The image is a composite. The top portion shows a panoramic view of a city at night, with various buildings and lights visible against a dark sky. The bottom portion shows two fluffy kittens sleeping peacefully on a dark, textured surface. The text is centered over the middle of the image, which has a dark red, textured background.

QtWebEngine – Taming the Beast

Agenda



- The Big Picture
- Minimal Example
- Graphics Stack
- Debugging & Profiling
- New Features
- Outlook

- Built on top of Chromium
- Introduced in Qt 5.4 (Qt for Device Creation: 5.3)
- Platforms
 - Windows (Visual Studio 2013)
 - OS X
 - Linux
 - Qt for Device Creation

Architecture

Executable

QtWebEngineProcess

Qt5WebEngine

Qt5WebEngineWidgets

Qt5WebEngineCore

Chromium

Complexity

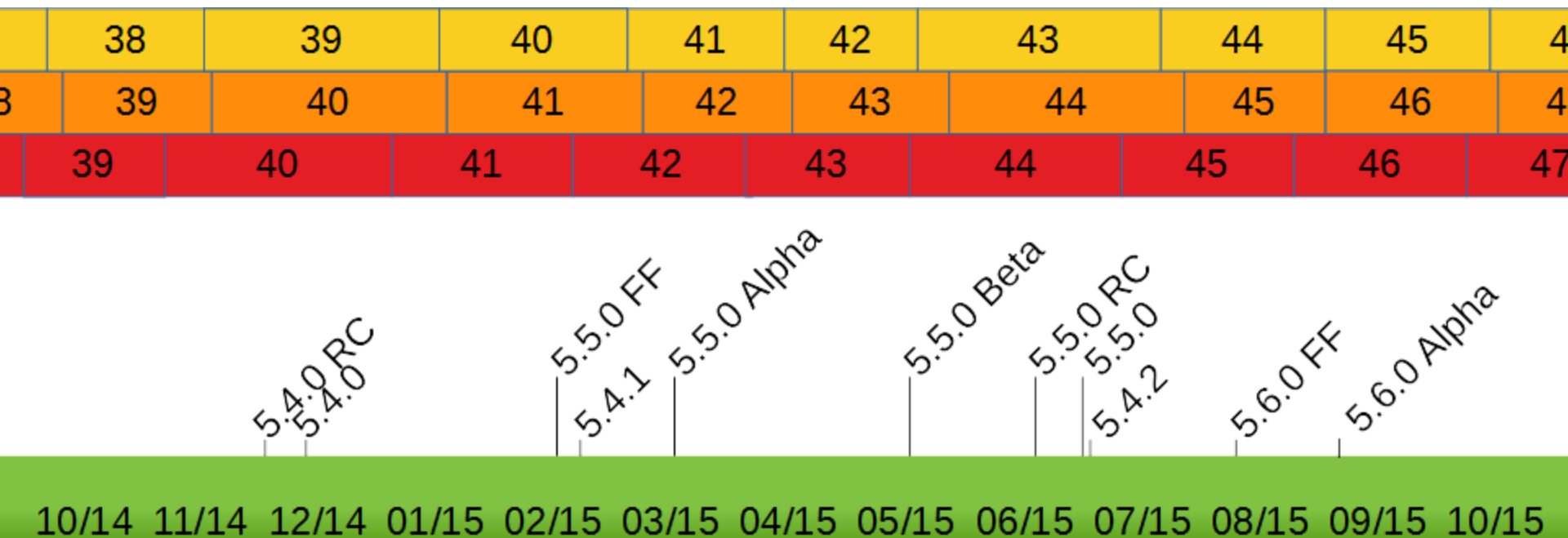


SLOC	Directory	SLOC-by-Language (Sorted)
9189194	src_3rdparty	cpp=5101622,ansic=3079988,asm=513055,python=294317
...		
12062	src_core	cpp=12062
3895	src_webengine	cpp=3895
3622	src_webenginewidgets	cpp=3622
...		

