Tutorial:
PREESM - Dataflow Programming of Multicore DSPs

Karol Desnos, Clément Guy, Maxime Pelcat

EDERC 2014 Conference, Milan, September 11th

PREESM

http://preesm.sourceforge.net/website

- Eclipse-based Tool
- Written in Java and Xtend
- Using
  - Eclipse Modeling Framework,
  - Eclipse Graphiti,
  - Eclipse CDT
- Compatible and tested on Linux and Windows
- Release 2.0.0 on sept 2014

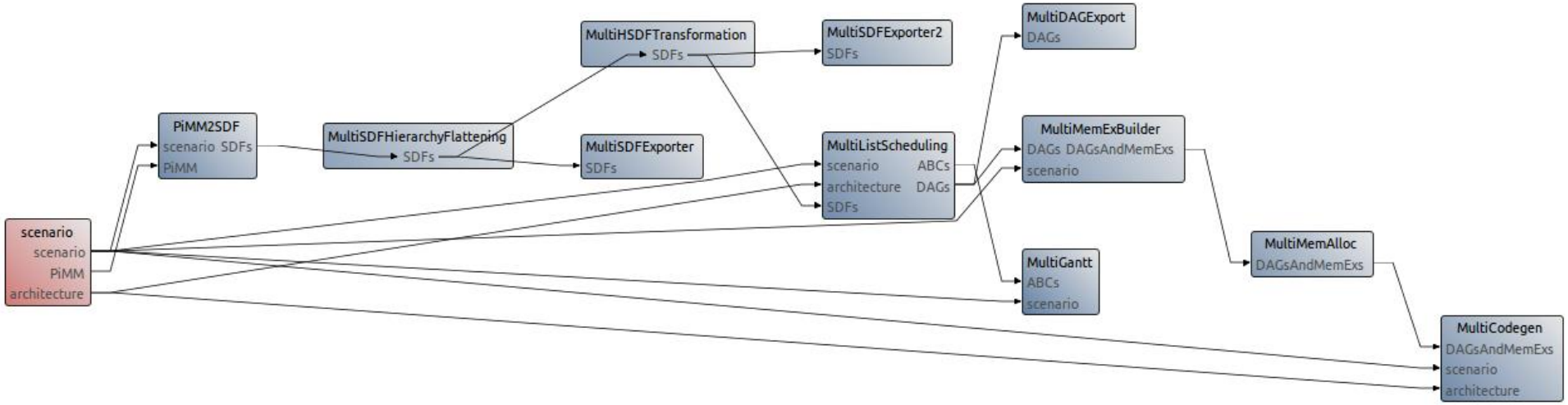
PREESM

- Started in 2007
- In collaboration with Texas Instruments France
- 16 contributors
- Academic collaborations
  - LAAS
  - University of Maryland
  - ENIS
  - Abo Akademi

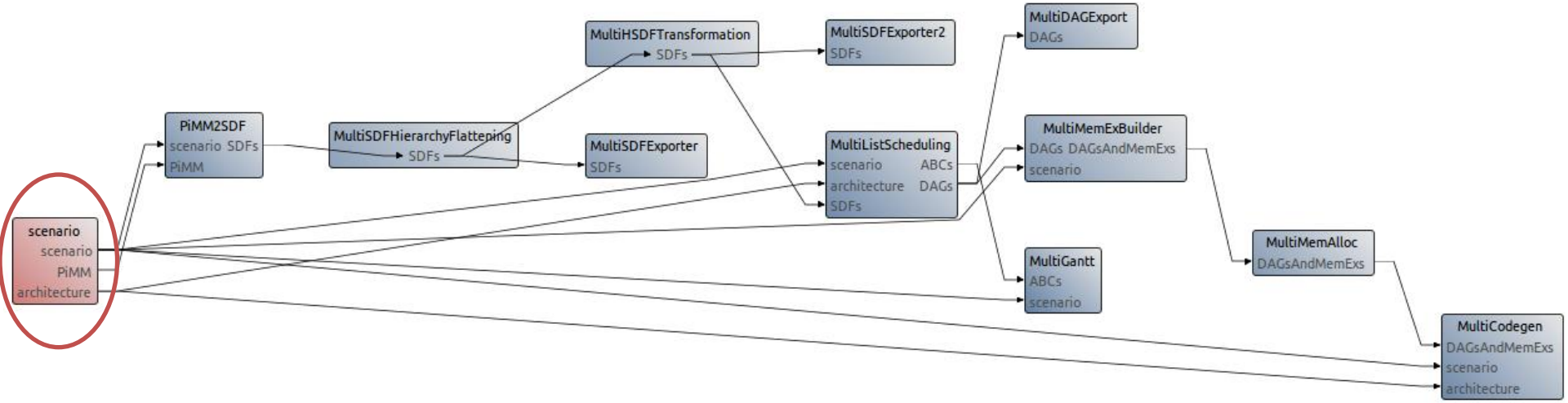
PREESM Workflow

- Preesm offers Editors
  - Algorithm
  - Architecture
  - Scenario
- And can run a Workflow
  - Transformations of models

