Qt is not QuickTime

C++ GUI Toolkit

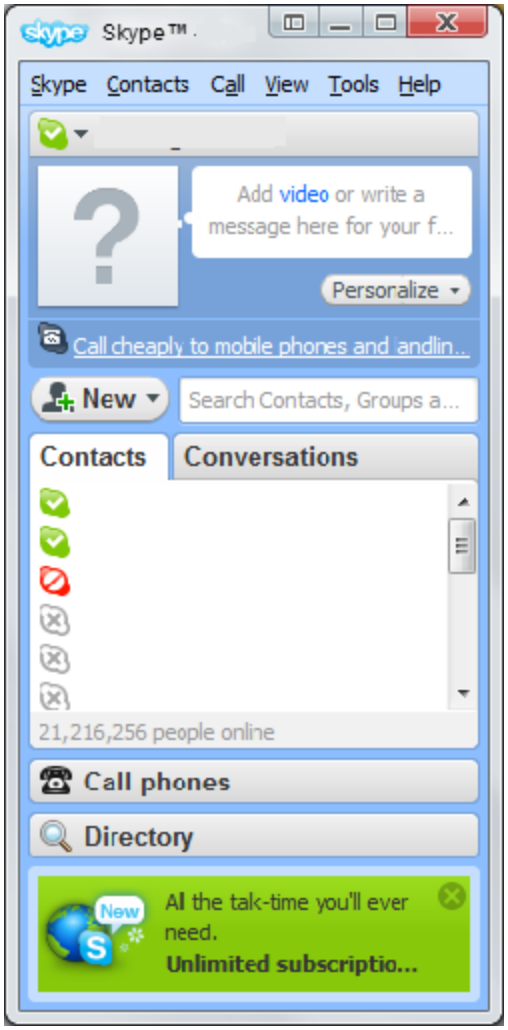
Runs Everywhere and it ain't Java

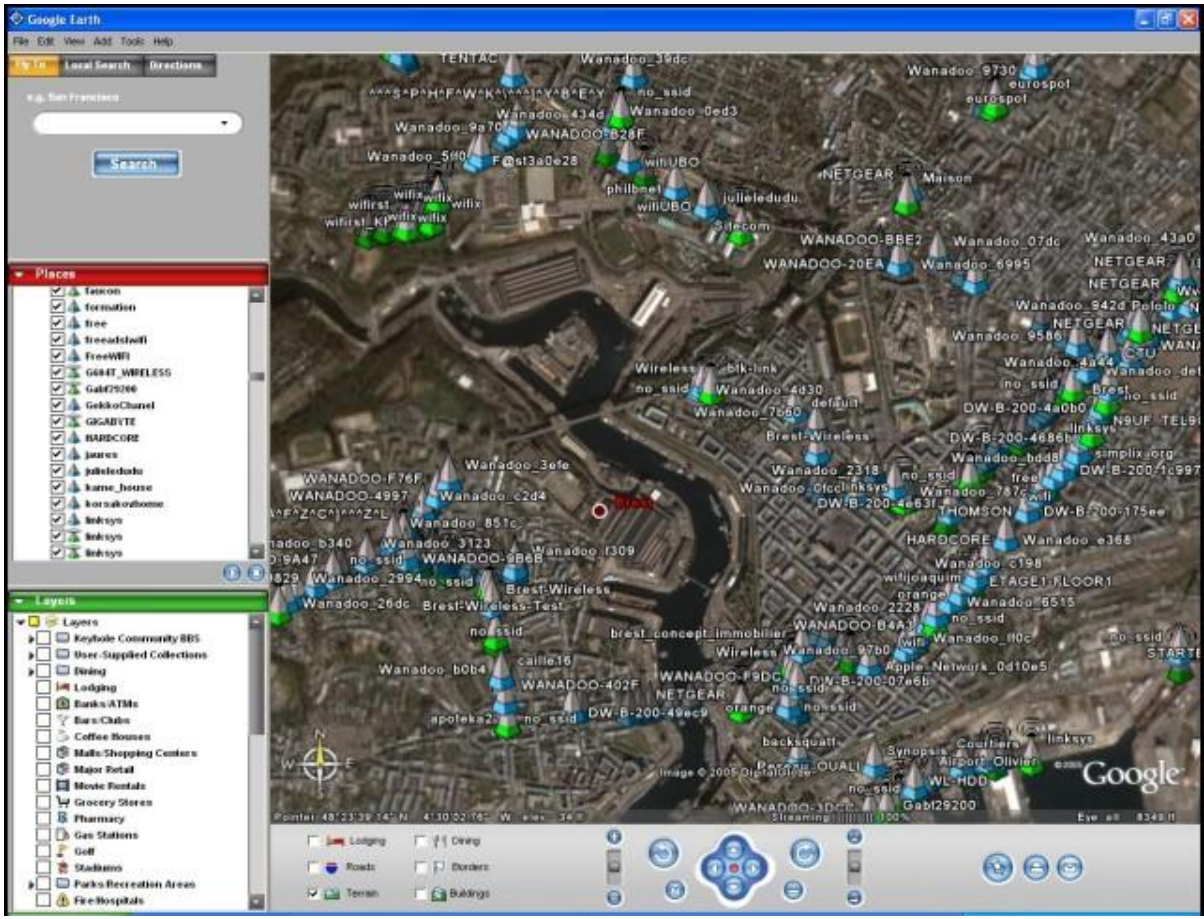
(GNU/Linux, Mac, Windows, Nokia Symbian and MeeGo Device, *nix, Partial Ports for Android, Kindle, iPhone exist)

History / TimeLine

- Haavard Nord and Eirik Chambe-Eng - Two Guys in a Garage - *1991*
- Cute and Trolls aka TrollTech - *1994*
- KDE – Qt's largest customer - *1998*
- Nokia acquires Trolltech – *2008*
- PyQt - 2008
- Qt goes LGPL and free for commercial use - *2009*
- Huge intellectual investment (read 700+ Engineers) behind the toolkit.

