VA Smalltalk Update

John O’Keefe
Chief Technical Officer
Instantiations, Inc.
Recent Events

- Completed first year as pure Smalltalk company
  - # of users and revenues continue to grow
- Growing Engineering staff
  - 4 engineers joined since Smalltalks 2010
  - Using contractors for additional capacity
Recent Events
Continued

• University Outreach
  • Hasso Plattner Institute Bachelor Project
    • GTK+ Bindings for Linux
  • Interested in more

• User Outreach
  • Conference participation
    • ESUG, Smalltalk Solutions, Smalltalks
  • VA Smalltalk Forum -> VA Smalltalk Google Group
    • Aggregated on [Smalltalk] http://forum.world.st
    • Previous forum content still available (static)

• New “Videos and Podcasts” pages on website
Recent Release History

- **V8.0 (May 2009)**
  - Seaside, Tabbed Browsers, Documentation delivery system
- **V8.0.1 (November 2009)**
  - Seaside update, ‘cdecl’ calling convention
- **V8.0.2 (May 2010)**
  - Seaside update, GLORP
- **V8.0.3 (February 2011)**
  - Seaside update, GUI improvements, added documentation
- **V8.5.1 (August 2011)**
  - See following slides
VA Smalltalk V8.5
August 2011

• Development Tools
  • Code Assist (code completion)

• Infrastructure
  • Logging Framework
  • Preference Settings Framework
  • Deprecation Exception

• Graphics and Windowing
  • Rebar Control
  • TIFF CCITT T.4 bi-level encoding

• Web Interface
  • Grease 1.0.5+ / Seaside 3.0.5+
  • HTTP Chunked Transfer Encoding
Development Tools
Code Assist

• **Problem:** Developers can create many more classes and methods than we can hold names for in our minds

• **Solution:** Use the computer’s calculating capabilities to predict names as they are being entered in the code editor
Development Tools
Code Assist

• General and extensible content assist framework
  • Customizable popup list displays actionable suggestions
  • Layout management functions controls popup location
  • APIs and Extension Points allow for customization

• Numerous configuration options
  • Enable and configure auto-popup of completion suggestions
  • Visibility settings (i.e. Show only Public method suggestions)
  • <Tab> or <Enter> to accept completion suggestion
  • Automatic completion if single available suggestion
Development Tools
Code Assist

- **Context-Sensitive Auto-Completion**
  - Supported in Browsers, Workspaces and Debuggers
  - Suggestions for Methods, Symbols/Atoms, and Variables
  - Uses parse tree analysis and type reconstruction to provide most relevant suggestions

- **Smart Suggestion Sorting**
  - *Variable Suggestions*: Locals before pseudo variables before pool variables
  - *Method Suggestions*: Public before Private
  - *Class Suggestions*: Classes that extend SubApplication after other classes
Development Tools
Code Assist

- Suggestion descriptions provide additional details
  - *Methods*: Class in which the method is defined or common superclass
  - *Scoped Pool Variables*: Value of the pool variable
  - *Unscoped Pool Variables*: Pool Dictionary in which it is defined

- Suggestions offered for ambiguous receivers
  - *OrderedCollection new add: #(1 2) be<Activate CA>*
    - Does the user want to complete *before*: for the receiver: *(1 2)*?
    - Does the user want to complete *add:before*: for the receiver: *OrderedCollection new*?
    - Only the user knows; both suggestions are offered
Development Tools
Code Assist

• Parentheses auto-inserted for method completions
• Method override suggestions
  • Requesting Code Assistance at the top of a code browser will offer methods available to be overridden
  • Arguments are automatically inserted upon completion of an overridden method
Infrastructure
Logging Framework

- **Problem**: The product currently contains many one-off logging solutions, but no centralized facility useable by product developers and customers.
- **Solution**: Provide a Logging Framework to standardize the definition, use, and output formats of logging.
Log4s is based on ideas from the popular Java logging framework *log4j*
Example of application logging requirement
- A banking company is required to keep monthly logs of all foreign transactions and weekly logs of all foreign transactions greater than $10000
Infrastructure
Logging Framework

• Example of application logging solution
  • Make .ini file entries
    [log4s]
    createLogger=('vast')

    dailyRollingFileAppender=(ForeignTxns, root, c:\logs\foreignTxns.log, false, info, EsPatternLayout, '%d [%c] %o', true, topOfMonth)

    dailyRollingFileAppender=(BigForeignTxns, vast, c:\logs\bigForeignTxns.log, false, warn, EsPatternLayout, '%d %c %o', true, topOfWeek)

