

QtQuick under the Hood

Thomas McGuire
thomas@kdab.com



- 1) QML Engine
- 2) Optimizations

- 1) QML Engine
- 2) Optimizations

1) QML Engine

QML File Loading

```
import QtQuick 2.0

Rectangle {
    color: "lightsteelblue"
    Text {
        anchors.centerIn: parent
        text: "Hello DevDays!"
    }
}
```

```
import QtQuick 2.0

Rectangle {
    color: "lightsteelblue"
    Text {
        anchors.centerIn: parent
        text: "Hello DevDays!"
    }
}
```

QML elements are C++ classes (QObject)

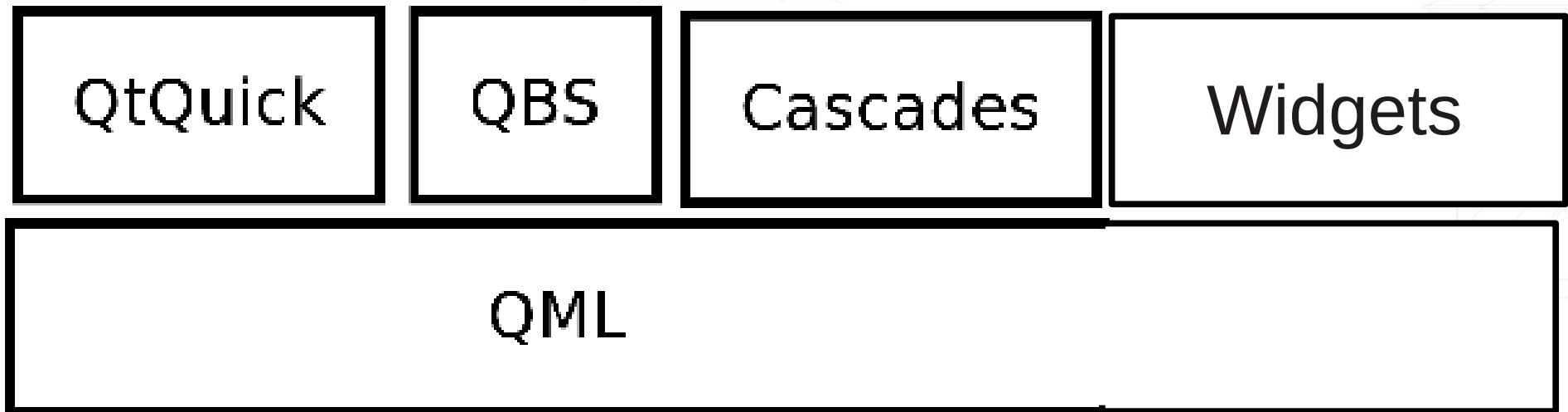
```
import QtQuick 2.0

Rectangle {
    color: "lightsteelblue"
    Text {
        anchors.centerIn: parent
        text: "Hello DevDays!"
    }
}
```

QML properties are QObject properties

- Elements: `qmlRegisterType()`
- Properties: `Q_PROPERTYs` from MOC
 - Also: Signals and Slots
- QML Parser
 - instantiates classes
 - assigns properties

- QML is UI-agnostic
- QQuickView
 - uses QQmlEngine to create root object
 - renders root object with scenegraph engine



1) QML Engine

QML / C++ Interaction

•QML

```
Rectangle {  
    objectName: "myRect"  
    width: 300  
    property int area: width * height  
    function sayHello()  
        { console.log("Hello"); }  
}
```

•C++

```
QObject *rect = rootObject->findChild("myRect");  
qDebug() << rect->property("width")  
         << rect->property("area");  
QMetaObject::invokeMethod(rect, "sayHello");
```