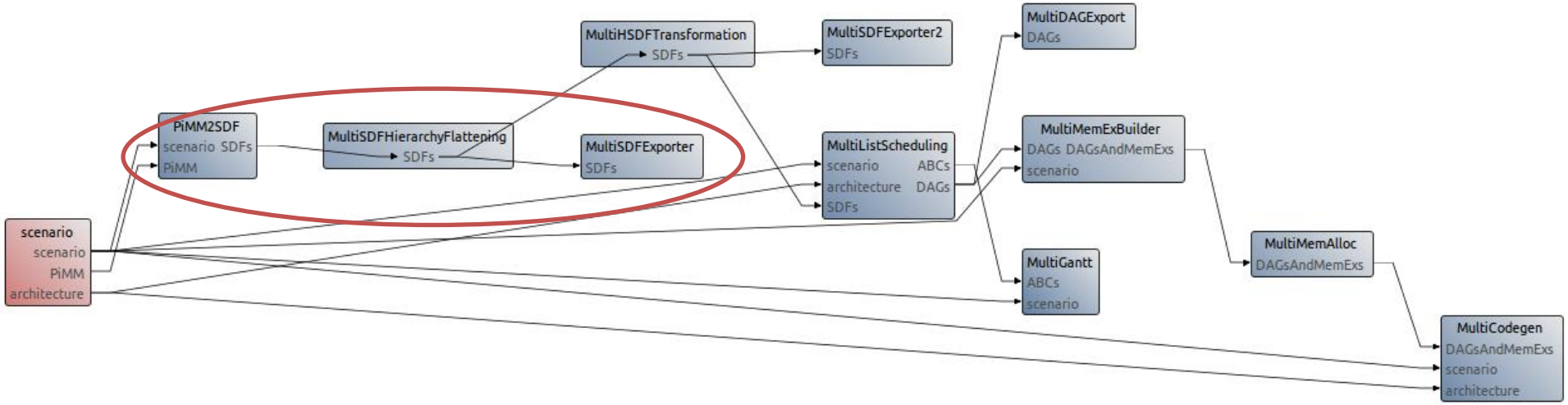
PREESM Workflow



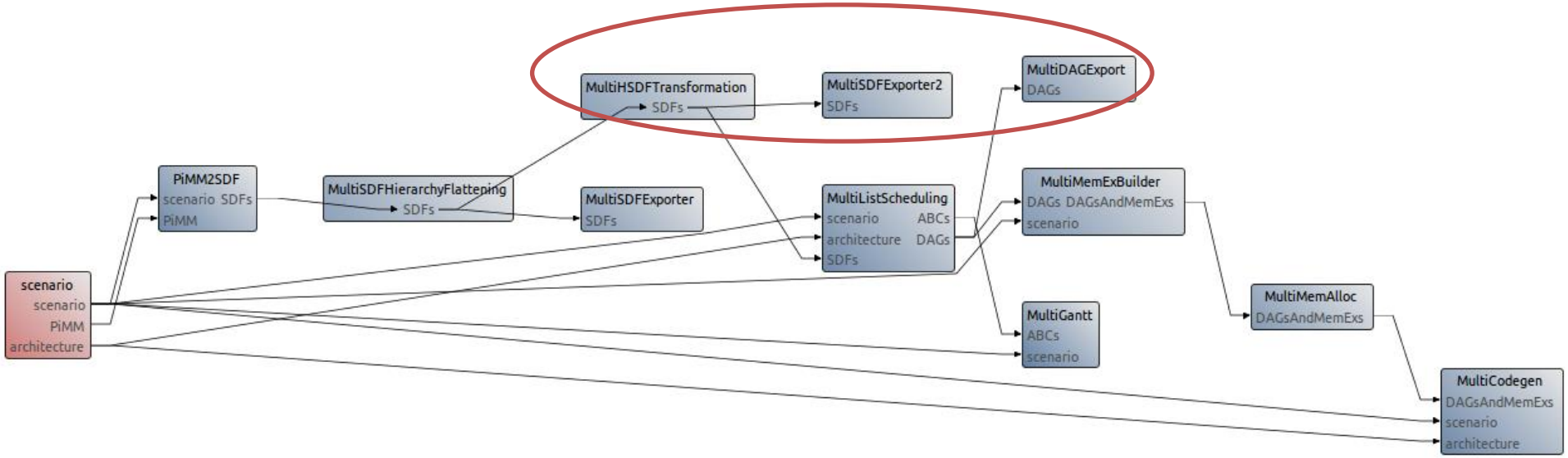
PREESM Workflow



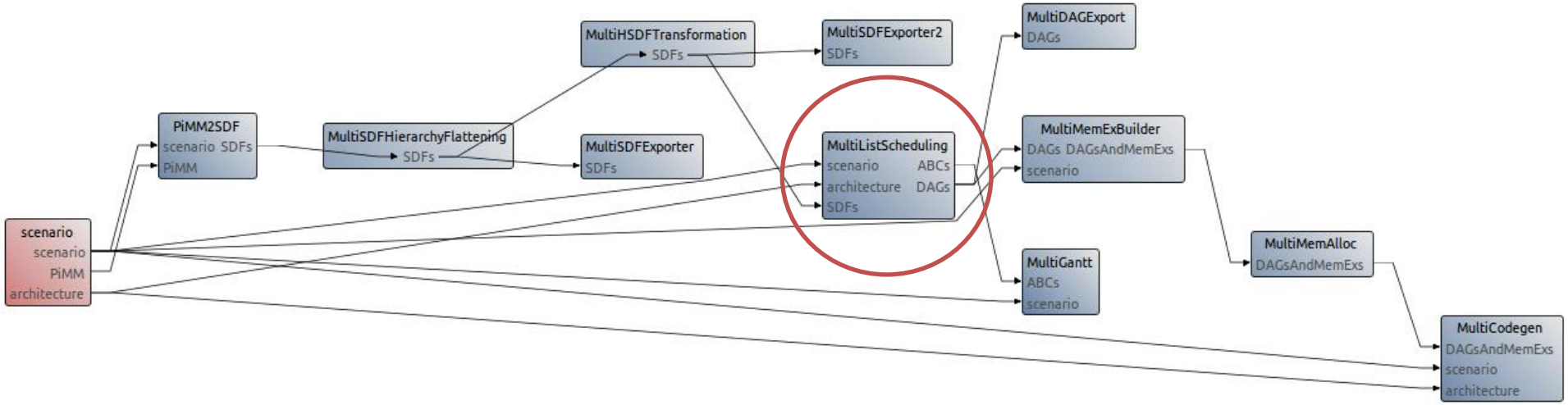
PREESM Workflow



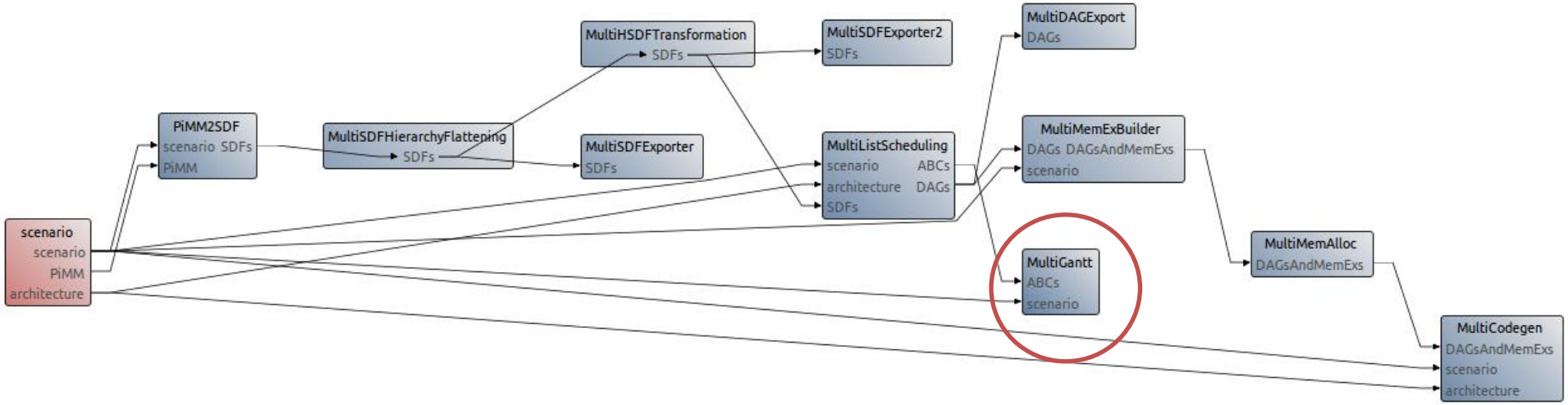
PREESM Workflow



