



FZI

Forschungszentrum Informatik

an der Universität Karlsruhe

Microelectronic System Design (SIM)



SCE

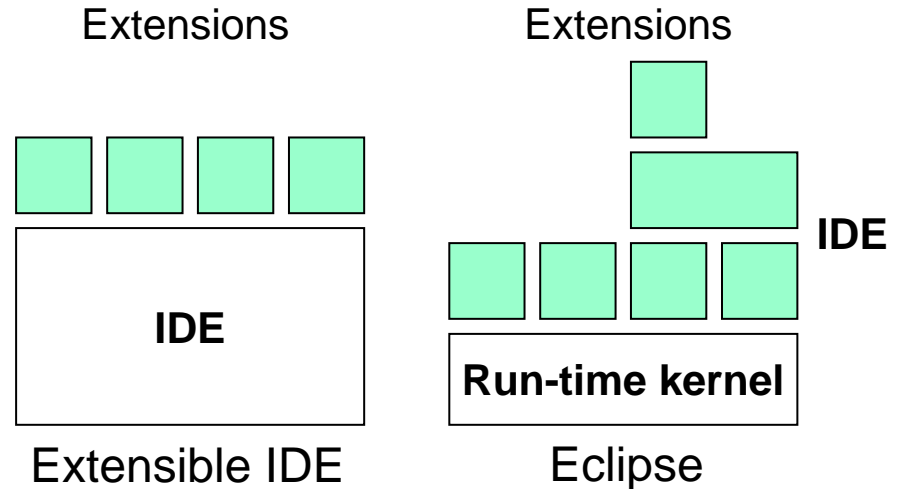
A SystemC Environment for Eclipse

Alexander Viehl, Oliver Bringmann

{viehl | bringman}@fzi.de

- Motivation
- Overview of Current Features
- Example: Visualization
- More to come - what could be done

Motivation



- Widely used platform and framework for software and application development
- Well known to many developers
- Open Source under the *Common Public License (CPL)*
- Modular and extensible by plug-ins
- Free available plug-in for C/C++ development (CDT)
- Complex inheritance between plug-ins possible

Overview of Current Features

- Project management
 - Add, Import, Export, new SystemC Project, new SystemC File
 - Wizards
- Configuration
 - Standard e.g. compiler and libraries
 - Syntax coloring and highlighting
 - Visualization properties
- Visualization
 - Perspective
 - Editor (syntax coloring and highlighting)
 - Outline View (hierarchy tree and structural visualization)
- Model execution
 - Compile using defined compiler and libraries
 - Execute using OSCI simulator

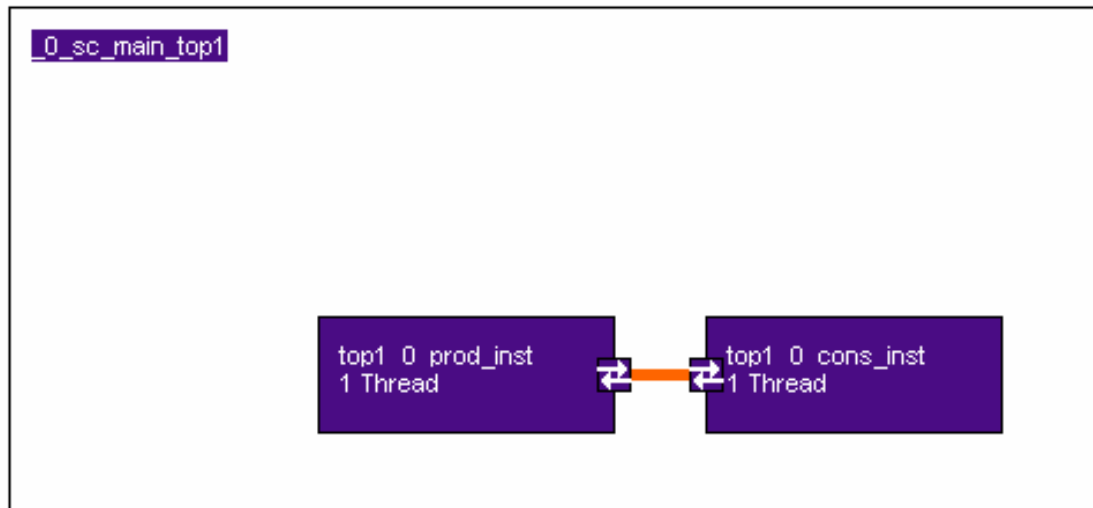
- Syntax highlighting
 - Color configuration relating to categories
 - Extensible due to extensibility of SystemC
- Tree viewer
 - Browsing of structural hierarchies
 - Fast access for analytical tools
- Structural representation
 - SystemC style graphical representation of system model
 - Using ports, channels, interfaces
 - Color configuration of elements
 - Zoom in/out of hierarchical structures

