

Embedded Domain Specific Languages in Clojure

Dimitry Gashinsky (@digash)

Pico Quantitative Trading LLC.

18 August 2010

Introduction

Main Part

Questions

Notes

Used to be by day

Coder for financial, security and technology industries.

Used to be at night

Lisp enthusiast and computational philosopher.

Now in twilight

VP of engineering for Pico Quantitative Trading, which includes all of the above.



Figure: R-LISP Book (1991)

To symbolically solve this:

$$\int_0^y \cos(2x) dx. \quad (1)$$

Type this at the REPL:

```
1: int(cos(2x),x,y,2y);
```

Get this solution:

$$\frac{\text{SIN}(4*Y) - \text{SIN}(2*Y)}{2}$$

MAGIC !

Definition of embedded (a.k.a. internal¹) DSL

Enumerating some of the important attributes:

- targeted towards specific problem

- limited in the scope as oppose to general purpose language

- usually not turing complete

- self-documenting

¹[[<http://martinfowler.com/articles/languageWorkbench.html>][Language Workbenches: The Killer-App for Domain Specific Languages?]]

Enumerating more of the important attributes:

- bottom-up design²

- parasitic in its nature

 - feeds on host abstract syntax tree (AST)

 - hides in the host's syntax

 - reproduces more quickly and in greater numbers than its hosts

²Programming Bottom-Up

When in need of wisdom use a good book²

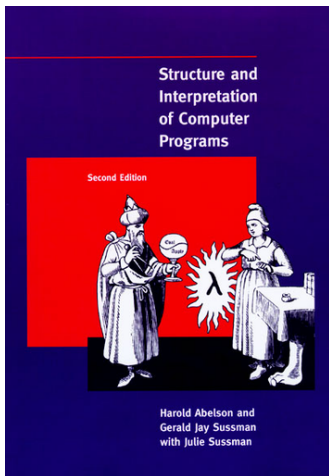


Figure: Structure and Interpretation of Computer Programs

Metalinguistic Abstraction (a.k.a. DSL)

Establishing new languages

- a powerful strategy for controlling complexity
- particularly important to computer programming, because we can implement these languages

Implementing languages is not that scary encouragement

“The evaluator [or compiler], which determines the meaning of expressions in a programming language, is just another program.”

Maxwell's Equations of Software! – Alan Kay inspiration

if either side of the equation is defined at all.

Example

```
f:  λ[[x;y]:cons[car[x];y]]
fn: (LAMBDA (X Y) (CONS (CAR X) Y))
arg1: (A B)
arg2: (C D)
args: ((A B) (C D))

evalquote[(LAMBDA (X Y) (CONS (CAR X) Y)); ((A B) (C D))] =
λ[[x;y]:cons[car[x];y]]((A B);(C D))=
(A C D)
```

evalquote is defined by using two main functions, called eval and apply. apply handles a function and its arguments, while eval handles forms. Each of these functions also has another argument that is used as an association list for storing the values of bound variables and function names.

```
evalquote[fn;x] = apply[fn;x;NIL]
```

where

```
apply[fn;x;a] =
[atom[fn] → [eq[fn,CAR] → car[x];
eq[fn,CDR] → cdr[x];
eq[fn,CONS] → cons[car[x];cdr[x]];
eq[fn,ATOM] → atom[car[x]];
eq[fn,EQ] → eq[car[x];cdr[x]];
T → apply[eval[fn;a];a]];

eq[car[fn];LAMBDA] → eval[caddr[fn];pairlis[cadr[fn];x;a]];
eq[car[fn];LABEL] → apply[caddr[fn];x;cons[cons[cadr[fn];
cdr[fn];a]]]

eval[e;a] = [atom[e] → cdr[assoc[e;a]];
atom[car[e]] →
[eq[car[e];QUOTE] → cdr[e];
eq[car[e];COND] → evcon[cdr[e];a];
T → apply[car[e];evalis[cdr[e];a];a]];
T → apply[car[e];evalis[cdr[e];a];a]]

pairlis and assoc have been previously defined.
evcon[c;a] = [eval[car[c];a] → eval[cadr[c];a];
T → evcon[cdr[c];a]]

and
evalis[m;a] = [null[m] → NIL;
T → cons[eval[car[m];a];evalis[cdr[m];a]]]
```

Figure: LISP 1.5 Programmer Manual page 13

Why Rich should have all the fn?

"We come to see ourselves as designers of languages, rather than only users of languages designed by others."

"... computer science itself becomes no more (and no less) than the discipline of constructing appropriate descriptive languages."

