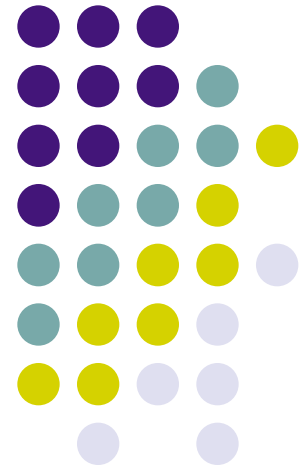


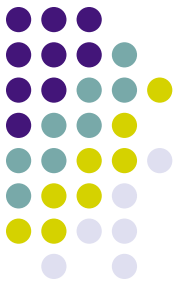
# Cross-Platform Apps

---

Glenn Reid  
October 2006  
v1.0



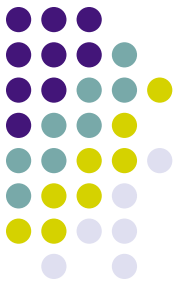
# Who Am I?

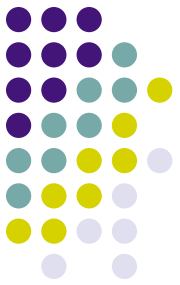


- Long-time App developer
  - **Apple:** iMovie, iPhoto
  - **Adobe:** Illustrator
  - **Fractal:** Painter
  - **NeXT:** PasteUp

# Perspective on Apps

- Motivation for building Apps
- Platforms
- Building and Delivering
- What' s Different Today





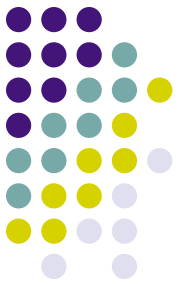
# What is an App?

- A Solution to a Problem
  - Does something “useful”
  - Does something “hard”
  - Runs on some platform(s)
  - Has some “market”

# App Abstractions

- User interface
- Storage
- “Features”

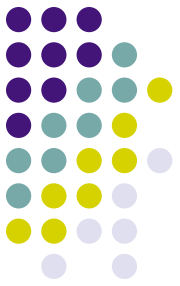




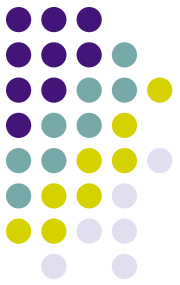
# Platforms

- Windows XP/Vista
- MacOS X
- Linux / X-windows
- Java / Swing
- Browser / IE / Javascript / AJAX
- Palm OS
- Windows CE
- Flash / Flex
- Xbox
- PlayStation
- Symbian OS

# How is Platform Chosen?



- By the customer, ultimately
- The market penetration of the platforms
- The software developers



# Cross-Platform Apps

- Do they even exist any more?
  - Yes, more than ever before
- Does Microsoft care?
  - Yes
  - Linux, AJAX, Ruby on Rails, Flash are symptoms





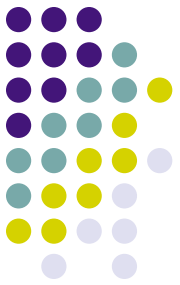
# Cross- vs. Multi-Platform

- Old Way of Thinking: Cross-Platform (pick 2)
  - Windows / Mac
  - Windows / Sun
  - Windows / Linux
  - Windows XP / Windows CE
  - Windows 98 / Win32
  - English / Japanese



# Platform-Agnostic

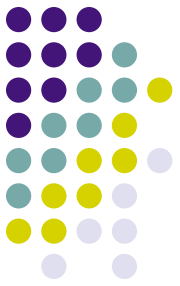
- If you want “Photoshop”
  - You don’t want to have to choose by platform
  - You might want it on your mobile device
  - You might change your mind later
- Software Should Be Like Music (was)
  - Music should not be like Software (is)
  - “Sorry, but Frank Sinatra never recorded *My Way* for output on those speakers.”



# Platform-Agnostic

- “Write Once, Run Anywhere”
  - Works for the *language* but not the *platform*
  - Sun had to write an entire “cross-platform” layer
  - Java/Swing “operating system” is cumbersome
  - Common denominator isn’t interesting

# What is an “Operating System”?

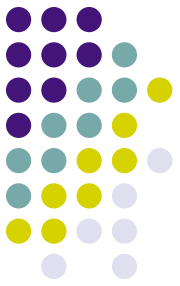


- A program launcher
- A file system
- A *tool kit* for applications
  - This is also the definition of a *platform*



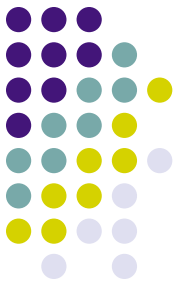
# BrOwSer

- Browsers have become OS' s
  - Application Launchers
  - A tool kit for applications
  - *No file system access*
- They just aren' t very good OS' s
  - Tool kits are primitive
  - Applications are Javascript or ActiveX
- Why?!?