    The pattern of %d %c %o will log the time, the logger name, and the transaction object.
Infrastructure
Logging Framework

• Define `Transaction >> printLog4s`
  `^ String streamContent: [:stream | 
    stream nextPutAll: self payee; space; nextPutAll: 
    self amount printString ]`

• The application code might look like this:
  `logIfNeeded: aTransaction`

    aTransaction isForeign ifTrue: [
      EsLogManager
      info: 'Foreign txn'
      object: aTransaction. "goes to root logger"
    aTransaction amount > 10000 ifTrue: [
      EsLogManager
      loggerNamed: 'vast'
      warn: 'Large Foreign txn'
      object: aTransaction ] ].
Infrastructure
Logging Framework

- Result
  - `foreignTxns.log` will have output like this:
  - `bigForeignTxns.log` will have output like this:
    '28 Aug 2011 08:15:07,000 vast Fred Smith 23645.23'
Infrastructure Preference Settings Framework

- **Problem**: Existing customization preference settings are not well-managed or well-documented. In some cases, they are completely hidden.

- **Solution**: Provide a Preference Settings Framework to standardize the management and documentation of preferences.
Infrastructure Preference Settings Framework

- Preferences are held in `.ini` file
  - Major groupings are stanzas -- generally named the same as the application they relate to
  - `keyword=value` entries hold the settings
  - Values are typed
    - Simple: boolean, decimal, directory, file, fraction, integer, multiLineString, number, point, string
    - Complex: array, range
Infrastructure
Preference Settings Framework

- Preference settings are managed by Applications
- The user implements 3 methods to support preference settings for an application
  - `validSettings`
  - `currentSettings`
  - `setCurrentSettings`
- Preference settings are read from the `.ini` file
  - At image startUp
  - When an application is loaded
Infrastructure Preference Settings Framework

- **validSettings**
  - Provides valid element types for keywords within stanzas
  - Example

```smalltalk
validSettings ^ LookupTable new
  at: self symbol asString
  put: (LookupTable with: 'fplevel' -> (0 inclusiveTo: 9) elementType: self integerType));
  at: 'VAST 5.0 Compatibility'
  put: (LookupTable with: 'installPath' -> self directoryType));
  yourself
```
Infrastructure
Preference Settings Framework

- **currentSettings**
  - Provides current and default setting values for keywords within stanzas
  - Example

```smalltalk
currentSettings ^ LookupTable new
  at: self symbol asString put: (LookupTable with: 'bitmapPath' -> (Array with: CgScreen bitmapPath with: #()));
  yourself
```
Infrastructure
Preference Settings Framework

- **setCurrentSettings**
  - Transfers setting values to their ‘used’ location for the application
  - Example

```smalltalk
setCurrentSettings
| addr |
addr := self settingFor: 'ServerAddress'.
addr isEmpty ifTrue: [ addr := nil ].
EmLibrary
  defaultName: (self settingFor: 'DefaultName');
  serverAddress: addr;
  openReadOnly: (self settingFor: 'OpenReadOnly')
```
Infrastructure
Deprecation Exception

• **Problem**: Frameworks evolve and some API methods are destined to be removed in favor of replacement methods, but customers may have code dependencies.

• **Solution**: Mark methods as deprecated and gather usage information
Infrastructure
Deprecation Exception

• Technique
  • Categorize method as ‘Deprecated’
  • Replace method content with:

    self
    deprecated: 'explanation string'
    in: 'version string'.
    ^ ...invocation of alternate implementation...

• Example

    asDecimal
    ^ self
    deprecated: 'Replaced by #asScaledDecimal'
    in: 'V8.5';
    asScaledDecimal
Infrastructure
Deprecation Exception

- *Deprecation* is subclass of *Warning*
- Default handling based on preference
  - *Logging*: write message to in-memory log
  - *Showing*: write message to Transcript (TranscriptTTY in runtime)
  - *-raising*: open MessagePrompter allowing continue or terminate
- Documentation provides sample code to dump in-memory log
Graphics and Windowing
Rebar Control

- **Problem**: Windows customers want Toolbars with adjustable content.
- **Solution**: Add support for Rebar control
Graphics and Windowing
Rebar Control