Don't

myr

Object	Type
0x639610	QQuickRootItem
myRect	QQuickRectangle_QML_0

Static Properties Dynamic Properties Methods Inbound Connections

Search

Property	Value	Type	Class
activeFocus	false	bool	QQuickItem
anchors	0x2e4e440 (...)	QQuickAnch...	QQuickItem
antialiasing	false	bool	QQuickItem
baseline		QQuickAnch...	QQuickItem
baselineOffset	0	double	QQuickItem
border	0x2c4f5b0 (...)	QQuickPen*	QQuickRectangle
bottom		QQuickAnch...	QQuickItem
children		QQmlListPro...	QQuickItem
childrenRect	0x0 400x400	QRectF	QQuickItem
clip	false	bool	QQuickItem
color	#b0c4de	QColor	QQuickRectangle
data		QQmlListPro...	QQuickItem
enabled	true	bool	QQuickItem
focus	false	bool	QQuickItem
gradient		QQuickGradi...	QQuickRectangle
height	400	double	QQuickItem
horizontalCenter		QQuickAnch...	QQuickItem
implicitHeight	0	double	QQuickItem
implicitWidth	0	double	QQuickItem
layer	0x2e6d800 (...)	QQuickItemL...	QQuickItem
left		QQuickAnch...	QQuickItem
objectName	myRect	QString	QObject
opacity	1	double	QQuickItem
parent	0x639610 (Q...	QQuickItem*	QQuickItem
pos	0x0	QPointF	QQuickItem
radius	0	double	QQuickRectangle
resources		QQmlListPro...	QQuickItem
right		QQuickAnch...	QQuickItem
rotation	0	double	QQuickItem
scale	1	double	QQuickItem
smooth	true	bool	QQuickItem
state		QString	QQuickItem
states		QQmlListPro...	QQuickItem
test	42	int	QQuickRectangle_QML_0
top		QQuickAnch...	QQuickItem
transform		QQmlListPro...	QQuickItem
transformOrigin	Center	TransformOri...	QQuickItem
transformOriginPoint	200x200	QPointF	QQuickItem
transitions		QQmlListPro...	QQuickItem
verticalCenter		QQuickAnch...	QQuickItem
visible	true	bool	QQuickItem
visibleChildren		QQmlListPro...	QQuickItem
width	400	double	QQuickItem



myr

Object	Type
0x639610	QQuickRootItem
myRect	QQuickRectangle_QML_0

Static Properties Dynamic Properties Methods Inbound Connections

Search

Property	Value	Type	Class
activeFocus	false	bool	QQuickItem
anchors	0x2e4e440 (...)	QQuickAnch...	QQuickItem
antialiasing	false	bool	QQuickItem
baseline		QQuickAnch...	QQuickItem
baselineOffset	0	double	QQuickItem
border	0x2c4f5b0 (...)	QQuickPen*	QQuickRectangle
bottom		QQuickAnch...	QQuickItem
children		QQuickListPro...	QQuickItem
childrenRect	0x0 400x400	QRectF	QQuickItem
clip	false	bool	QQuickItem
color	#b0c4de	QColor	QQuickRectangle
data		QQuickListPro...	QQuickItem
enabled	true	bool	QQuickItem
focus	false	bool	QQuickItem
gradient		QQuickGradi...	QQuickRectangle
height	400	double	QQuickItem
horizontalCenter		QQuickAnch...	QQuickItem
implicitHeight		double	QQuickItem
implicitWidth		double	QQuickItem
layer	0x2e0d80 (...)	QQuickItemL...	QQuickItem
left		QQuickAnch...	QQuickItem
objectName	myRect	QString	QObject
opacity	1	double	QQuickItem
parent	0x639610 (Q...	QQuickItem*	QQuickItem
pos	0x0	QPointF	QQuickItem
radius	0	double	QQuickRectangle
resources		QQuickListPro...	QQuickItem
right		QQuickAnch...	QQuickItem
rotation	0	double	QQuickItem
scale	1	double	QQuickItem
smooth	true	bool	QQuickItem
state		QString	QQuickItem
states		QQuickListPro...	QQuickItem
test	42	int	QQuickRectangle_QML_0
top		QQuickAnch...	QQuickItem
transform		QQuickListPro...	QQuickItem
transformOrigin	Center	TransformOri...	QQuickItem
transformOriginPoint	200x200	QPointF	QQuickItem
transitions		QQuickListPro...	QQuickItem
verticalCenter		QQuickAnch...	QQuickItem
visible	true	bool	QQuickItem
visibleChildren		QQuickListPro...	QQuickItem
width	400	double	QQuickItem

Virtual Meta Object



1) QML Engine

JavaScript and Bindings


```
Text {
  anchors.centerIn: parent
  text: "Area: " + parent.height * parent.width
}
function sayHello() { console.log("Hello!"); }
MouseArea {
  onClicked: {
    sayHello();
    Qt.quit();
  }
}
```

- bound signal handlers, bindings, functions
- v8 + wrappers

- Property change → Binding update
- Changed signal
 - NOTIFY keyword
- captured properties connected
- declarative

