INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team DaRT

Dataparallelism for Real-Time

Lille - Nord Europe

THEME COM

Table of contents

DaRT is a common project with the University of Science and Technologies of Lille (USTL), via the Laboratory of Fundamental Computer Science of Lille (LIFL, associated to the CNRS as UMR 8022).

1. Team

Research Scientist

Samy Meftali [On leave from USTL (Associate professor)]
Abdoulaye Gamatié [Research scientist, UMR CNRS 8022]

Faculty Member

Jean-Luc Dekeyser [Team Leader, Professor, USTL (partly on leave at INRIA), HdR]
Pierre Boulet [Vice-Team Leader, Professor, USTL, HdR]
Cédric Dumoulin [Associate professor, USTL]
Anne Etien [Associate professor, USTL]
Frédéric Guyomarc'h [Associate professor, USTL (on leave at INRIA until august 2008)]
Philippe Marquet [Associate professor, USTL]
Smaïl Niar [Professor, UVHC, HdR]

Technical Staff

Thomas Legrand [Software Engineer, INRIA]
Alexis Muller [Expert Engineer, INRIA]
Emmanuel Leguy [Research Engineer, USTL]
Abdellatif Tinzefte [Expert Engineer, USTL]

PhD Student

Adolf Abdallah [French Minister grant]
Vincent Aranega [French Minister grant]
Mouna Baklouti [co-advising USTL-ENIS Sfax]
Hajer Chtioui [co-advising UVHC-ENIS Sfax]
Calin Glitia [INRIA Futurs grant]
Souha Kamoun [French Minister grant]
Imran Rafiq Quadri [HEC Pakistan and INRIA grant]
Wendell Rodrigues [VALEO, GPU Tech and Regional grant]
Safouan Taha [co-advising USTL-CEA, CEA grant]
Julien Taillard [VALEO and Regional grant]
Nicolas Wojcik [INRIA grant]
Huafeng Yu [INRIA Futurs and Regional grant]

Post-Doctoral Fellow

Rabie Ben Atitallah [Post-doctoral fellow, INRIA then UVHC]
César Moura [Post-doctoral fellow, INRIA, until september 2008]
Jean-Marie Mottu [Post-doctoral fellow, INRIA, from december 2008]
Sébastien Le Beux [Post-doctoral fellow, INRIA, until august 2008]

Visiting Scientist

Abou El Hassan Benyamina [Invited Assistant Professor at USTL, Associate Professor, University of Oran, Algeria]
Mohamed Abid [Professor, ENIS Sfax,Tunisia]
Vlad Rusu [Research scientist, INRIA Rennes-Bretagne Atlantique, HdR]

Administrative Assistant

Karine Lewandowski [INRIA]

2. Overall Objectives

2.1. Introduction

For the last few years we have seen the beginning of the "design gap". This gap is caused by the exponential growth of the integration rate of transistors on chips and the comparatively slower growth of the productivity of the integrated circuits designers. It is now impractical to fill a chip with custom designed logic. One has to reuse existing design parts or fill the chip area with memory (a good example of this evolution is the multi-core processors that include several existing processing cores instead of complexifying a single core). This evolution is clearly attested by the International Technology Roadmap on semiconductors.

In the same time, the computing power requirements of intensive signal processing applications such as video processing, voice recognition, telecommunications, radar or sonar are steadily increasing (several hundreds of Gops for low power embedded systems in a few years). If the design productivity does not increase dramatically, the limiting factor of the growth of the semiconductor industry will not be the physical limitations due to the thinness of the fabrication process but the economy! Indeed we ask to the system design teams to build more complex systems faster, cheaper, bug free and decreasing the power consumption...

We propose in the DaRT project to contribute to the improvement of the productivity of the electronic embedded system design teams. We structure our approach around a few key ideas:

- Promote the use of *parallelism* to help reduce the power consumption while improving the performance.
- *Use of MDE (Model Driven Engineering)* By separating the concerns in different models allowing reuse of these models and to keep them human readable.
- Propose an environment starting at the highest level of abstraction, namely the *system modeling* level.
- *Automate code production* by the use of (semi)-automatic *model transformations* to build correct by construction code.
- Develop *simulation techniques* at precise abstraction levels (functional, transactional or register transfer levels) to check the design the soonest.
- Prototype the resulting embedded systems of FPGA
- Promote *strong semantics* in the application model to allow verification, non ambiguous design and automatic code generation.
- Focus on a *limited application domain*, intensive signal processing applications. This restriction allows us to push our developments further without having to deal with the wide variety of applications.

