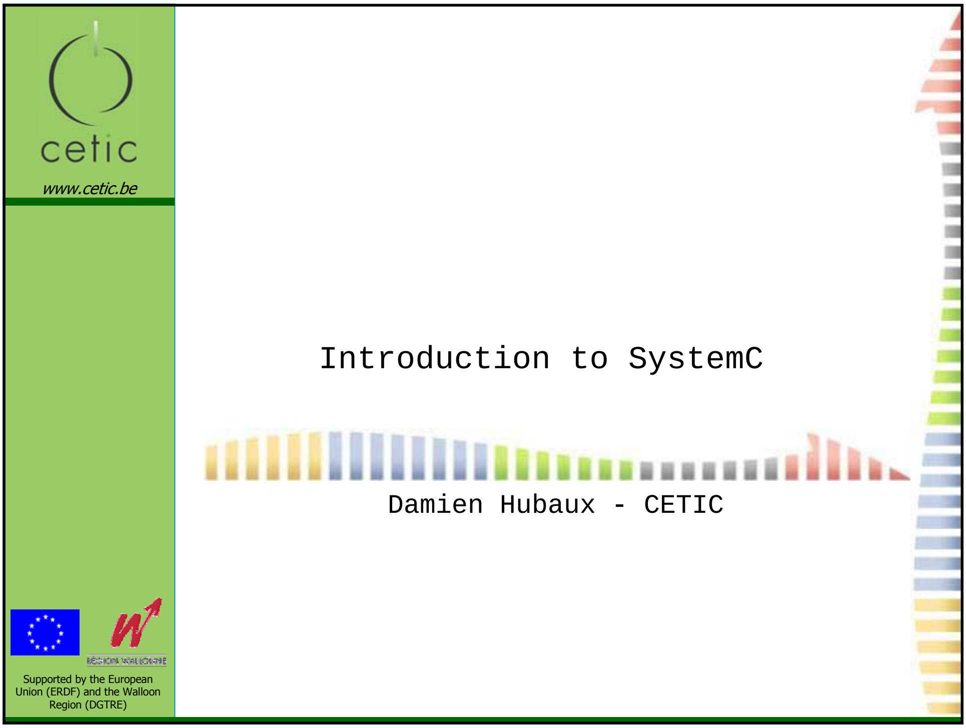
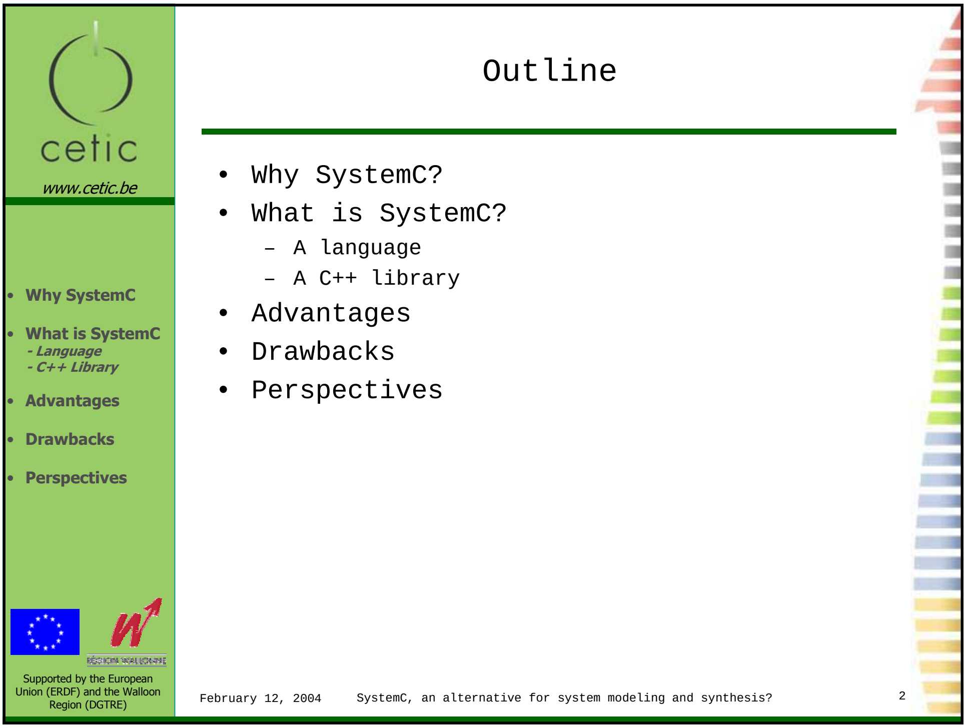


cetic
www.cetic.be

Introduction to SystemC

Damien Hubaux - CETIC

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)



