Web based teaching through real-time collaboration with ghcLiVE

Shae Erisson

Department of Computer Science and Information Systems
University of North Alabama

April 9, 2013

Google Summer of Code

- offers student developers stipends to write code for various open source projects.
- over 4,500 students with over 300 open source projects
- started in 2005, now in its eighth year.

Haskell

Haskell is a general purpose purely functional programming language with non-strict semantics and strong static typing.

Factorial in Haskell

```
factorial n = product [1..n]
```

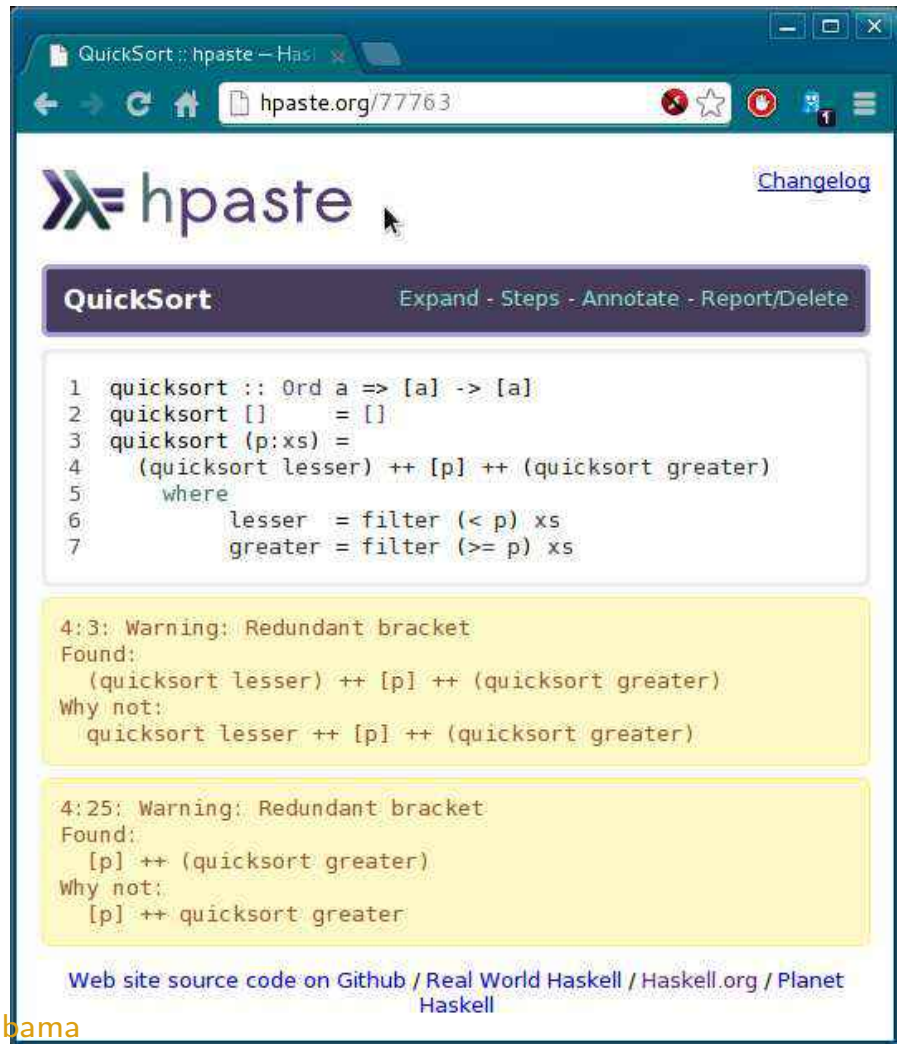
QuickSort in Haskell

```
quicksort :: Ord a => [a] -> [a]
quicksort []     = []
quicksort (p:xs) =
  (quicksort lesser) ++ [p] ++ (quicksort greater)
    where
        lesser  = filter (< p) xs
        greater = filter (>= p) xs
```

