The example Reference Manual

An example of Declt documentation system

Alexander Artemenko

This manual was generated automatically by Declt 3.0 "Montgomery Scott" on Mon Jan 25 10:54:41 2021 GMT+0.

Table of Contents

Copying

1 Introduction

This is the introduction to Declt (`https://www.lrde.epita.fr/~didier/software/lisp/declt/user/index.html`) example project. `Declt` is a documentation system used to build a Quickref (`https://quickref.common-lisp.net/`) — API reference for all libraries, included into the Quicklisp.

This project is a part of the CL Doc Systems (`https://cl-doc-systems.github.io/`) — attempt to create a comprehensive set of examples of different Common Lisp documentation systems.

1.1 Pros & Cons

1.1.1 Pros

- Generates manuals in Texinfo which can be converted to `HTML`, `PDF`, `DVI` and `PostScript`. Here is PDF version (`index.pdf`) of this site.
- Automatically embeds license information.
- Uses standard Texinfo format (`https://www.gnu.org/software/texinfo/manual/texinfo/texinfo.html`), which has good documentation. Theoretically, Texinfo should well suite for large documents.
- It is able to generate single or multi page `HTML` output. To switch the mode, change `single-page-p` variable in `docs/scripts/builder.lisp`.

1.1.2 Cons

- Works only under `SBCL`.
- Free form documentation chapters are limited by "Introduction" and "Conclusion".
- Texinfo is not so popular these days. Also, it is very verbose.
- There is no default CSS theme.
- I wasn't able to use cross referencing Texinfo tags from docstrings. Seems Texinfo markup does not work there. But it works in the introduction and conclusion. Here is example link to the [`do-the-job`], page 11, function.

2 Systems

The main system appears first, followed by any subsystem dependency.

2.1 `example`

Author Alexander Artemenko

License MIT

Description
This description will be used only if long-description is missing

Long Description
Test long description. Both descriptions will be shown in the docs.

Dependency
[`example/app`], page 5, (system)

Source [`example.asd`], page 7, (file)

Directory `/home/runner/work/declt/declt/`

2.2 `example/app`

Author Alexander Artemenko

License MIT

Dependencies

- [`example/utils`], page 5, (system)
- [`example/class`], page 5, (system)

Source [`example.asd`], page 7, (file)

Directory `/home/runner/work/declt/declt/`

Component
[`file-type.lisp`], page 7, (file)

2.3 `example/utils`

Author Alexander Artemenko

License MIT

Source [`example.asd`], page 7, (file)

Directory `/home/runner/work/declt/declt/`

Component
[`file-type.lisp`], page 7, (file)

2.4 `example/class`

Author Alexander Artemenko

License MIT

Source [`example.asd`], page 7, (file)

Directory `/home/runner/work/declt/declt/`

Component
[`file-type.lisp`], page 7, (file)

3 Files

Files are sorted by type and then listed depth-first from the systems components trees.

3.1 Lisp

3.1.1 `example.asd`

Location `/home/runner/work/declt/declt/example.asd`

Systems

- [`example`], page 5, (system)
- [`example/app`], page 5, (system)
- [`example/utils`], page 5, (system)
- [`example/class`], page 5, (system)

3.1.2 `example/app/file-type.lisp`

Parent [`example/app`], page 5, (system)

Location `app.lisp`

Packages [`example/app`], page 9,

Exported Definitions
[`foo`], page 11, (function)

3.1.3 `example/utils/file-type.lisp`

Parent [`example/utils`], page 5, (system)

Location `utils.lisp`

Packages [`example/utils`], page 9,

Exported Definitions
[`do-the-job`], page 11, (function)

Internal Definitions
[`concat`], page 13, (function)

3.1.4 `example/class/file-type.lisp`

Parent [`example/class`], page 5, (system)

Location `class.lisp`

Packages [`example/class`], page 9,

Exported Definitions

- [`admin`], page 11, (class)
- [`non-documented-user`], page 12, (class)
- [`user`], page 12, (class)

Internal Definitions

- [`email`], page 13, (method)
- [`inner-documented-user`], page 14, (class)
- [`is-admin`], page 13, (generic function)
- [`is-admin`], page 13, (method)

- [`is-admin`], page 13, (method)
- [`last-login-at`], page 13, (method)
- [`(setf last-login-at)`], page 13, (method)
- [`name`], page 13, (method)

4 Packages

Packages are listed by definition order.

4.1 example/app

This is docstring for the EXAMPLE/APP package.

The package contains a FOO function which does it's job by applying transformation to the first and second arguments.

Source [`file-type.lisp`], page 7, (file)

Use List `common-lisp`

Exported Definitions
[`foo`], page 11, (function)

4.2 example/utils

The utils package.

The only exported function DO-THE-JOB will be show in a separate section "Exported Definitions".

Source [`file-type.lisp`], page 7, (file)

Use List `common-lisp`

Exported Definitions
[`do-the-job`], page 11, (function)

Internal Definitions
[`concat`], page 13, (function)

4.3 example/class

This package demonstrates how Declt displays classes and generic functions.

The key consept is USER class

It is possible to check if user has admin privileges, using this IS-ADMIN function.

Right now, IS-ADMIN returns T only for objects of class ADMIN.