What can you make using Qt ?





- Skype, Google Earth, KDE, VLC Media Player and many more

== PyQt

Why PyQt is most popular language binding for making GUI Apps ?

- Python – fluid, concise, intuitive OOPS implementation (Perl OOPs anyone),
- Qt in Python – Write Less Code vis a vi C++,
- Compatible to Qt Philosophy aka classes, OOPs, Event Loop – Signal Slot,
- Riverside the company behind PyQt makes sure that Python API is in sync with Qt C++ API and up to date, (well almost)

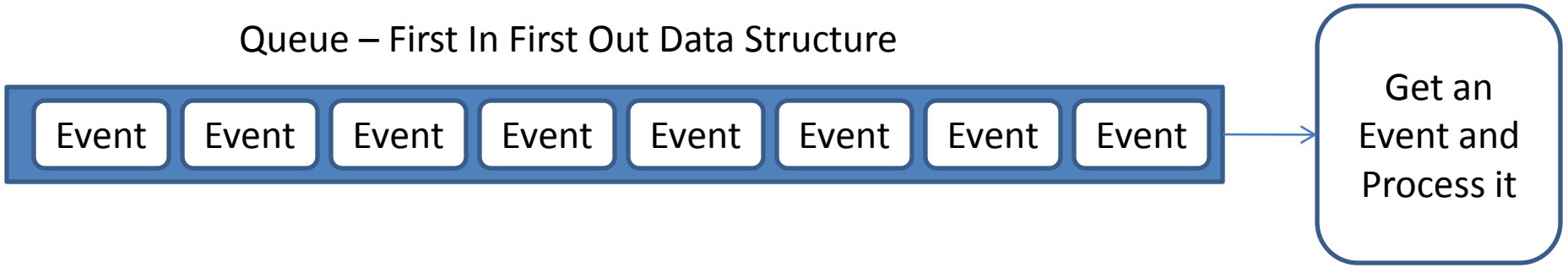
PyQt 101

1) print ”hello world”,
2) Event Loop Paradigm and GUI,
3) Function Callbacks || Signal-Slot,
4) Qt Designer – Creating To Do UI,
5) To Do Feature List and Code walk,
6) Stylesheet and To Do Look and Feel,
7) Resources to Learn more …

```
import sys
from PyQt4.QtCore import *
from PyQt4.QtGui import *

app = QApplication(sys.argv)
pushbutton = QPushButton(”Hello World”)
pushbutton.show()
app.exec_()
```

Event Loop Paradigm



Event == Mouse Clicks, Draw/Paint/Refresh Events, Resize, Keyboard Input, Quit Event. User actions or interaction with the application results in a event that's queued and eventually executed.

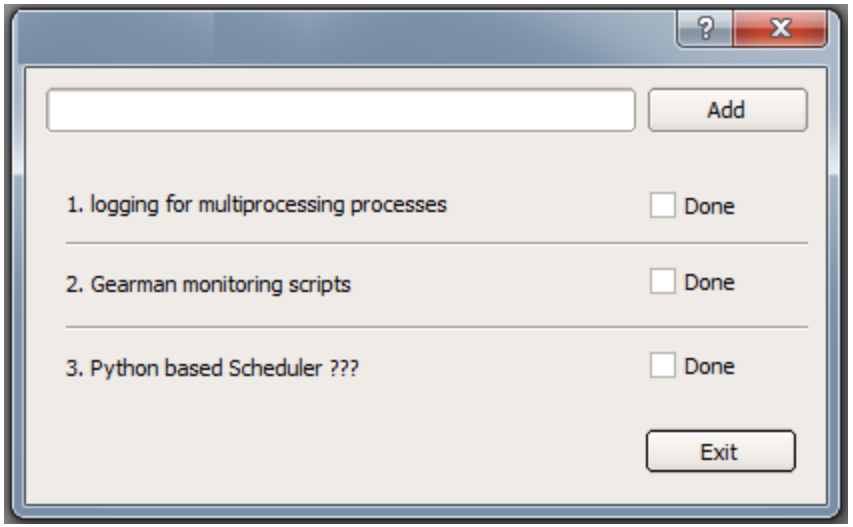
Why Event Loop Paradigm for GUI applications ?

a) External Device Inputs i.e. Mouse and Keyboard, are received by the OS and subscribed to by the application, (input stream async in nature)
b) GUI is nothing but a picture continuously refreshed/repainted so that it appears responsive, so instruction needs to be send to OS to periodically repaint,
c) Queuing and executing events as a stream is a logical solution when multiple asynchronous inputs exist. Thus Event loop is a natural fit for GUI frameworks.

Creating To Do UI using Qt Designer (Live)

To – Do Application

- Features
  - Add a task,
  - Show tasks pending,
  - Strike when task is done,
  - Persistent storage of tasks. (File/DB)



Code Walk of To Do
(Live)

http://github.com/versesane/pyqttodoapp

About Me :-)

- Yet Another Software Engineer
- Loves reading, programming and satisfying my curiosity
- Day Job @ 3PAR - (Python, Django)
- Qt Ambassador by Nokia – License to advocate and evangelize Qt
- Homepage => http://uptosomething.in