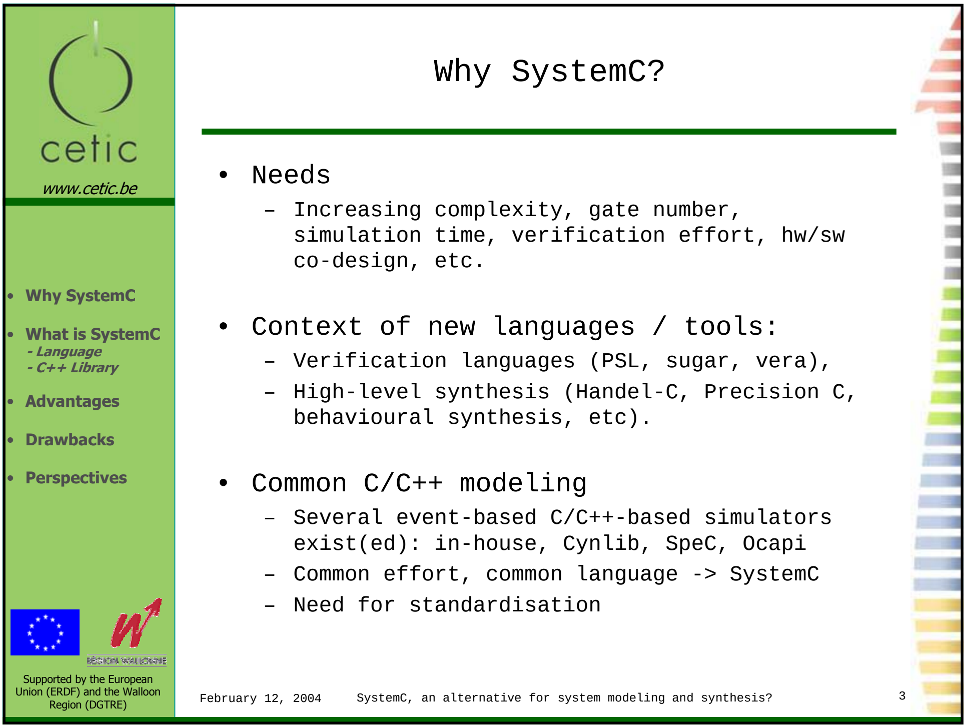
cetic
www.cetic.be

- Why SystemC
- What is SystemC
  - Language
  - C++ Library
- Advantages
- Drawbacks
- Perspectives

Outline

- Why SystemC?
- What is SystemC?
  - A language
  - A C++ library
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

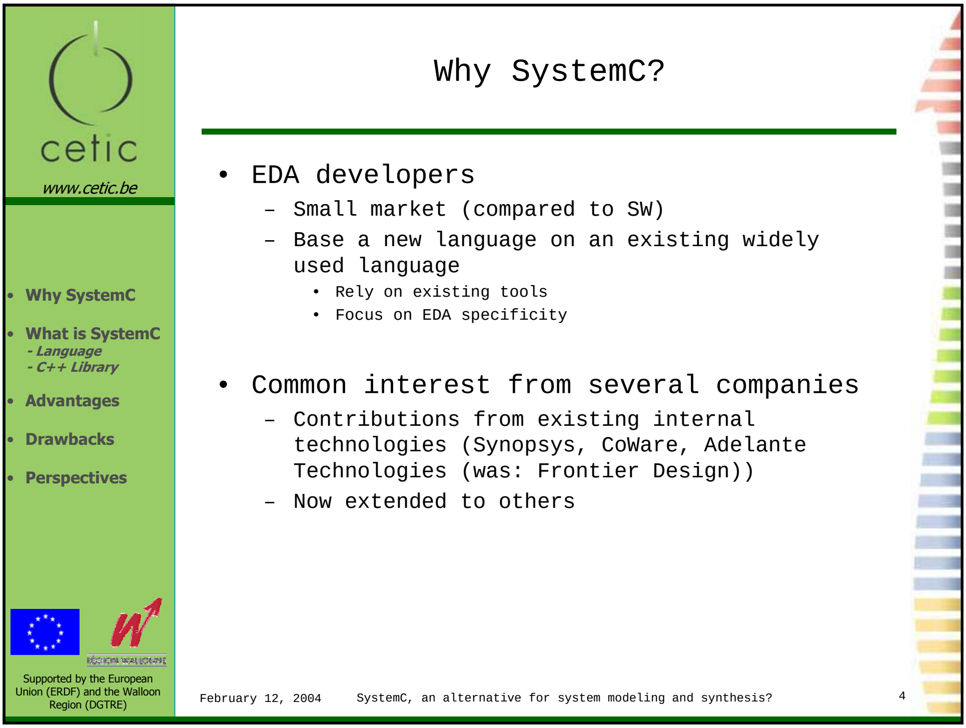


- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Why SystemC?