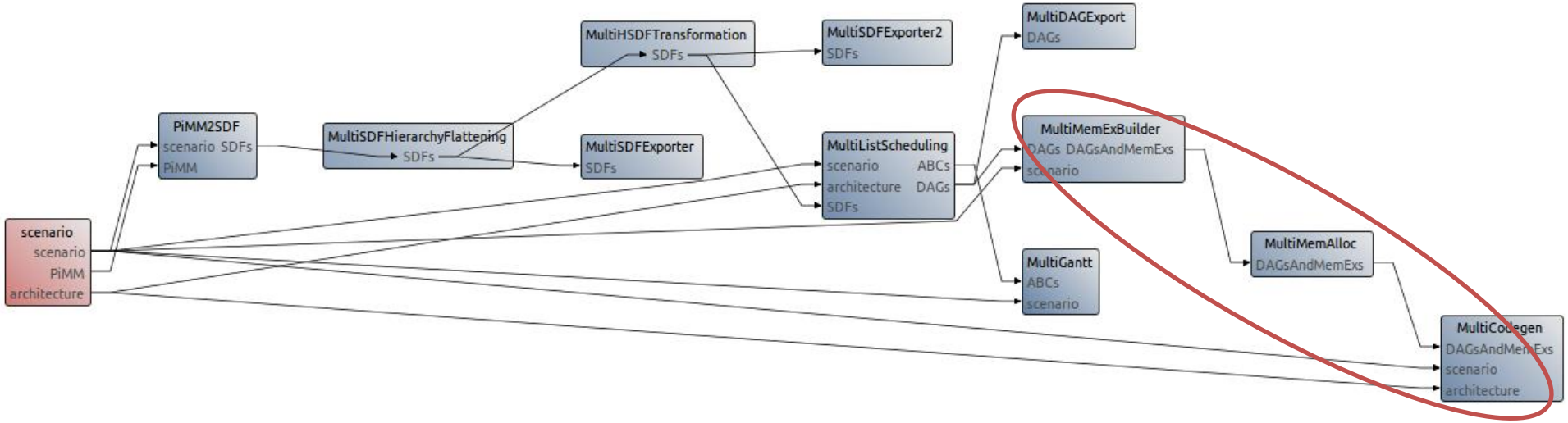
PREESM Workflow



PREESM Workflow



PREESM Workflow



PREESM Workflow

- A workflow runs typically within a few tens of seconds
- Algorithm: typically 10-1500 actors
- Architecture: typically 1-20 cores

