

Virtual Talk Series – DES Tools – CIF

Dennis Hendriks, Eclipse ESCET™ project lead, ESI (TNO) / Radboud University

April 21, 2022

CIF supports synthesis-based engineering of supervisory controllers

Supervisory controllers

- Cyber-physical systems, e.g., X-ray systems, lithography systems, industrial printers, waterway locks
- More and more complex control software (the cyber part), to ensure correct and safe behavior

Synthesis-based engineering (SBE)

- Combines model-based engineering with computer-aided design
- Synthesize correct-by-construction controllers, using hyper-automation

Main benefits

- Better quality controllers by reducing human error
- Reduced engineering effort by focusing on what is essential
- Supports large industrial systems by efficient controller engineering

For more information: <https://eclipse.org/escet/cif/synthesis-based-engineering>

CIF language

CIF is a powerful declarative modeling language for the specification of discrete event, timed, and hybrid systems as a collection of synchronizing automata

Finite state machines

- Automata
- Synchronizing events
- Point-to-point channels
- Non-determinism
- Monitors

Support for large systems

- Automaton definitions/instantiations
- Groups (+ definitions/instantiations)
- Imports

Extended finite state machines

- Data (variables, guards, updates)
- Discrete variables
- Algebraic variables (+ equations)
- Input variables
- Constants
- Rich built-in data-types
- Custom types / type declarations
- Initialization predicates
- Invariants
- Functions (also as data)

Timed state machines

- Time variable
- Continuous variables (+ equations)
- Urgency concepts

Other extensions

- Supervisory controller synthesis
- Stochastics
- SVG visualization/interaction
- Print output

For more information: <https://eclipse.org/escet/cif/language-tutorial>

CIF tools

The CIF tooling supports the entire development process of controllers

Specification

- Textual editor
- Conversion to yEd diagrams

Supervisory controller synthesis

- Data-based synthesis
- Event-based synthesis
- Conversion to Supremica

Simulation, visualization and validation

- Interactive simulation
- Automated simulation
- Powerful visualization features

Verification

- Controller property checker
- Conversion to mCRL2
- Conversion to UPPAAL

Implementation

- PLC code generation
- Java code generation
- C and Simulink code generation

Other

- State space generator (explorer)
- CIF to CIF transformations
- Merger
- Event disabler

For more information: <https://eclipse.org/escet/cif/tools>

CIF in action

The screenshot displays the Eclipse ESCET IDE interface for a CIF simulation. The main editor shows the following code:

```
15 group tank:
16   cont V = 10.0;
17   alg real Qi = controller.n * 5.0;
18   alg real Qo = sqrt(V);
19   equation V' = Qi - Qo;
20
21   svgout id "water" attr "height" value 7.5 * V;
22   svgout id "V" text value fmt("V = %.1f", V);
23   svgout id "Qi" text value fmt("Qi = %.1f", Qi);
24   svgout id "Qo" text value fmt("Qo = %.1f", Qo);
25 end
26
27 automaton controller:
28   alg int n;
29
30   location closed:
31     initial;
32     equation n = 0;
33     edge when tank.V <= 2 goto opened;
34
35   location opened:
```

The **Plot Visualizer** shows a graph of variables over time (0 to 35). The legend includes: controller.n (red), tank.Qi (green), tank.Qo (blue), tank.V (yellow), and tank.V (orange).

The **State Visualizer** table shows the current state:

Name	Value
time	39.599999999999994
controller	opened
controller.n	1
tank.Qi	5.0

The **SVG Visualizer** shows a schematic of a tank with a valve. The valve is labeled 'n' and 'Qi = 5.0'. The tank is labeled 'VC' and 'V = 9.0'. The outlet is labeled 'Qo = 3.0'.

The **Console** shows the following output:

```
ToolDef interpreter [TERMINATED after 42s 638ms] /CIFExamples-0.1.0.qualifier/hybrid/tank/tank.tooldef (started at 2020-05-07 14:28:53.278)

Transition: delaying for 3.1883805325514274 time units at time 36.918671836315674
Simulation was terminated per the user's request.
```

CIF is ready to use

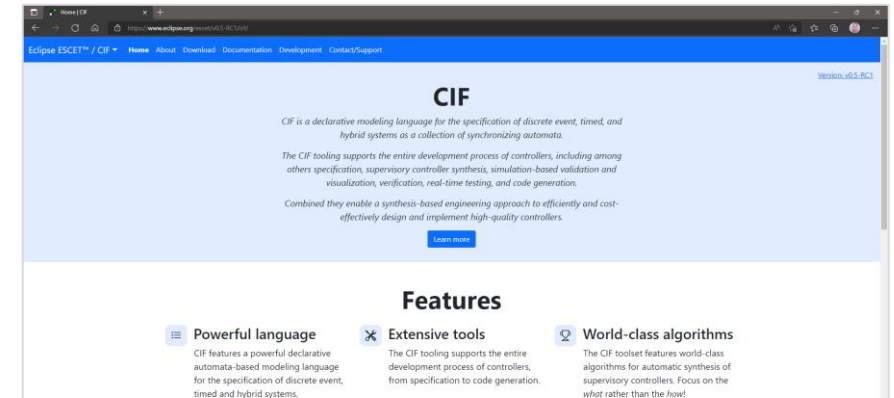
Long history

- 30+ years of academic and applied SBE research
- 15 years of CIF language/tool development (origin at TU/e)

