



Introduction



Reports



Analysis



Discoveries



Conclusion

The Quickref Cohort

~ ELS 2024 ~

Didier Verna

EPITA / LRDE

didier@lrde.epita.fr



lrde/~didier



@didierverna



didier.verna



in/didierverna

Preamble

What is a “cohort” ?

A group of individuals having a statistical factor (such as age or class membership) in common in a demographic study.

– *The Meriam-Webster Dictionary^a, definition 2.b.*

^a<https://www.merriam-webster.com/dictionary/cohort>

History

- ▶ **2010:** Declt project starts
 - ▶ *reference manual generator for Common Lisp libraries*
- ▶ **2013:** First stable release
- ▶ **2017:** Quickref project starts
 - ▶ *Declt (2.3) wrapper for the whole Quicklisp world*
- ▶ **Today:** more than 2000 libraries documented

Jump to: [F](#) [A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

Quickref

The Quickref Cohort is [here](#).

Reference manuals for Quicklisp libraries

Quicklisp version: 2023-10-01, 2016 manuals available.
 Documentation generated on Mon Feb 26 18:20:29 2024 GMT+0 with [Quickref 4.0 beta 2](#) "The Alternat" / Declt 4.0 beta 2 "William Flair",
 Tester: testlisp (2024 system) 1.0.

[Library Index](#) [Author Index](#)

Library Index

#

1am	3d-math	40erts-easl-system
3b-bmfont	3d-matrices	40erts-ci
3b-hdr	3d-quaternions	40erts-doc
3b-sew	3d-spaces	40erts-logging
3dgi-shader	3d-transforms	40erts-sysnk
3bmd	3d-vectors	ci-4500
3bz		

A

a-ci-logger	ci-annot-prove	asdl-finalizers
able	ci-annot-revlist	asdl-fingert
absl	ci-arc4lun	asdl-manager
abstract-classes	ci-ansi-term	asdl-package-system
access	ansi-test-harness	asdl-system-connections
accimotion	ci-ansi-test	asdl-viz
ci-acronyms	anrk	assne
action-fst	anypool	asrl
ci-actors	ci-appertum-stream	asson p
ad hoc	ci-apple-plat	assertion-error
adapt	apply-argv	assoc-villa
adtpi-subcommands	april	ci-association-rules
adp	arc-compact	astroids

Pressure Points

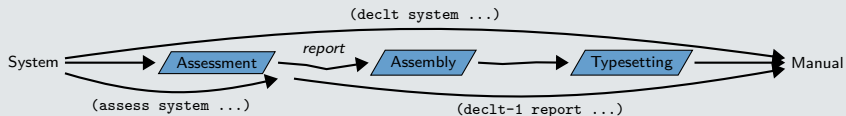
1. Quickref runs Declt in separate processes (cf. ELS 2019)
 - ▶ Library information not directly available
 - ▶ Author index not straightforward to build

⇒ Author information extracted in an *ad hoc* way
2. Declt generates Texinfo files
 - ▶ Subsequent rendering in PDF, HTML, etc.
 - ▶ But people want other formats or direct HTML output

⇒ Incentive to design an abstract representation

The Declt Pipeline (ongoing)

Current state: stage 1 complete (Decl 4.0b1)



1. **Assessment stage:** library $\rightarrow$ *report*
 - ▶ Collected (cross-referenced) information, unordered
2. **Assembly stage:** report $\rightarrow$ *script*
 - ▶ Organized content, format-independent
3. **Typesetting stage:** script $\rightarrow$ *manual*
 - ▶ Formatted reference manual

Consequences

- ▶ Author index much easier to generate
 1. `(declt:assses library)` → report
 2. Extract author information from report
 3. Continue with `(declt-1 report)`

And a new idea...



What if we dumped all Quicklisp library reports to files, and used them for other purposes than documentation ?