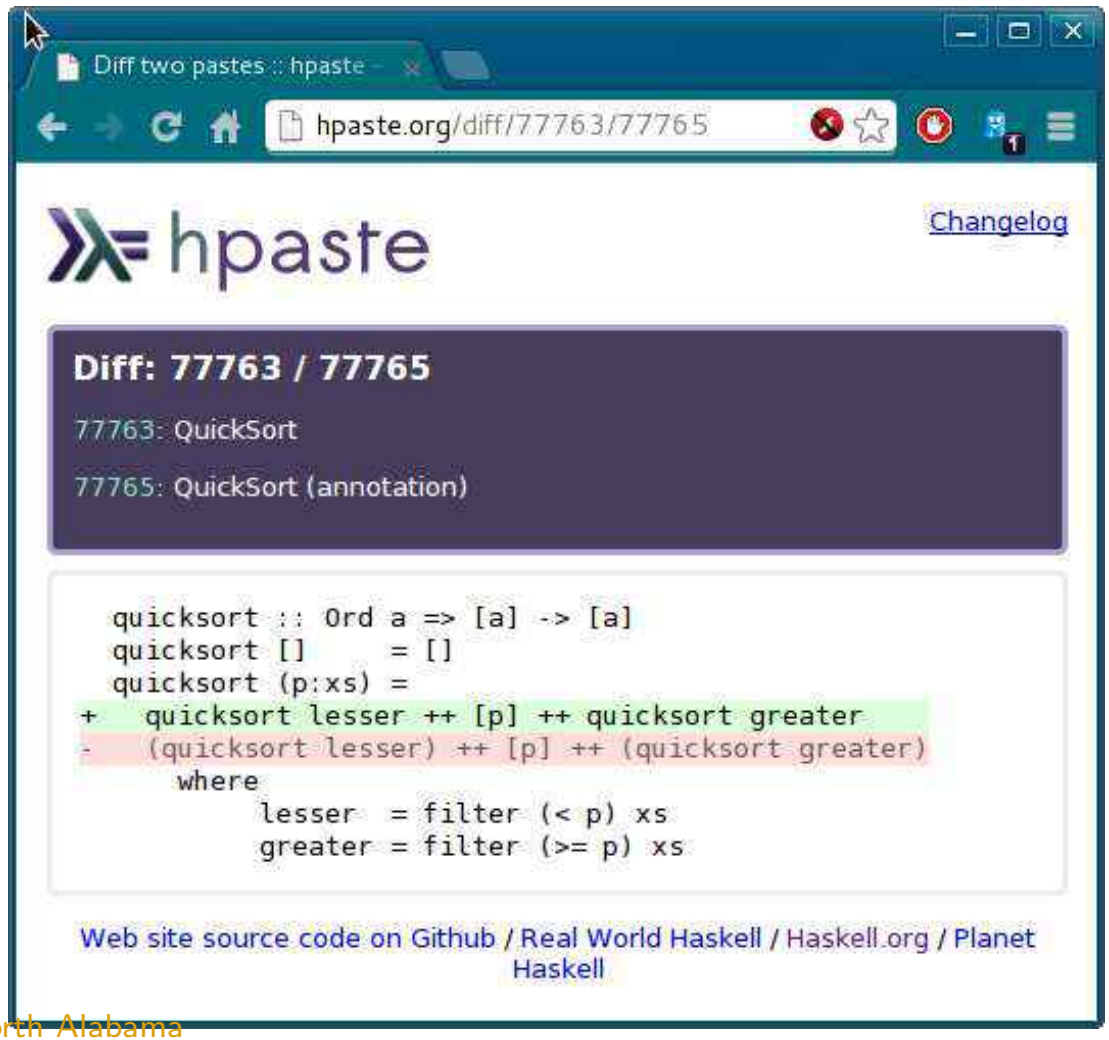
Internet Relay Chat

```
["Haskell - the language of ICFP winners 3 years running", "Dist
15:25  dons> ?wiki Monad
15:25  lambdabot> http://www.haskell.org/haskellwiki/Monad
15:25  dons> gives a few examples
15:26  sorear> So (with overloaded null operator) we can have e.g
               echo = putStrLn getStrLn
15:26  sorear> that desugars as echo = (ap putStrLn getStrLn)
15:26  sorear> the (I wish) bind instance triggers
15:26  sorear> echo = getStrLn >>= putStrLn
15:27  sorear> If it can be made to work... all the niceties of a
               full-blooded strict language
15:28  sorear> getStrLn = case getChar of { '\n' -> [] ; x -> x :
               getStrLn }
15:28  sorear> hmm, case is a problem
15:28  wy> I wonder why if (1<0) then a1+a2 else 2 work if a1 and a2
           are Natual type. There is a fromInteger in the instance.
           But why does if...then...else call that function?
15:29  sorear> sure... (1 < 0) === (fromInteger 1 < fromInteger 0)
15:29  sorear> (1 < 0) :: (Num a, Ord a) => a
15:29  sorear> oops
15:30  wy> sorear: but that has nothing to do with the return type
15:30  sorear> (Num a , Ord a) => Bool
 [15:30] [dons{+6e1}] [7:#haskell{+ns}] [Act: 2,3,10,14,16,20]
[#haskell]
```

hpaste


QuickSort :: hpaste – Has
hpaste.org/77763
hpaste
Changelog
QuickSort
Expand - Steps - Annotate - Report/Delete
1 quicksort :: Ord a => [a] -> [a]
2 quicksort []   = []
3 quicksort (p:xs) =
4   (quicksort lesser) ++ [p] ++ (quicksort greater)
5   where
6   lesser = filter (< p) xs
7   greater = filter (>= p) xs
4:3: Warning: Redundant bracket
Found:
(quicksort lesser) ++ [p] ++ (quicksort greater)
Why not:
quicksort lesser ++ [p] ++ (quicksort greater)
4:25: Warning: Redundant bracket
Found:
[p] ++ (quicksort greater)
Why not:
[p] ++ quicksort greater
Web site source code on Github / Real World Haskell / Haskell.org / Planet Haskell

hpaste

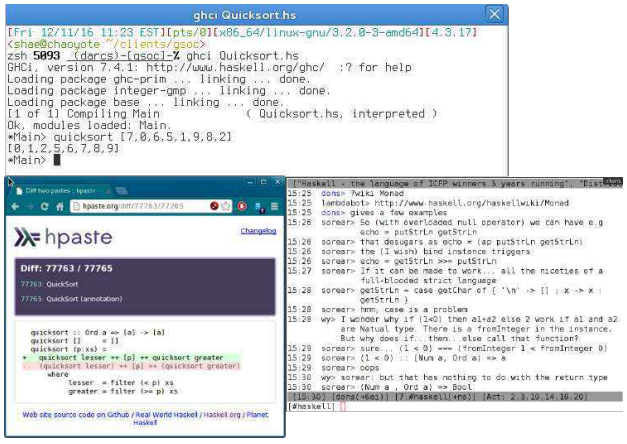

Diff two pastes :: hpaste
hpaste.org/diff/77763/77765
hpaste
Changelog
Diff: 77763 / 77765
77763: QuickSort
77765: QuickSort (annotation)
quicksort :: Ord a => [a] -> [a]
quicksort [] = []
quicksort (p:xs) =
+ quicksort lesser ++ [p] ++ quicksort greater
- (quicksort lesser) ++ [p] ++ (quicksort greater)
where
lesser = filter (< p) xs
greater = filter (>= p) xs
Web site source code on Github / Real World Haskell / Haskell.org / Planet Haskell

