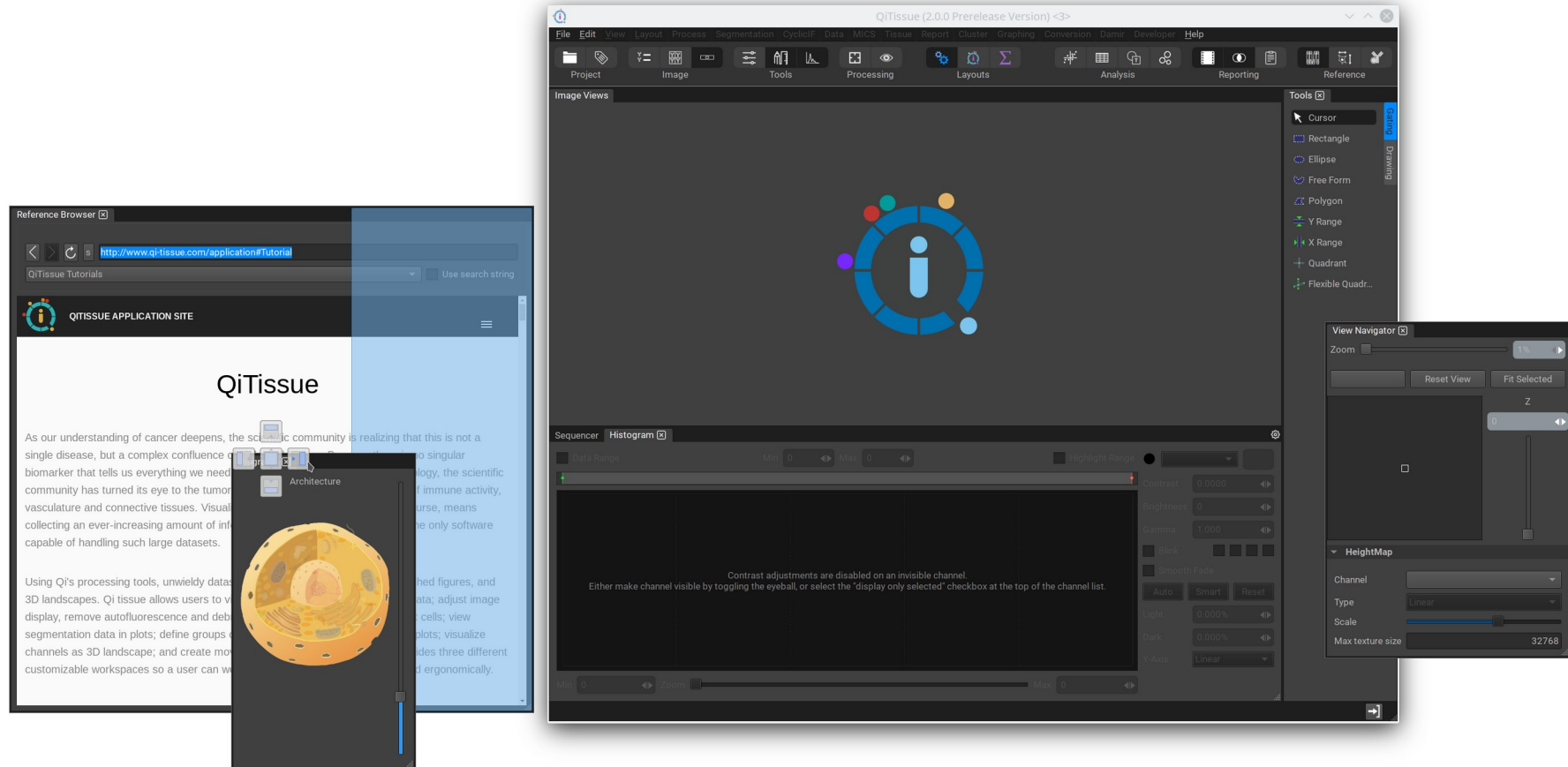
#4 Customization of every internal widget