- Specialized container for widgets
- Holds bands
  - Bands may be moved and resized
  - Band may hold
    - Gripper (for resizing)
    - Child widget
    - Label
    - Bitmap
Graphics and Windowing
TIFF CCITT T.4 bi-level encoding

• **Problem**: Customers are receiving FAX documents encoded as TIFF CCITT T.4 that they can’t process
• **Solution**: Add support for CCITT T.4 bi-level encoding (read and write) to existing TIFF support
Web Interface
Seaside 3.0.5+ / Grease 1.0.5+

- **Problem**: Seaside and Grease development has progressed since VA Smalltalk V8.0.3
- **Solution**: Port Seaside 3.0.5 and Grease 1.0.5 (both with updates through 8/9/2011 – exact level identified in configuration maps)
Web Interface
HTTP Chunked Transfer Encoding

• **Problem**: Customers need to maintain persistent connections for HTTP with dynamically generated content (size not known when connection established).
• **Solution**: Add support for HTTP Chunked Transfer Encoding. Chunked HTTPRequests and HTTPResponses (including web services) can be processed.
Support

- 60+ bug fixes and minor enhancements
Looking to the Future
Future Releases

- Release schedule is about twice a year
  - Next release is planned for end of Q1/2012
  - Current information available in Product Roadmap
- Content based on requirements from:
  - Direct customer interactions
  - Forums
  - Support cases
  - Internals
V8.5.1
Planned Content

• GUI Look and Feel
  • Windows Controls
    • Rebar improvements
    • ProgressBar improvements
    • DateAndTimePicker
  • Support for .PNG files

• Web interface
  • Seaside 3.0.6.3 (early)
  • Seaside 3.x (later)
  • SST Servlet multipart form support
V8.5.1
Planned Content

• Development Tools
  • Code Assist improvements
  • SUnit Expected Failure support

• Middleware
  • GLORP improvements
  • MQ currency

• Platform
  • Windows Services control moved from C to Smalltalk
V8.5.1
Planned Content

• Performance and Scalability
  • SST Lightweight Marshaling improvements

• Installation
  • Changing to standard platform installer
    • Windows: InstallShield
    • Unix: .rpm, .deb, ...
Priority Technologies

- Internationalization
- Web interface
- Middleware
- GUI Look and Feel
- Development Tools
- Security
- Performance and Scalability
- Platforms
- External Interfaces
- Other
Future Releases
Candidate Items

- Internationalization
  - Full Unicode/UTF-8 (including VM)
- Web interface
  - Seaside 3.x
  - Continuation support
  - Web services tooling improvements
  - Web services debugging tools/doc
  - Validating XML parser
Future Releases
Candidate Items

• GUI Look-and-Feel
  • GTK+ 2.x on Linux
  • Windows Common Controls additions
  • Icon/image support enhancements
  • Backport widgets from add-ons

• Development Tools
  • Redesigned Change Browser & Merge Tool
  • Additions to Code Assist
  • Monticello Importer
Future Releases
Candidate Items

• Infrastructure
  • Settings Dialogs to complement Settings Framework
  • Consolidate most settings using Settings Framework

• Middleware
  • DB2 Stored Procedures improvements
  • GLORP infrastructure improvements
  • GLORP Programmer’s Reference
  • Active Records built on GLORP
  • TCP/IP V6
Future Releases
Candidate Items

• Security
  • “Basic” security framework (consolidate existing OpenSSL wrappers)
  • “Full” security framework -- OpenSSL 1.0 wrappers

• Performance and Scalability
  • Incremental garbage collection
  • 64-bit Smalltalk
  • Class library performance hotspots
  • Integrate KES/Stats tool for object monitoring
Future Releases
Candidate Items

• External Interfaces
  • JNIPort
  • .NET/C# Other

• Class Libraries
  • Collection hashing policies
  • Collection sorting policies

• Other
  • We’re always looking for suggestions
How Do You Get VA Smalltalk?

- Download evaluation copy

- Buy development licenses

- Download development build
  - Announced in VA Smalltalk Google Group

- Be a committer on an Open Source project
  - [http://www.instantiations.com/company/open-source.html](http://www.instantiations.com/company/open-source.html)

- Work for an educational institution
  - [http://www.instantiations.com/products/academic-license-program.html](http://www.instantiations.com/products/academic-license-program.html)
Contact us

- General information
  - info@instantiations.com
- Sales
  - sales@instantiations.com
- Support
  - support@instantiations.com
- Me
  - john_okeefe@instantiations.com