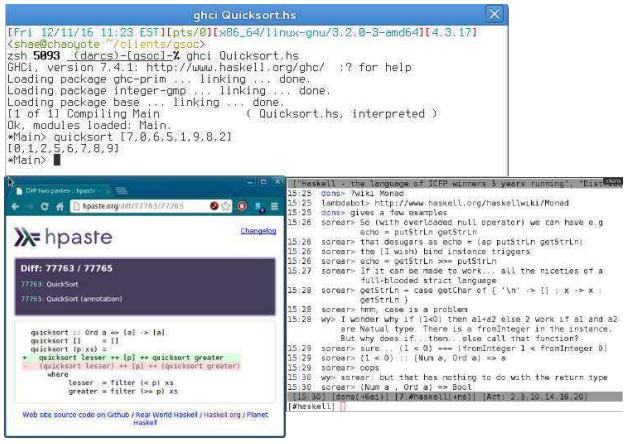
ghci

ghci Quicksort.hs

```
[Fri 12/11/16 11:23 EST][pts/0][x86_64/linux-gnu/3.2.0-3-amd64][4.3.17]
<shae@chaoyote:~/clients/gsoc>
zsh 5093 (darcs)-[gsoc]-% ghci Quicksort.hs
GHCi, version 7.4.1: http://www.haskell.org/ghc/  :? for help
Loading package ghc-prim ... linking ... done.
Loading package integer-gmp ... linking ... done.
Loading package base ... linking ... done.
[1 of 1] Compiling Main             ( Quicksort.hs, interpreted )
Ok, modules loaded: Main.
*Main> quicksort [7,0,6,5,1,9,8,2]
[0,1,2,5,6,7,8,9]
*Main>
```

The Problem is ...



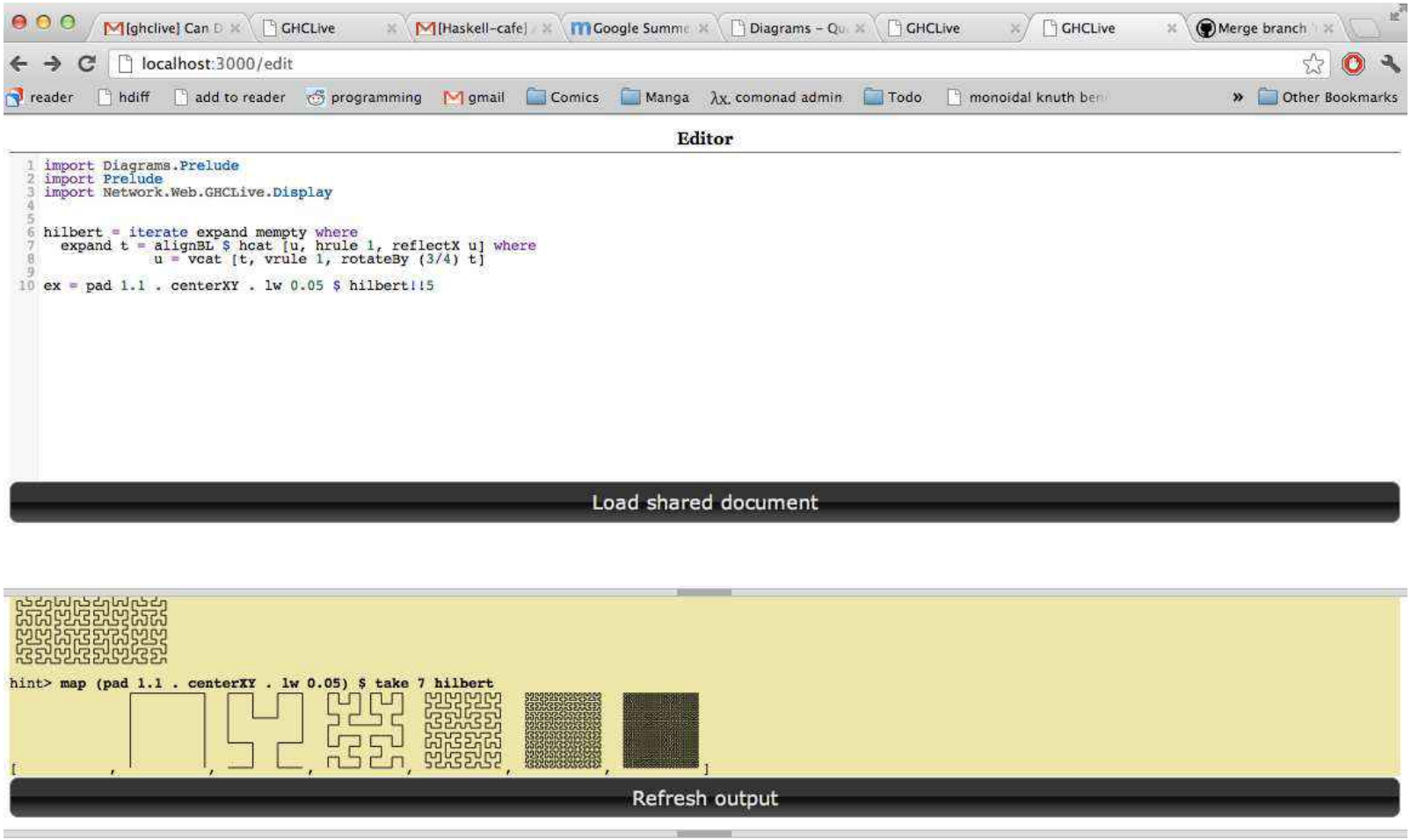


How to debug source you can't see?