- Missing Changed Signal
- Unnecessary Changed Signal
- Updates in hidden items



The Kraken

1) QML Engine

???

```
states: State {
    name: "resized";
    when: mouseArea.pressed
PropertyChanges {
    target: rect;
    color: "blue";
    height: container.height
    }
}
```

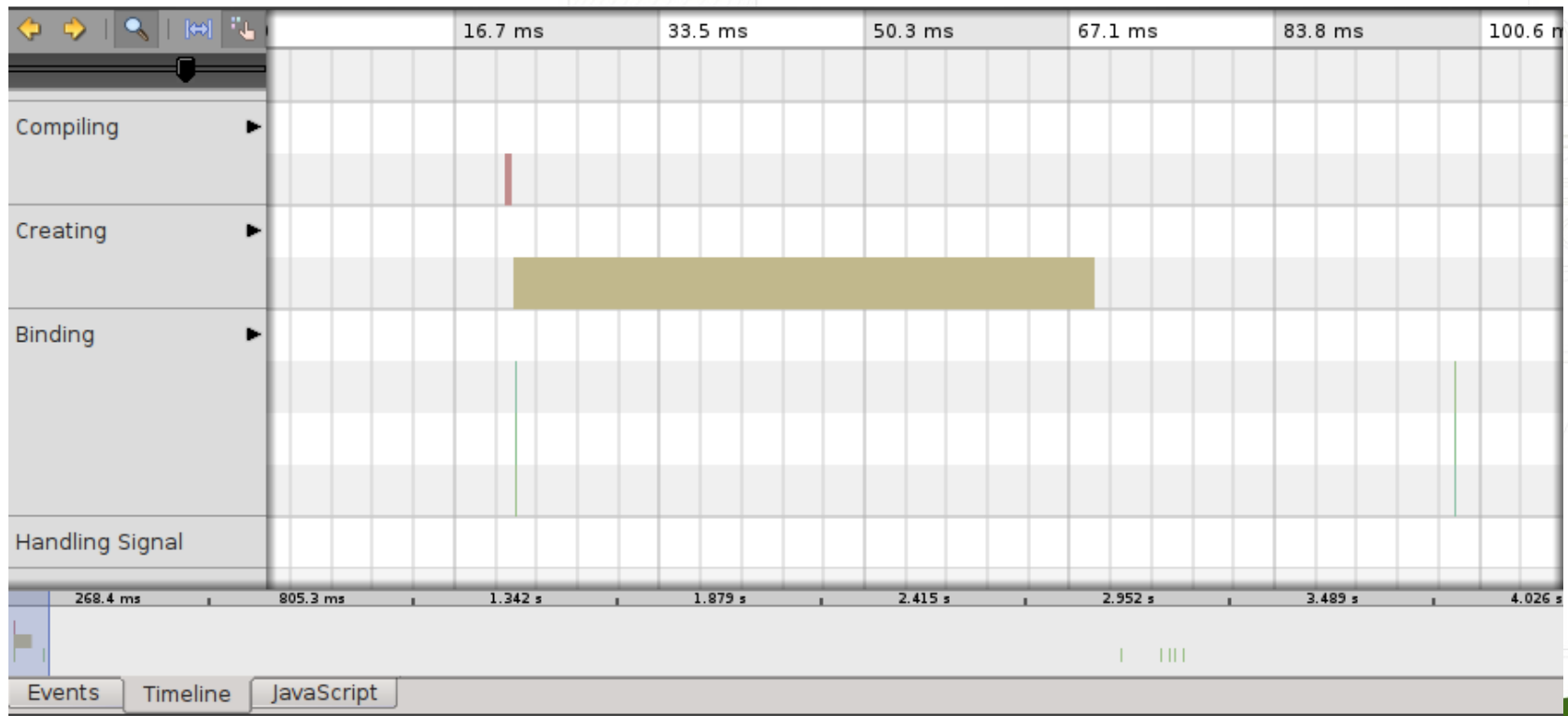
- Custom syntax
- Compile & Run
- PropertyChanges, ListModel, Connections, ...
 - Not: PropertyAnimation
- QQmlCustomParser private API

QML Engine

- QML loading
 - creates C++ objects
 - assigns to values / creates binding objects
 - creates virtual meta objects
 - creates bound signal handlers
- JS via v8 (bindings, functions, signal handlers)
- Bindings updated via NOTIFY signals
- Custom Parsers

- 1) QML Engine
- 2) Optimizations**

- Use QML Profiler
 - Bindings? Painting? Loading?