All these ideas are implemented into a prototype co-design environment based on a model driven engineering approach, Gaspard. This open source platform is our test bench and is freely available. To help the designer, such an environment should evaluate several architectural solutions as well as several application specifications with regard to their performance and cost. We are able to estimate metrics from SystemC simulations and the refactoring algorithm defined for the transformation of loops to particular multiprocessors are the first steps for exploration. Automatic exploration system based on multi-objective methods has to transform the SoC description (size, network, memory, association). The space of solutions is huge and a fast simulation in SystemC at a high abstraction level is a good opportunity to reduce the space in a short delay. After that, a precise simulation at low level in SystemC or even in VHDL can start to refine the solution.
The main technologies we promote are UML 2 [52], MDE [82] and Eclipse EMF [66] for the modeling and model handling; Array-OL [60], [61], [58], [57] and synchronous languages [55] as computation models with strong semantics for verification; SystemC [80] for the simulation; OpenMP for the shared memory parallel execution; VHDL for the synthesis; and Java to code our prototypes.

2.2. Highlights of the year

The Gaspard2 platform is now compliant to the OMG standard MARTE profile, which is dedicated to the design of embedded and real-time systems. It therefore becomes the first tool compiling high-performance system on chip models, fully specified in MARTE, into SystemC, VHDL, OpenMP Fortran and C, and synchronous data flow languages. A few tutorials and demonstrations about the new platform version have been given by the DaRT project-team members at the following international scientific events: ICESCA'08, FDL'08 and IDT'08.

3. Scientific Foundations

3.1. Introduction

ISP Intensive Signal Processing

SoC System-on-Chip

The main research topic of the DaRT team-project concerns the hardware/software codesign of embedded systems with high performance processing units like DSP or SIMD processors. A special focus is put on multi processor architectures on a single chip (System-on-Chip). The contribution of DaRT is organized around the following items:

Co-modeling for High Performance SoC design: We define our own metamodels to specify application, architecture, and (software hardware) association. These metamodels present new characteristics as high level data parallel constructions, iterative dependency expression, data flow and control flow mixing, hierarchical and repetitive application and architecture models. All these metamodels are implemented with respect to the MARTE standard profile of the OMG group, which is dedicated to the modeling of embedded and real-time systems.

Model-based optimization and compilation techniques: We develop automatic transformations of data parallel constructions. They are used to map and to schedule an application on a particular architecture. This architecture is by nature heterogeneous and appropriate techniques used in the high performance community can be adapted. New heuristics to minimize the power consumption are developed. This new objective implies to specify multi criteria optimization techniques to achieve the mapping and the scheduling.

SoC simulation, verification and synthesis: We develop a SystemC based simulation environment at different abstraction levels for accurate performance estimation and for fast simulation. To address an architecture and the applications mapped on it, we simulate in SystemC at different abstraction levels the result of the SoC design. This simulation allows us to verify the adequacy of the mapping and the schedule, e.g., communication delay, load balancing, memory allocation. We also support IP (Intellectual Property) integration with different levels of specification. On the other hand, we use formal verification techniques in order to ensure the correctness of designed systems by particularly considering the synchronous approach. Finally, we transform MARTE models of data intensive algorithms in VHDL, in order to synthesize a hardware implementation.

3.2. Co-modeling for HP-SoC design

Keywords: *MDE*, *Metamodel*, *Model*, *Modeling*, *Transformation*, *UML.*

The main research objective is to build a set of metamodels (application, hardware architecture, association, deployment and platform specific metamodels) to support a design flow for SoC design. We use a MDE (Model Driven Engineering) based approach.

3.2.1. Foundations

3.2.1.1. System-on-Chip Design

SoC (System-on-Chip) can be considered as a particular case of embedded systems. SoC design covers a lot of different viewpoints including the application modeling by the aggregation of functional components, the assembly of existing physical components, the verification and the simulation of the modeled system, and the synthesis of a complete end-product integrated into a single chip.

