Oh my…. Maya is Qt!

Kristine Middlemiss, Autodesk

Developer Consultant, Autodesk Developer Network

Biography

Autodesk

Topics

- Introducing Qt
- How Qt fits into Maya
- Ways to work with Qt
  - Qt Designer with Maya Commands
  - Maya API
  - PyQt
  - Qt SDK

Welcome to Qt in Maya!

- Autodesk Maya 2011 is built upon the Qt UI Toolkit 4.5.
- This powerful cross-platform toolkit brings with it a wealth of new features and capabilities.
- We are extremely excited to bring this new functionality to you!

Qt (Cute)

- Application and GUI framework
- Cross-platform
- Trolltech bought by Nokia
- Qt licensing
  - GNU LGPL
  - Commercial

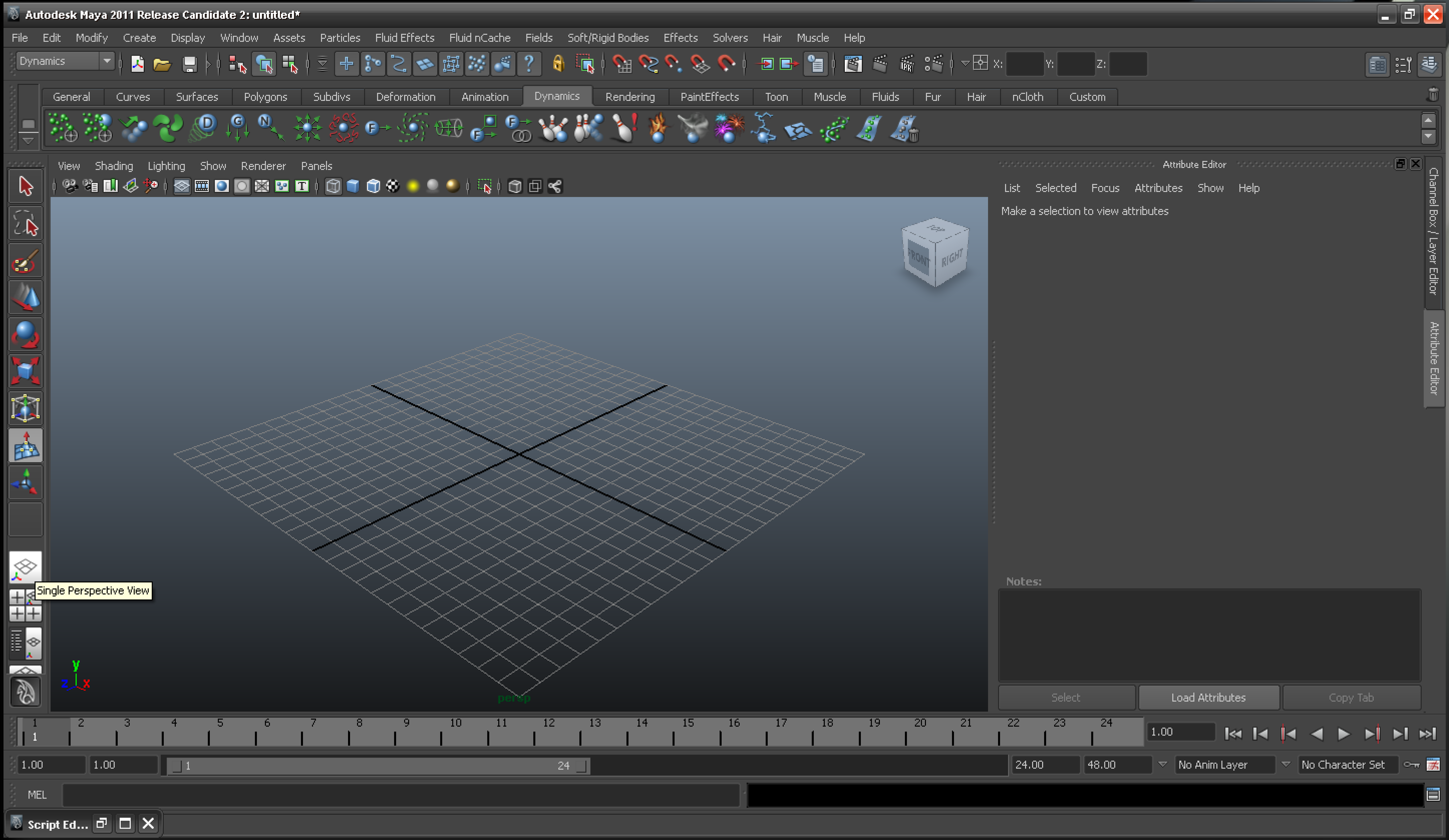
Qt

Code less.
Create more.
Deploy everywhere.

Why Qt?

- Create compact, high-performance apps
- Focus on innovation, not infrastructure coding
- Speed development and lower training costs – intuitive and easy-to-use
- Fully access code for easy debugging and customization

What does it look like?


Autodesk Maya 2011 Release Candidate 2: untitled*

Make a selection to view attributes

File Edit Modify Create Display Window Assets Particles Fluid Effects Fluid nCache Fields Soft/Rigid Bodies Effects Solvers Hair Muscle Help
Dynamics
General Curves Surfaces Polygons Subdivs Deformation Animation Dynamics Rendering PaintEffects Toon Muscle Fluids Fur Hair nCloth Custom
View Shading Lighting Show Renderer Panels
Attribute Editor
List Selected Focus Attributes Show Help
Notes:
Select
Load Attributes
Copy Tab
Single Perspective View
persp
Channel Box / Layer Editor
Attribute Editor
1.00
24.00
48.00
No Anim Layer
No Character Set
MEL
Script Ed...

