Scwm:
Scheme Constraints Window Manager

Greg J. Badros
gjb@cs.washington.edu
University of Washington, Seattle
Linux Users of Victoria
4-May-1999

What is Scwm?

A free, fantastically configurable, fully-programmable, self-documenting, modular X11 window manager that embeds Guile/Scheme, and integrates with GNOME, GTk+, and KDE.

Overview of Talk

- History
- Design principles
- Demonstration
- Implementation
- Scheme and Scwm
- Future work
- Questions

What is a window manager?

- Mostly-ordinary X11 client application
- Decorates windows
- Permits manipulation of windows
  - moving, resizing, (de)iconifying
  - virtual desktops/workplaces
- Global X11 key and mouse bindings
- Inter-process communication via windows

Conceiving Scwm

Wouldn't it be great if:

- fvwm2 had a real language; or
- GWM embedded Guile/Scheme instead of WOOL

- Window manager functionality would be new primitives in the language....

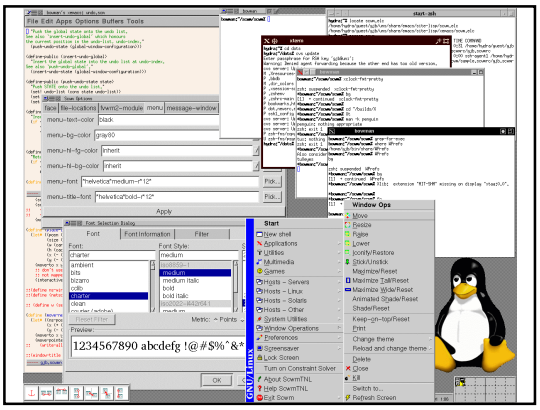
Birth of Scwm

- Maciej Stachowiak, September 1997
  fvwm2 ⇒ Scwm-0.1
- Used GNU Guile v1.2
- A Turing-complete window manager!!!
- Most fvwm2 functionality

Design Principles

- Make it work early and always
- Replace large features one at a time
- Migrate features to Scheme level
- Be modular—pay only for what you use
- Be extensible—plan for the future

4-May-1999 G. Badros, Scwm 6



Scwm Implementation

- 24,000 NCNB lines of C
- 15,000 lines of Scheme with Scwm
- Guile: 42,000 lines of C; 9,000 of Scheme
- Cassowary: 7,000 lines of C++

4-May-1999 G. Badros, Scwm 8

Downsides of Scwm

- Requires many skills of developers
- On the large side: 480k binary
- ~3 MB process image at minimal startup
- Integrates tightly with Guile, other code
- BUT... libraries are shared, Guile (1.8MB w/o Scwm) is slimming down
- Non-trivial to build... lots of dependences

4-May-1999 G. Badros, Scwm 9

.scwmrc configuration file

- Evaluated by Guile
- Sets up the initial environment
  - decorations
  - key, mouse bindings
  - run clients
  - read configuration
  - ...virtually anything you want!!!

4-May-1999 G. Badros, Scwm 10

Scwm: A WM that can add 2+2

(+ 2 2) ⇒ 4
(* 2 (+ 2 1)) ⇒ 6
(interactive-move) ⇒ #<unspecified>

- `scwmexec ´(+ 2 2)´; scwmrepl`
- GNU Emacs' scwm-mode
  Has integrated help!

4-May-1999 G. Badros, Scwm 11

Interactive features

```
(select-window-interactively)
  ⇒ #<window 113246217: "xterm">

(display-message-briefly "Hello world!")

(bind-key ´all "C-S-M-t" ns-goto-cutbuf-url)

(bind-mouse ´root 1
  (lambda () (popup-menu root-start-menu))
```


Scheme

- R4RS = Revised^4 Report on Scheme
  (now R5RS is the standard)
  - Only 50 pages describing the whole language
  - Strong theoretical foundations
  - Becoming a favourite introductory language
- Guile/Scheme
  - GNU project's scripting language
  - Many users, active development