The model driven engineering is appropriate to deal with the multiple abstraction levels. Indeed, a model allows several viewpoints on information defined only once and the links or transformation rules between the abstraction levels permit the re-use of the concepts for a different purpose.

3.2.1.2. Model-driven engineering

Model Driven Engineering (MDE) [82] is now recognized as a good approach for dealing with System on Chip design issues such as the quick evolution of the architectures or always growing complexity. MDE relies on the model paradigm where a model represents an abstract view of the reality. The abstraction mechanism avoids dealing with details and eases reusability.

A common MDE development process is to start from a high level of abstraction and to go to a targeted model by flowing through intermediate levels of abstraction. Usually, the high level models contain only domain specific concepts, while technological concepts are introduced smoothly in the intermediate levels. The targeted levels are used for different purposes: code generation, simulation, verification, or as inputs to produce other models, etc. The clear separation between the high level models and the technological models makes it easy to switch to a new technology while re-using the previous high level designs. Transformations allow to go from one model at a given abstraction level to another model at another level, and to keep the different models synchronized

In an MDE approach, a SoC designer can use the same language to design application and architecture. Indeed, MDE is based on proved standards: UML 2 [51] for modeling, the MOF (Meta Object Facilities [78]) for metamodel expression and QVT [79] for transformation specifications. Some profiles, i.e. UML extensions, have been defined in order to express the specificities of a particular domain. In the context of embedded system, the MARTE profile in which we contribute follows the OMG standardization process.

3.2.1.3. Models of computation

We briefly present our main models of computation that consist of the Array-OL language and the synchronous model. The former allows us to express the parallelism in applications while the latter favors the formal validation of the design.

Array-OL. The Array-OL language [60], [61], [58], [57] is a mixed graphical-textual specification language dedicated to express multidimensional intensive signal processing applications. It focuses on expressing all the potential parallelism in the applications by providing concepts to express data-parallel access in multidimensional arrays by regular tilings. It is a single assignment first-order functional language whose data structures are multidimensional arrays with potentially cyclic access.

The synchronous model. The synchronous approach [55] proposes formal concepts that favor the trusted design of embedded real-time systems. Its basic assumption is that computation and communication are instantaneous (referred to as “synchrony hypothesis”). The execution of a system is seen through the chronology and simultaneity of observed events. This is a main difference from visions where the system execution is rather considered under its chronometric aspect (i.e., duration has a significant role). There are different synchronous languages with strong mathematical foundations. These languages are associated with tool-sets that have been successfully used in several critical domains, e.g. avionics, nuclear power plants.

In the context of the DaRT project, we consider declarative languages such as Lustre [59] and Signal [74] to model various refinements of Array-OL descriptions in order to deal with the control aspect as well as the temporal aspect present in target applications. The first aspect is typically addressed by using concepts such as mode automata, which are proposed as an extension mechanism in synchronous declarative languages. The second aspect is studied by considering temporal projections of array dimensions in synchronous languages

based on clock notion. The resulting synchronous models are analyzable using the formal techniques and tools provided by the synchronous technology.

3.2.2. Contributions of the team

Our proposal is partially based upon the concepts of the "Y-chart" [67]. The MDE contributes to express the model transformations which correspond to successive refinements between the abstraction levels.

Metamodeling brings a set of tools which enable us to specify our application and hardware architecture models using UML tools, to reuse functional and physical IPs, to ensure refinements between abstraction levels via mapping rules, to initiate interoperability between the different abstraction levels used in a same codesign, and to ensure the opening to other tools, like verification tools, thought the use of standards.

The application and the hardware architecture are modelled separately using similar concepts inspired by Array-OL to express the parallelism. The placement and scheduling of the application on the hardware architecture is then expressed in an association model.

All the previously defined models, application, architecture and association, are platform independent. No component is associated with an execution, simulation or synthesis technology. Such an association targets a given technology (OpenMP, SystemC/PA, VHDL, Lustre, etc.). Once all the components are associated with some technology, the deployment is realized. This is done by the refinement of the association model to the deployed model first, and then to further abstraction levels (e.g. polyhedron, loop in figure 1).