» Brand New Look and Feel!

UI Demo

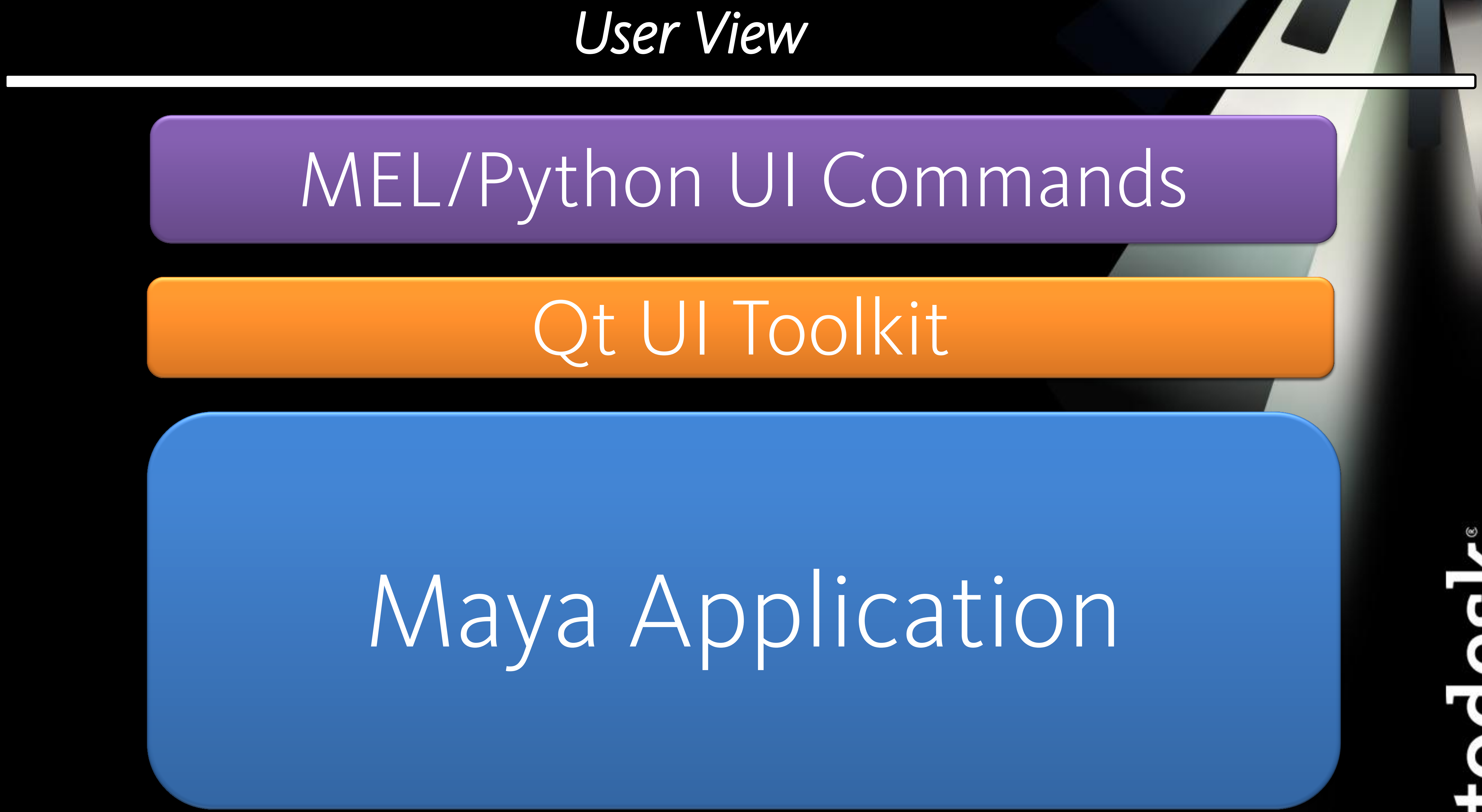
What does Qt mean for an:

- Artist
  - Customize at your finger tips
- Scripter
  - Some new MEL commands
- API Developer
  - This is where the fun begins!

Has MEL Gone Away?

- NO!!!

How does Qt fit in with MEL?



User View

MEL/Python UI Commands

Qt UI Toolkit

Maya Application

Previous Internal UI Tool Kit

Windows	Mac	Linux
• Qt toolkit Windows Native UI	• Qt toolkit Carbon	• Qt toolkit Motif

Extending Maya Commands

- New UI Maya commands:
  - dockControl
  - toolBar
  - treeLister
  - loadUI
- Check out
  - What's New in MEL and Python