Examples for the tutorial

Algorithms

Sobel filter: edge detection

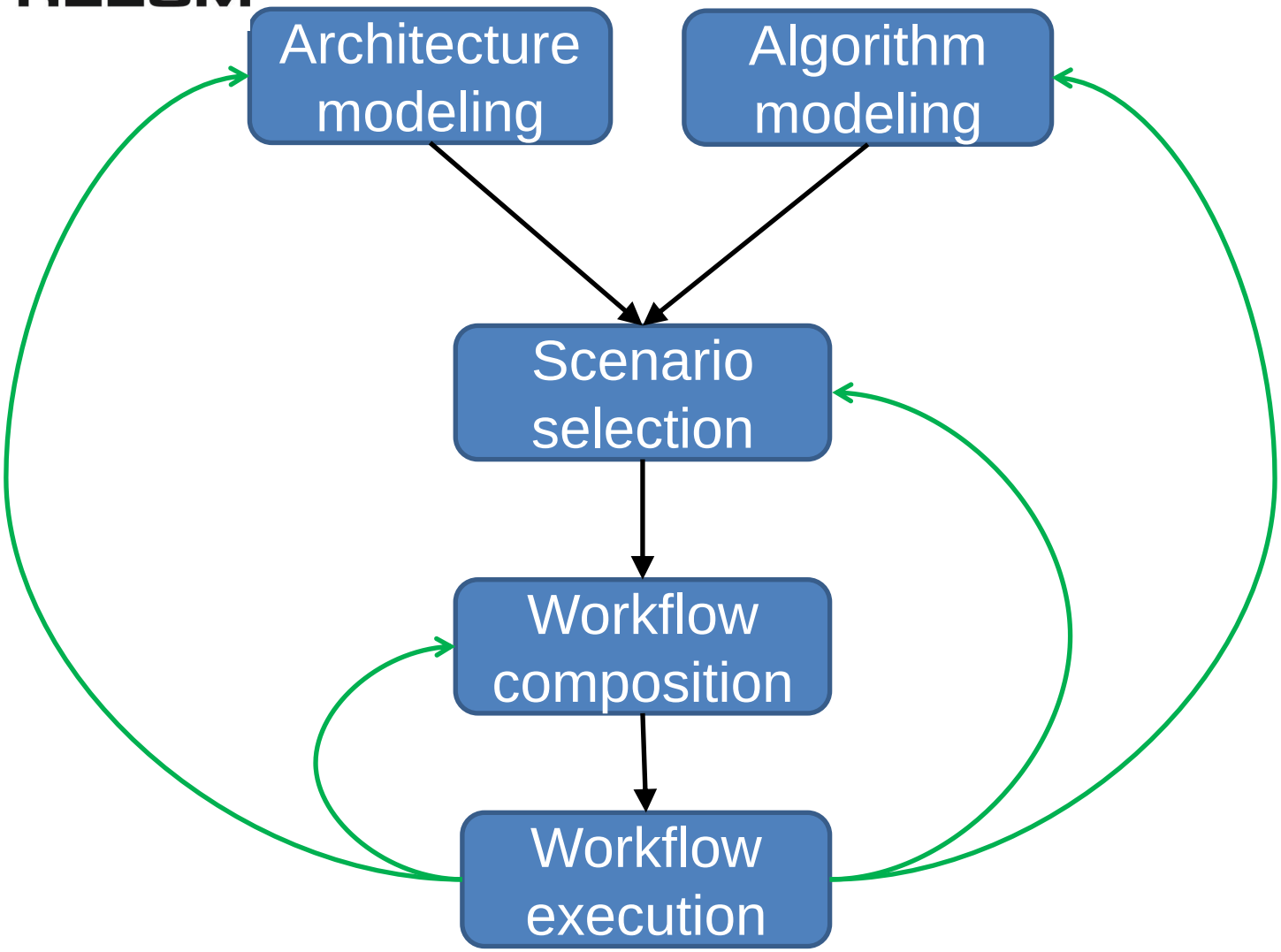
Stereo matching: disparity map

Architectures

Intel i7 quad-core

TI Shannon (C6678)

