Hat: Windows and WIMP

Neil Mitchell

Progress Updates

I have:

- Ported the Hat tools to Windows
- Written Hat-make
- Library-fied some Hat tools
- Written Hat-Gui

Hat Windows Port

- Works reasonably well
- Windows dislikes:
  - Terminal escape codes
  - Spawning xterms
  - Shell script
  - Some C functions
  - Bad FilePath manipulations

Hat Windows Port (2)

- Does not include:
  - hat-explore - advanced escape codes, looks pretty hard to get round
  - hat-anim - escape codes, might be able to work round
  - hat-detect - tries to spawn an xterm
- Everything else works (pretty much)

Hat-Make

- Hmake doesn’t work on Windows
- Hat-make uses hat-trans to compile the files
  - Hat-trans has *serious* FilePath vs Windows issues (they remain unfixed)
  - A very hacky program
- Cabal hat the way forward?

Hat tools as a library

- Have converted hat-stack, hat-cover

```
hatCover :: FilePath -> [String] ->
                        IO [(String, [Interval LC])]

hatStack :: FilePath ->
                        IO (Maybe (String, [StackEntry]))
```

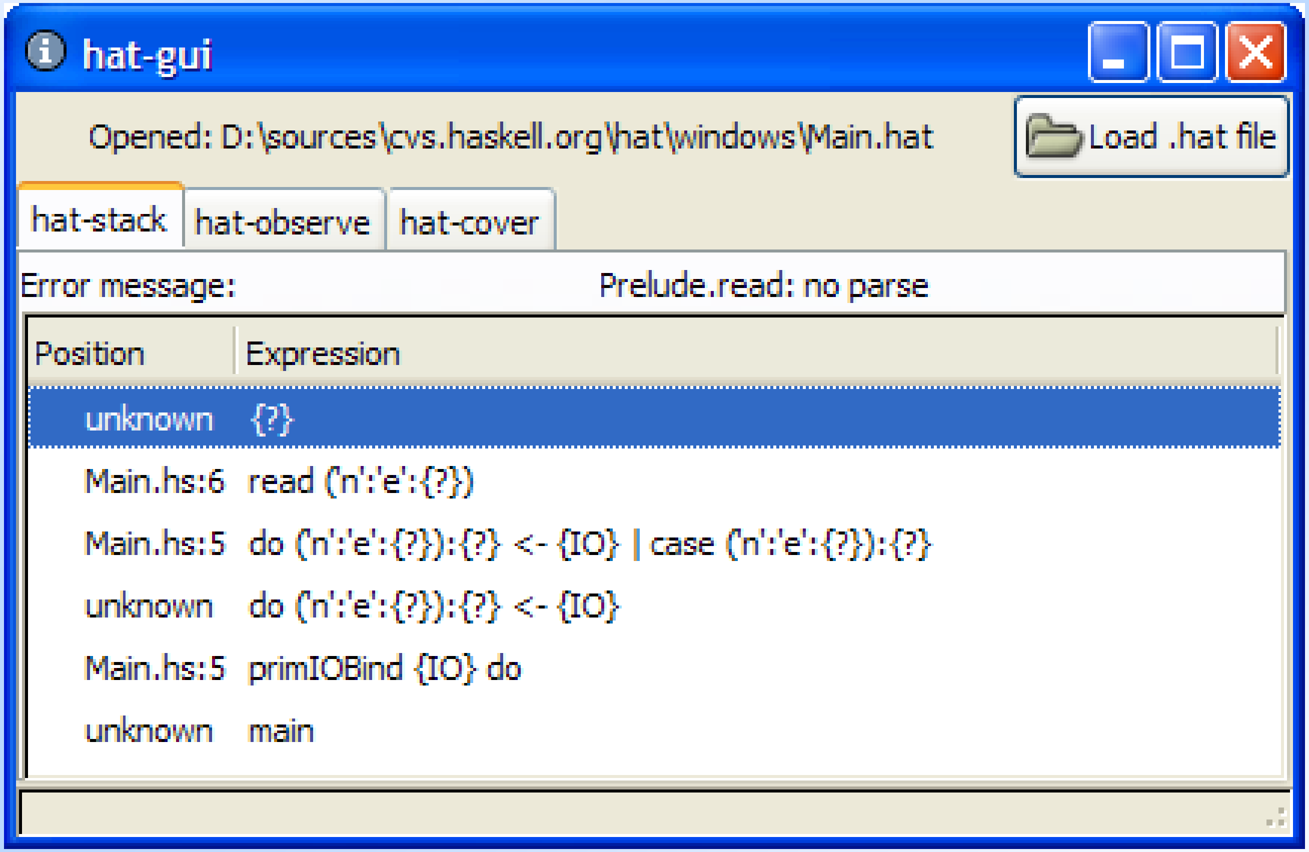
- Anyone want to help?