The simulation results can lead to a refinement of the application, the hardware architecture, the association and the deployment models. We propose a methodology to work with these models. The stages of design are:

1. Separation of application and hardware architecture modeling.
2. Association with semi-automatic mapping and scheduling.
3. To achieve the deployment, selection of IPs for each element of application/architecture models.
4. Automatic generation of the various platform specific simulation or execution models.
5. Automatic simulation or execution code generation with calls to the IPs.
6. Refinement at the highest level taking account of the simulation results.

3.2.2.1. High-level modeling in Gaspard2

In Gaspard2, models are described by using the recent OMG standard MARTE profile combined with a few native UML concepts (see Figure 1). The former Gaspard2 profile is no longer used. Today, the Hardware Resource Model (HRM) concepts of MARTE enable to describe the hardware part of a system. The Repetitive Structure Modeling (RSM) concepts allow one to describe repetitive structures. Finally, the Generic Component Modeling (GCM) concepts are used as the base for component modeling.

The above concepts are expressive enough to permit the modeling of different aspects of an embedded system:

- functionality (or applicative part): the focus is mainly put on the expression of data dependencies between components in order to describe an algorithm. Here, the manipulated data are mainly multidimensional arrays. Furthermore, a form of reactive control can be described in modeled applications via the notion of execution modes. This last aspect is modeled with the help of some native UML notions in addition to MARTE.
- hardware architecture: similar mechanisms are also used here to describe regular architectures in a compact way. Regular parallel computation units are more and more present in embedded systems, especially in SoCs. HRM is fully used to model these concepts.
- association of functionality with hardware architecture: the main issues concern the allocation of the applicative part of a system onto the available computation resources, and the scheduling. Here also, the allocation model takes advantage of the repetitive and hierarchical representation offered by MARTE to enable the association at different granularity levels, in a factorized way.

In addition to the above usual design aspects, Gaspard2 also defines a notion of *deployment* specification (see Figure 1) in order to generate compilable code from a SoC model. The corresponding package defines concepts that (i) enable to describe the relation between a MARTE representation of an elementary component (a box with ports) to a text-based code (and Intellectual Property - IP, or a function with arguments), and (ii) allow one to inform the Gaspard transformations of specific behaviours of each component (such as average execution time, power consumption...) in order to generate a high abstraction level simulation in adequacy with the real system.

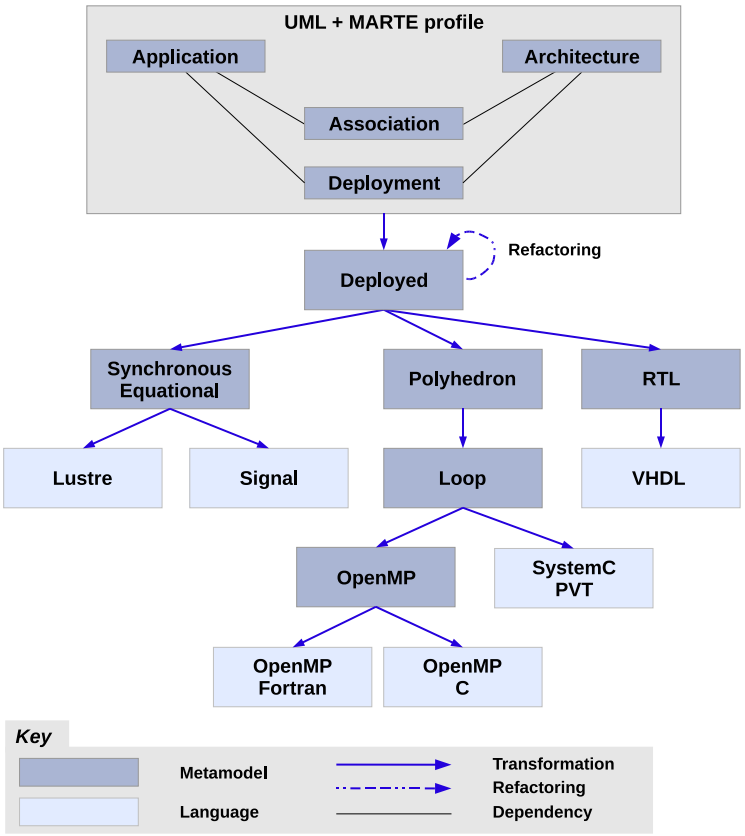

Figure 1. Overview of the design concepts.