- Needs
  - Increasing complexity, gate number, simulation time, verification effort, hw/sw co-design, etc.
- Context of new languages / tools:
  - Verification languages (PSL, sugar, vera),
  - High-level synthesis (Handel-C, Precision C, behavioural synthesis, etc).
- Common C/C++ modeling
  - Several event-based C/C++-based simulators exist(ed): in-house, Cynlib, SpeC, Ocapi
  - Common effort, common language -> SystemC
  - Need for standardisation



- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Why SystemC?

- EDA developers
  - Small market (compared to SW)
  - Base a new language on an existing widely used language
    - Rely on existing tools
    - Focus on EDA specificity
- Common interest from several companies
  - Contributions from existing internal technologies (Synopsys, CoWare, Adelante Technologies (was: Frontier Design))
  - Now extended to others

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

What is SystemC?

- OSCI: Open SystemC Initiative
- OSCI’s Language Reference Manual (LRM)
  - A Hardware Description language
  - A C++ library that implements hardware concepts (clock, ports, concurrency, etc)
- The reference implementation of this language
  - Reference!!! (one of the possible...)
  - Open-source, freely available

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

What is SystemC?

- Associated libraries (from OSCI)
  - Extension are possible
  - Master-Slave library: abstract communications
  - Verification library: offers to SystemC more verification features (assertion, introspection, etc.)
- Contributions of the OSCI working groups
  - Language
  - Transaction Level Modeling
  - Verification
  - Synthesis

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

What is SystemC?

- There are (have been) 2 (successive) "opinions" about SystemC
- RTL modeling using C++
  - What is the benefit compared to my favourite HDL?
  - (Looks suspect to HW designers)
- System level modeling
  - Between system (C++, Matlab, etc), software, and hardware designers (HDL)
    - C/C++ is possible
    - RTL is possible
    - Everything between
    - Mixing algorithmic and RTL
    - Mixing HW and SW

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Outline

- Why SystemC?
- What is SystemC?
  - A language
  - A C++ library
- Advantages
- Drawbacks
- Perspectives

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

A Hardware Description Language

- You can write (RTL) SystemC without really knowing C++
- Let's make a short comparison:

VHDL / SystemC

ENTITY Mouse IS

SC_MODULE(Mouse)

PROCESS Phone (ring)

SC_METHOD(Phone);
sensitive<<ring;

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Declaration

```
entity my_module is
[ports]
end my_name


architecture my_arch of
   my_module is
  [components]
  [types]
  [signals]
begin
  [processes (all)]
  [combinatorial]
end
```

```
SC_MODULE(my_module)
{
 [ports]
 [signals]
 [modules]
 [processes (decl.)]
 SC_CTOR(my_module){
  [processes
   (sensitivity)]
 }
};


[processes
   (implementation)]
```

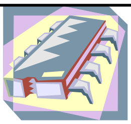
cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Entity / Ports



```
entity my_name is
port(
  ... [ports]
  );
end my_name
```

```
SC_MODULE(my_name)
{
  [ports]
};
```

```
port (
 input: port1 std_logic;
 output: port2 std_logic;
);
```

```
{
  sc_in<bool> port1;
  sc_out<bool> port2;
  ...
};
```

