Lua for Molecular Biology

Yutaka Ueno
Neuroscience,
AIST Tsukuba, Japan

Lua is good in Molecular biology for:
1. programming tasks
2. database management tasks
3. development of algorithms

Current Projects
1. sequence annotation
2. molecular simulation
3. image processing
Processing Sequence Annotation Data using the Lua Programming Language

Yutaka Ueno, Masanori Arita, Toshitaka Kumagai, Kiyoshi Asai

Computational Biology Research Center (CBRC) AIST

Genome Informatics 14 (2003) 164-175.

http://www.jsbi.org/journal/GIW03/GIW03F016.html

GUPPY : Genetic Understanding Perspective Preview sYstem

• An sequence map viewer program
GUPPY home page

Methods and GUPPY script files are provided

http://staff.aist.go.jp/yutaka.ueno/guppy
The basic annotation data format in Lua

```lua
one = {
    {symbol = "orfD", pos1 = 18, pos2 = 48, category = 1},
    {symbol = "orfE", pos1 = 58, pos2 = 78, category = 3},
    {symbol = "gene1", pos1 = 88, pos2 = 188;}
    {symbol = "5'utr", pos1 = 88, pos2 = 94},
}
```

A Lua program to list subsidiary annotations

```lua
num = getn(one)
print(" total", num)

for idx = 1, num do
    cnt = one[idx]
    if (cnt[1]) then
        print(idx, cnt[1].symbol)
    end
end

---- print(one[3][1].symbol)
```
In Perl Language

```perl
@one = (
    {symbol => "orfD", pos1 => 18, pos2 => 48, category => 1},
    {symbol => "orfE", pos1 => 58, pos2 => 78, category => 3},
    {symbol => "gene1", pos1 => 88, pos2 => 188,
      child => [
        {symbol => "5'utr", pos1 => 88, pos2 => 94},
      ],
    },
);

$num = $#one + 1;

print "total $num
";

for ($idx = 0; $idx <$num; $idx++) {
    $cnt = $one[$idx];
    if ($cnt->{child}[0]) {
        print "$idx child $cnt->{child}[0]{symbol} 
";
    }
};

### $one[2]{child}[0]{symbol}

**PRO** : Widely accepted

**CON** : Difficulties in hierarchical data with a notion of "reference"
In Python Language

```python
one = [
    {'symbol': 'orfD', 'pos1': 18, 'pos2': 48, 'category': 1},
    {'symbol': 'orfE', 'pos1': 58, 'pos2': 78, 'category': 3},
    {'symbol': 'gene1', 'pos1': 88, 'pos2': 188,
     'child': [
         {'symbol': '5'utr', 'pos1': 88, 'pos2': 94},
     ]},
],

num=len(one)  ## 3

print "total",num

for idx in range(num):  ## 0,1,2
    cnt=one[idx]
    if cnt.get('child'):
        print idx," ",
        cnt['child'][0]['symbol'],"\n"

### one[2]['child'][0]['symbol']

**PRO**: Rappidly growing community in scientific applications

**CON**: Unusual indentation rule
In Ruby Language

```ruby
one = [
    {'symbol' => "orfD", 'pos1' => 18, 'pos2' => 48, 'category' => 1},
    {'symbol' => "orfE", 'pos1' => 58, 'pos2' => 78, 'category' => 3},
    {'symbol' => "gene1", 'pos1' => 88, 'pos2' => 188, 'child' => [
        {'symbol' => "5'utr", 'pos1' => 88, 'pos2' => 94},
    ],
},
]

num = one.size

print("total ", num, "\n")

for idx in 0..num-1
    cnt = one[idx]
    if cnt['child']
        print(idx, " ",
            cnt['child'][0]['symbol'], "\n")
    end
end

## one[2]['child'][0]['symbol']

**PRO**: Modern programming technology

**CON**: Involving tricky object oriented programming topics
```
### Comparison of Languages

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>Java</th>
<th>Lisp</th>
<th>Basic</th>
<th>Perl</th>
<th>Tcl</th>
<th>Python</th>
<th>Ruby</th>
<th>Lua</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. dynamic data</td>
<td></td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>. auto memory</td>
<td>○</td>
<td>○</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>2. variables</td>
<td>○</td>
<td>○</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>. numericals</td>
<td>○</td>
<td>○</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>3. syntax</td>
<td>○</td>
<td>○</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>. ease of use</td>
<td>○</td>
<td>○</td>
<td></td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
</tbody>
</table>

