Scintillating! A Modernized Text Editor for VA Smalltalk

Seth Berman
Software Engineer
Instantiations, Inc.

instantiations
VA Smalltalk

Requirements

- Provide a modern text editor
  - Additional visual cues and styling
  - Take advantage of the latest technologies
- Minimize change to our existing system
  - Maintain full API compatibility with existing editor
  - Structural compatibility with our widget frameworks
- Must be inline with our cross-platform roadmap
  - GTK
- New capability must be made accessible to our customers

Scintilla

- Free library providing functionality to help build text editors
- Provides numerous features specific to source code editing
- Initial release in 1999
- Active community
- Used by Code::Blocks, Notepad++, TortoiseSVN
- Cross-Platform support

Scintilla Integration in VAST

- Integrated into our Common Widgets Framework
- New CwScintillaEditor widget integrated
- Offers full API support for Scintilla 3.3.3 (the latest)
- Compatibility methods implemented to provide 100% API capability with our existing legacy editor components

instantiations
VA Smalltalk

Direct2D/DirectWrite

- Microsoft's technology to provide higher quality font rendering
- Hardware-Accelerated Rendering
  - Offloads many aspects of rendering to the GPUs
- Windows 7 and above
- How noticeable a difference? Depends on
  - Font type and style
  - Monitor type and size
  - Your eyes and/or attention to detail

instantiations
VA Smalltalk

Internationalization

- Double Byte Support
  - Japanese, Chinese and Korean DBCS
- Input Method Editor (IME) Support
  - Used to assist typing in languages with thousands of characters

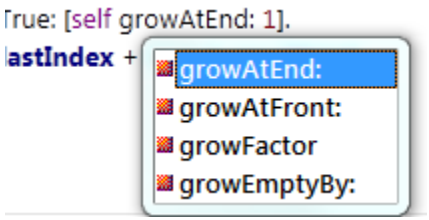
Text Editor Basics

- Auto-Indent
- Keyboard Shortcuts
  - <Tab> to indent, <Shift+Tab> to unindent
  - <Alt+Up/Down> to move blocks of selected text one line at a time
- DragNDrop to relocate blocks of selected text
- Margin Area

Multiple Undo/Redo

- Finally!
- No hard limit (only memory)
- Supports coalescing
  - Combine contiguous insertions and deletions into single undo operations
- APIs for user-defined coalescing

Code Completion



- Scintilla offers a code completion popup and API
- We hooked it up to our code completion engine
- Users have the option to select which popup they prefer
  - Basic (Minimal styling capability)
  - Extended (Maximum styling capability)
  - Scintilla (Somewhere in the middle)

instantiations
VA Smalltalk

Syntax Color Highlighting

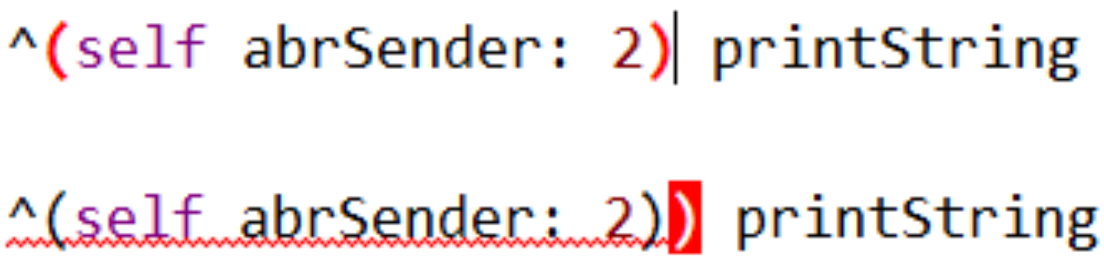
- Scintilla Lexer defines how a specified range of text is to be colored
  - Has lexer support for 80+ languages
  - Smalltalk is one of them, but it was too simplistic
  - Provides hooks to allow us to write our own custom lexer in Smalltalk (Container-Defined Styling)
- Container-Defined Styling
  - Scintilla specifies what needs to be styled via events
  - VAST's custom styler defines how the character range is to be styled

VAST Custom Styler

- Comes in 2 flavors
  - Method styler to style text in browsers and debuggers
    - Optimized to style methods
  - Snippet styler to style text in inspectors and workspaces
    - Optimized to style snippets of code
    - Adds some fuzzy logic rules
    - Now we can offer color in our inspectors and workspaces
- Styler uses a custom token scanner instead of parse trees
  - Enables real-time coloring
  - Much more flexible then our previous parse-tree implementation

Bracket Highlighting

- Stylization used to indicate matching characters for ()[]{}
- Separate style used to highlight unmatched characters