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Hierarchy

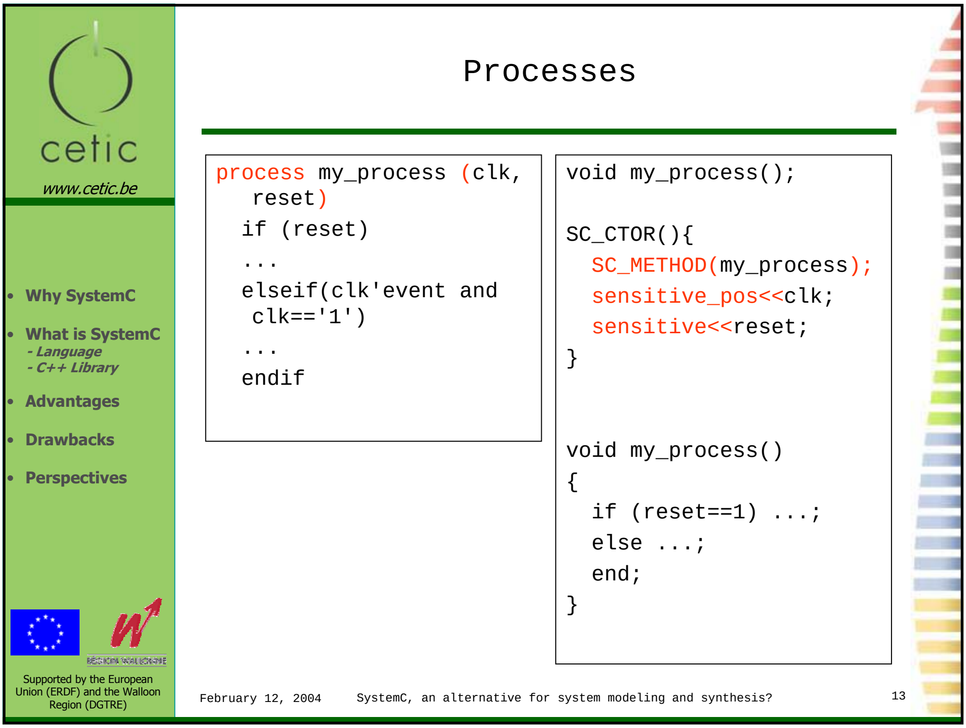
```
architecture my_arch
   of my_module is

component child
  port(...);
end component;

Begin
child_inst : child
  port map(
  in => signal1;
  ...
  );
...
```

```
#include "child.h" //decl

SC_MODULE(my_module)
{
 child* child_inst;
 SC_CTOR{
  child_inst = new child;
  child_inst.in->(signal1);
  ...
 }
};
```



cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

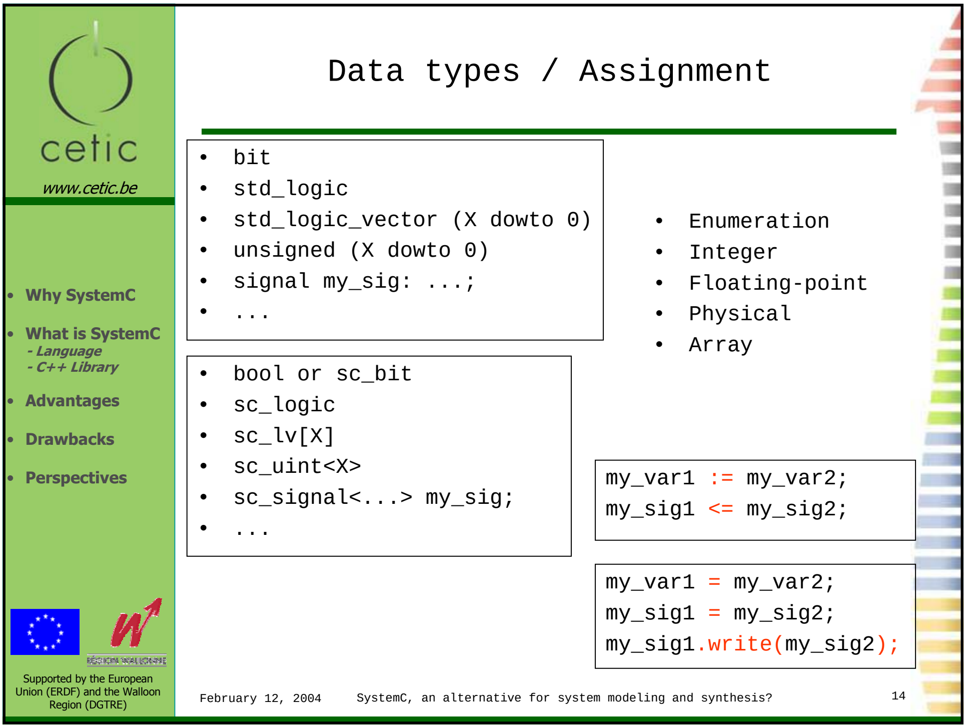
Processes

```
process my_process (clk,
   reset)
  if (reset)
  ...
  elseif(clk'event and
   clk=='1')
  ...
  endif
```

```
void my_process();

SC_CTOR(){
  SC_METHOD(my_process);
  sensitive_pos<<clk;
  sensitive<<reset;
}


void my_process()
{
  if (reset==1) ...;
  else ...;
  end;
}
```



cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Data types / Assignment

- bit
- std_logic
- std_logic_vector (X dowto 0)
- unsigned (X dowto 0)
- signal my_sig: ...;
- ...