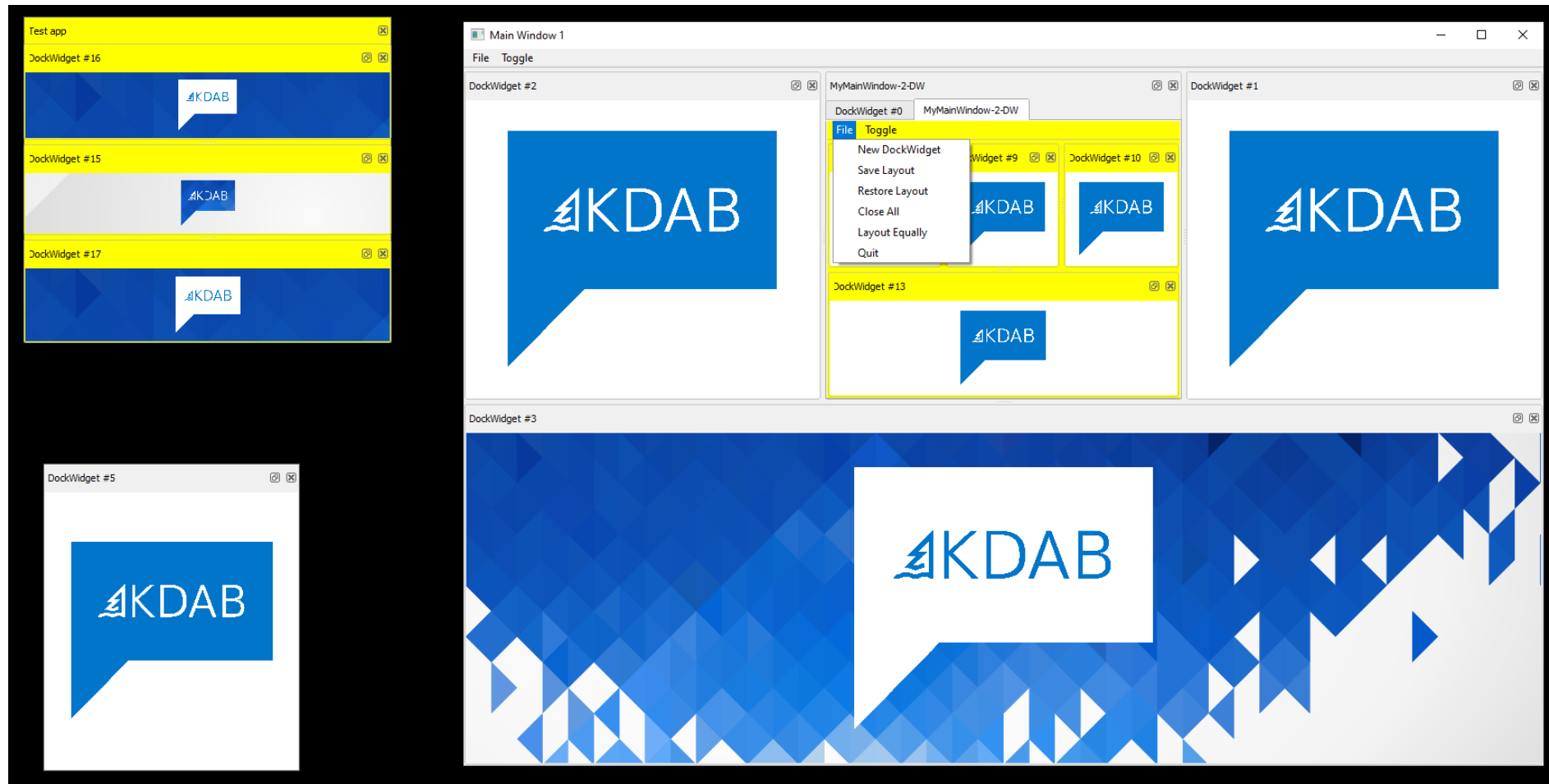
#4 Customization of every internal widget (cont.)

- QStyle still works, but more limited
- Add a button to a QTabBar
- But what's better than a custom title bar ?

