From tinyscheme to tr7

How to lose time for glory

Conclusion

- Full C implementation of R7RS small
  - Only 2 files: tr7.h & tr7.c
- Only R7RS small + SRFI 136, nothing more
- 115 kilobyte (AMD64, -Os, stripped)
- Open source, BSD-0
  - https://gitlab.com/jobol/tr7

FOSDEM'22

Is it a joke? No!

- One more Scheme interpreter! Is it a joke? Why Scheme? Why one more? NIH syndrome?
  - easy integration (only 2 “C” files, fast start)
  - implementation of strict R7RS small
  - development made for fun and by interest at waste time

demo

```
$ git clone https://gitlab.com/jobol/tr7.git
$ cd tr7
$ ls
$ make OFLAGS=-Os
$ ls
$ strip tr7i
$ ls -l tr7i
$ curl -s http://synthcode.com/scheme/irregex/irregex-0.9.10.tar.gz | tar zx
$ ./tr7i
tr7> (load "irregex-0.9.10/irregex.scm")
tr7> (irregex-split '(+ space) " fee fi  fo\tfum\n")
tr7> (irregex-extract '(+ numeric) "192.168.0.1")
tr7> (irregex-replace/all '(+ space) "one two three" "_")
tr7> (exit)
$ TR7INIT=irregex-0.9.10/irregex.scm ./tr7i
tr7> (irregex-extract '(+ numeric) "192.168.0.1")
```

FOSDEM'22

Existing interpreters

- From implementation link of https://schemers.org
- The interpreters written in C implementing R7RS are:
  - chibi-scheme
  - Foment
  - Picrin
- All of them have many files and rely on some scheme part

From tinyscheme to tr7

- At start (February 2021): I wanted to modify tinyscheme to implement R7RS
- Alas! It took much more time than expected but like a drug addict I continued to shoot up changes
- Now (January 2022): Not much of tinyscheme remains
  - tinyscheme: 5884 lines env.
  - tr7: 15849 lines env.
- Still addict? Yeeeeesssss!

Pointer tagging

- The 3 lower bits of the pointer are used to differentiate primitive types:
  - 0: pairs
  - 1 & 5: integers (30 or 62 bits)
  - 2: cells
  - 3: constants, characters, operators
  - 4: doubles
  - 6, 7: unused spare

Library not standalone

- Include file tr7.h without #if on features
- The standalone interpreter/REPL program tr7i is a separate C file, examples/tr7i.c
- Selection of features at compilation have no impact on header, avoiding mismatches

Self include!

- In order to deliver a single file, the operators and symbols are in tr7.c itself and is included for:
  - Declaring operator constants
  - Declaring symbol constants
  - Predefined symbols
  - Builtin operations definition and dispatch

Fast start

- Predefined static symbols
- No scheme file to read compile interpret
- More statically defined constants is possible but it is really not obvious how to do it keeping things simple, easy to read and not using a processing step (only C compile stuff)

Optimisations, call/cc, ...

- Fast reuse of stack except when call/cc is active
- Use of C tail call optimization
- Simplification of the continuation state
- Further optimization will require evaluation of arguments from last to first

C interface

- Expected count of arguments
- Expected type of arguments
- Arguments received as an array
- Closure by function

Forecast

1) End of February: read of big strings, deliver `read-error`, `file-error`, `include-library-declarations`
2) End of March: string intraline spaces escape, `scheme-report-environment`, `null-environment`, clean import of built-in `(scheme ...)`
3) End of May: cleanup, comment, documentation
4) End of September: optimization with semi-compilation and hygienic `define-syntax` with `syntax-error`, reasonable debugging and error report
5) End of December: numeric tower, final?

FOSDEM'22

Why?

- I really ask me why? I know the purpose but:
  - Gimp is switching to python and will probably never upgrade its scheme engine
  - Supposing someone needs an extension language for a project, will s-he mind to use a scheme engine?
- Okay but fun... fun to go to bed late and to wake up at 2am with something in mind.

Post-conclusion

- Near completion but not 100% complete
  - But really usable and functional
- Achievement expected before end 2022
  - Goals: conformity and efficiency
- Repo at https://gitlab.com/jobol/tr7
  - Star it to get in touch
  - Help and feedback more than welcome
- Contact jobol@nonadev.net