Mature and proven technology

- Thoroughly tested tools
- Extensive documentation
- Successful case studies in many domains
 - E.g., healthcare, automotive, infrastructure, semiconductor
- Proven real-world maturity
 - E.g., car drove in real traffic (cruise control)
 - E.g., bridge passed site acceptance test

For more information: <https://eclipse.org/escet/cif>



CIF

Learn more about CIF from the extensive documentation that covers both the CIF language and tools:

- [CIF documentation home page](#)
- [CIF synthesis-based engineering](#)
- [CIF language tutorial](#)
- [CIF language reference](#)
- [CIF tools](#)
- [CIF examples](#)
- [CIF release notes](#)

CIF is developed as part of the Eclipse ESCET open-source project

Eclipse ESCET project

- An Eclipse Foundation project since 2020
- Home to CIF, and some other tools (Chi and ToolDef)
- Learn more at: <https://eclipse.org/escet>

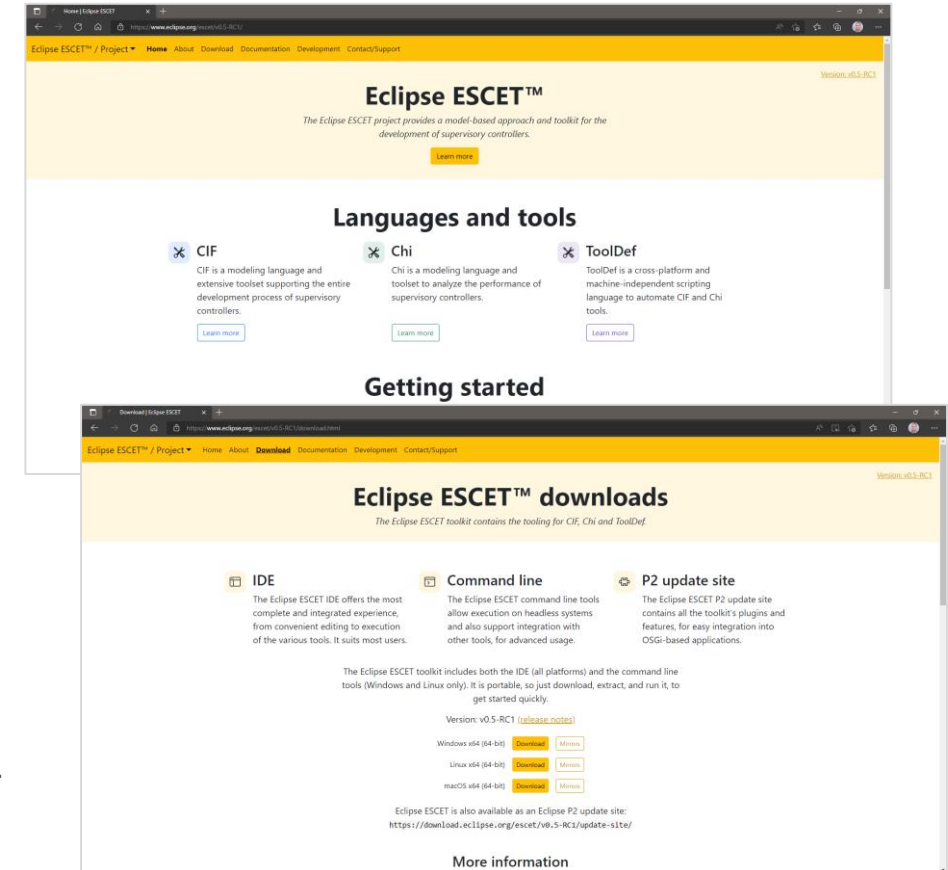
Download for free

- Eclipse ESCET IDE and command line tools
- Windows, Linux, macOS
- Portable: extract and use

Active development

- Quarterly releases
- 100+ end user visible improvements and fixes since April 2021

You can get started today!





Eclipse ESCET and ESCET are trademarks of the Eclipse Foundation. Eclipse and the Eclipse Logo are registered trademarks of the Eclipse Foundation. Other names may be trademarks of their respective owners.

Copyright (c) 2010, 2022 Contributors to the Eclipse Foundation.



Radboud University