SIZE is another issue in the implementation for a High speed interactive computer graphics.
LOCUS       HUMHA2WC     2226 bp    DNA             PRI       09-NOV-1994
DEFINITION  Human gene for aquaporin-2 water channel.
ACCESSION   D31846
NID         g567249
KEYWORDS    aquaporin-2 water channel.
SOURCE      Homo sapiens DNA.
            Homo sapiens
            Eukaryotae; mitochondrial eukaryotes; Metazoa; Chordata;
            Vertebrata; Eutheria; Primates; Catarrhini; Hominidae; Homo.
REFERENCE   1 (bases 1 to 2226)
            Uchida,S., Sasaki,S., Fushimi,K. and Marumo,F.
            J. Biol. Chem. 269 (38), 23451-23455 (1994)
            MEDLINE 94375443
            Submitted (17-Jun-1994) to DDBJ by:
            Shinichi Uchida
FEATURES             Location/Qualifiers
            source 1..2226
            /organism="Homo sapiens"
            TATA_signal 545
            exon 574..1027
            /number=1
            CDS join(668..1027,1095..1259,1327..1407,1465..1674)
            /codon_start=1
            /product="human aquaporin-2 water channel"
            /db_xref="PID:g567250"
            /translation="MWELRSIAFSRAVFAEFLATLLFVFFGLGSL
            AMAFGLGTLVQALGHISGAHINPAVTAVACLVGCHVSVL
            YNYLVFPPAKLSERTHISISNOTCORRECT"
            intron 1028..1094
            /number=1
            exon 1095..1259
            /number=2
            polyA_signal 2221
BASE COUNT      412 a    686 c    666 g    462 t
ORIGIN      Chromosome 12.
            1 aagcttaatg attatggtt gattagctgc aagaatgcaaa
            2 cacaacacctt tatgca
HUMHA2WC={
  LOCUS = "HUMHA2WC", bp=2226, DNA = "PRI", date="09-NOV-1994",
  DEFINITION = "Human gene for aquaporin-2 water channel.",
  ACCESSION = "D31846",
  NID = "g567249",
  KEYWORDS = "aquaporin-2 water channel.",
  SOURCE = "Homo sapiens DNA.",
    ORGANISM = "Homo sapiens", taxon =
      "Eukaryotae; mitochondrial eukaryotes; Metazoa; Chordata;"
      .. "Vertebrata; Eutheria; Primates; Catarrhini; Hominidae; Homo.",
  REFERENCE = {
    [ 1 ] = { loc= "(bases 1 to 2226)",
    AUTHORS = "Uchida,S., Sasaki,S., Fushimi,K. and Marumo,F.",
    TITLE = "Isolation of human aquaporin-CD gene",
    JOURNAL = "J. Biol. Chem. 269 (38), 23451-23455 (1994)",
    MEDLINE = "94375443", }, },
  COMMENT = "Submitted (17-Jun-1994) to DDBJ by:
    .. "Shinichi Uchida"
  FEATURES = {
    {k="source", loc= {1,2226}, organism="Homo sapiens"},
    {k="TATA_signal", loc= {545}},
    {k="exon", loc= {574,1027}, number=1},
    {k="CDS", loc={tpg="join";\{668,1027\},\{1095,1259\},\{1327,1407\},\{1465,1674\}\},
      codon_start=1,
      product="human aquaporin-2 water channel",
      db_xref="PID:g567250",
      translation="MWELRSIAFSRAVFAEFLATLLFVFFGLGS"
        .. "AMAFGLGTVQLHGISHGPHTPVATVCVGCHVSL"
        .. "YNVLFPPAKSLSRVLKTHISISNOTCORRECT"},
    {k="intron", loc= {1028,1094}, number=1},
    {k="exon", loc= {1095,1259}, number=2},
    {k="polyA_signal", loc=\{2221\}}, },
  COUNT = \{ a = 412 , c = 686 , g = 666 , t = 462 \},
  ORIGIN = "Chromosome 12."
  seqence = "aagcttaatgatttatgggtgattagctgcaagaatgcaagcacagaaga"
    .. "cacaaccttttatgc"}
Processing annotations

Suppose if we need to merge two annotation data differently formatted...

1. Data Rearrangement:
   Picking-up, grouping, sorting, comparing, ...

2. Coordinate Translation:
   GenBank data are annotated by its 'locus' coordinate starting from 1 ...

3. On-Demand Editing:
   Adding, or modifying annotation is the biological objective
Implementation

An in-house Graphics and GUI library (ASHLEY)
- ANSI-C : 33,000 lines
  - Linux/X-Window, Windows, MacOS Classic & Carbon

Lua 4.0.1
- patch for fgets() to support foreign CR LF.
- support $endinput

Lua code 6,400 lines
ANSI C 4,900 lines
- - - source code is available
Bermuda Principles for Human Genome

• 1996 - Bermuda international meeting for the genome project agreed to formalize the conditions of data access:
  • Primary genomic sequence should be in the public domain
  • Sequence data should be released as soon as possible (24 hour)
  • Annotation should be submitted immediately to public databases

• URLs:
  • Heritage of Humanity (by Dr. John Salston)
    • http://mondediplo.com/2002/12/15genome
  • Bermuda Principles
    • http://www.gene.ucl.ac.uk/hugo/bermuda.htm
A Persistent Large Table on Disk

A virtual memory based large table
A huge table of lua whose part is in the disk
  - several GB of data
  - read only access (DVD-ROM)
  - update journal would be nice

A Simple implementation in Lua by swapping out unused table data does work fine.
Conclusion

• Computational tasks to visualize annotation data for genetic sequences involve:
  • (1) data rearrangement, (2) coordinate translation (3) local editing.
• Those tasks are greatly aided by a programming language that provides the necessary functions:
  • (1) handling of data containers, (2) symbolic references, (3) a simple programming syntax.
• Lua language was successfully applied to GUPPY, a sequence visualization program with arrangement of annotation data and a flexible layout.