3.2.2.2. Intermediate concept modeling

According to Figure 1, one can notice that Gaspard2 targets different technologies for various purposes: formal verification, high-performance computing, simulation and hardware synthesis. This is achieved via model transformations that relate intermediate representations towards the final target representations. Below are briefly presented these intermediate representations:

- A metamodel based on polyhedra (Polyhedron in Figure 1). It presents the association mechanism from the architecture viewpoint. Instead of having concepts to indicate the placement of tasks and data arrays as in the Deployed metamodel (*i.e.* allocation and distribution), in the Polyhedron metamodel, the data arrays are contained into memories and the tasks are linked to the processors. In order to faithfully represent the repetitions of these distributed elements, the *polyhedron* mathematical concept is used. A polyhedron is basically a set of linear equations and inequalities. In literature,

several works on parallel scheduling rely on them because they enable to determine an efficient execution of loop iterations on processors. A whole set of theories and *tools* are available to generate optimized code out of such a representation (our implementation uses the CLooG tool [53]).

- A metamodel based on loops (Loop in Figure 1). It is very close to the Polyhedron metamodel. The unique difference is the representation of the task repetition. Instead of using a polyhedron, the repetition is represented by a *LoopStatement*. It corresponds to the pseudo-code structure that for a given processor index, goes all over the repetition index of the associated tasks [54]. The SystemC code is directly generated from this metamodel whereas another intermediary metamodel is used to target OpenMP technologies.
- A metamodel for procedural language with OpenMP (OpenMP in Figure 1). It is inspired by the ANSI C and Fortran grammars and extended by OpenMP statements [81]. The aim of this metamodel is to use the same model to represent Fortran and C code. Thus, from an OpenMP model, it is possible to generate OpenMP/Fortran or OpenMP/C. The generated code includes parallelism directives and control loops to distribute task (IPs code) repetitions over processors [34].
- A RTL metamodel (RTL in Figure 1). It gathers the necessary concepts to describe hardware accelerators at the RTL (Register Transfer Level) level, which allows the hardware execution of applications. This metamodel introduces, *e.g.*, the notions of *clock* and *register* in order to manipulate some of the usual hardware design concepts. The RTL metamodel is independent from any Hardware Description Languages (HDL) such as VHDL [50] or Verilog [85] with the corresponding IPs. However, it is precise enough to enable the generation of synthesizable HDL code [48].
- Synchronous metamodel (Synchronous Equational in Figure 1). It allows a description of applications at the functional level. It relies on the concepts of *Signal*, *Equation* and *Node*. A *Signal* corresponds to a variable. An *Equation* is a relation defining the outputs in term of a function of the inputs. A *node* is a system of equations over signals that specifies relations between values and clocks of the signals. From this metamodel, either Lustre or Signal synchronous language an be generated with the good IPs, allowing one to check functional properties of applications described with MARTE [44].

3.3. Model-based optimization and compilation techniques

Keywords: *Compilation*, *Data-parallelism*, *Heuristics*, *Mapping*, *Optimization*, *Power Consumption*, *Scheduling*.

3.3.1. Foundations

3.3.1.1. Optimization for parallelism

We study optimization techniques to produce "good" schedules and mappings of a given application onto a hardware SoC architecture. These heuristic techniques aim at fulfilling the requirements of the application, whether they be real time, memory usage or power consumption constraints. These techniques are thus multi-objective and target heterogeneous architectures.

We aim at taking advantage of the parallelism (both data-parallelism and task parallelism) expressed in the application models in order to build efficient heuristics.

Our application model has some good properties that can be exploited by the compiler: it expresses all the potential parallelism of the application, it is an expression of data dependencies –so no dependence analysis is needed–, it is in a single assignment form and unifies the temporal and spatial dimensions of the arrays. This gives to the optimizing compiler all the information it needs and in a readily usable form.

