Stak Scheme updates

@raviqqe

March 17, 2024

Contents

- Progress
  - Library system fixes
- Future work

Library system update

`import`

- `(rename ...)`
  - Renames each imported symbol.
- `(prefix ...)`
  - Puts a prefix to imported symbols.

`export`

- `(rename ...)`
  - Renames an exported symbol.

Library system bug fixes

- Macro expansion
- Quotes

Macro expansion

- Literals in macros need to be matched on expansion.
- Previously, they were not properly matched across library boundaries.

```
(define-syntax cond
  (syntax-rules (else =>)
    ((_ (else result1 result2 ...))
      (begin result1 result2 ...))
```

- The literals need to be exported too.

```
(export
  ; ...

  else
  =>)
```

Quotes

- `'`, `` ` ``, `,`, and `,@`
- `(quote ...)` is actual syntax.
- Quoted symbols can be compared by `eq?` for equality.

```
'(1 2 3)
`(1 ,x 3)

'(x y z) ; <- ?
```

Quotes

- Are quoted symbols need to be different across libraries?
  - The answer seems to be no according to the other implementations.

```
(define-library (foo bar)
  (import (scheme base))

  (begin
    (define x '(x y z))

    ; This is `(define x (quote (x y z)))`.
    ; But is it `(define foo/bar/x (quote (foo/bar/x foo/bar/y foo/bar/z)))`?
    ))
```

Others

- Removal of hacks
  - Parsing "keywords"
    - `syntax-rules`, `define-syntax`, `...`, etc. are normal macros now.
  - Built-in symbols now have string representations.
- Compiler optimization
  - Integration test: 3 hours -> 1.5 hours

Future work

- Compiler performance improvements
- `eval` procedure

Summary

- Fixing a library system is fun!