- ▶ *A cohort of more than half a million programmatic definitions!*

 Plan

Declt Reports

Cohort Analysis

Funny discoveries

Declt Reports

```
#<DECLT.ASSESS:REPORT {70069BE4D3}>
```

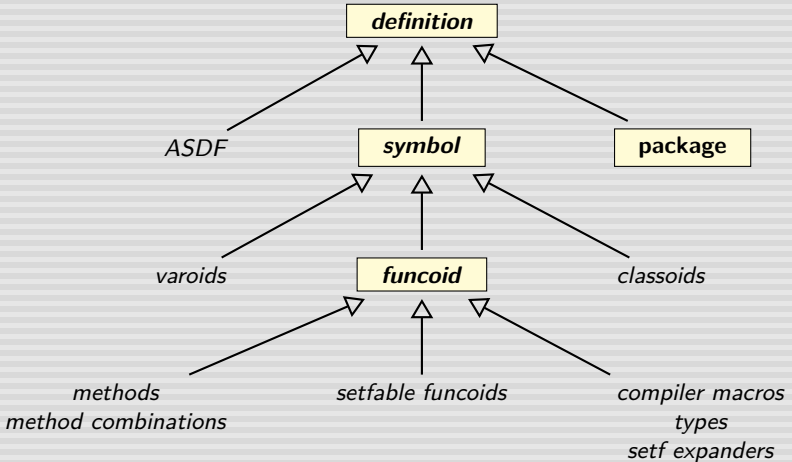
```
-----  
Class: #<STANDARD-CLASS DECLT.ASSESS:REPORT>  
-----
```

```
Group slots by inheritance [ ]  
Sort slots alphabetically [X]
```

All Slots:

```
[ ] CONTACTS          = (("Didier Verna" . "didier@didierverna.net"))  
[ ] COPYRIGHT-YEARS   = "2010-2013, 2015-2022"  
[ ] DEFINITIONS       = (#<DECLT.ASSESS:SYSTEM-DEFINITION declt.core>  
                        #<DECLT.ASSESS:MODULE-DEFINITION src>  
                        ...)  
[ ] LIBRARY-NAME      = "Declt"  
[ ] LIBRARY-VERSION   = "4.0 beta 2 \"William Riker\""  
[ ] LICENSE           = (:BSD "The BSD License" "Permission to use, ...)  
[ ] SYSTEM-NAME       = "net.didierverna.declt"  
[ ] TAGLINE           = "Documentation Extractor from Common Lisp to Texinfo"
```


Definitions Hierarchy (excerpt)



Example: Generic Function Definition

```
#<DECLT.ASSESS:GENERIC-FUNCTION-DEFINITION {70060D9653}>
```

```
-----  
Class: #<STANDARD-CLASS DECLT.ASSESS:GENERIC-FUNCTION-DEFINITION>  
-----
```

```
Group slots by inheritance [ ]
```

```
Sort slots alphabetically [X]
```

All Slots:

```
[ ] COMBINATION = #<DECLT.ASSESS:COMBINATION-DEFINITION STANDARD>  
[ ] EXPANDER-FOR = NIL  
[ ] EXPANDERS-TO = NIL  
[ ] FOREIGN = NIL  
[ ] HOME-PACKAGE = #<DECLT.ASSESS:PACKAGE-DEFINITION DECLT.ASSESS>  
[ ] METHODS = (#<DECLT.ASSESS:METHORD-DEFINITION VALUE-TYPE> ...)  
[ ] OBJECT = #<STANDARD-GENERIC-FUNCTION DECLT.ASSESS:VALUE-TYPE (2)>  
[ ] SETF = NIL  
[ ] SOURCE-FILE = #<DECLT.ASSESS:LISP-FILE-DEFINITION symbol>  
[ ] SYMBOL = DECLT.ASSESS:VALUE-TYPE  
[ ] UID = 51
```

Records Dumping (beta!)

```
("net.didierverna.declt"  
  (:CONTACTS 1)  
  ...  
  (:SYSTEM "net.didierverna.declt.assess"  
    :DOCSTRING 44 :DEPENDENCIES 2 :CHILDREN 2  
    :DEFSYSTEM-DEPENDENCIES 0)  
  ...  
  (:PACKAGE "NET.DIDIERVERNA.DECLT.ASSESS"  
    :DOCSTRING 39  
    :EXTERNAL-SYMBOLS 169 :INTERNAL-SYMBOLS 119  
    :USE-LIST 2 :USED-BY-LIST 1)  
  ...  
  (:CLASS "GENERIC-FUNCTION-DEFINITION"  
    :DOCSTRING 154  
    :DIRECT-SUPERCLASSES 1 :DIRECT-SUBCLASSES 1  
    :DIRECT-METHODS 11  
    :DIRECT-SLOTS 3)  
  ...  
  (:GENERIC-FUNCTION "DOCUMENT"  
    :DOCSTRING 45 :METHODS 39)  
  ...)
```