- Enumeration
- Integer
- Floating-point
- Physical
- Array

- bool or sc_bit
- sc_logic
- sc_lv[X]
- sc_uint<X>
- sc_signal<...> my_sig;
- ...

```
my_var1 := my_var2;
my_sig1 <= my_sig2;
```

```
my_var1 = my_var2;
my_sig1 = my_sig2;
my_sig1.write(my_sig2);
```

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Outline

- Why SystemC?
- What is SystemC?
  - A language
  - A C++ library
- Advantages
- Drawbacks
- Perspectives

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

A C++ library

- Classes that implement hardware data-types and concepts
- A simulation kernel: registering functions in order to allow "parallel" execution.
- Possibility to extend with own data-types, concepts
  - Support for own methodology
  - Support for own libraries
  - Support all C/C++ you want

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

A C++ libray

Some C++ constructs are replaced by macro's

The module properties and functions are in fact inherited.

HDL: access through ports

C++: member access

```
SC_MODULE(my_module)
{
  SC_CTOR(my_module){};
};
```

```
class my_name: public
   sc_module
{
public:
 my_module():sc_module()
 {...};
};
```

cetic

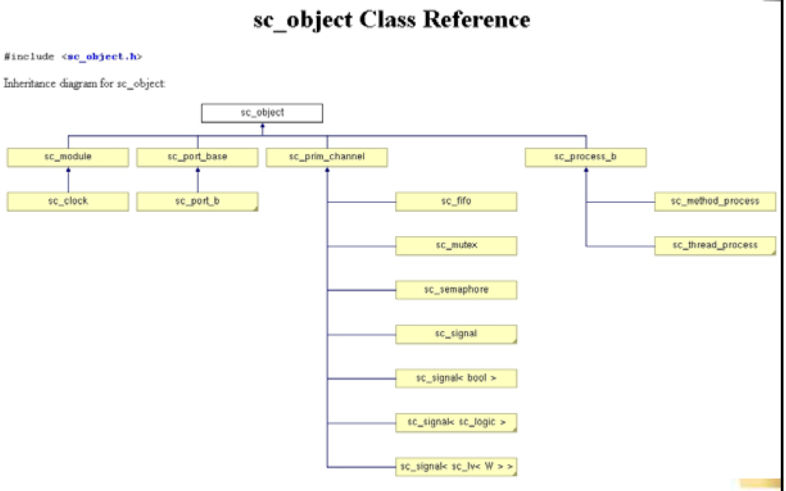
www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

SystemC internal: sc_object

http://www.iro.umontreal.ca/~chareslu/systemc-2.0.1/



sc_object Class Reference

`#include <sc_object.h>`

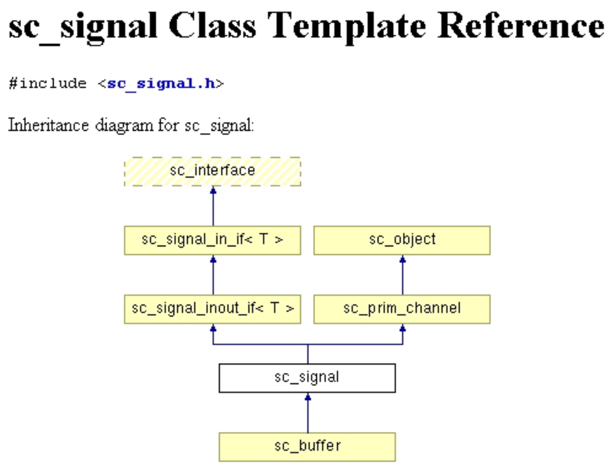
Inheritance diagram for sc_object:

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

SystemC internal: sc_signal<T>



sc_signal Class Template Reference

`#include <sc_signal.h>`

Inheritance diagram for sc_signal:

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Outline

- Why SystemC?
- What is SystemC?
  - – A language
  - – A C++ library
- Advantages
- Drawbacks
- Perspectives