Hat-Gui

- Not a brand new design
  - The old tools repackaged
  - Less console, more WIMP
- Designed as easy to use

- Demonstration (technical circumstances permitting)

Screenshots 1

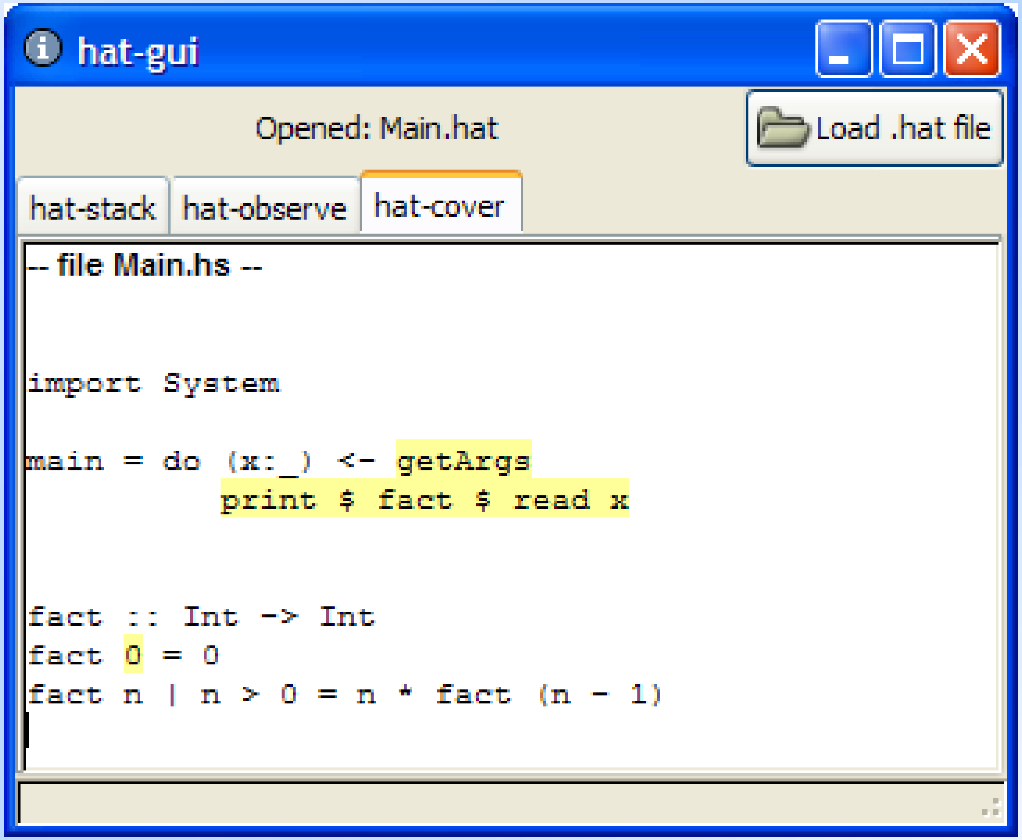


Opened: D:\sources\cvs.haskell.org\hat\windows\Main.hat

Error message: Prelude.read: no parse

Position	Expression
unknown	{?}
Main.hs:6	read ('n':'e':{?})
Main.hs:5	do ('n':'e':{?}):{?} <- {IO} | case ('n':'e':{?}):{?}
unknown	do ('n':'e':{?}):{?} <- {IO}
Main.hs:5	primIOBind {IO} do
unknown	main

Screenshots 2



```
-- file Main.hs --


import System

main = do (x:_) <- getArgs
          print $ fact $ read x


fact :: Int -> Int
fact 0 = 0
fact n | n > 0 = n * fact (n - 1)
```

Hat-Gui todo

- Lots of things left to do
  - Integrate all the tools
  - Link them in interesting ways
  - Not that hard to do
- Work suspended for now
  - An undergrad at Durham hopes to work on it

Windows Hat Users?

- At least 2! (but sadly not me)
- Bug reports always about Hat, not the port
  - Parsing problems (1e12)
  - Library support (Data.Map)
- Seem enthusisatic about Hat-Gui
  - *"wow, that looks great!"*

The Way Forward: Cabal

- Common infrastructure for building Haskell programs
- Supports GHC/Hugs, preprocessors
- Generates .exe, haddock, hoogle
- Bundling hat support in is sensible
- Will be used for the base libaries

My suggestions

- Easy to install
  - Windows installer, Linux packages etc
- Easy to generate .hat files
  - 100% library coverage would be nice
  - Integration in Haskell tools (WinHugs)
- Make it easy to discover the tools
  - GUI
  - Tutorial style documentation

Conclusions

- Plenty of Windows users want Hat
- Hat-make is just not good enough
- Hat-Gui could be very nice