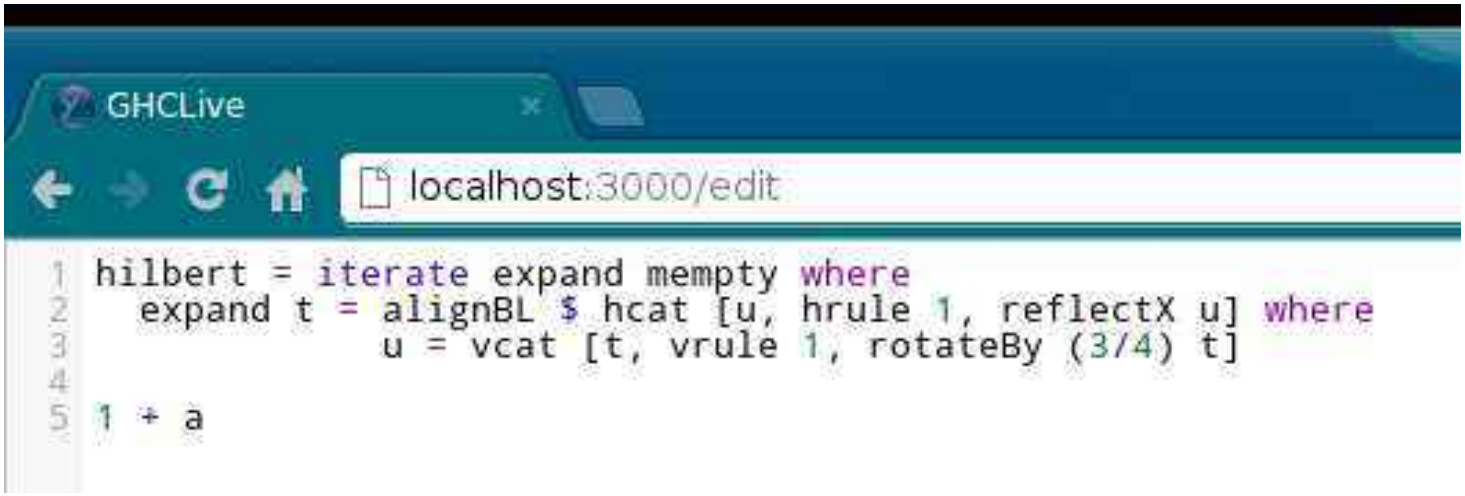
<novice>I still get those same errors I showed you before.
<expert>...but the errors you posted *still* don't match with the file you posted.
<expert>Are you definitely hitting "save" in your text editor?

- It turned out that the editor used by the novice was not saving at all.