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

SystemC advantages

- Open-source reference implementation
  - OSCI released not only the LRM, but an open-source reference implementation that everyone can freely download and use.
- Some (open-source) software development tools allow to create a HW simulation environment
  - Simulation = compile a C++ program
  - Large number of software development tools, from vendors and Open-source
  - (but need more discussion: later)

cetic

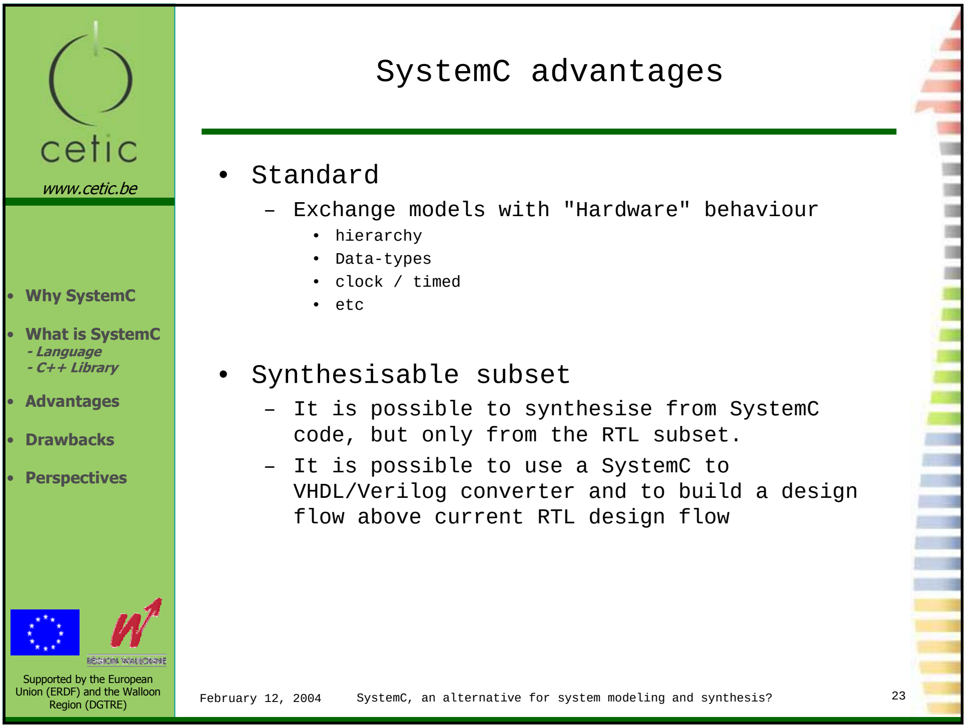
www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

SystemC advantages

- C++ library / Object-Oriented
  - Can add own data types
  - Can add own library in order to support own methodology
  - Allows more than RTL. You can use all C++ you want (not for synthesis)
  - SystemC capabilities can be extended. There are add-on libraries:
    - Master-Slave: an abstract communication scheme
    - Verification: add assertion based verification to SystemC
- Verification
  - System testbench can be applied at each level of abstraction



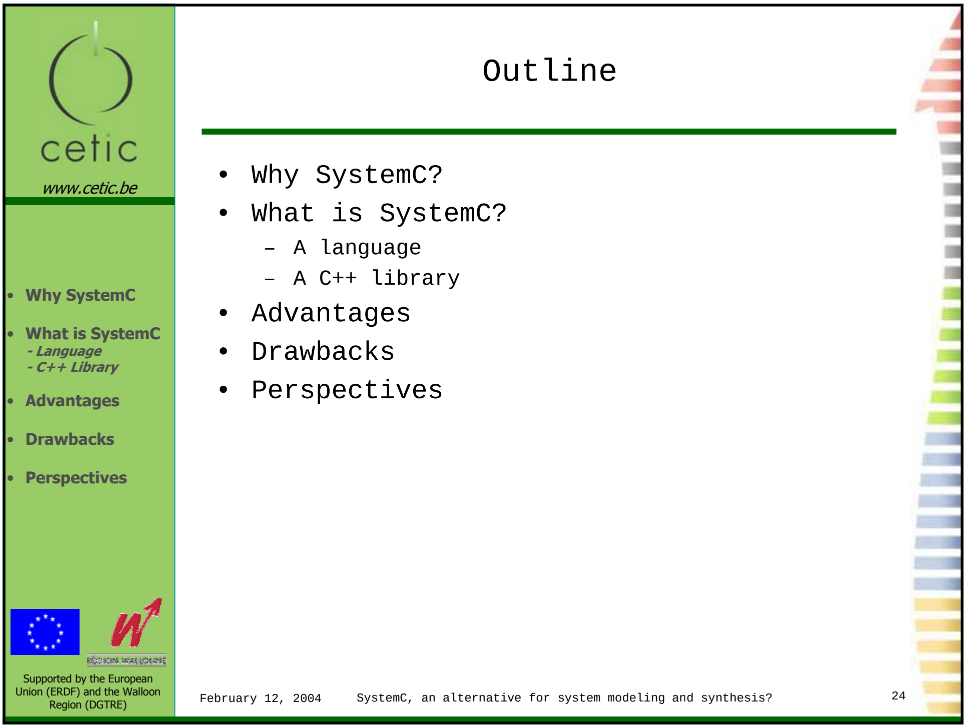
cetic
www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