Maya Commands Demo

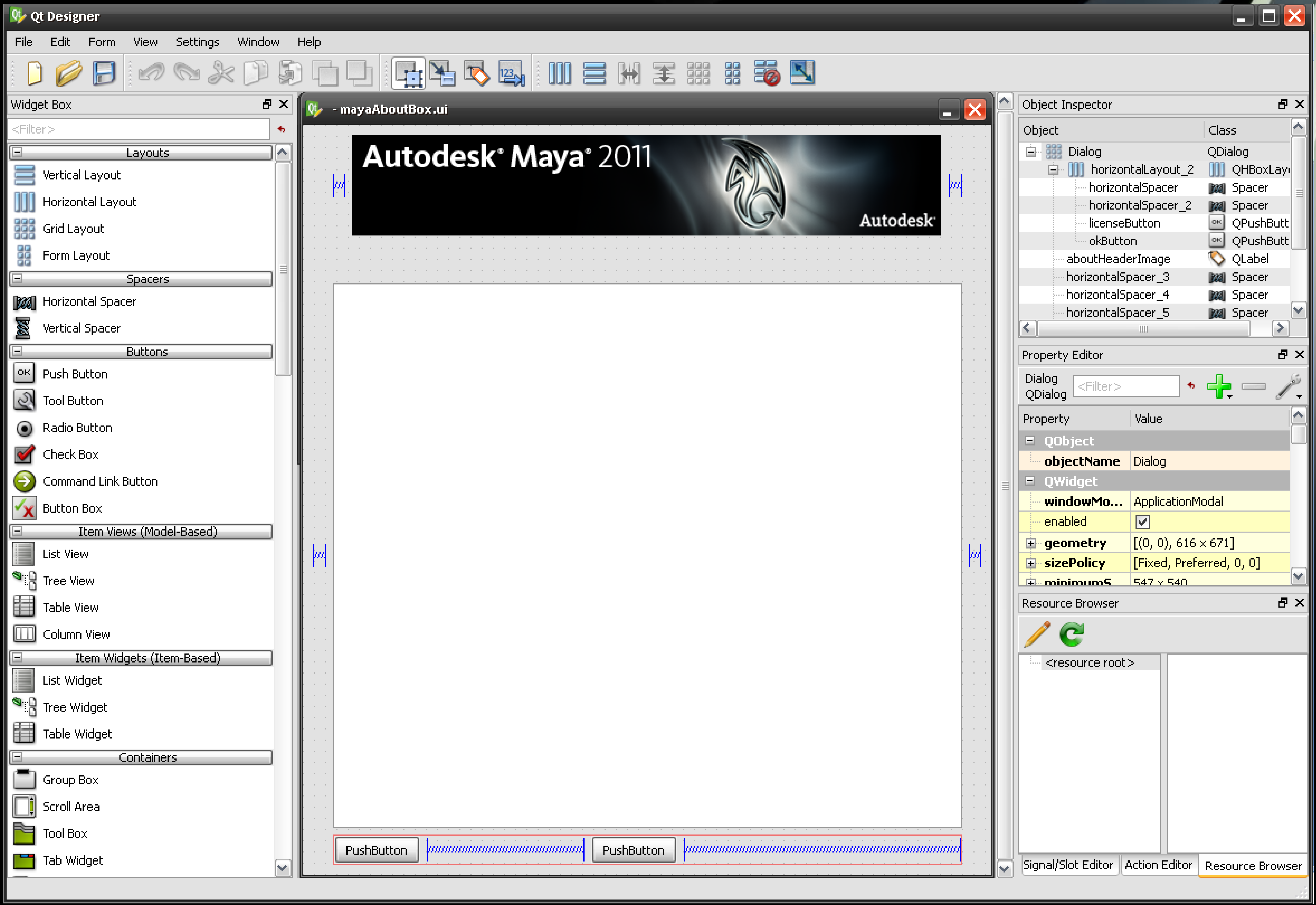
Ways to Integrate Qt with Maya

Ways to Integrate Qt with Maya

1. Qt Designer with Maya Commands
2. Maya API
3. PyQt
4. Qt SDK

Qt Designer with Maya Commands

Qt Designer – Nokia Tool



Object	Class
Dialog	QDialog
horizontalLayout_2	QHBoxLay
horizontalSpacer	Spacer
horizontalSpacer_2	Spacer
licenseButton	QPushButt
okButton	QPushButt
aboutHeaderImage	QLabel
horizontalSpacer_3	Spacer
horizontalSpacer_4	Spacer
horizontalSpacer_5	Spacer

Dialog
QDialog

Property	Value
QObject	
objectName	Dialog
QWidget	
windowMo...	ApplicationModal
enabled	☑
geometry	[(0, 0), 616 x 671]
sizePolicy	[Fixed, Preferred, 0, 0]
minimumS	547 x 540

Qt Designer

- Visual GUI design of dialogs and windows
- Layout widgets
- Edit widget properties
- Add tool bars, menus, and actions
- Connect predefined SIGNALS and SLOTS
- Save file format is XML
- uic – User Interface Compiler will generate code

Qt Designer

- Qt Designer: *. ui
- MEL command: loadUI
  - string $dialog1 = `loadUI -f "C:/mydialog.ui"`;
  - showWindow $dialog1;
- Advantages
  - Easy interface building
  - New control added
  - More functionality and flexibility: i.e. docking windows
- Embed MEL/python command
  - create dynamic property on widget
  - put MEL/python command as a string

Qt Designer Demo

Maya API

Maya API

- Qt Utility Class: **MQtUtil**
- Provides basic methods for accessing the Qt controls
- Allow user to look up Maya controls by name and retrieve its Qt pointer

MQtUtil Class

```
static QWidget* findControl(MString controlName, const QWidget* ancestor);
static QWidget* findLayout(MString layoutName, const QWidget* ancestor );
static QAction*  findMenuItem(MString itemName);
static QWidget* findWindow(MString windowName);
static MString fullName(const QObject* uiElement);
static QList<QObject*>  getLayoutChildren(const QWidget* layout);
static QObject*  getParent(const QObject* uiElement);
static QWidget *  mainWindow ()
static MNativeWindowHdl  nativeWindow (const QWidget *control)
static MString toMString(const QString& qstr);
static QString  toQString(const MString& mstr);
```

Safety First

- Safest way create your own Qt windows
- It’s not recommended to use the Qt API to modify existing Maya UI elements

Working with Qt Examples

- Qt Examples in the devkit
  - saveSwatchesCmd
  - qtForms
  - helixQtCmd
- No Visual Studio Project
- Use Makefile.qt
  - Download Qt package
  - Launch command prompt window
  - Go to devkit/plug-ins directory
  - Execute Command: nmake -f Makefile.qt

Maya API Qt Demo

PyQt

How does the Qt SDK Differ?