Higher-order functions

```
(for-each iconify (list-all-windows))
⇒ #<unspecified>

(map window-title (list-all-windows))
⇒ ("Xterm" "xlogo" "Emacs" "Emacs" "FvwmPager")

(list-all-windows
      #:only (window-title?? "Emacs"))
⇒ ("Emacs" "Emacs")
```


Anonymous functions

```
(define (push-window-right win)
  (move-window-relative win 1 0))
(map push-window-right
  (list-all-windows))

(map (lambda (win)
           (move-window-relative win 1 0))
  (list-all-windows))
```


Hooks for extensibility

```
(add-hook! change-desk-hook
  (lambda (old new)
    (display-message-briefly
      (string-append "Desk #"
        (number->string new)))))
```

Also remove-hook!, add-timer-hook!

Extensible GUI Configurability

```
(define-scwm-option *menu-font* "fixed"
  "The font to use for menu items."
   #:group ´menu
   #:type ´font
 ...)
```

Scheme evaluation

(* 2 (+ 2 1)) ⇒ 6

Procedure * (multiplication) will be applied to arguments: 2 and (+ 2 1)

Arguments are evaluated first (eagerly): 2 ⇒ 2, (+ 2 1) ⇒ 3

Then * is invoked on 2 and 3

4-May-1999 G. Badros, Scwm 18

Prefix vs. infix

- Infix: 2 * (1 + 2)
  - Group explicitly to override precedence
  - Operator must be repeated when applied to more than 2 arguments
- Prefix: (* 2 (+ 1 2))
  - Lots of parentheses
  - (+ 1 2 3 4 5) ⇒ 15
- **Same execution order!**

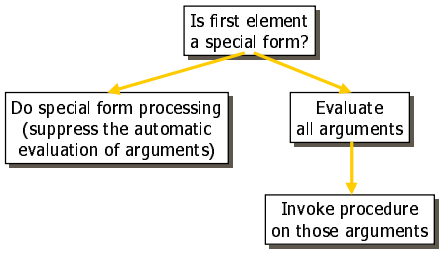
4-May-1999 G. Badros, Scwm 19

Scheme literals

- Numbers: 2 ⇒ 2
- Strings: "foo" ⇒ "foo"
- Symbols: sum ⇒ #<procedure sum>
- Lists: (sum 7 8) ⇒ 15
  Evaluate by procedure application
- Suppress evaluation by quoting
  ´(sum 7 8) ⇒ (sum 7 8)

4-May-1999 G. Badros, Scwm 20

List evaluation



4-May-1999 G. Badros, Scwm 21

Quoting

(quote (+ 1 2 3)) ⇒ (+ 1 2 3)

- quote is a special form that returns its single argument without evaluation
- ´(+ 1 2 3) ⇒ (+ 1 2 3)
  Shorthand for above
- (eval ´(+ 1 2 3)) ⇒ 6
  eval evaluates its single argument
- (eval ´´(+ 1 2 3)) ⇒ (+ 1 2 3)

4-May-1999 G. Badros, Scwm 22

Other special forms

```
(define-public (move-viewport x y)
  "Move the viewport of the virtual desktop by X, Y."
  (let ((pos (viewport-position)))
    (set-viewport-position!
        (+ x (car pos)) (+ y (cadr pos)))))
```

viewport-position returns list (x-offset y-offset), e.g. (50 80)
(car ´(50 80)) ⇒ 50 ;; the number 50
(cdr ´(50 80)) ⇒ (80) ;; the list with one element 80
(cadr ´(50 80)) ≈ (car (cdr ´(50 80))) ⇒ 80

4-May-1999 G. Badros, Scwm 23

Future work

- Release v1.0... Real Soon Now!
- More documentation
- More fvwm2 code cleanup
- Move more functionality out of core
- Event rewrite
- Decoration rewrite
- Constraints research

4-May-1999 G. Badros, Scwm 24

More information...

- `www.cs.washington.edu/homes/gjb`
  `ph.mit.edu/pub/scwm`
- **Full distribution, supporting software**
- **Online help (HTML format)**
- **Introductory tutorial**
- **Mailing list archives**
  majordomo@ph.mit.edu, scwm-discuss

4-May-1999 G. Badros, Scwm 25