SystemC advantages

- Standard
  - Exchange models with "Hardware" behaviour
    - hierarchy
    - Data-types
    - clock / timed
    - etc
- Synthesisable subset
  - It is possible to synthesise from SystemC code, but only from the RTL subset.
  - It is possible to use a SystemC to VHDL/Verilog converter and to build a design flow above current RTL design flow



cetic
www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Outline

- Why SystemC?
- What is SystemC?
  - A language
  - A C++ library
- Advantages
- Drawbacks
- Perspectives

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

SystemC drawbacks

- Synthesis
  - Tool support, model availability, etc
- Some awkward constructs
  - Data-type conversion, casts
- Compilation / runtime errors:
  - C++ show-up!
  - Obscure messages, core dump, etc.
  - SystemC syntax checking would be better than C++ syntax checking
- Debugging
  - Tools debug C++, not SystemC

cetic

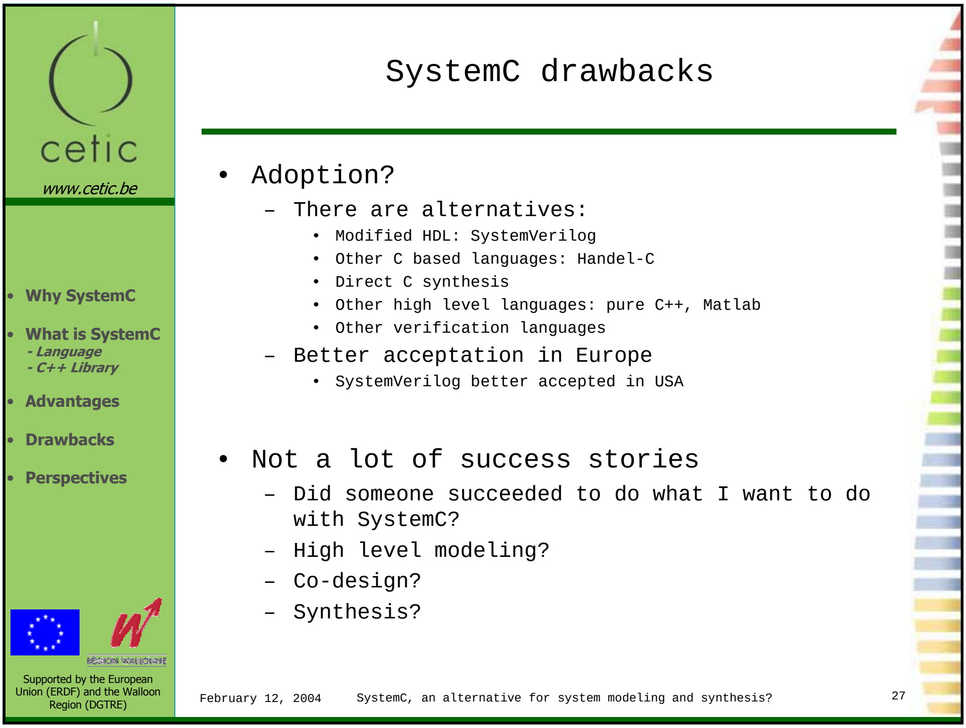
www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