Short list of features useful for DSLs:

- `fn #()`

- `unquote` and `unquote-splicing`

- code as data

- macrology

configuration language

Every application needs a DSL

Market Data Message Specification

The packet header has the following format:

PACKET HEADER			
Name	Length	Format	Notes
Sequence Number	8	Numeric	The sequence number of the first message in the packet. If the packet contains more than one message, subsequent message sequence numbers are derived implicitly.
Message Count	2	Numeric	The number of message blocks contained in the packet.
Session Identifier	1	Alphanumeric	Number of the current daily session. This field will begin as '0' and increment thereafter if the feed is restarted during the trading day.

Figure: One small piece of huge spec

Market Data Message Description Language

```
(defmessage PacketHeader
  [SequenceNumber      8 :numeric
    "The sequence number of the first message in
    the packet. If the packet contains more than
    one message, subsequent message sequence numbers
    are derived implicitly."]
  [MessageCount        2 :numeric
    "The number of message blocks contained in
    the packet."]
  [SessionIdentifier 1 :alphanumeric
    "Number of the current daily session. This field
    will begin as 0 and increment thereafter if the
    feed is restarted during the trading day."])
```

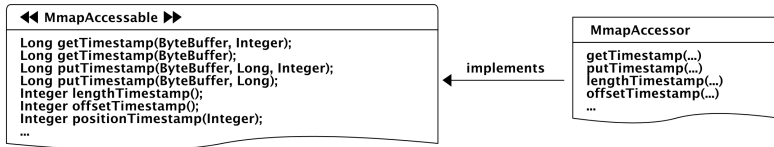

Memory Map Description Language

```
(defmmmap MarketDataMmap
  [Timestamp      8 :integer]
  [TradeTime      8 :integer]
  [QuoteCondition 4 :character]
  [TradeCondition 4 :character]
  [Bid            4 :decimal]
  [Ask            4 :decimal]
  [Last          4 :decimal]
  [BidSize       4 :integer]
  [AskSize       4 :integer]
  [LastSize      4 :integer]
  [TotalAmount   8 :integer]
  [Volume        4 :integer]
  [Currency      3 :character]
  [Latency       8 :integer]
  [Symbol        12 :string])
```

Macroexpanded defmmap



Figure: defmmap macroexpanded



Memory Map Accessable Java API

```
public interface com.picotrading.tech.madase.mmap.MarketData {
    public Long getTimestamp(java.nio.ByteBuffer, Integer);
    public Long getTimestamp(java.nio.ByteBuffer);
    public Long putTimestamp(java.nio.ByteBuffer, Long, Integer);
    public Long putTimestamp(java.nio.ByteBuffer, Long);
    public Integer lengthTimestamp();
    public Integer offsetTimestamp();
    public Integer positionTimestamp(Integer);
    public Long getQuoteTime(java.nio.ByteBuffer, Integer);
    public Long getQuoteTime(java.nio.ByteBuffer);
    public Long putQuoteTime(java.nio.ByteBuffer, Long, Integer);
    public Long putQuoteTime(java.nio.ByteBuffer, Long);
    public Integer lengthQuoteTime();
    public Integer offsetQuoteTime();
    public Integer positionQuoteTime(Integer);
    public Long getTradeTime(java.nio.ByteBuffer, Integer);
    public Long getTradeTime(java.nio.ByteBuffer);
}
```

Memory Map Accessor Java API

```
public final class com.picotrading.tech.madase.mmap.MarketDataMmapAccessor {
    public com.picotrading.tech.madase.mmap.MarketDataMmapAccessor(long address, int length);
    public com.picotrading.tech.madase.mmap.MarketDataMmapAccessor(long address, int length, int position);
    public Object length();
    public Integer positionSymbol(Integer position);
    public Integer offsetSymbol(Integer position);
    public Integer lengthSymbol(Integer position);
    public String putSymbol(java.nio.ByteBuffer, String);
    public String putSymbol(java.nio.ByteBuffer, String, Integer position);
    public String getSymbol(java.nio.ByteBuffer);
    public String getSymbol(java.nio.ByteBuffer, Integer position);
    public Integer positionInternalLatency(Integer position);
    public Integer offsetInternalLatency(Integer position);
    public Integer lengthInternalLatency(Integer position);
    public Long putInternalLatency(java.nio.ByteBuffer, Long value);
    public Long putInternalLatency(java.nio.ByteBuffer, Long value, Integer position);
    public Long getInternalLatency(java.nio.ByteBuffer);
    public Long getInternalLatency(java.nio.ByteBuffer, Integer position);
}
```

Questions?

<mailto:dimitry@gashinsky.com>

twitter:@digash

xmpp:i@digash.com

<http://blog.digash.com>

```
(defn slice [^ByteBuffer b n]
  (vec (for [p (range 0 (.capacity b) (length MarketDataMmap))]
    (let [^ByteBuffer b (.position b p)
          ^ByteBuffer s (.slice b)]
      (.limit s (length MarketDataMmap))))))
```

(+ 1 4)

ls