Total Physical Source Lines of Code (SLOC)	= 9,225,030
Development Effort Estimate, Person-Years	= 2,912.37

generated for qtwebengine-opensource-src-5.5.0
using David A. Wheeler's 'SLOCCount'

Release Schedules





Show me some code!



No OpenGL? Bummer!



- ANGLE
- Qt Quick 2D Renderer
- Software Emulation

Agenda



- The Big Picture
- Hello World!
- Graphics Stack
- **Debugging & Profiling**
- New Features
- Outlook

JavaScript Console Demo



Chrome DevTools



```
$ QTWEBENGINE_REMOTE_DEBUGGING=3999 ./myapp  
$ chrome http://localhost:3999
```

Command Line Arguments



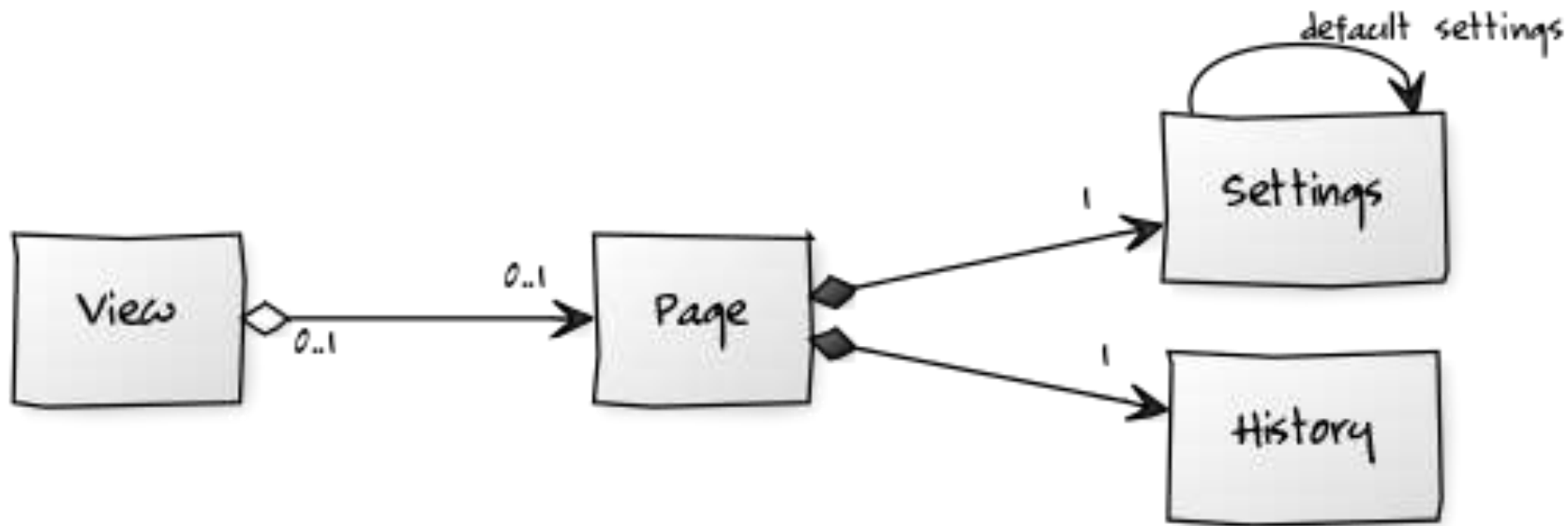
```
--remote-debugging-port=3999  
--use-data-dir <dir>  
--single-process  
--show-fps-counter  
...
```