Example: Syntax Highlighting

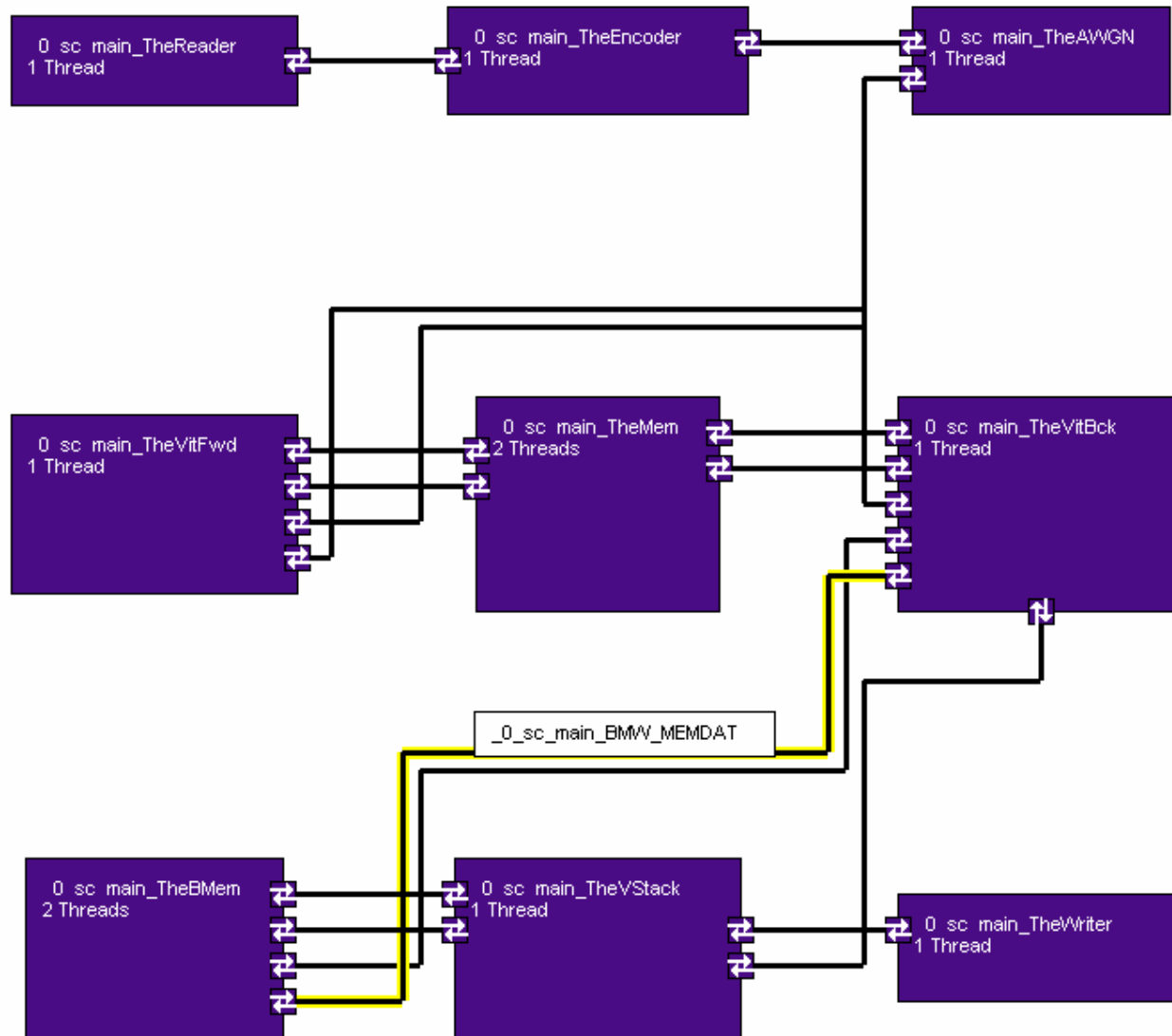
```
//begin of module awgn
SC_MODULE(awgn)
{
    sc_out<unsigned long> sout;
    sc_in<unsigned long> sin;
    double mean;
    double variance;
    double amplitude;
    /*int numsym;
    long seed;*/
    unsigned int rn;
    unsigned int range;

    SC_CTOR(awgn) {
        SC_METHOD(vcc_run);
        sensitive << sin;
        dont_initialize();
        mean=Mean;
        variance=Variance;
        amplitude=Amplitude;
        numsym=1 << NumBits;
        seed=1234567890;
        range=32767;
        rn=0;
    };
};
```

Example: Structural Visualization



Example: Structural Visualization



More to come – what could be done

- Extending CDT
 - Managed makefile
 - Refactoring
 - SystemC code completion
- SystemC specific refactoring
 - Interfaces
 - SC_METHOD → SC_THREAD
 - SC_THREAD → SC_HAS_PROCESS
- Code generation
 - Using e.g. external UML2 tools
 - Graphical modeling using e.g. GEF/EMF
 - Interfaces and structure from SPIRIT
- Providing API for inclusion of user specific tools and annotations
 - Transaction recording e.g. DUST
 - Refinement and mapping
 - Visualization of analytical results
 - Linter
- Graphical user interface for debugging