#5 Having no title bar

[Show demo](#)

#6 MainWindows within MainWindows

[Show demo](#)

#7 Focus Scope

- Popular in QML
- QWidget focus is per Window
- But dock widgets are kind of an intermediate between top-level and a child widget, would be nice to keep track of focus
- Not too hard to implement...

[Show demo](#)

#8 Supporting Windows Aero-snap

- Many docking systems miss this
- Linux ??
- macOS: 5.15.1, due to QTBUG-85105



Qt / QTBUG-85105

QWindow::startSystemMove() does not work

#9 Layout saving/loading

- Stored in JSON
- Developers can ship layout presets
- Restore can use relative sizes so it looks good on bigger/smaller screens
- Partial save/restore

#10 Misc

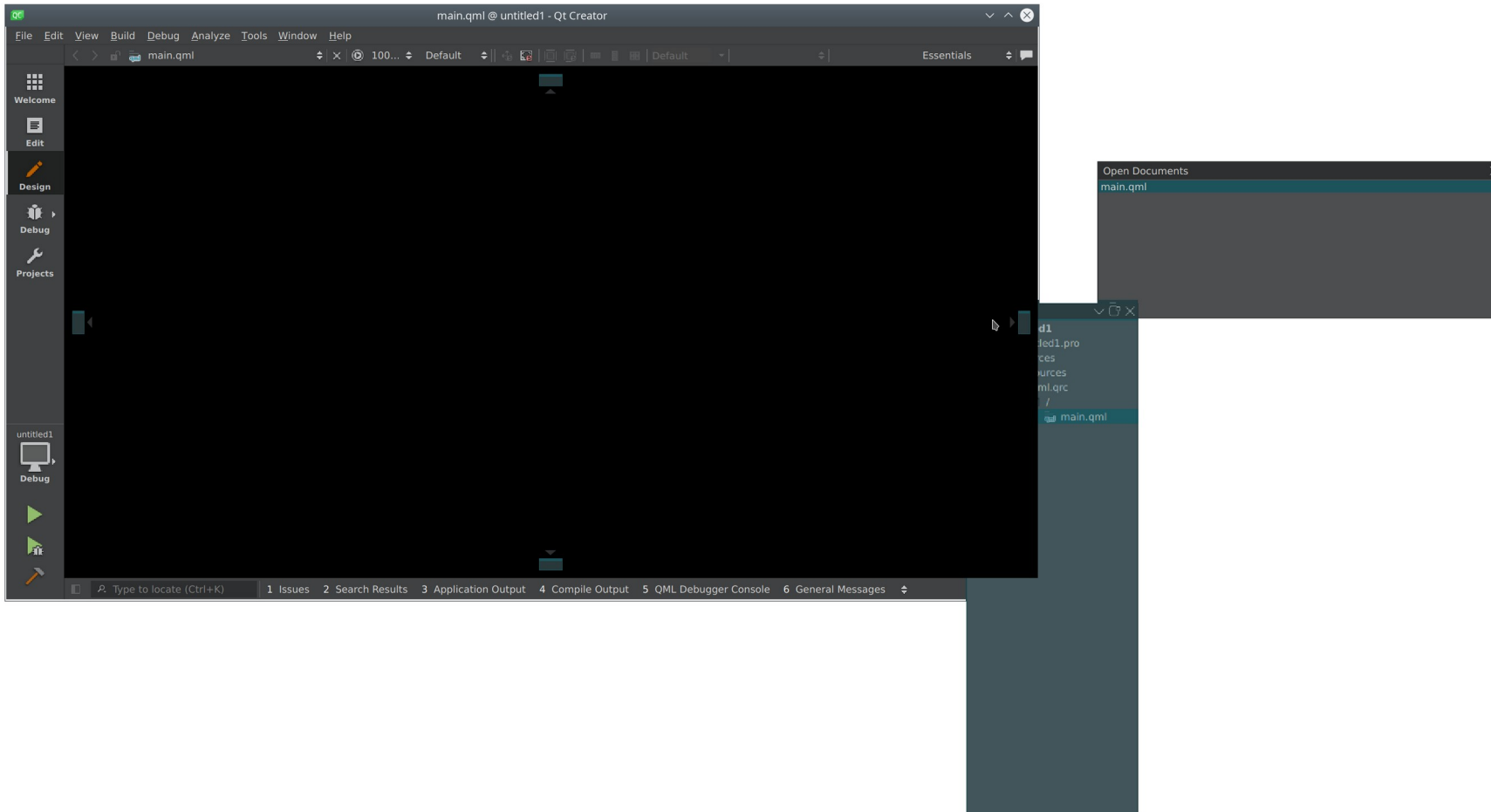
- Python bindings (thanks renato.araujo@kdab.com)
- Having a close button on each tab
- Lazy resize
- Allow dock widgets to be non-closable, or non-dockable
- Double-click to maximize
- and more...