- Python binding to Qt library
- External API from Autodesk, created by Riverbank Computing Limited
- Provides developers with the tools to develop and deploy applications

Why PyQt?

- Combines all the advantages of Qt and Python
- Has all the power of Qt, but is able to exploit it with the simplicity of Python.
- You don’t need to learn a new language if you already working in with Maya Python

What is PyQt?

- Is a set of Python bindings for Qt
- There are two sets of bindings:
  1. PyQt v4 supports Qt v4
  2. The older PyQt v3 supports Qt v3 and earlier
- Main Modules:
  1. QtCore
  2. QtGui

Installing PyQt on your computer

- Download PyQt (PyQt modules)
- Download SIP (C++ interface code for Python)
- Building Qt
- Building SIP
- Building PyQt

- Provide pyqtmaya2011.pdf

Qt SDK

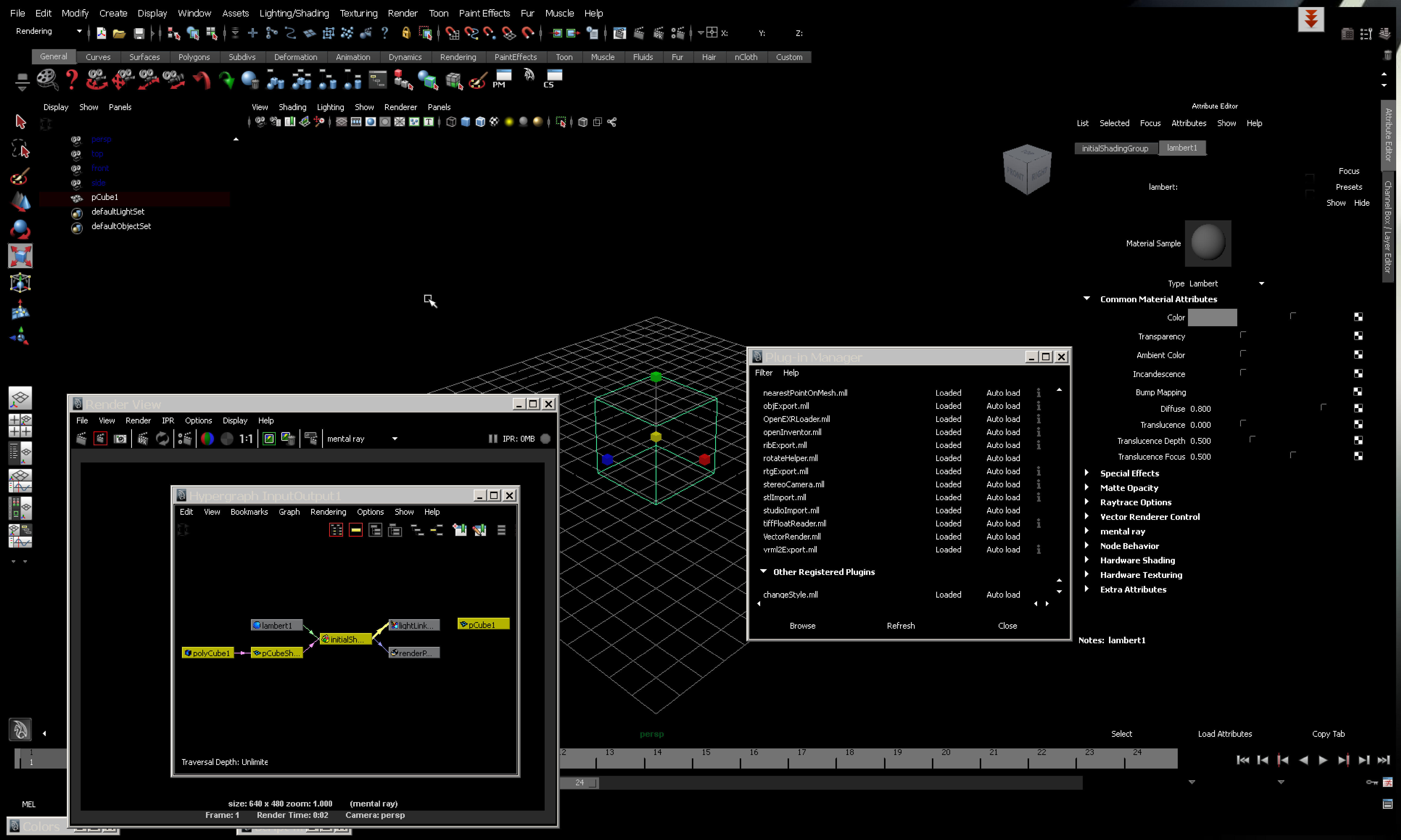
How does the Qt SDK Differ?

- External API from Autodesk
- The technology is just embedded in Maya
- Provides developers with the tools to develop and deploy applications
- Maya leverages this toolkit

Installing Qt on your computer

- Decide on Qt license for Maya plug-ins
- Download and install Qt source 4.5 Mingw (required for Qt)
- Rebuild Qt from source
- Configure
- Set Paths
- Build plug-ins

What CAN it look like?