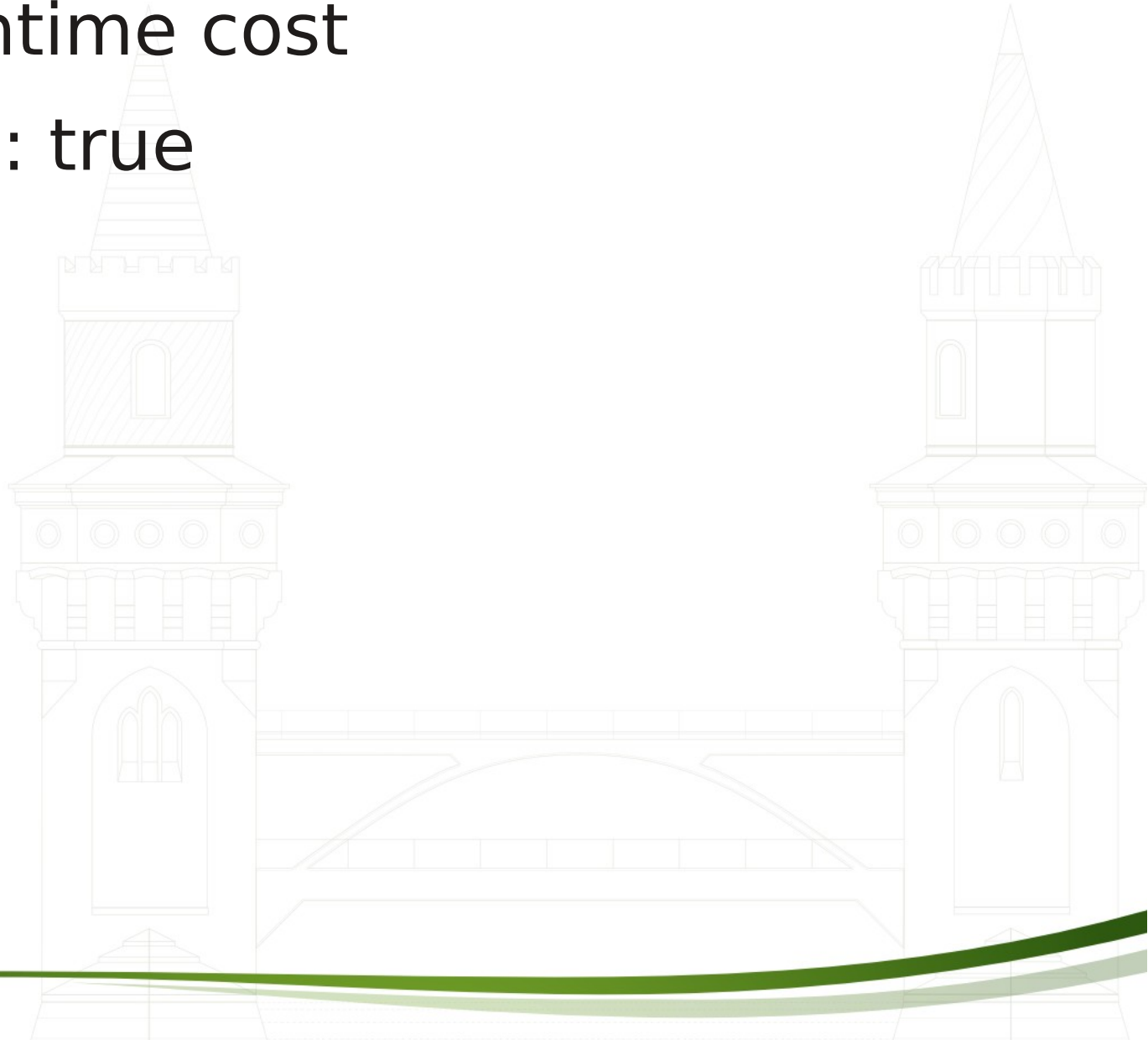
- **Virtual Machine Environment**

- 2 phase loading

- **compiling** - creates bytecode instructions
- **creating** - executes bytecode

