ÉTOILÉ STATUS UPDATE

FOSDEM 2012

http://www.etoileos.com
What is it?

• Étoilé is a user environment designed from the ground up around the things people do with computers: create, collaborate, and learn.
Goals

• Composite Document
• Collaboration
• Persistence & Versioning
• Clean, consistent and plastic UI
• Fast and Easy Development
Release Status

- 0.4.1 released in March 2009
- not supported anymore
- 0.4.2 released now
- EtoileFoundation released this week
- new LanguageKit release soon
- more to come
What’s new since last year?
Theming News

- Slow progresses…
- Resurrected Nesedah theme
  - still very much work-in-progress
- Menu theming support in GNUstep
GNUstep runtime
libobjc2 (GNUstep)

• Version 1.6 released in November
• Inspired by the Étoilé ObjC runtime
• Based on the new Apple runtime API
• Compatible with the old GNU runtime API
ObjC 2 Features

- Full ObjC 2 Support with Clang (partial with GCC 4.6)
  - Non-fragile Instance Variables
  - Properties
  - Fast Enumeration and Proxy
  - Blocks
  - etc.
New Features

- ARC and Boehm GC support
- Associated References
- Prototype support
- Blocks as methods
More Features

- Improved LLVM optimizations
- Small Objects hidden in pointers
- Property Type Encoding access
- Unified Exception Model for ObjC++
- More efficient low-memory profile
ObjC2JS
ObjC to JS

- Compiles C and Objective-C to JavaScript
- Walk the Clang AST to output JS
- Supports C pointer manipulation
  - based on AddressOf objects
  - pointer arithmetic just works…
  - `char *str = “bla”; char **ptr = &(str++);`
What is supported?

• array, structure, enum, union
• using WebGL ArrayBuffer objects
• heap and stack allocation
• static variables
• arithmetic overflow
• classes, objects and messages
Peculiarities

• Automatic Array bounds checking

• Garbage collected memory

• Integer-to-pointer casts are supported

• Dereferencing such a pointer causes a run-time error

• Pointer-to-integer casts are supported
Future Work

• Basic Foundation classes NSArray, NSString etc. implemented in JavaScript

• Basic Graphics classes NSView, NSBezierPath etc. based on the Canvas

• A event loop based on ETWebApp

• Click a button to turn a Étoilé application into a web app!
SourceCodeKit
SourceCodeKit

- Clang-driven
- Code Indexing
- Syntax Highlighting
- Code completion
- Now ARC compatible
- CodeMonkey will probably use it
Syntax Highlighting

SourceCodeKit vs VIM
DocGenerator
What’s New?

• API diagrams based on Graphviz
  • include classes, categories and protocols
• README, INSTALL and NEWS listed in the sidebar when available
  • use Discount to render them to HTML
• Global variable doc support
LanguageKit
What’s new?

• Boehm-based GC and ARC support
• Calling C functions without FFI
  • In Smalltalk just do C sqrt: 42 for sqrt(42)
• Small objects (strings, ints, etc.) support in libobjc2
• ObjC compatible-block
What’s new?

- dealloc to clean up objects in Smalltalk
- On the way
  - Debugging
  - AMD64 ABI
EtoileFoundation
EtoileFoundation

- Prototypes, traits
- Collection Protocols
- High-order Messaging
- FAME-inspired Metamodel
- Mirror-based Reflection
- UTI, UUID, History, Socket etc. classes
What’s new?

- Reworked collection protocols
- Rewritten prototype support
  - based on libobjc2 1.6
  - -clone, -slotValueForKey: etc.
- Rewritten Trait support
  - including new collection traits
What’s new?

• Rewritten UUID generation code
  • eliminate potential collisions on Linux in a tight loop
• GC and ARC compatible
• Much better API documentation
• Full trait implementation including
  • Composite trait and the Flattening Property that goes along
  • Exclusion and Aliasing operators
  • Mixin-style composition as an option
  • Thread-safe
Trait Example

NSDicionary *aliasedMethods =

    [NSDictionary dictionaryWithObjectsAndKeys: @"lost:", @"wanderWhere:"];

[[self class] applyTraitFromClass: [BasicTrait class]
excludedMethodNames: S(@"isOrdered")
    aliasedMethodNames: aliasedMethods
    allowsOverride: YES];
Collection Traits

- Protocols
  - ETCollection
  - ETCollectionMutation
- Traits
  - ETCollectionTrait
  - ETMutableCollectionTrait
CoreObject
What is it?

- An Object Store with advanced revision control e.g. selective undo
- We use it as a to build
  - Generic Object Manager
  - Compound Document Editor
- More in the CoreObject talk later
EtoileUI
What’s New?

- New aspect repository as generic model for styling, object palettes, template pickers etc.
- Redesigned copy support around ETCopier and a new `-copyWithCopier`:
  - aliasing is now symmetric between the source and original object graph
What’s New?

• Finished automatic layout update
• Performance improvement (display, reload)
• More control over object insertion in an item tree
Pick and Drop Control

- Drop operation types can now be controlled by both source and destination
- Metadatas can be pushed on a pickboard
- More control over the precise object
  - pushed on the active pickboard
  - inserted in the drop target
CoreObject Integration

• Many changes to support persistency
  • metamodel, new UI object base class, etc.

• History Browser

• Worktable
  • an experimental Compound Document Editor
Demo
Work-in-progress

- ParserKit
  - a new PEG-based parser
  - inspired by OMeta and PetitParser
- XMPPKit
  - extensive cleaning underway
Étoilé Developers

Eric Wasylishen  ♫ CoreObject, Theming, AppKit
Christopher Armstrong  ♫ CoreObject
Niels Grewe  ♫ DBusKit
Mathieu Suen  ♫ ParserKit
Alessandro Sangiuliano  ♫ XMPPKit
David Chisnall  ♫ libobjc2, LanguageKit, SourceCodeKit
Quentin Mathé  ♫ EtoileUI, CoreObject, DocGenerator

http://www.etoileos.com