File Edit Modify Create Display Window Assets Lighting/Shading Texturing Render Toon Paint Effects Fur Muscle Help
Rendering
General Curves Surfaces Polygons Subdivs Deformation Animation Dynamics Rendering PaintEffects Toon Muscle Fluids Fur Hair nCloth Custom
Display Show Panels
View Shading Lighting Show Renderer Panels
persp
top
front
side
pCube1
defaultLightSet
defaultObjectSet
persp
MEL
Render View
File View Render IPR Options Display Help
1:1
mental ray
IPR: 0MB
Hypergraph InputOutput1
Edit View Bookmarks Graph Rendering Options Show Help
lambert1
initialSh...
lightLink...
pCube1
polyCube1
pCubeSh...
renderP...
Plug-in Manager
Filter Help
Browse Refresh Close
Attribute Editor
List Selected Focus Attributes Show Help
initialShadingGroup
lambert1
Focus
Presets
Show Hide
Attribute Editor
Channel Box / Layer Editor
Select Load Attributes Copy Tab
12 13 14 15 16 17 18 19 20 21 22 23 24
24

Traversal Depth: Unlimite

size: 640 x 480 zoom: 1.000 (mental ray)
Frame: 1 Render Time: 0:02 Camera: persp

nearestPointOnMesh.mll	Loaded	Auto load
objExport.mll	Loaded	Auto load
OpenEXRLoader.mll	Loaded	Auto load
openInventor.mll	Loaded	Auto load
ribExport.mll	Loaded	Auto load
rotateHelper.mll	Loaded	Auto load
rtgExport.mll	Loaded	Auto load
stereoCamera.mll	Loaded	Auto load
stlImport.mll	Loaded	Auto load
studioImport.mll	Loaded	Auto load
tiffFloatReader.mll	Loaded	Auto load
VectorRender.mll	Loaded	Auto load
vrml2Export.mll	Loaded	Auto load

Other Registered Plugins

changeStyle.mll	Loaded	Auto load

lambert:

Material Sample

Type Lambert

Common Material Attributes

Color
Transparency
Ambient Color
Incandescence
Bump Mapping
Diffuse 0.800
Translucence 0.000
Translucence Depth 0.500
Translucence Focus 0.500

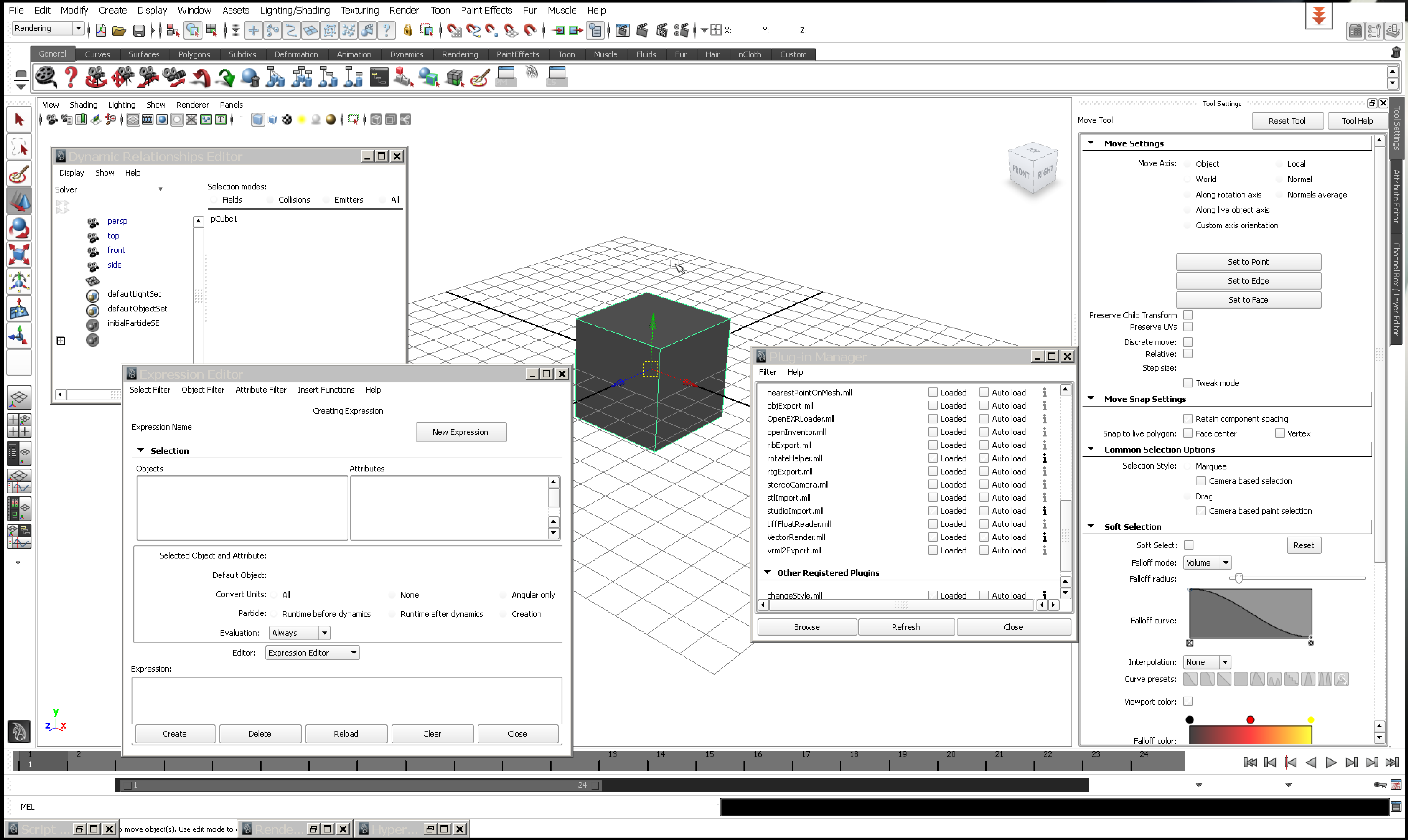
Special Effects
Matte Opacity
Raytrace Options
Vector Renderer Control
mental ray
Node Behavior
Hardware Shading
Hardware Texturing
Extra Attributes