QML_COMPILER_DUMP=1 to see VME instructions

- Startup vs runtime cost
- asynchronous: true

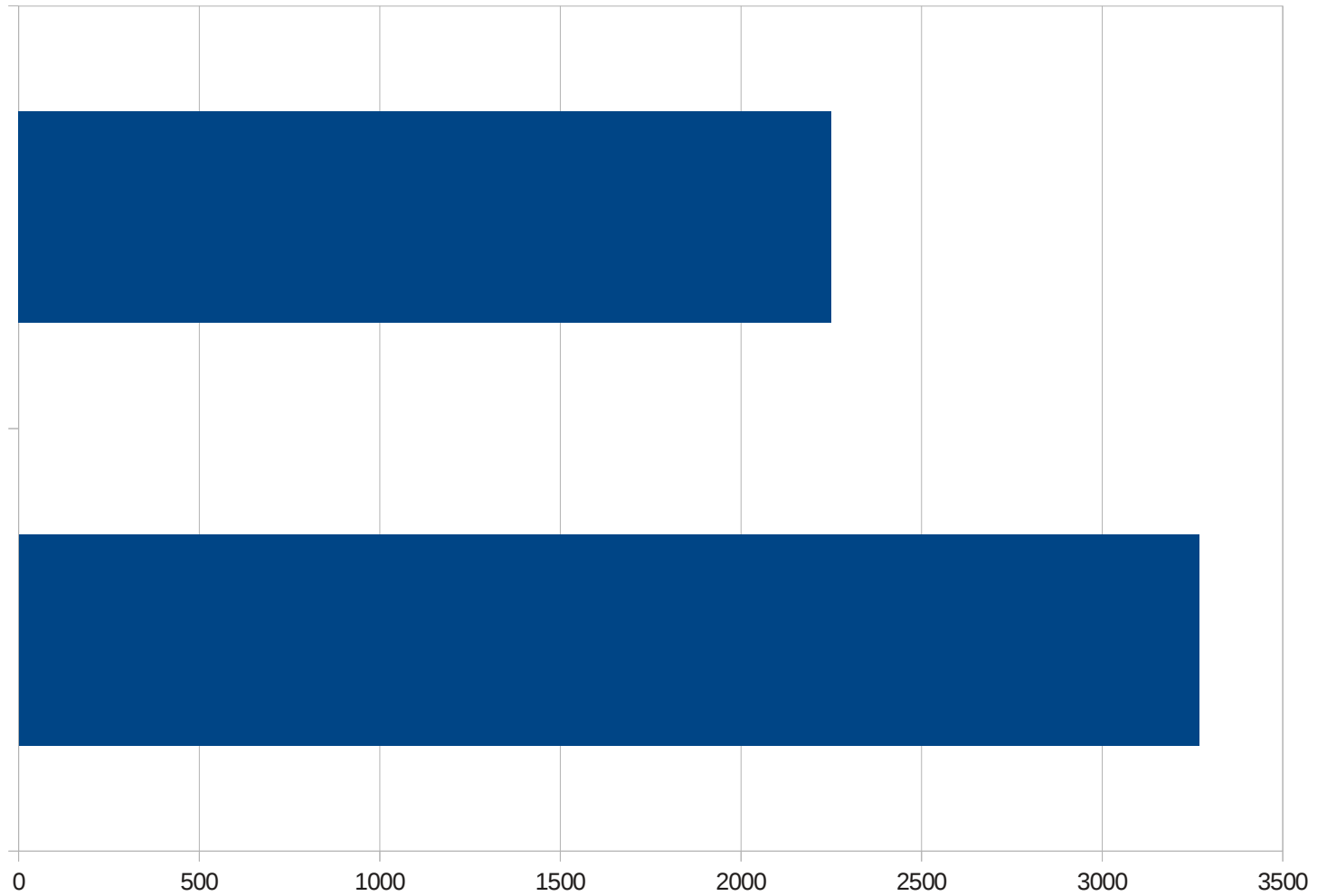


- Also known as
 - optimized binding
 - compiled binding
- Bytecode interpreter
 - No JS engine
- For simple bindings