Agenda

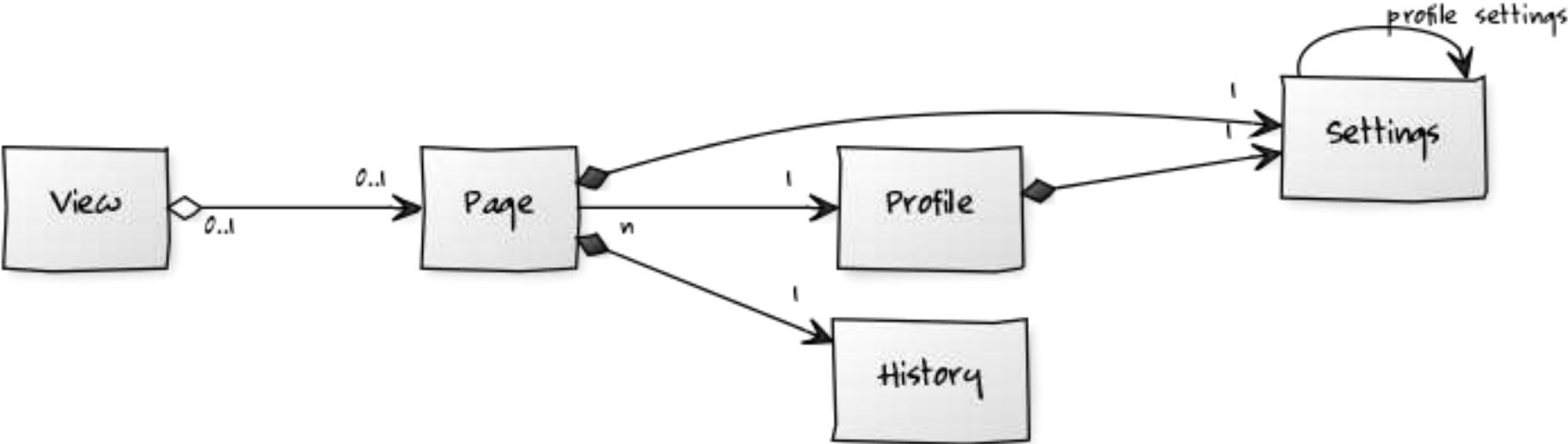


- The Big Picture
- Hello World!
- Graphics Stack
- Debugging & Profiling
- **New Features**
- Outlook

Architecture (Qt 5.4)



Architecture (Qt 5.5)



WebEngineProfile (Qt 5.5)



- Isolate group of pages
 - Common settings
 - Cookies
 - Content scripts
 - List of visited links
- “offTheRecord” switch

```
page.runJavaScript("document.title",  
    [](QVariant r) { qDebug() << r.toString(); }  
);
```

```
view.runJavaScript("document.title",  
    function(result) { console.log(r); }  
);
```

JavaScript Injection Demo



- Based on API for Chromium's support for *Content Script Extensions*
- Add, find, remove scripts `QWebEngineScriptCollection`
- Script properties `QWebEngineScript`
 - Injection point: `DocumentCreation`, `DocumentReady`, `Deferred`
 - Isolation through worlds: `MainWorld`, `ApplicationWorld` ...
 - `runOnSubFrames`

- Bi-directional channel between QObject's and JS
 - Dynamically created objects in JS proxy QObject's in C++
 - Asynchronous where possible
 - Supports different protocols
- Perfect match for QWebEngine
 - `page.webChannel` uses Chromium IPC (Qt 5.5)

ImageAnalyzer Demo



- WebRTC
 - Secure P2P networking
 - Audio, Video
- HTML5 Geolocation
 - Using QtLocation backend

`navigator.webkitGetUserMedia`

`navigator.geolocation`

Agenda



- The Big Picture
- Hello World!
- Graphics Stack
- Debugging & Profiling
- New Features
- **Outlook**

- Intercept URL requests `UrlRequestInterceptor`
- Install URL scheme handlers `UrlSchemeHandler`
- Manage Cookies `CookieStoreClient`
- Support for `QNetworkProxy::applicationProxy`

UrlRequestInterceptor demo



Plugins (5.6)



- NPAPI is dead, long live Pepper (PPAPI)!
- Say hello to Adobe Flash 😊

Summary



- You need OpenGL for optimal performance
- Debugging: Use console API, or Chromium's devtools
- Follow Chromium/Chrome news
- Talk to us!



Thank you!

www.qt.io

kai.koehne@theqtcompany.com