Notes: lambert1

- » Interesting…

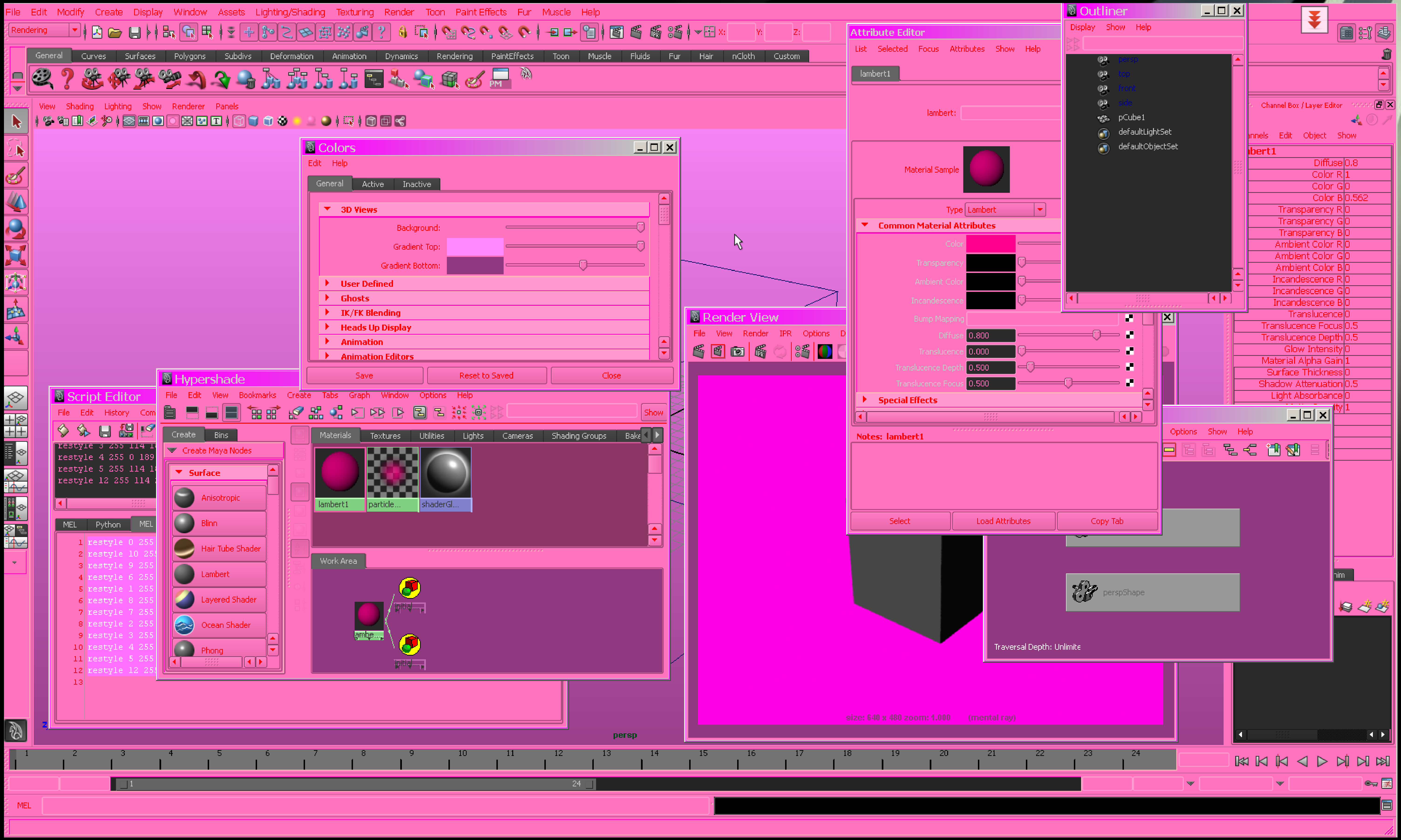
Autodesk

What CAN it look like?



- Google and I would get along…

What CAN it look like?



```
restyle 3 255 114 1
restyle 4 255 0 189
restyle 5 255 114 1
restyle 12 255 114
```

```
restyle 0 255
restyle 10 255
restyle 9 255
restyle 6 255
restyle 1 255
restyle 8 255
restyle 7 255
restyle 2 255
restyle 3 255
restyle 4 255
restyle 5 255
restyle 12 255
```

lambert:

Type: Lambert

Diffuse: 0.800

Translucence: 0.000

Translucence Depth: 0.500

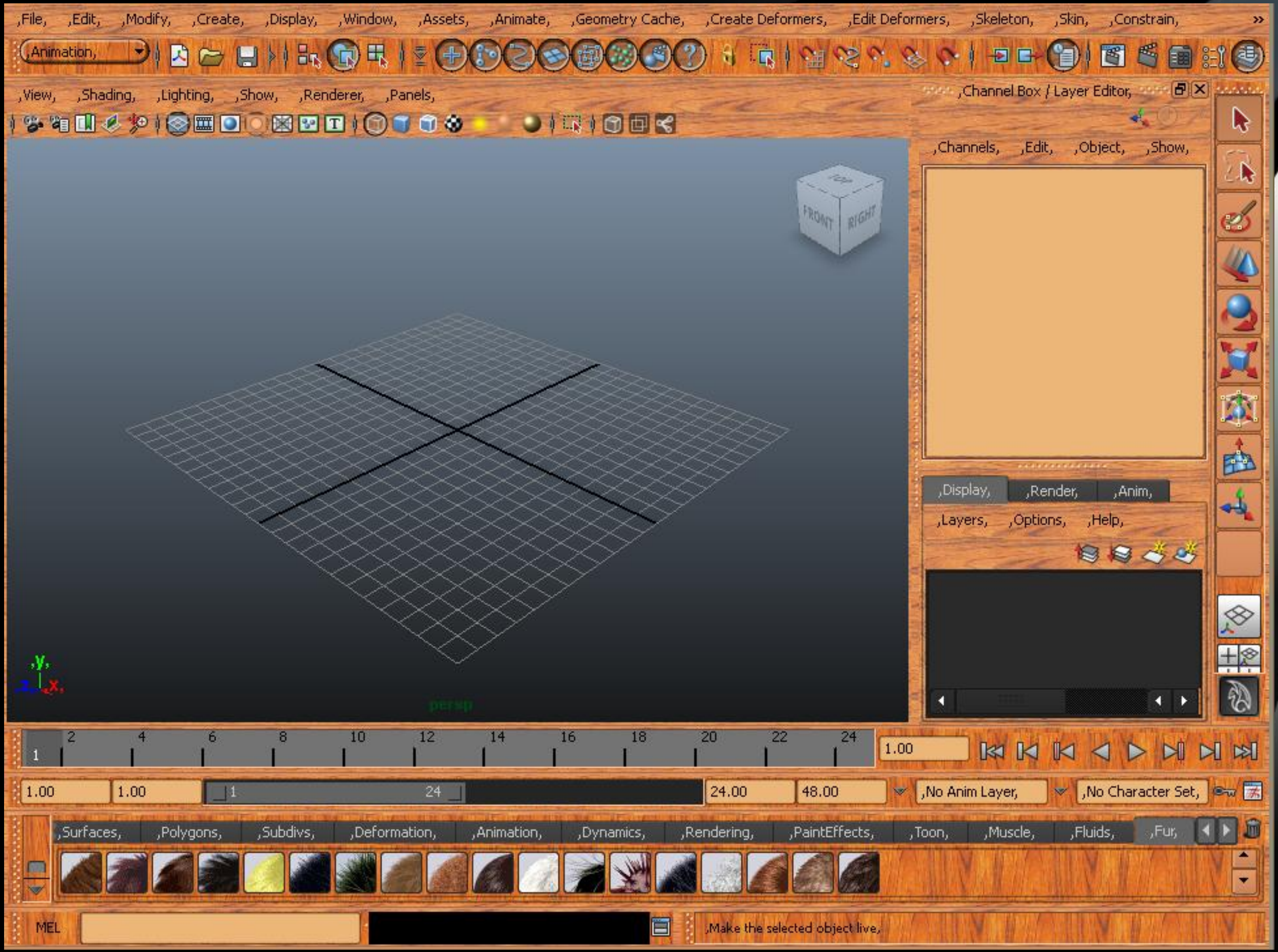
Translucence Focus: 0.500

Traversal Depth: Unlimite

bert1	
Diffuse	0.8
Color R	1
Color G	0
Color B	0.562
Transparency R	0
Transparency G	0
Transparency B	0
Ambient Color R	0
Ambient Color G	0
Ambient Color B	0
Incandescence R	0
Incandescence G	0
Incandescence B	0
Translucence	0
Translucence Focus	0.5
Translucence Depth	0.5
Glow Intensity	0
Material Alpha Gain	1
Surface Thickness	0
Shadow Attenuation	0.5
Light Absorbance	0
ity	1

» Yikes!!!!!!

What CAN it look like?


,File, ,Edit, ,Modify, ,Create, ,Display, ,Window, ,Assets, ,Animate, ,Geometry Cache, ,Create Deformers, ,Edit Deformers, ,Skeleton, ,Skin, ,Constrain,
Animation,
,View, ,Shading, ,Lighting, ,Show, ,Renderer, ,Panels,
,Channel Box / Layer Editor,
,Channels, ,Edit, ,Object, ,Show,
,Display, ,Render, ,Anim,
,Layers, ,Options, ,Help,
FRONT RIGHT TOP
persp
1 2 4 6 8 10 12 14 16 18 20 22 24
1.00
1.00 1.00 1 24 24.00 48.00
,No Anim Layer, ,No Character Set,
,Surfaces, ,Polygons, ,Subdivs, ,Deformation, ,Animation, ,Dynamics, ,Rendering, ,PaintEffects, ,Toon, ,Muscle, ,Fluids, ,Fur,
MEL
,Make the selected object live,

- No comment ;)