QML_BINDINGS_DUMP=1 to see v4 instructions
QML_DISABLE_OPTIMIZER=1 to disable

v4 Bindings

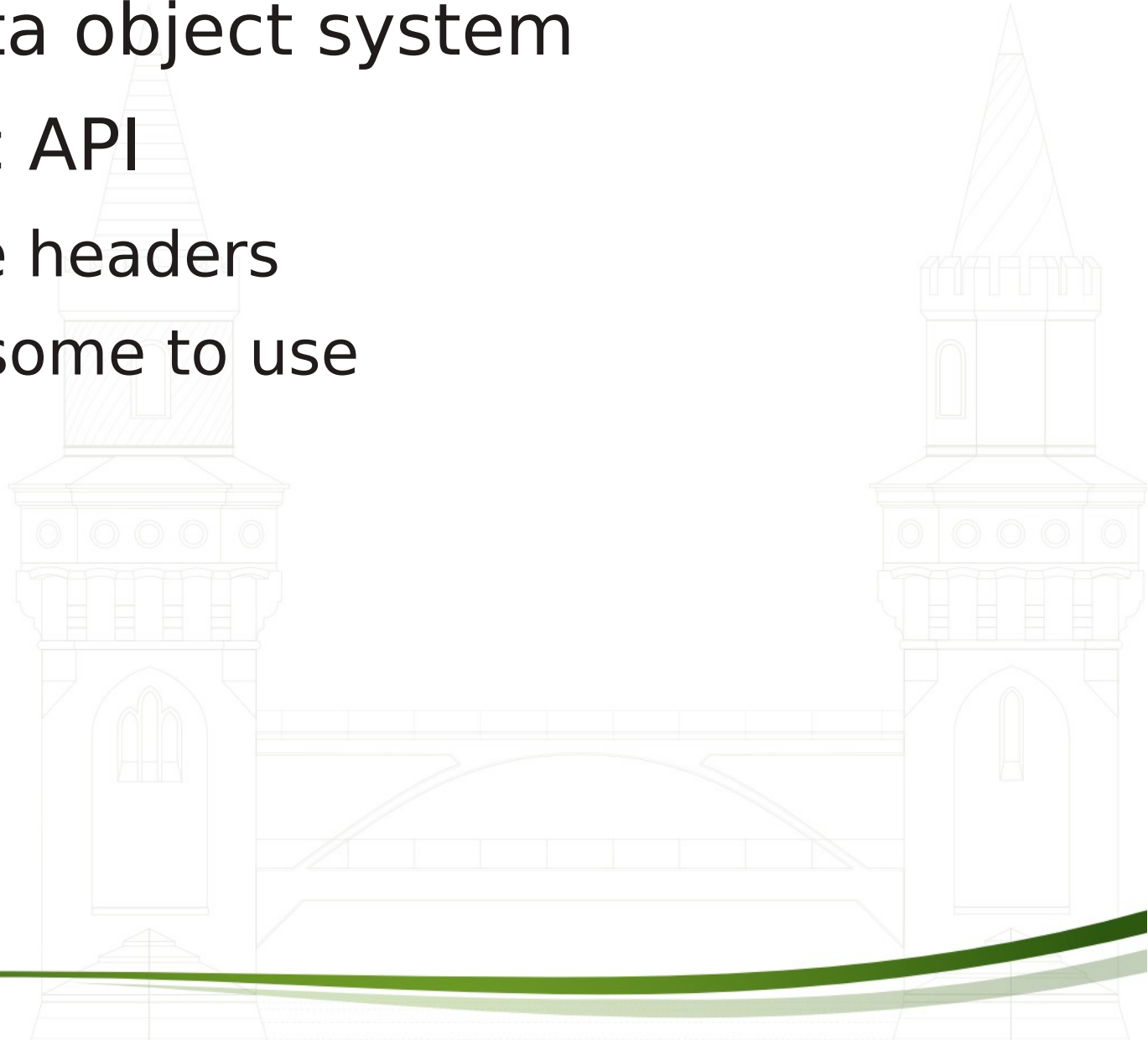
v8 Bindings

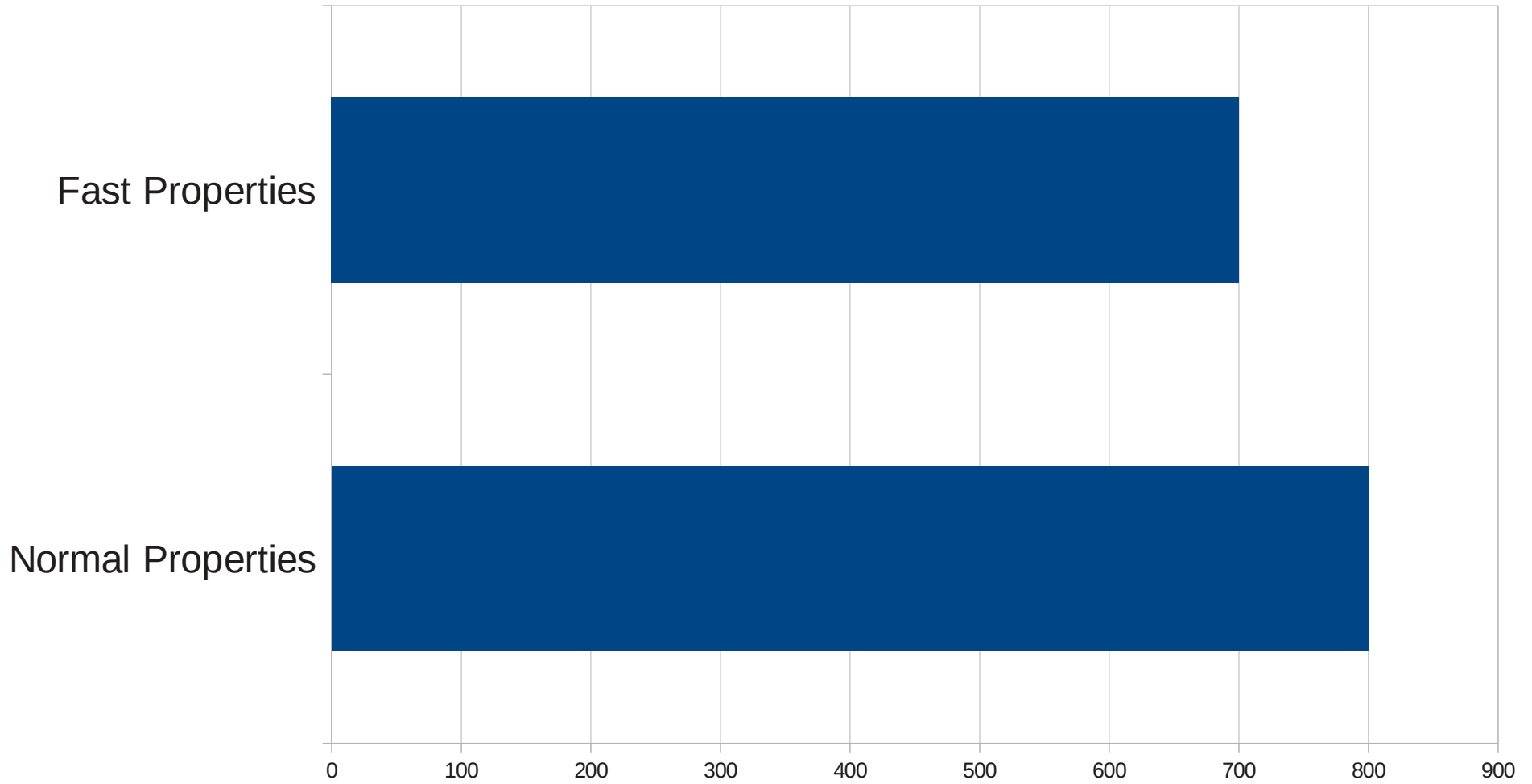


- QV4M
- Optimized for QML
- LLVM, MASM or VME
 - works on iOS



- Bypasses meta object system
- Not yet public API
 - needs private headers
 - a bit cumbersome to use





- qmlmin

- reduces size of QML file
- no changes in loading speed
- useful when embedded into executable

- FINAL keyword

- allows QML easier property lookup
- Not allowed to override property in subclass

Optimizations

- QML profiler
- VME
- v4
- Fast Properties
- FINAL

- My blog articles “QML Engine Internals”
 - <http://www.kdab.com/category/blogs/qmlengineseries/>
- Qt docs
 - <https://qt-project.org/doc/qt-5.0/qtquick-performance.html>
 - UI-related tips as well
- The sources
 - <http://www.kdab.com/~thomas/devdays2012/>

```
Item {
    id: endOfTalk
    Text {
        text: "Questions?"
    }

    contactInfo: [
        Twitter { "@thomas_mcguire",
                  "@KDABQt" }
        AskNow {},
        Email { "thomas@kdab.com" }
    ]
}
```