

What's New In Cincom Smalltalk

Alan Knight (knight@acm.org)
Cincom Systems of Canada

About the Talk

- Some New and Interesting Stuff
 - ⇒ Things already shipping
 - ⇒ Things in the next release
 - ⇒ Longer term goals
 - ⇒ Non-technical items (consolidation, releasing)
- Both major features and selected smaller items
- Not nearly comprehensive
- Some technical detail

ObjectStudio 8

- ➔ Separate product
 - ⇒ Shipping now-ish
- ➔ Full Re-implementation
 - ⇒ ObjectStudio semantics
 - ⇒ ObjectStudio class libraries
 - ⇒ Native Widgets (Windows)
 - ⇒ Access to all VisualWorks facilities
 - ⇒ Greatly improved VM performance

Object Studio 8 – Technical Challenges

- VM issues for native widgets
- Different semantics
 - ⇒ Methods, assigning to parameters, {}
- File-based
- Solutions
 - ⇒ Namespaces for classes
 - ⇒ Different compiler, and code rewriting
 - ⇒ Message rewriting, and code transformation

ObjectStudio 8 – ObjectStudio Users

- ➔ Ability to use VisualWorks tools where they are an improvement
- ➔ Access to class libraries
 - ⇒ Web/Application Server
 - ⇒ Web Services and Networking
 - ⇒ Database
- ➔ VM performance
- ➔ Continuing Development

ObjectStudio 8 – VisualWorks Users

- Native Widgets
- Vista certification pending
- COM and ActiveX support
- “Legacy” connects
 - ⇒ APPC
 - ⇒ EHLLAPI

Virtual Machine

- ➔ Intel Mac OS X Support

 - ⇒ Shipped in 7.5

- ➔ General Mac OS X Stability and Performance

 - ⇒ Improved in 7.5

 - ⇒ Much continuing work

 - ⇒ Single threaded VM

Mac OS X Virtual Machine

- ➔ Single Threaded VM
 - ➔ Worse performance right now
 - ➔ Better performance once optimized
 - ➔ Much more stable
 - ➔ Much easier interface to Mac system code

Virtual Machine

- ➔ VM as DLL
 - ⇒ Smalltalk doesn't have to be in charge
 - ⇒ Detailed planning stages
- ➔ 64-bit stability and exploiting features
 - ⇒ E.g. hashing
- ➔ Socket and THAPI performance
- ➔ Meta-issues
 - ⇒ Source code control
 - ⇒ Consolidate platform-specific code
 - ⇒ Regression test suite

Aside: Hashing

- ➔ Major rework of hashes
 - ➔ Analyze existing hashes for better distribution
 - ➔ String hash that takes into account all characters
 - ➔ Single level hash for symbols
 - ➔ Note: Hash Analysis Tool in public repository

Widgetry

- ➔ Entirely new windowing framework
- ➔ In progress for a long time
- ➔ Version 1.0 shipping shortly
 - ⇒ “Service Pack” release
 - ⇒ Supported product

Widgetry - Benefits

- ➔ Simpler, cleaner model
 - ➔ “Wrapper” system not comprehensively updated for many years
- ➔ More functionality
- ➔ Better compatibility with other systems
- ➔ Easier to support native widgets (eventually)

Widgetry - Issues

- ➔ Documentation not shipping with 1.0
 - ⇒ Some web resources available
- ➔ No UIBuilder or migration support
- ➔ Tools migration slower than planned
- ➔ Suitable for
 - ⇒ Early adopters
 - ⇒ Those with sophisticated UI requirements that can't be met with "Wrapper"

Seaside

- ➔ Major new initiative
- ➔ Potentially significant opening in web development space
 - ⇒ J2EE, .NET, Web Services extremely heavyweight
 - ⇒ Lighter frameworks, esp. Ruby on Rails
- ➔ Outreach to non-Smalltalk developers
- ➔ We can do better

Seaside - Web

- ➔ Already a working VisualWorks port
 - ⇒ Michel Bany
- ➔ Integration with VisualWorks
 - ⇒ Access to facilities
 - ⇒ Supported
 - ⇒ Cleaner portability
 - ⇒ Simpler underlying layer – fewer choices
 - ⇒ Easy to use out of the box