Rapid prototyping process



Algorithm modeling

PiSDF

Parameterized

Dynamism

interfaced

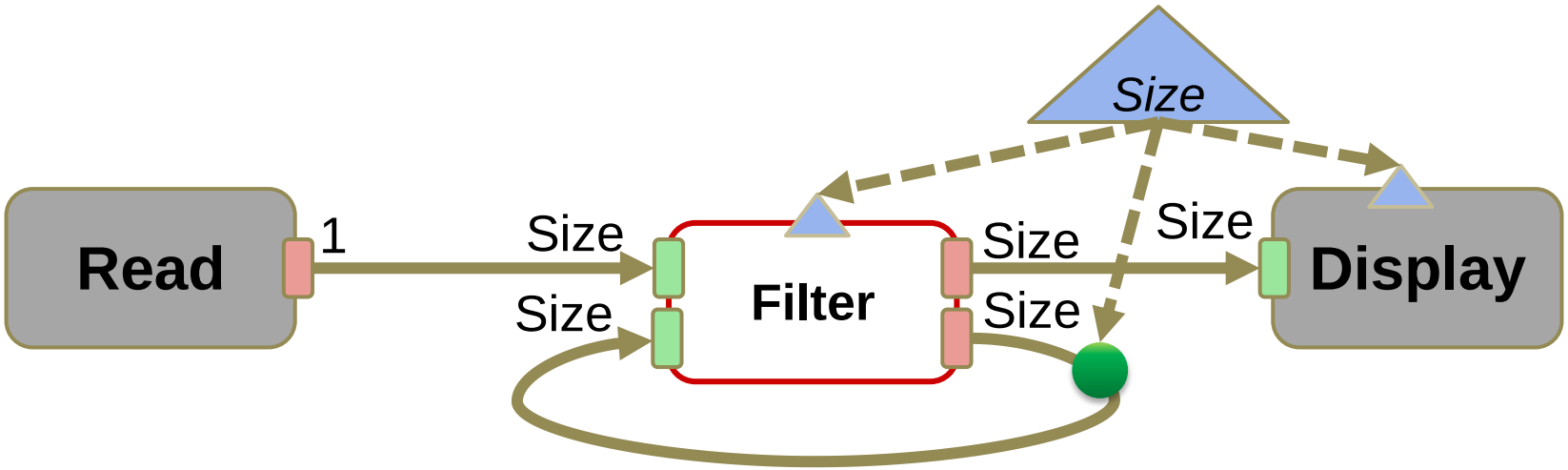
Hierarchy & Composition

Synchronous **D**ata-**F**low

Actors & Fifos

Sobel filter

Stereo matching