# Browser Apps: *Why?!*

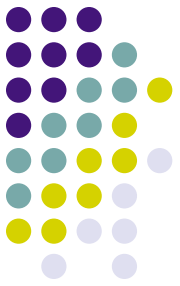
- It's "easy" — lots of GUI tools for UI
- It's cross-platform (mostly)
- Developers are lazy



# Doing Something “Hard”

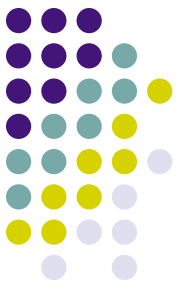
- If the platform makes it easy, I can't build a product on it
- Developers need “added value” somewhere
- Competitive advantage == solving hard problems
- Sometimes platform improvements kill products/companies

# Adobe



- Builds “core technology” X-platform layers
- Imaging libraries
- Font handling
- Graphics acceleration
- Silly things like abstracting rectangles, handling strings, newlines vs. CRLF.

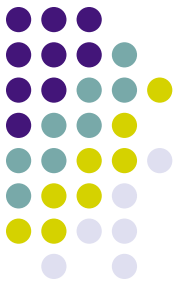




# Cool Platform Features

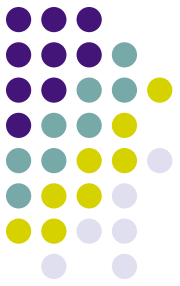
- Can I even use them in a X-platform app?
- Standard technologies vs. differentiation
- *Apps* compete on features
- *Platforms* compete on performance, stability
- Adding “features” to an OS is a little silly
  - Apple does this: nobody uses them except Apple
- Games: better/faster on one platform?

# Games

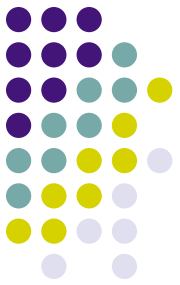


- Interesting Category
  - use *no* OS features
  - Use performance features extensively

# Why Apple now builds Apps



- Nobody was using Cool Features of the OS
  - All significant Mac Apps were cross-platform
  - Cross-platform apps didn't use Cool Features
- In order to show off the Cool Features...
  - Build your own apps!

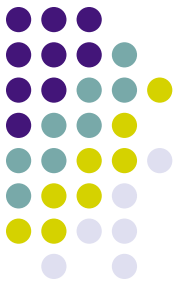


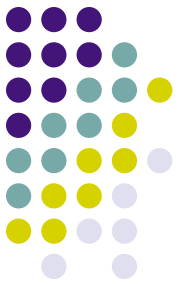
# Platforms vs. Apps

- Microsoft and Apple compete with ISV's
  - Build Platforms and Apps that show them off
  - This is now accepted practice
- Third-Party Software vendors are declining
  - Domain-specific expertise (Legal, Graphics, etc)
- Web-Based Software is proliferating

# Who are the Tools For?

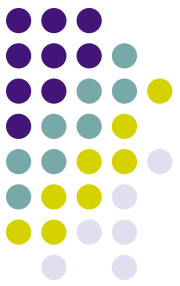
- For Microsoft App Teams?
- For Multi-Platform developers?





# #if ... #else ... #endif

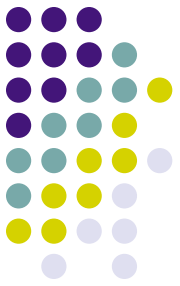
- One Source File, Multiple Platforms?
- What is a “project”?
  - Collection of Source Files
  - Graphical Assets
  - Configuration Settings
  - Specification of Resulting “App”



# “Add to Project...”

- What a pain in the neck
- “Add to 5 Totally Different Projects”
  - Makefile for UNIX
  - Cocoa IDE for Mac version
  - What Else?
- Intelligent Directories
  - Mac Version
  - Windows Version
  - Compile everything in here, please!

# Projects, Source Code, IDE' s



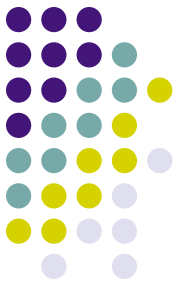
- It' s very hard to share code across platforms
- IDE' s / Makefiles / Projects are difficult to manage
- It' s far more than just the “runtime”
- C# vs. C++ vs. Java vs. PHP vs. Whatever





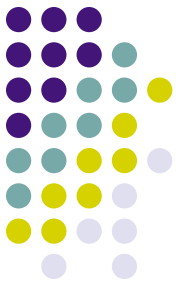
# “Making things Easy”

- For Whom?
- At What Cost?
- Can I really Use the “Easy” Stuff?
- Do I have a standards-based alternative?



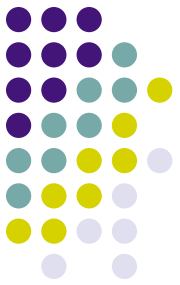
# Multiple Views

- Easy In, Easy Out
- Multiple Representations
  - This file is “in”, that one is “out” (debug build)



# What is an App?

- A Solution to a Problem
  - Does something “useful”
  - Does something “hard”
  - Runs on some platform(s)
  - Has some “market”



# The Market Chooses

- Compelling Apps drive platform choice
  - gaming machines
  - Photoshop machines
  - Autocad machines
- Compelling Apps don't use platform features
- Be “best delivery platform” for good ideas
- Consider iTunes for Windows
  - 95% of iPod owners run Windows

# Thanks

- Hopefully Added some Perspective
- Happy to Chat afterwards
- Thanks to William Parkhurst