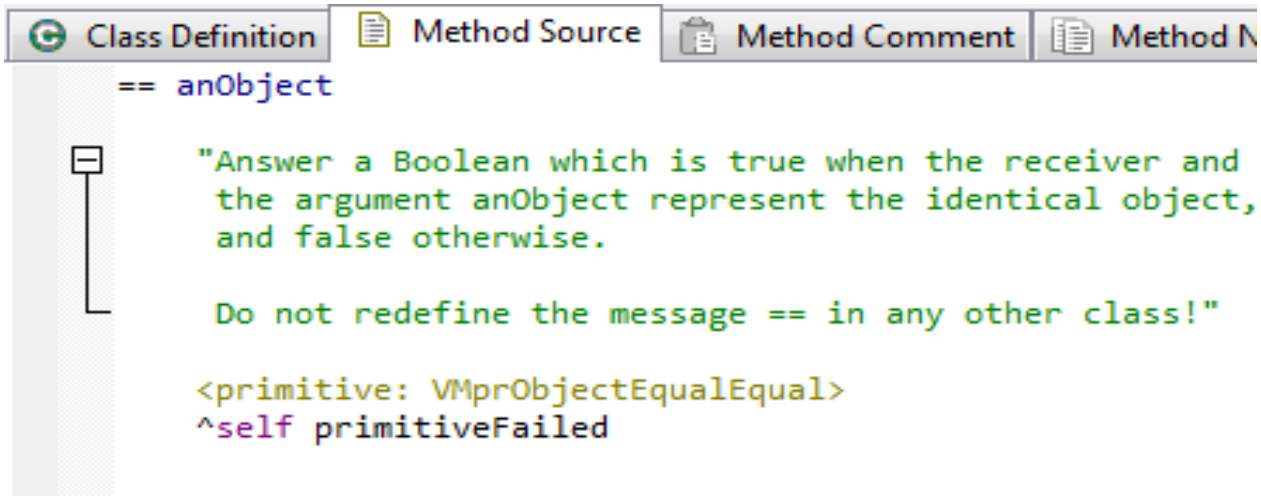


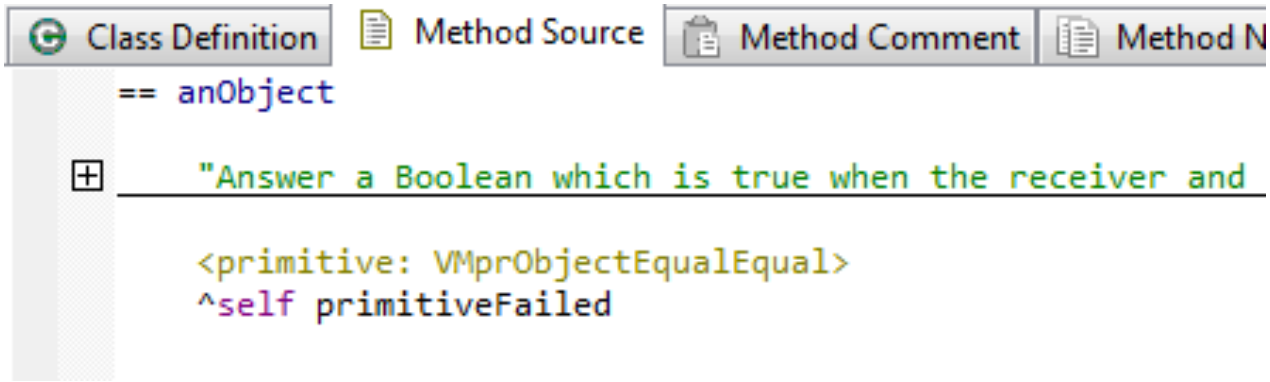
instantiations
VA Smalltalk

Smart Highlighting

- Adding stylization to a selected word and any matching word in the source
- Useful seeing local variable usage
- Adds extra decoration to highlighting browsers
- Added logic to handle block argument highlighting
- Stylization applied behind text so syntax color highlighting shows through

Code Folding Cont...





instantiations
VA Smalltalk

Line Numbers

- Available in Browser/Debuggers/Workspaces/Inspectors
- Configurable across different window types
- Line Number margins are dynamically sized

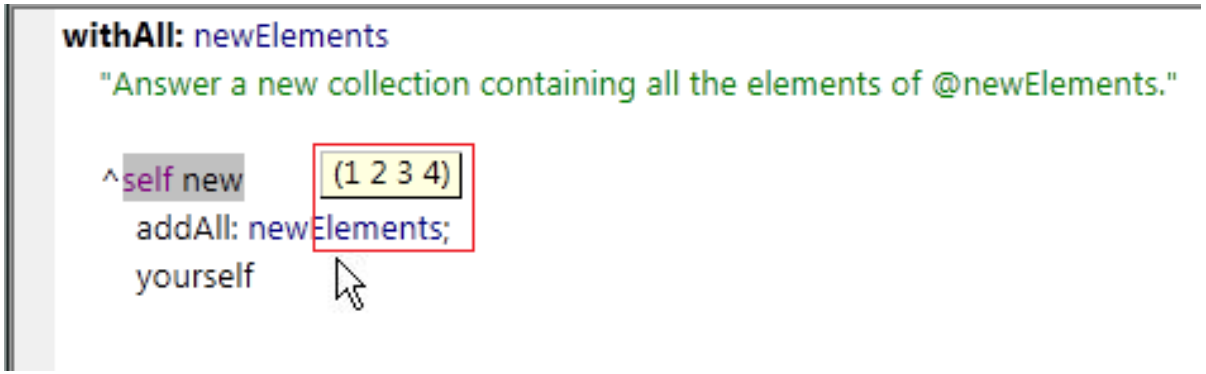
Breakpoint Management

- Persistent Breakpoint margin
- New Breakpoint icons
- Multiple Breakpoints / Line support
- Stylizes Breakpoint regions

instantiations
VA Smalltalk

Debugger Calltips

- Hover mouse over variables and globals displays a calltip showing it's value
- Clicking a calltip will bring up an inspector
- Long calltips are formatted to single-line



instantiations
VA Smalltalk

Error/Warning/Info Indicators

- Stylized Squiggle lines underneath text to indicate
  - Fatal Errors
  - Errors
  - Warnings
  - Info
- Calltips provide information about the indicator
- Incremental compiler runs in the background to identify issues in real-time
  - Collects information from parse trees and the styler

Questions?

instantiations
VA Smalltalk