Multi-user editor and compiler in a web browser

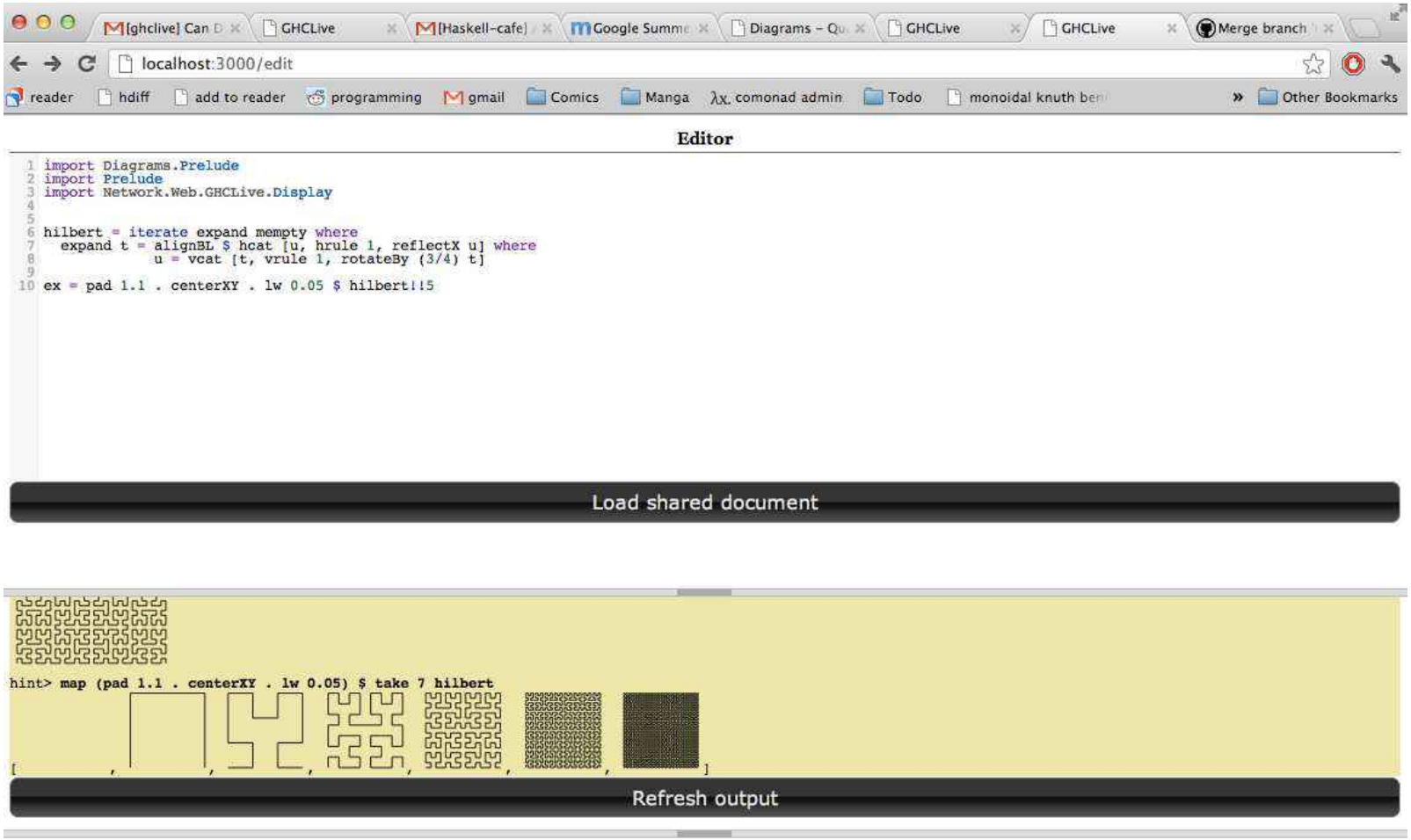


Source widget

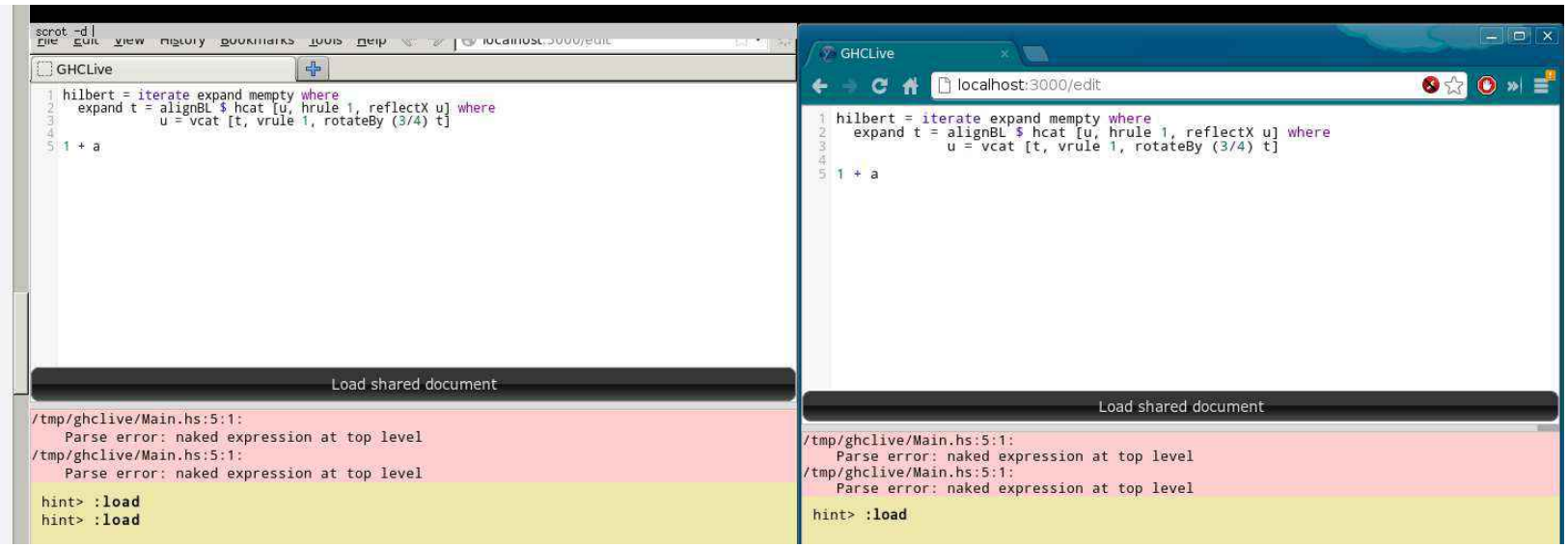


Haskell source code is edited in the top pane.

