

# Thoughts about Avalon

Matthias Hopf  
SuSE Labs  
Novell





# What the fuzz?

---

- Raised user expectations
- Quartz Extreme + Avalon ahead
- Needed hi-level changes should be announced early
- Clone everything? No! But...
- No “outrunning without catching up”
- Backwards compatibility
  
- Start of discussion: NOW



# Being First: Quartz Extreme

---

- Actually the first widely accepted composition based windowing system
- OpenGL / PDF / etc. output drivers
- Application space libraries
  - Vector drawing
  - Image manipulation
  - Video
  - Color management

# The Mists of Avalon

---

- Larger Impact than Quartz due to M\$
- Fact sheet about Avalon:
  - Resolution and device independent graphics
  - Improved precision: double vs. float
  - Improved color space: scRGB vs sRGB
  - Retained mode rendering + scene graph
- Stricter separation of UI and code (XAML)
- Application space libraries
  - Similar to MacOS



# Our Position

---

- Composite
  - Equivalent to competitors, both feature-, and memory hog wise
  - Multiple composition managers?
- Render
  - Has enough potential for 2D widgets
  - Still being enhanced slowly
- EXA
  - Mid-term solution for aged XAA
- Xgl
  - More equivalent to competitors

- Mostly up to par or even superior
  - Clearly separated and extensible composition
  - Several features missing (most application space)
- Issues
  - Still experimental
  - OpenGL driver issues
  - Binary only drivers vs. GPL
  - OpenGL vs. Windows Vista?  
The Future does not shine very bright right now



# Future (1) – Retained Mode

---

- Retained composition
- Retained refresh: needs scene graph
- On Xserver side?
  - Increases Xserver memory footprint, complexity
  - Difficult: stability + extensibility
  - We don't want server side fonts (again)
  - Acceleration (display lists) easier
- On application side?
  - Similar to Avalon
  - Not our business?



## Future (2) – Accessibility

---

- Render upscaled window contents
- Support for non-bitmap content
- Scene graph on the server? (again)
- Advanced Expose events?
  - Could be hidden by toolkit
  - Memory management in composite manager even more tricky
  - Larger footprint per window during zoom



## Future (3) – Antialiasing

---

- Prerendering
  - Already done for Render (e.g. glyphs)
  - Incorrect for transformed windows (esp. sub-pixel)
- Minification during composite
  - Only way to do it correctly
  - Implicitly for all primitives
  - Increased memory footprint, needed bandwidth
  - Even higher on added sub-pixel rendering
  - Support by graphics hardware?  
Or only through pixel shaders?
- Combination of both?

## Future (4) – 3D Widgets

---

- Easily usable substitute for OpenGL context
- Similar to Avalon's Viewports
  - Retained mode (scene graph)?
  - Xserver or application space?
- Create new extension for integration?
  - Should be easy to integrate in Xgl
  - Will be difficult to integrate in Xorg
- Or just use OpenGL in client library?
  - Not accelerated without pBuffers / FBOs
  - But direct rendering possible in the far future

## Future (5) – Calibration

---

- Color correction
  - Embedded in Quartz + Avalon
  - Application space library
  - Profile to be exported by Xserver
  - Needs system-wide interface (printers etc.)
- Geometry correction
  - Viewing plane distortion  
e.g. keystone correction or PowerWalls
  - Job of Composite manager  
Possible in current plugin system?

## Future (6) – Dynamic Reconfiguration

---

- X has been notorious for its static-only configuration
- Dynamic Reconfiguration already state-of-the-art on all other platforms
- Slowly changing, but...
- Each driver does it differently
- Interface should be common + extensible
- Backwards compatible (Randr, others?)
- Interface proposal pending



# Discussion?

---

- General direction?
  -
- Points missed?
  -
- Nontrivial aspects?
  -
- Backwards compatibility?
  -