3.3.1.2. Transformation and traceability

Model to model transformations are at the heart of the MDE approach. Anyone wishing to use MDE in its projects is sooner or later facing the question: how to perform the model transformations? The standardization process of Query View Transformation [79] was the opportunity for the development of transformation engine as Viatra, Moflon or Sitra. However, since the standard has been published, only few of investigating tools, such as ATL[1] (a transformation dedicated tool) or Kermeta [2] (a generalist tool with facilities to manipulate models)

[1] http://www.eclipse.org/m2m/atl

are powerful enough to execute large and complex transformations such as in the Gaspard2 framework. None of these engine is fully compliant with the QVT standard. To solve this issue, new engine relying on a subset of the standard recently emerged such as QVTO [3] and smartQVT. These engine implement the QVT Operational language .

Traceability may be used for different purposes such as understanding, capturing, tracking and verification on software artifacts during the development life cycle [68]. MDE has as main principle that everything is a model, so trace information is mainly stored as models. Solutions are proposed to keep the trace information in the initials models source or target [86]. The major drawbacks of this solution are that it pollutes the models with additional information and it requires adaptation of the metamodels in order to take into account traceability. Using a separate trace model with a specific semantics has the advantage of keeping trace information independent of initial models [70].

3.3.2. Contributions of the team

We focus on two particular subjects in the optimization field: data-parallelism efficient utilization and multi-objective hierarchical heuristics.

3.3.2.1. Data-parallel code transformations

We have studied Array-OL to Array-OL code transformations [58], [83], [63], [62], [64] [69]. Array-OL allows a powerful expression of the data access patterns in such applications and a complete parallelism expression. It is at the heart of our metamodel of application, hardware architecture and association.

The code transformations that have been proposed are related to loop fusion, loop distribution or tiling but they take into account the particularities of the application domain such as the presence of modulo operators to deal with cyclic frequency domains or cyclic space dimensions (as hydrophones around a submarine for example).

We pursue the study of such transformations with three objectives:

- Propose utilization strategies of such transformations in order to optimize some criteria such as memory usage, minimization of redundant computations or adaptation to a target hardware architecture.
- Stretch their application domain to our more general application model (instead of just Array-OL).

3.3.2.2. Multi-objective hierarchical scheduling heuristics

When dealing with complex heterogeneous hardware architectures, the scheduling heuristics usually take a task dependence graph as input. Both our application and hardware architecture models are hierarchical and allow repetitive expressions. We propose a Globally Irregular, Locally Regular (GILR) combination of heuristics to allow to take advantage of both task and data parallelism [75] and have started evaluating multi-objective evolutionary meta-heuristics in this context. These evolutionary meta-heuristics deal with the irregular (task parallelism) part of the design [56] while we have proposed a heuristic to deal with the regular part (data parallelism) [76].

Furthermore, local optimizations (contained inside a hierarchical level) decrease the communication overhead and allow for a more efficient usage of the memory hierarchy. We aim at combining the data-parallel code transformations presented before and the GILR heuristics in order to deal efficiently with the data-parallelism of the application by using repetitive parts of the hardware architecture.

3.3.2.3. Transformation techniques

In 2006,in front of the absence of transformation tool that supports external black box calls (e.g. native function calls), recursive rule call, rule inheritance and integration of imperative code, we developed, our own tool MoMoTE. MoMoTE is a Java framework defined to enhance model to model transformations. An other tool MoCode has also been defined, for the model to text transformation (i.e. the code generation). It relies on JET

[2] http://www.kermeta.org
[3] http://www.eclipse.org/m2m/qvto/doc

principles. It takes in charge the calls of the right templates as well as the link between the JET and the Ecore worlds.

We also focus on another difficulty encountered in the design of the Gaspard2 supply chain: document and specify transformations. Based on the principle that a model transformation specification is it-self a model, we proposed TrML, a UML profile dedicated to transformation. TrML is a graphical representation of the transformation to increase documentation, exchange around transformations. This notation is independent from any existing transformation engines and focus on what should be transformed, rather than how to implement the transformation.

3.4. HP-SoC simulation, verification and synthesis

Keywords: *SystemC*, *TLM*.

Many simulations at different levels of abstraction are the key of an efficient design of embedded systems. The different levels include a functional (and possibly distributed) validation of the application, a functional validation of the application and an architecture co-model, and a validation of a heterogeneous specification of an embedded system (a specification integrating modules provided at different abstraction levels).