Multi-user editor and compiler in a web browser

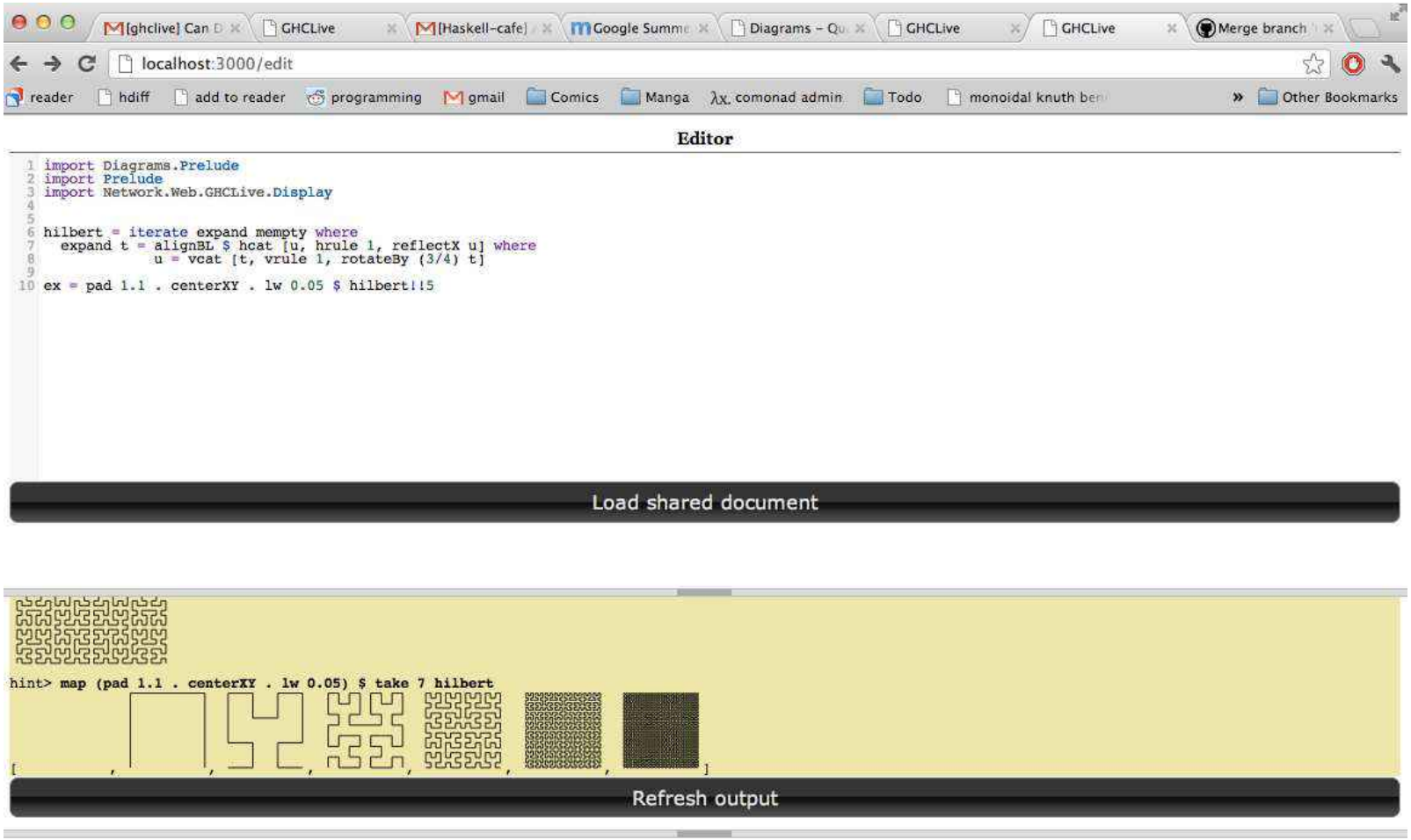


synchronized source widget

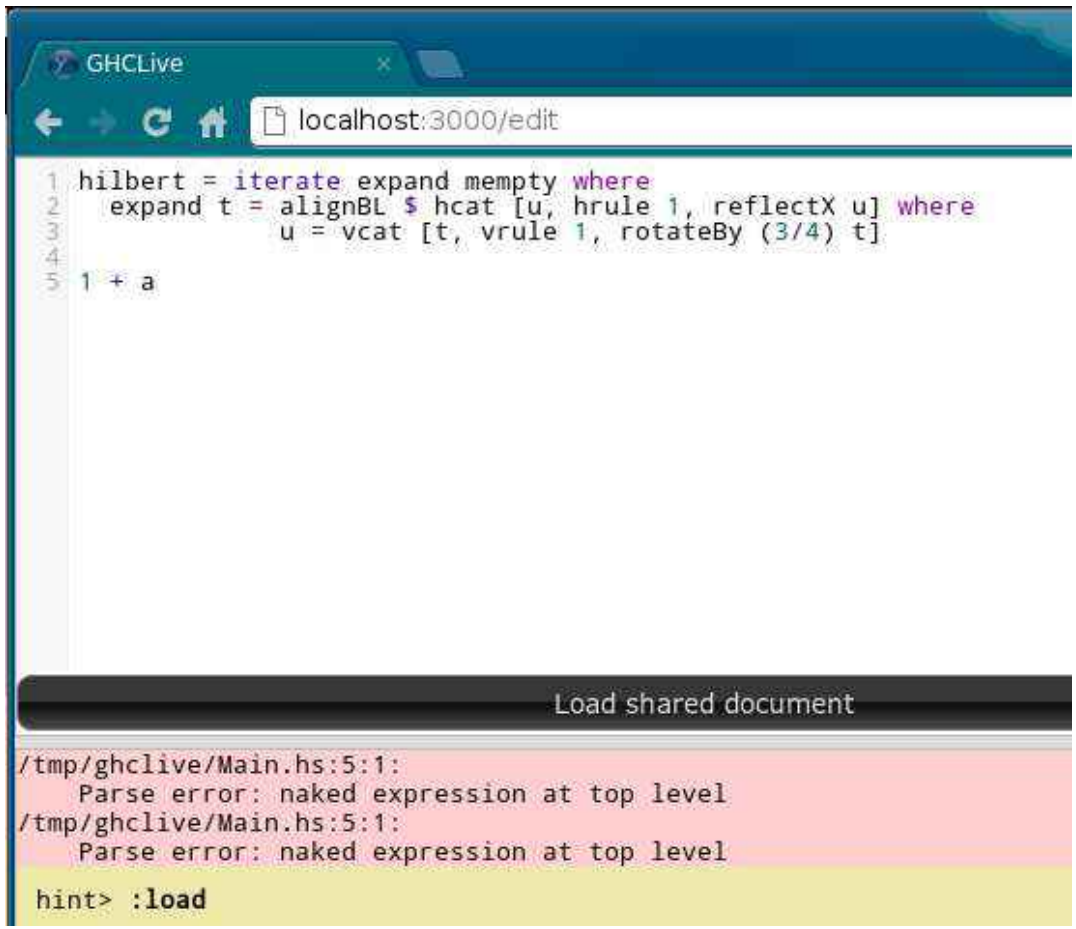


As the source code changes in the shared editor, every attached client is updated in real-time.