Be careful with OS differences! (Or use KDDockWidgets)

#1 Please support Window Managers with no compositing

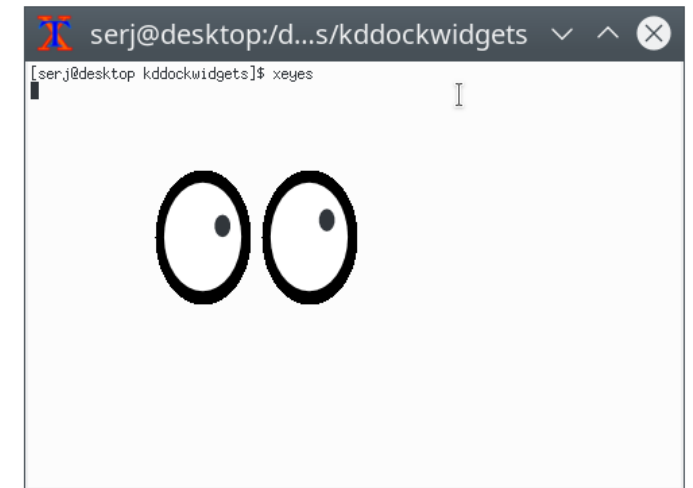
- Mostly a concern in X11
 - Sometimes no support for translucent windows (In the sense of `QWindow::setOpacity()`)
- Bad graphics drivers, virtual machines, old hardware
- `QX11Info::isCompositingManagerRunning()`

#1 QtCreator “QML Designer” showing black



#1 Compositing (cont.)

- Windows: Always enabled on Windows 10
- Solution: Use `QWidget::setMask()` instead of `setOpacity()`
- X11 doesn't need fancy compositing



#2 Doing proper custom title bars (aka non-native stuff)

- ...without killing the native window frame
- Makes the window very hard to resize
 - QDockWidget: QWidgetResizeHandler
 - KDDW:Linux|macOS: own global event filter with proper margin
 - KDDW:Windows win32 code
 - Maybe QWindow::startSystemResize() nowadays (kinda)



#2 Doing proper custom title bars (cont.)

- don't kill screen edge snapping
 - Called “Aero-snap” on Windows
 - KDDW: win32 API

#2 Doing proper custom title bars (cont.)

- Don't kill drop shadow
 - Windows: Not implemented but doable
 - Linux: maybe don't bother. Or do it yourself in the client area + `Qt::FramelessWindowHint`.

#3 Wayland

- No support
- Missing ways to know if there's a drop area below
- Or even basic things like moving and resizing a window programmatically (`QWindow::startSystemMove()`)
- Might need a wayland extension

Layouting Engine

Layouting Engine

- Like QSplitter but on steroids
- Recursively-nested multi-splitter
- min/max/size-policy constraints, resizing windows
- Extensible with fancy features
- `kddockwidgets/src/private/multisplitter/Item.cpp`

Layouting Engine

- Not based QWidget or QLayout
- 5000 LOC of pure QtCore, reusable code, hence a separate library
 - libkddockwidgets_multisplitter.so
 - libkddockwidgets.so

QtQuick/QML Support ?