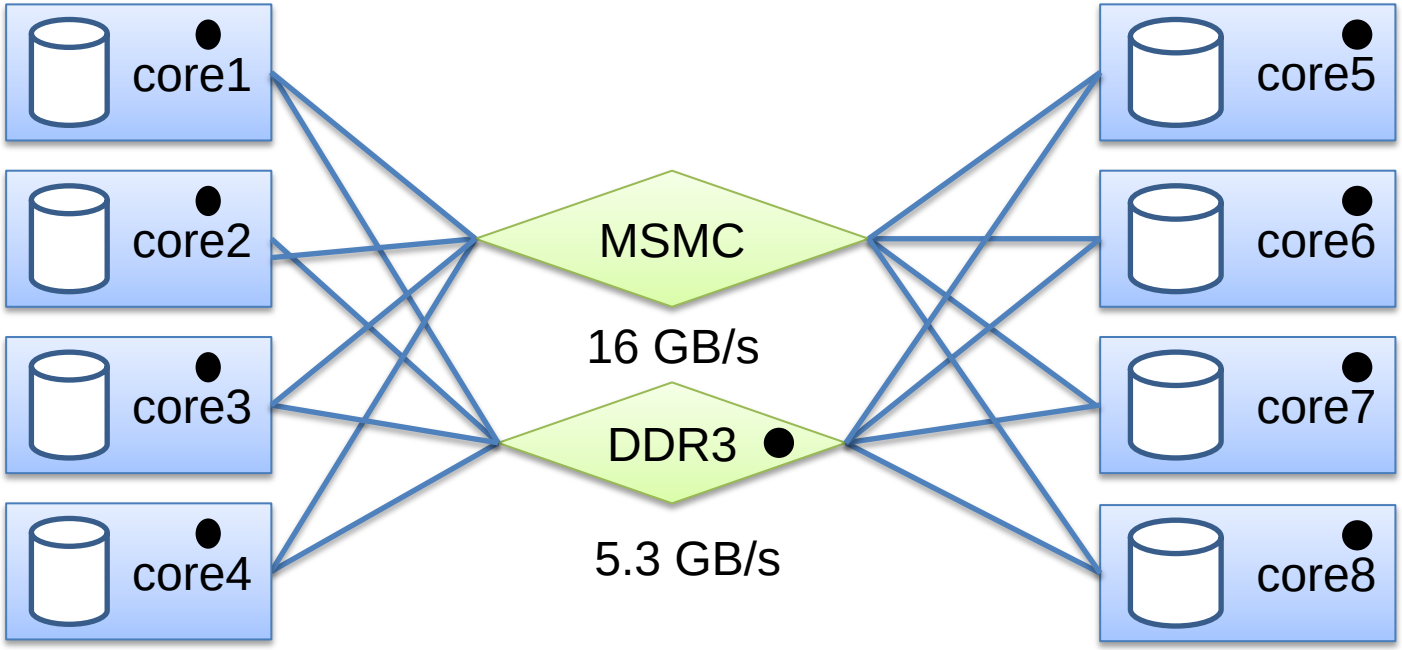


S-LAM

System-Level Architecture Model

- Processing elements
- Communication nodes

Intel i7 quad-core
TI Shannon (C6678)



Scenario selection

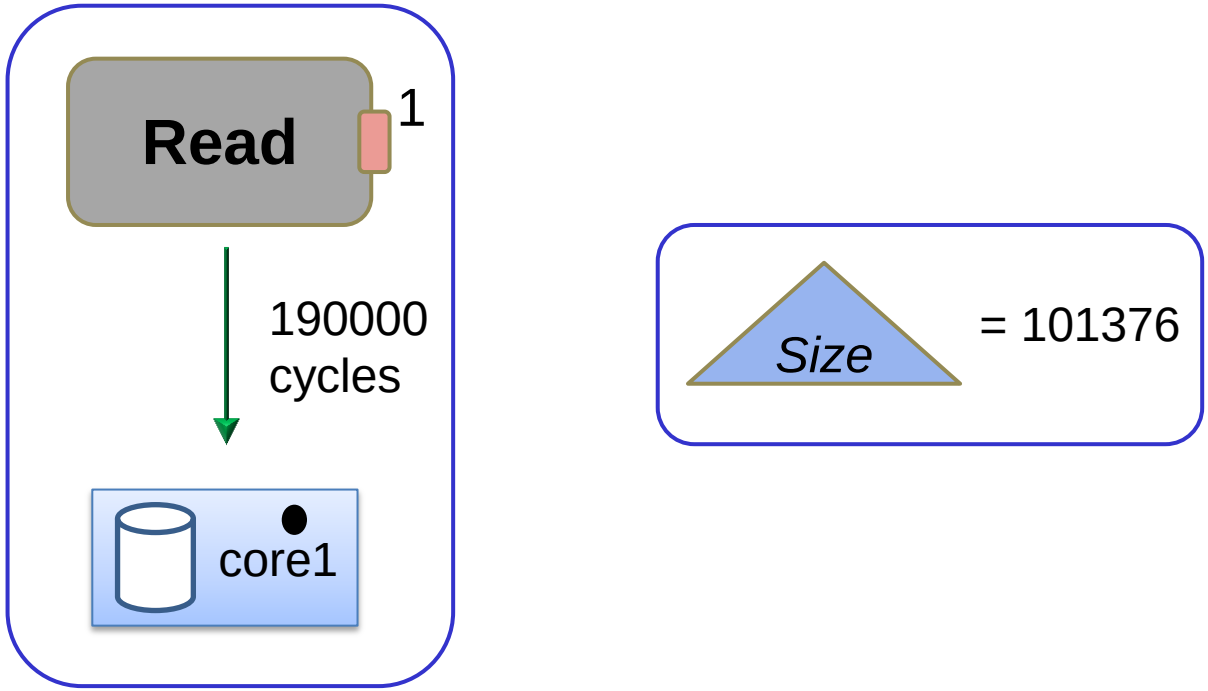
Link between **algorithm** & **architecture**