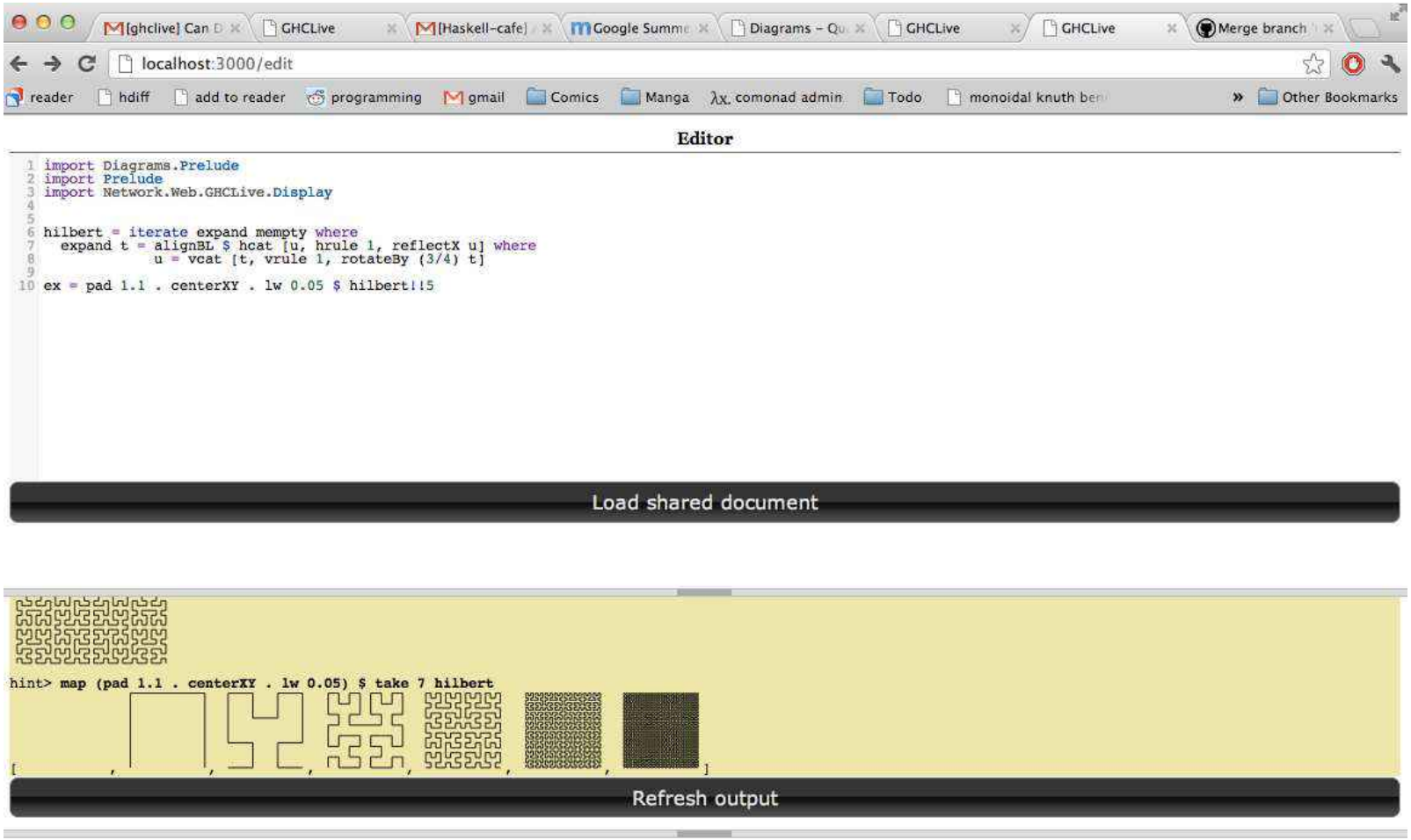
Multi-user editor and compiler in a web browser