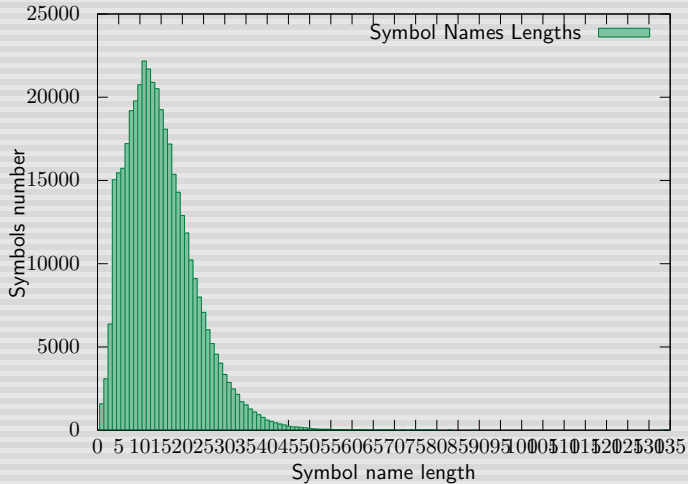
 Plan

Declt Reports

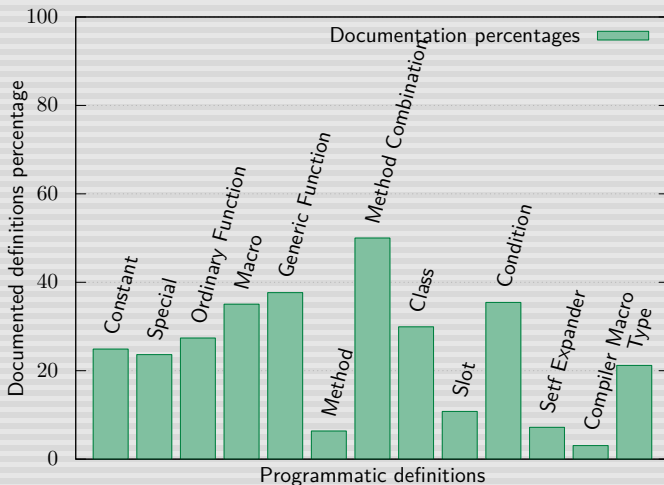
Cohort Analysis

Funny discoveries

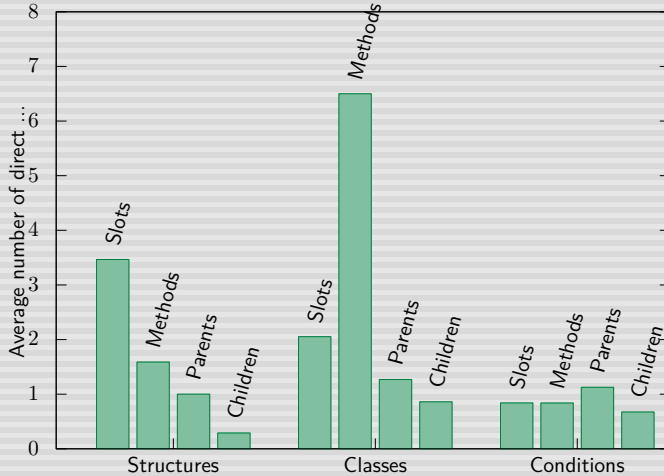
Symbols



Documentation



Classloids



 Plan

Declt Reports

Cohort Analysis

Funny discoveries

Docstring Length Record

- ▶ For a macro
- ▶ 27253 characters / 550 lines
- ▶ Hand written – In fact, a complete user manual
- ▶ DEFBINARY
- ▶ lisp-binary / Jeremy Phelps

Symbol Length Record

- ▶ A function name
- ▶ 135 characters
- ▶ FORMAT'ed & INTERN'ed
- ▶ |semantic-checker for character class with clauses ((#\0 #\9) (#\a #\f) (#\
- ▶ ESRAP-PEG / Michael Raskin

Symbol Cardinality Record

- ▶ A class name
- ▶ 13 components (incl. 3 empty ones)
- ▶ Hand written
- ▶ ARC-CONSTRAINT-TANGENT-TO--CENTER-ON--THROUGH-POINT--SELECT-SIDE
- ▶ Gendl / Dave Cooper



Conclusion

- ▶ More than half a million ASDF / programmatic entities available
- ▶ Tarball & Sample statistical charts available
<https://quickref.common-lisp.net/cohort/>
- ▶ Updated along with the Quickref documentation
- ▶ **Please advise: serialization**
Can drop original objects, but...
 - ▶ need to maintain cross-references between definitions
 - ▶ potential problems with lambda-lists, EQL-specializers, *etc.*