Source [`file-type.lisp`], page 7, (file)

Use List `common-lisp`

Exported Definitions

- [`admin`], page 11, (class)
- [`non-documented-user`], page 12, (class)
- [`user`], page 12, (class)

Internal Definitions

- [`email`], page 13, (generic function)
- [`email`], page 13, (method)
- [`inner-documented-user`], page 14, (class)

- [`is-admin`], page 13, (generic function)
- [`is-admin`], page 13, (method)
- [`is-admin`], page 13, (method)
- [`last-login-at`], page 13, (generic function)
- [`last-login-at`], page 13, (method)
- [`(setf last-login-at)`], page 13, (method)
- [`(setf last-login-at)`], page 13, (generic function)
- [`name`], page 13, (generic function)
- [`name`], page 13, (method)

5 Definitions

Definitions are sorted by export status, category, package, and then by lexicographic order.

5.1 Exported definitions

5.1.1 Functions

`do-the-job` *FIRST SECOND* [Function]

The function does the job.

It CONCATENATES first and second arguments
calling internal function concat.

On this multiline we'll check how does documentation system processes docstrings.

NOTE: pay attention the Texinfo markup in the second paragraph does not supported in docstrings and I was not able to make CONCATENATES word bold :(

Package [`example/utils`], page 9,

Source [`file-type.lisp`], page 7, (file)

`foo` *FIRST* **&key** *OTHER* [Function]

This is example function.

* FIRST - Just a first argument.
* OTHER - Optional keyword argument. Default is 100500.

Returns: A string with first and other concatenated.

Internally it calls EXAMPLE/UTILS:DO-THE-JOB
to do the real job.

Note, that the link above is broken, but Coo does not warn us when building the docs. Sphinx issues a warning inn such case.

Package [`example/app`], page 9,

Source [`file-type.lisp`], page 7, (file)

5.1.2 Classes

`admin` () [Class]

Admins should have additional priveleges.

Package [`example/class`], page 9,

Source [`file-type.lisp`], page 7, (file)

Direct superclasses
[`user`], page 12, (class)

Direct methods
[`is-admin`], page 13, (method)

`non-documented-user` () [Class]

Package [`example/class`], page 9,

Source [`file-type.lisp`], page 7, (file)

Direct superclasses
[`user`], page 12, (class)

`user` () [Class]

All users in the system have this class.

Last login slot is updated automatically.

Package [`example/class`], page 9,

Source [`file-type.lisp`], page 7, (file)

Direct superclasses
`standard-object` (class)

Direct subclasses

- [`admin`], page 11, (class)
- [`non-documented-user`], page 12, (class)
- [`inner-documented-user`], page 14, (class)

Direct methods

- [`is-admin`], page 13, (method)
- `last-login-at` (method)
- [`last-login-at`], page 13, (method)
- [`email`], page 13, (method)
- [`name`], page 13, (method)

Direct slots

`name` [Slot]

A full username.

Type `string`

Initargs `:name`

Readers [`name`], page 13, (generic function)

`email` [Slot]

Correct email address.

Type `string`

Initargs `:email`

Readers [`email`], page 13, (generic function)

`last-login-at` [Slot]

Readers [`last-login-at`], page 13, (generic function)

Writers [`(setf last-login-at)`], page 13, (generic function)

5.2 Internal definitions

5.2.1 Functions

`concat` *FIRST SECOND* [Function]

This function is not exported and should not be showed in the API reference.

Package [`example/utils`], page 9,

Source [`file-type.lisp`], page 7, (file)

5.2.2 Generic functions

`email` *OBJECT* [Generic Function]

Package [`example/class`], page 9,

Methods

`email` (*USER* `user`) [Method]

Correct email address.

Source [`file-type.lisp`], page 7, (file)

`is-admin` *USER* [Generic Function]

Returns t if user can modify the system.

Package [`example/class`], page 9,

Source [`file-type.lisp`], page 7, (file)

Methods

`is-admin` (*USER* `admin`) [Method]

`is-admin` (*USER* `user`) [Method]

`last-login-at` *OBJECT* [Generic Function]
`(setf last-login-at)` *NEW-VALUE OBJECT* [Generic Function]

Package [`example/class`], page 9,

Methods

`last-login-at` (*USER* `user`) [Method]

automatically generated reader method

Source [`file-type.lisp`], page 7, (file)

`(setf last-login-at)` *NEW-VALUE* (*USER* `user`) [Method]

automatically generated writer method

Source [`file-type.lisp`], page 7, (file)

`name` *OBJECT* [Generic Function]

Package [`example/class`], page 9,

Methods

`name` (*USER* `user`) [Method]

A full username.

Source [`file-type.lisp`], page 7, (file)

5.2.3 Classes

`inner-documented-user ()` [Class]

This class only to demostrate how Declt's separates exported symbols from internal.

It will not be shown in the separate section "Internal Definitions".

Package [`example/class`], page 9,

Source [`file-type.lisp`], page 7, (file)

Direct superclasses
[`user`], page 12, (class)

6 Conclusion

Declt can be used be used when you need to generate API reference for third-party libraries as Quickref (https://quickref.common-lisp.net/) does for all Quicklisp libraries.

Ability to generate docs in different formats also might be interesting.

But the lack of markup support for docstrings and cross-reference helpers along with limited ability to create free form documentation chapters makes Declt useless for documenting 40Ants projects (https://40ants.com).

Appendix A Indexes

A.1 Concepts

A.2 Functions

A.3 Variables

A.4 Data types