- Execution times
- Execution constraints
- Simulation information

Enables separation of concerns

Scenario example

Sobel filter on Intel i7 quad-core



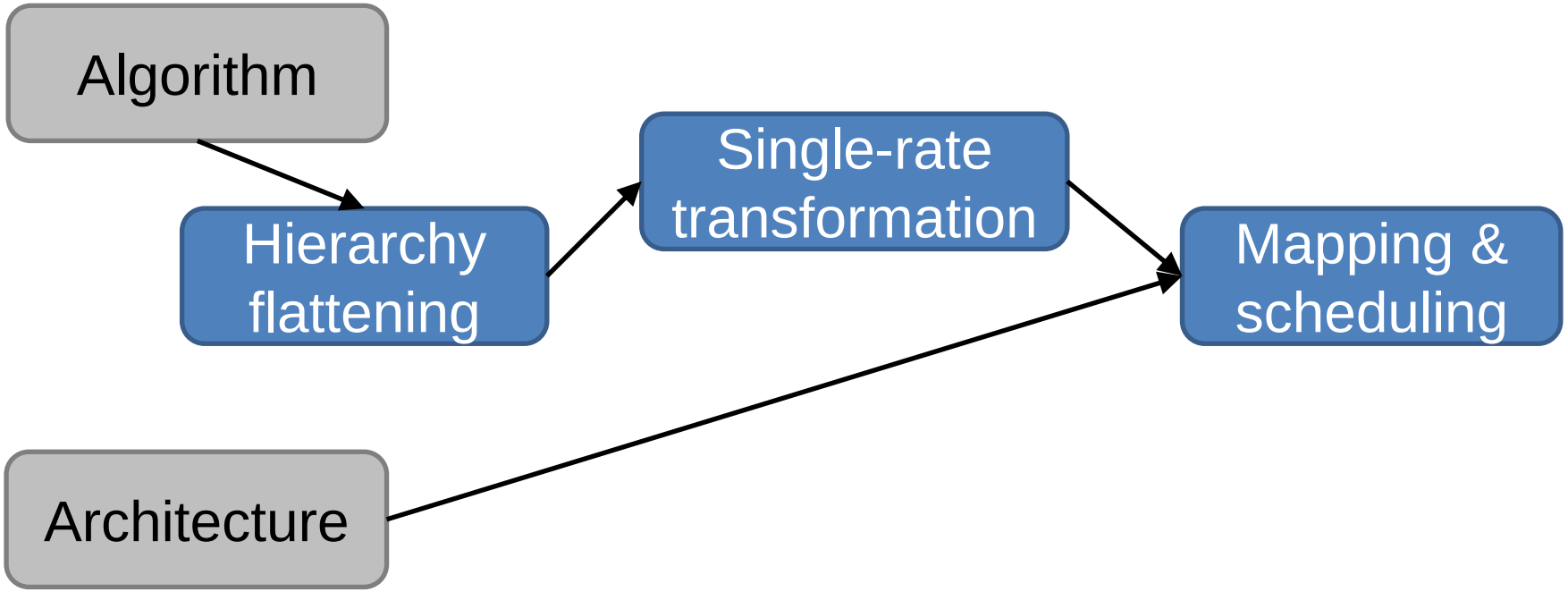
Rapid prototyping tasks

- Scheduling
- Code generation
- Memory optimization
- Vizualization tools
- …

Workflow examples

2 workflows

- Scheduling
- Scheduling + code generation



Small application on CPU

What about more realistic cases?