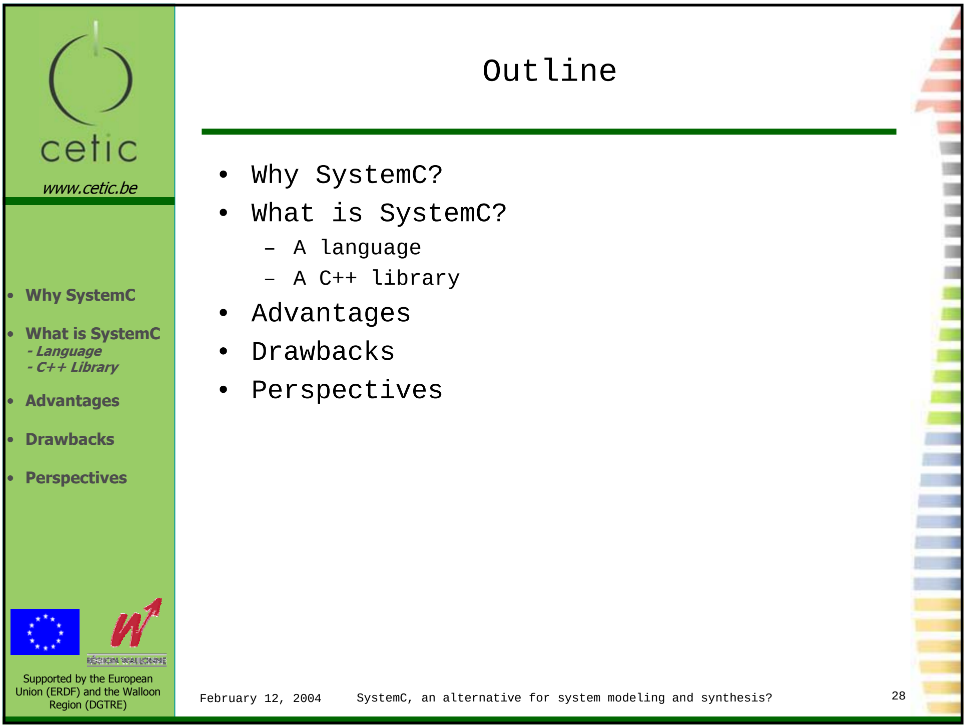
SystemC drawbacks

- SystemC : C++ faster than VHDL?
  - If you make high level simulation: Yes
    - (methodology is different, comparison is not fair)
  - Not guaranteed with RTL code
    - Beware that the OSCI reference simulator is not optimised for speed -> vendor SystemC kernel?
- Some limitations
  - No dynamic instantiation of modules (logic from a purely hardware point of view)



SystemC drawbacks

- Adoption?
  - There are alternatives:
    - Modified HDL: SystemVerilog
    - Other C based languages: Handel-C
    - Direct C synthesis
    - Other high level languages: pure C++, Matlab
    - Other verification languages
  - Better acceptation in Europe
    - SystemVerilog better accepted in USA
- Not a lot of success stories
  - Did someone succeeded to do what I want to do with SystemC?
  - High level modeling?
  - Co-design?
  - Synthesis?



Outline

- Why SystemC?
- What is SystemC?
  - A language
  - A C++ library
- Advantages
- Drawbacks
- Perspectives

cetic

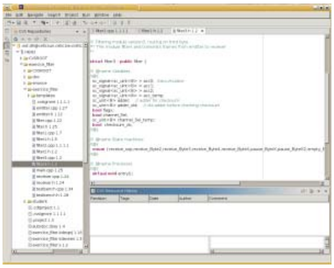
www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Perspectives

- Tools
  - EDA use custom languages, custom tools
  - SystemC allows to rely heavily on software tools (compare number of HDL users to software users…)
  - Ex: Eclipse IDE
    - A free OO IDE: common facilities built-in (user interface, file handling, editor, CVS, etc)
    - Allows to add custom HDL support (parser, etc)
    - HDL specific tools remain (synthesis, ad-hoc modeling, verification and methodology)



cetic

www.cetic.be

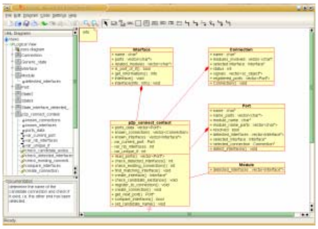
- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Perspectives

- CASE tools
  - Versionning (also works with VHDL)
  - Documentation: generate (HTML) documentation from code
  - UML: standard graphical representation
  - UML: coding rules, partial C++ code generation

- State machines
- Classes
- Sequence diagrams
- Structured classes (new in UML2)



cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

RÉGION WALLONNE

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Perspectives

- SystemC is C++ but codesign (HW-SW) is not trivial: you have to set-up your how methodology (but easier with a single language)
- Extended synthesisable subset
  - Templates
  - Inheritance
- Use / Adoption
  - See Doulos survey...

cetic

www.cetic.be

- Why SystemC
- What is SystemC
  - *- Language*
  - *- C++ Library*
- Advantages
- Drawbacks
- Perspectives

RÉGION WALLONNE

Supported by the European Union (ERDF) and the Walloon Region (DGTRE)

Follow up

- Conclusion?
  - first see other presentations...