SoCs are more and more complex and integrate software parts as well as specific hardware parts (IPs, Intellectual Properties). Generally before obtaining a SoC on silicium, a system is specified at several abstraction levels. Any system design flow consists in refining, more or less automatically, each model to obtain another, starting from a functional model to reach a Register Transfer Level model. One of the biggest design challenges is the development of a strong, low cost and fast simulation tool for system verification and simulation.

The DaRT project is concerned by the simulation at different levels of abstraction of the application/architecture co-model and of the mapping/schedule produced by the optimization phase.

3.4.1. Foundations

3.4.1.1. Abstraction levels

Currently, Transaction Level Modeling, TLM, is being used in the industry to solve a variety of practical problems during the design, development and deployment of electronic systems.

The TLM 2.0 standard appeared during the very few last years. It consists in describing systems according to the specifications of the TLM abstraction levels. At these levels, function calls simulate the behavior of the communications between architecture components.

Nowadays, this modelling style is widely used for verification and it is starting to be used for design at many major electronic companies. Recently, many actions and challenges have been started in order to help to proliferate TLM. Thus, several teams are working to furnish to designers standard TLM APIs and guidelines, TLM platform IP and tools supports. SystemC is the first system description language adopting TLM specifications. Thus, several standardization APIs have been proposed to the OSCI by all the major EDA and IP vendors. This standardization effort is being generalized now by the OSCI / OCP-IP TLM standardization alliance, to build on a common TLM API foundation. One of the most important TLM API proposals is the one from Cadence, distributed to OSCI and OCP-IP. It is intended as common foundation for OSCI and OCP-IP allowing protocol-specific APIs (e.g. AMBA, OCP) and describing a wide range of abstraction levels for fast and efficient simulations.

3.4.1.2. Dynamic reconfiguration - FPGA

Current FPGAs support the notion of Partial Dynamic Reconfiguration which allows part of the FPGA to be reconfigured on the fly hence introducing the idea of virtual hardware. Partial Reconfiguration allows swapping of tasks (mutually exclusive)depending upon user requirements and Quality of service needs. Using such a technology permits to optimize energy consumption and the area in the system. It allows also to have very flexible systems, adaptable for large application classes.

3.4.1.3. Verification

Our privileged basis for verification is the reactive synchronous domain. Over the last two decades several formal verification technologies have been provided by a very active research community in this domain. Among the available tools, we can mention efficient compilers that act more than usual compilers in that they address more static analysis issues. There are also various model-checkers that use both symbolic representations and non symbolic ones. Some of these model-checkers offer facilities that go beyond verification by enabling the synthesis of (discrete) controllers. Finally, these synchronous technologies give the opportunity in some cases to perform a functional simulation of the described systems.

3.4.2. Contribution of the team

The results of DaRT simulation package concerns mainly the PVT and the CABA levels. We also propose techniques to interact with IPs specified at other level of abstraction (mainly RTL).

3.4.2.1. Co-simulation in SystemC

From the association model, the Gaspard2 environment is able to automatically produce SystemC simulation code. The MDE techniques offer the transformation of the association model to the SystemC Gaspard2 model. During this transformation the data parallel components are unrolled and the data dependencies between elementary tasks become synchronisation primitive calls.

The SoC architecture is directly produced from the architecture model. A module in SystemC simulates the behaviour of tasks mapped to a particular processor. Other modules contain the data parallel structures and are able to answer to any read/write requests. The communications between tasks and between tasks and memories are simulated via communication modules in SystemC. These last modules produce interesting results concerning the simultaneous network conflicts and the capacity of this network for this application. The Loop metamodel allows automatic SystemC code generation. The association model is first transformed into a model of this Loop metamodel and this model is then automatically transformed into SystemC code. This development is integrated in the Gaspard2 prototype and uses the MoMoTE tool (see the software section). The produced simulation code is based on SystemC IPs assembling. These IPs are available in the Gaspard library in both TLM and CABA levels. They represent all the usual architecture components such as processors (ARM, MIPS, ..etc), memories, caches, buses, NoCs, etc.