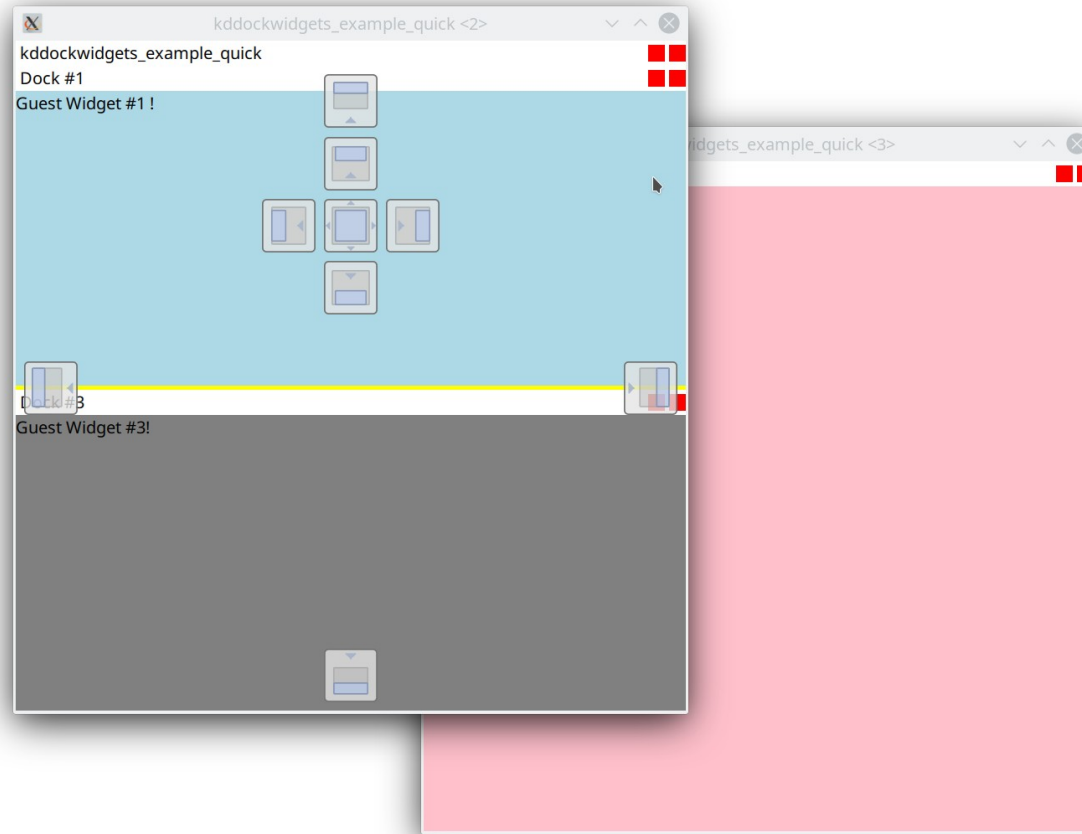
QtQuick/QML support ?

- Yes, but without having a separate codebase
- Reuse “90%” of the existing code and only write glue
- Adapters, wrappers, .qml for GUI
- Reuse the existing tests (6k lines of testing code)

QtQuick/QML support ?

- Most code is in classes that don't depend on QWidget
- QWidget specific classes are small
- Layouting engine is already separate from all the docking.
Fully reusable. (5k)

QtQuick/QML support ?



Sleeping good at night

Sleeping good at night

- 200 automated tests
 - Drag windows around, DND
 - 6k LOC (30% of the code)
 - Prevented loads of regressions
- Fuzzer caught some crashes
- Running with ASAN and crazy checks in the CI

Sleeping good at night

- Tests that depends on focus are flaky
 - Might fail you get a notification
- Offscreen QPA
 - `./my_app -platform offscreen`
 - parallelization your machine has 8 cores but only 1 mouse cursor
 - `ctest -j8`

Roadmap

Roadmap

- QtQuick
 - Maybe in December
- QtQuick on EGLFS
 - No multi-window support
 - Use regular QQuickItems with higher Z
- Wayland ??
- You tell me

Final Words

Final words

- Fun project to work on
 - Big features can be done with small patches in a few hours
 - Bug-fixes with oneliners
 - BIG focus on Quality
 - 200 tests gives a good sense of stability
 - It's more about polished fancy features now



– Pulp Fiction

Thank you! - Questions ?