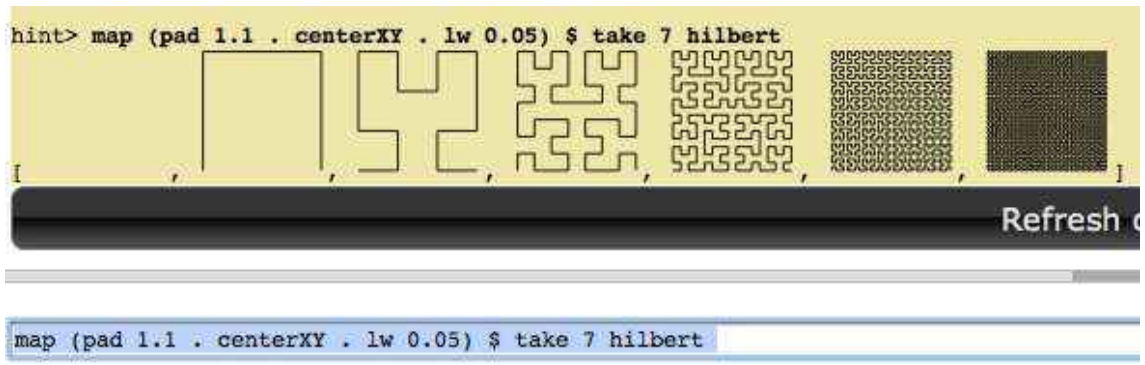
compiler error



Multi-user editor and compiler in a web browser

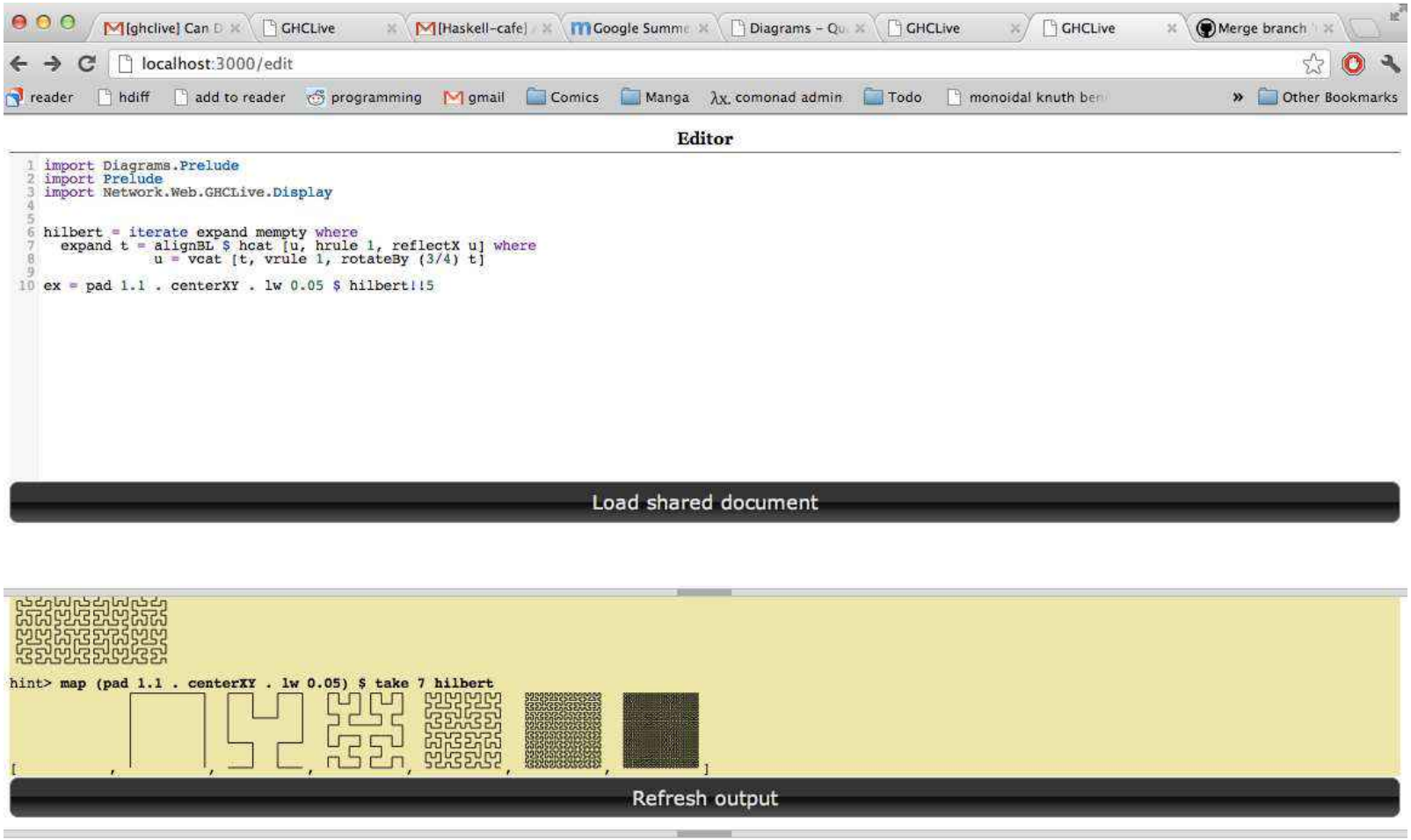


ghcLiVE REPL input widget

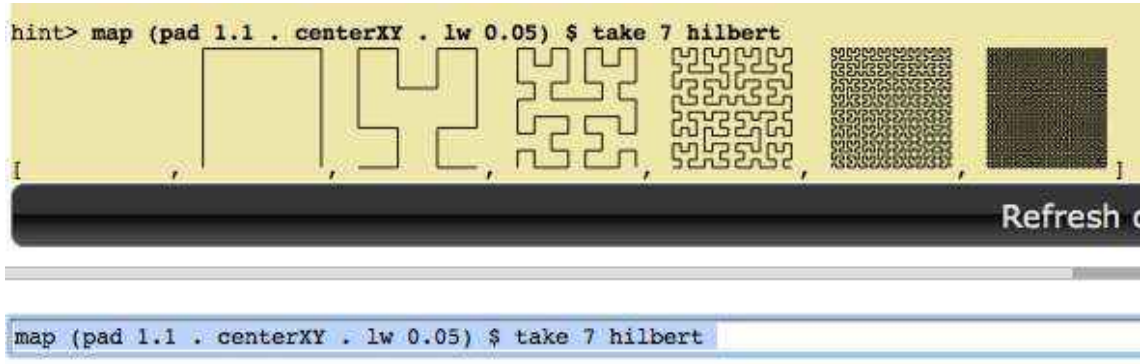


The input pane allows you to execute the loaded source code.

Multi-user editor and compiler in a web browser



ghcLiVE output widget



The output pane supports both text and Simple Vector Graphics output.

Dependencies

- Glasgow Haskell Compiler has a complicated API (ghc-api)
- Hint is a user-friendly wrapping of the ghc-api
- Yesod web framework for Haskell (similar to Ruby on Rails)
- JavaScript source code editor from codemirror.net
- WebSockets don't work with older browsers

Architecture

- Three threads
  - webserver
  - compiler
  - editor synchronization
- just under 300 lines to sew together the compiler, webserver and synchronizer
- just under 200 lines for document state and synchronization

Design choices

- Desktop only
- No security
- No accounts
- No state

Current state

- Working prototype
- Several known bugs
- Interested academic users

Future work

- Fix bugs
- Work with interested educational users
- Generalize to languages other than Haskell?
- Voice chat functionality with Google Talk?

Questions?

Thank you!

- http://github.com/shapr/ghclive
- http://ghclive.wordpress.com

Shae Erisson
Dept. of Computer Science & Information Systems
University of North Alabama
`http://www.ScannedInAvian.com/`