Seaside - Technical

- ➔ Running on Opentalk-HTTP
- ➔ Opentalk
 - ⇒ General distributed programming and network protocol support
 - ⇒ Basis for e.g. Web Services, CORBA
- ➔ Kept strongly in sync with Squeak version
 - ⇒ Tools for supporting easier interchange
- ➔ Relational Database support

Aside: Networking improvements

- ➔ Much better support for streaming large content
 - ⇒ HTTP Serving
 - ⇒ Mail messages (SMTP, POP, IMAP)
- ➔ Improved mail support in general
- ➔ Firewall/NAT improvements

Seaside - Database

- ➔ Be able to use Relational DBs easily
 - ⇒ Connectivity - MySQL support
- ➔ Parallels Ruby on Rails “ActiveRecord”
 - ⇒ Uses Glorp as underlying mapping layer
 - ⇒ Automatic for simple schemas
 - ⇒ Be able to use database information
 - ⇒ Scale up to more complex schemas
 - ⇒ Better performance
 - ⇒ Much more about this tomorrow

Deployment

- ➔ Make it much easier
 - ⇒ Various Runtime Packager improvements
 - ⇒ Make Runtime Packager less necessary
 - ⇒ Move functionality into base, be dynamic
 - » Headless
 - » Runtime
 - » Error Logging
- ➔ Smaller base image
- ➔ Alternative ways of running code

Scripting

- Do more without requiring the environment
 - ⇒ One liners from the command line
 - ⇒ Write scripts in files
 - ⇒ Entire programs in files
- Scripting extensions
- System more tolerant of headless mode
- Standard I/O
- Experimental Phases
 - ⇒ “Scripting Support” in public repository

Internationalization

- Everything should work, regardless of locale and character set.
- Areas
 - ⇒ Character display/fonts
 - ⇒ Character input
 - ⇒ Filesystems
- Mac OS X input manager
 - ⇒ Only Mac OS 9 right now
- Current Windows “Unicode Support”

Store – Shadow Compilation

➔ Shadow compilation

⇒ “Atomic loading”

⇒ Shipping in 7.5, but turned off

⇒ Can't yet load everything

» Method order dependency

» Custom compilers

» Semantics of the code (e.g. system overrides)

Store (7.5)

- Merge tool UI overhaul
- MSAccess officially supported
 - ⇒ StoreForMSAccess
- Logging
 - ⇒ Records what was loaded from the database
 - ⇒ By a 7.5 or later client
 - ⇒ Table TW_LoadRecord
- Overrides within Bundles
 - ⇒ Setting – required for ObjectStudio 8

DLL/CC

- ➔ Major overhaul
- ➔ Still in early stages
- ➔ Entirely new parser (SMACC based)
- ➔ Lighter weight
- ➔ Get rid of the strange classes
 - ⇒ Loading/saving issues
- ➔ Examining C++ connectivity options

DLL/CC - #linkedIn

- ➔ Small, but interesting, in 7.5
- ➔ If the library name is #linkedIn, it looks for the library in the current executable.
- ➔ Very useful for e.g. libc functions

Some small items

- ➔ Dependents, events, removed from base collections (7.5)
 - ⇒ They aren't supported anyway
 - ⇒ Large performance increase
 - ⇒ Avoids bottleneck
- ➔ Databases
 - ⇒ ODBC connection pooling, charsets
 - ⇒ Oracle scrollable cursors

More smaller items

- ➔ Gecko support for web browser plugin (preview, mostly because of doc)
- ➔ Many COM connect improvements
- ➔ UUID generation
- ➔ Announcements
- ➔ / and \ as Filename messages
- ➔ #any, #sort, #sorted
- ➔ Splash screen off by default

Upcoming

- ➔ Many browser and tool improvements
 - ⇒ See Travis for details
- ➔ DefaultPackageNamespaces
 - ⇒ Set a package's #namespace property
 - ⇒ Class extensions are compiled in that namespace
 - ⇒ The “inspect” problem.

The End

➔ Questions?