Welcome to the Sim-Diasca 2.4.7 Documentation Overview Page

Public Version

We recommend to have a look at the Sim-Diasca General-Purpose Presentation first, in order to discover the engine.

Afterwards, one may refer to the installation guide below.

Alternatively, much shorter versions thereof are:

- either to follow these quickstart instructions:

```
# Ensure that a recent Erlang version is installed, then:
$ git clone https://github.com/Olivier-Boudeville-EDF/Sim-Diasca.git
$ cd Sim-Diasca && make test
```

- or to have a look at the streamlined executable procedure on which we rely for continuous integration

The following public documentation for **Sim-Diasca** version 2.4.7 is available from here:

The **Sim-Diasca Technical Manual**	The richest detailed source of information regarding the engine (except the code itself of course).	[html]	[pdf]
The **Sim-Diasca Installation Guide**	A much detailed installation walkthrough, if needed.	[html]	[pdf]
The **Sim-Diasca Modeller Guide**	Hints to better design simulation models.	[html]	[pdf]
The **Sim-Diasca Developer Guide**	General recommendations when developing services offered by the engine (also applicable for model implementation).	[html]	[pdf]
The **Sim-Diasca Dataflow HOWTO**	For the very specific use cases where a dataflow paradigm would be more suitable than a multi-agent one.	[html]	[pdf]
The **Sim-Diasca Coupling HOWTO**	When having to couple third-party models, instead of writing them directly as Sim-Diasca ones.	[html]	[pdf]

Seasoned Sim-Diasca users may just bookmark the Sim-Diasca Cheat Sheet.

As an example, this IFL2021 article discusses an application case (the ACME simulator) and, starting from its section 5, offers a walk-through of the full corresponding software stack. See also this overview of the metaprogramming taking place with Sim-Diasca.

One may also be interested in the interface of the public main project and in its wiki.

A mirrored documentation of the lower layers used by Sim-Diasca is available

from here as well:

The **Traces Manual**	To benefit from convenient, distributed traces in order to collect all information of interest from the simulation.	[html]	[pdf]
The **WOOPER Manual**	So that the model implementation can rely on a suitable object-oriented paradigm.	[html]	[pdf]
The **Myriad Manual**	To take advantage of a general-purpose toolbox providing a range of built-in transverse services.	[html]	[pdf]

Seasoned WOOPER users may just bookmark the WOOPER Cheat Sheet.

Finally, regarding the overall implementation language, Erlang, one may refer to its searchable online documentation, including its module index.

This public overview page for Sim-Diasca version 2.4.7 has been last updated on Tuesday, February 13, 2024.