Wood Panel Demo

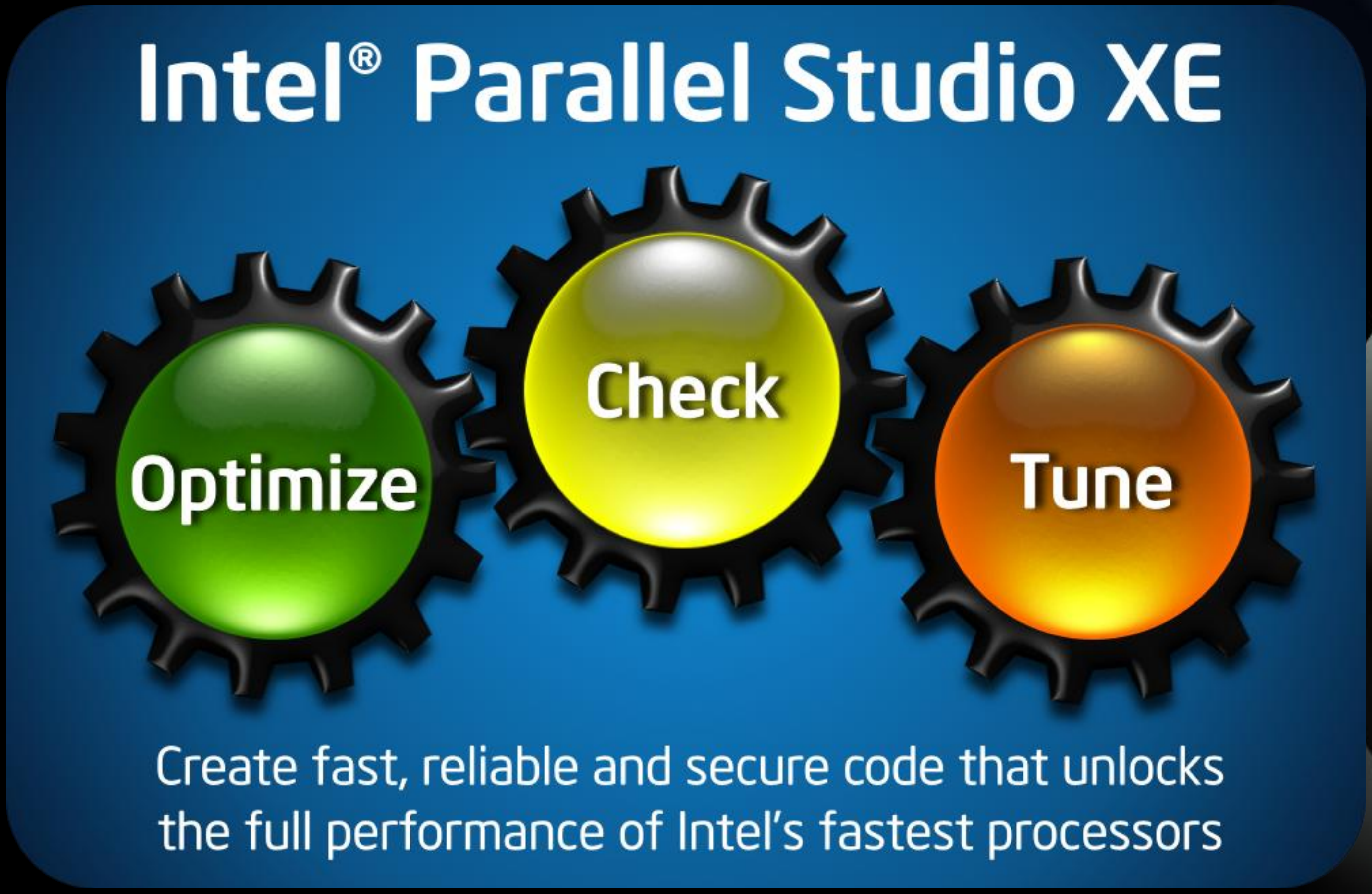
Summary

- Artist
  - Customize at your finger tips
- Scripter
  - Some new MEL commands
- API Developer
  - This is where the fun begins!

Summary

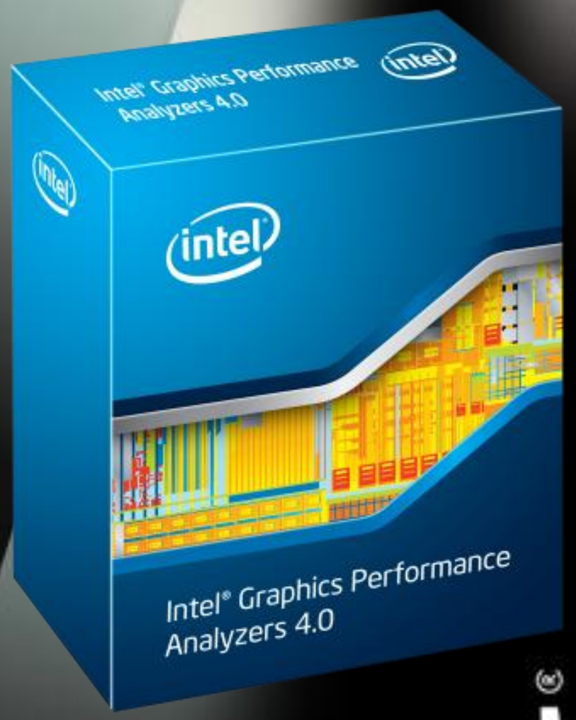
1. Qt Designer with Maya Commands
2. Maya API
3. PyQt
4. Qt SDK

Accelerating Your Development & Your Games



Intel® Parallel Studio XE

Create fast, reliable and secure code that unlocks the full performance of Intel's fastest processors



www.intel.com/go/parallel

One integrated toolkit to develop, optimize, check & tune

Developing & Playing Games On Intel

Intel® Core™ based clients

Intel® Xeon® based workstations

Smart Performance
Brilliant Visuals

Workload Optimized
Innovation Platforms

- Intel® HD Graphics 3000 – An entirely new gaming experience with processor based graphics

- Adaptive performance super charges your gaming experience with Intel® Turbo Boost Technology 2.0

New

- Intel® HD Graphics P3000 – delivering professional graphics features, functions & performance

- Processing, memory & bandwidth capacity that helps to accele- rate your innovations

Autodesk®

Q & A

Image courtesy of Firaxis Games

Autodesk®

Autodesk®