

Forum:[Hack Forums](#)

Thread: [R5RS Scheme \(Racket/LISP\) S-expression to XML converter](#)

MiguelSanchez - March 5, 2022, 10:14 am

I'm going to make a nice animated SVG generator, rather than outputting html I'm going to output normal s-expressions which I've just writing homomorphically to XML anyway to preserve a direct mapping.

It's just this way I'll have macros and quasiquotes in HTML and SVG.

It's short

```
(define (->str s) (if (string? s) (string-append "\"" s "\"") (if (number? s)(number->string s)
(symbol->string s))))
(define (pair->xml p) (string-append " " (symbol->string (car p)) "=" (->str (cdr p)) ))

(define (s-expression->xml-inner exp) (if (string? exp) exp (string-append "<" (symbol->string (car
exp)) (apply string-append (map pair->xml (cadr exp))) ">" (s-expression->xml (caddr exp)) "</"
(symbol->string (car exp)) ">")))
(define (s-expression->xml exps) (if (null? exps) "" (string-append (s-expression->xml-inner (car
exps)))))
```

```
(define (demo duration) `(svg
  ((viewbox . "0 0 10 10") (xmlns . "http://www.w3.org/2000/svg"))
  (rect
    ((width . 7) (height . 10))
    (animate ((attributeName . "width") (values . "7;10;7") (dur . ,duration) (repeatCount .
"indefinite"))))))
```

```
(display (s-expression->xml (demo "5s")))
```

I've got a basic animated SVG demo function named demo, it just takes duration as a variable so I could show the syntax for unquoting a variable in the quasiquote, everything following ` is a quote, it means I'm talking about the literal symbols not what they represent, unless preceded by a comma (like duration is) which means I'm referring to the variable duration's assigned value not the literal name 'duration.

I'll turn this into a tight banner making library soon, and I'll be back for business (with free vouch copies of custom banners when I do)