- Execution on DSP (C6678)
- Stereo matching algorithm

Rapid prototyping for multicore DSPs

- High-level modeling of **parallelism**
- **Automatic** mapping
- **Automatic** scheduling
- **Automatic** code generation
- **Advanced** memory optimization
- **Bridges** to UML MARTE, SDF3 & Orcc

PREESM is constantly improving

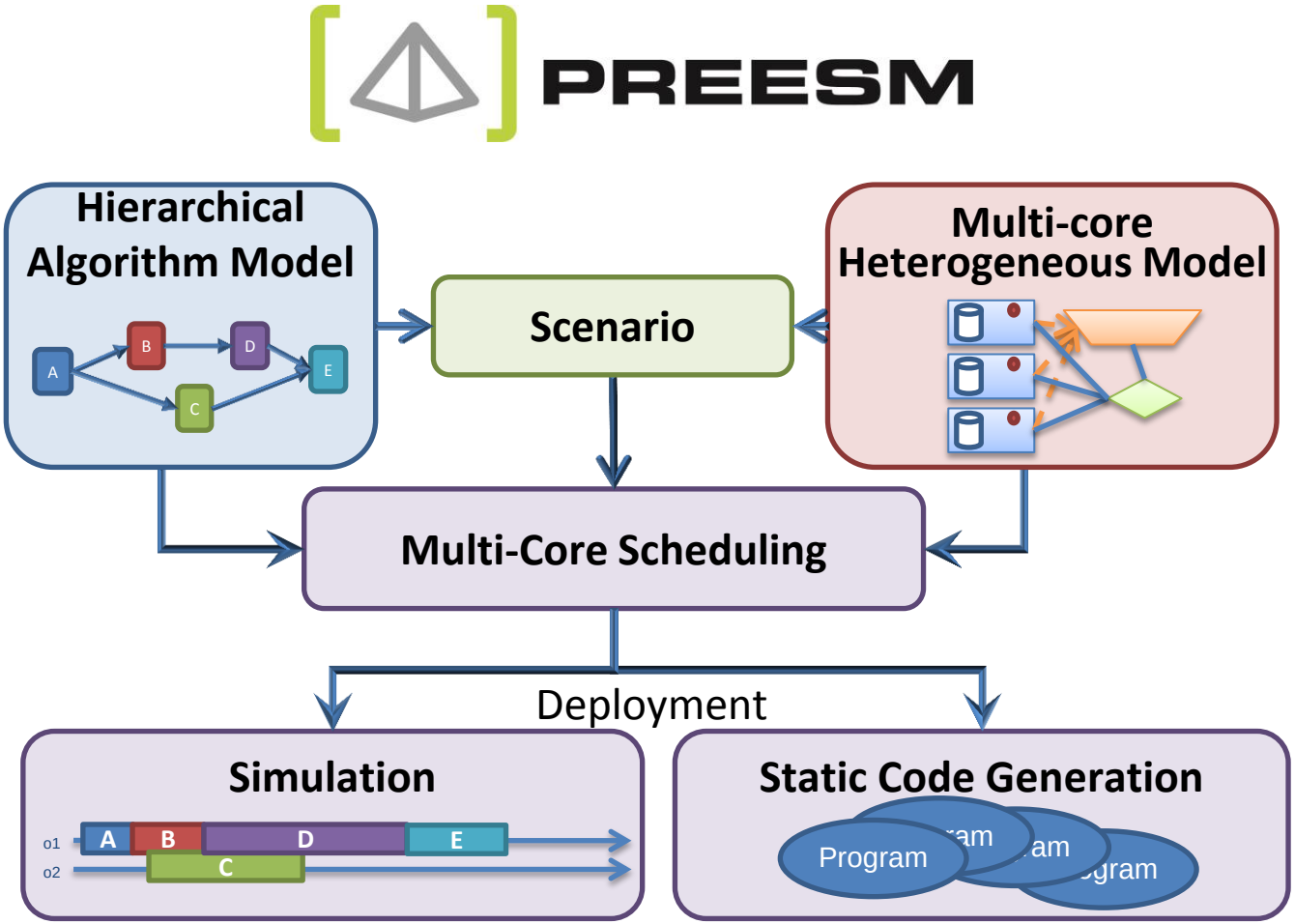
Research tool

- New models & features
- Regular enhancements

Incoming features

- Parameter-dependent timings
- Distributed memory management
- Bridge to DIF from Univ. of Maryland
- GUI enhancements (workflow scripts)
- …

Thank you for your attention



preesm.sourceforge.net/website/ Twitter: @PreesmProject