3.4.2.2. TLM: Transactional level modelling

Due to all TLM's benefits, we defined a TLM metamodel as a top level point for automatic transformations to both simulation and synthesis platforms. Our TLM metamodel contains the main concepts needed for verification and design following the Cadence API proposal. But, as we are targeting multi-language simulation platforms, the meta model is completely independent from the SystemC syntax. It is composed mainly by two parts: architecture and application. This clear separation between SW and HW parts permits easy extensions and updates of the meta model.

- The architecture part contains all necessary concepts to describe HW elements of systems at TLM levels. The SW part is mainly composed of computation tasks. They should be hierarchical and repetitive. A set of parameters could be attached to each task in order to specify the scheduling dependently of the used computation model.
- Thus this metamodel keeps hierarchies and repetitions of both the application and the architecture. This permits to still benefit from the data parallelism as far as possible in the design (simulation and synthesis flow). In fact, the designer can choose to eliminate hierarchies when transforming the TLM model into a simulation model, and to keep it when transforming into a synthesis model.

In order to keep our design flow coherent, we choose to use two significant simulation levels. Each of them has special advantages.

The main objectives of the PVT level are fast verification of system functionalities and monitoring of the contentions in the interconnection network. Complementary to this level, the CABA level is used to accurately estimate the execution time and power consumption. At the PVT level, details related to the computation and communication resources are omitted. The software application is executed by an instruction-accurate Instruction Set Simulator. Transactions are performed through channels instead of signals. At the CABA level, hardware components are implemented at the cycle accurate level for both processing and communication parts. Communication protocol and arbitration strategy are specified as well. Simulation at the PVT level permits a rapid exploration of a large solution space by eliminating non interesting regions from the DSE process. The solutions selected at this level are then forwarded to a new exploration at the CABA level. At each level, the exploration is based on developed performance and power estimation tools. Code generation at both of those levels needs parameter specifications for execution time, power estimation, and platform configurations. These parameters are specified at the deployment phase.

3.4.2.3. Verification

Guaranteeing the correctness of systems is a highly important issue in the Gaspard2 design methodology. This is required at least for their validation. In order to provide the designer with the required means to cope with validation, we propose to bridge the gap between the Gaspard2 design approach and validation techniques for SoCs by using the synchronous approach and test-based techniques.

We have already defined a synchronous dataflow equational model of Gaspard2 specification concepts. The resulting model is then usable to address various correctness issues: causality analysis that enables to detect erroneous data dependencies (i.e. those which lead to cycles) in specifications, clock synchronizability analysis when such a system model is to be considered on a deployment platform, etc. This analysis relies on formal tools.

4. Application Domains

4.1. Intensive signal processing

Keywords: *multimedia*, *telecommunications*.

The DaRT project-team aims at improving the design of embedded systems with a strong focus on intensive signal processing applications.

This application domain is the most intensive part of signal processing, usually composed of systematic signal processing followed by intensive data processing. The systematic signal processing mainly consists of a chain of filters and regular processing applied on the input signals independently of the signal values. It results in a characterization of the input signals with values of interest. The intensive data processing applies irregular computations on these values of interest. Those computations may depend on the signal values.

Below are three application examples from our industrial partners.

Software Radio Receiver. This application is structured in a front end systematic signal processing including signal digitalizing, channel selection, and application of filters to eliminate interferences. These first data are decoded in a second and more irregular phase (synchronization, signal demodulation...).

Sonar Beam Forming. A classical sonar chain consists in a first and systematic step followed by a more general data processing. The first step provides frequency and location correlations (so called *beam*) from a continuous flow of data delivered by the hydrophones (microphones disposed around a submarine). It is based on signal elementary transformations: FFT (Fast Fourier Transformation) and discrete integration. The second step analyses a given set of beams and their history to identify temporal correlation and association to signal sources.

Video Encoder/Decoder. A video encoder works in a two-steps approach. The first part (from preprocessing to wavelet/cosine decomposition) is systematic. The second part of the encoder includes irregular processing (quantification, coding stages, motion detection). The decoder works the other way around: a first irregular phase is followed by a systematic phase. Recently we have used the H263 protocol.

4.2. Transport

Our contribution to the safety in transport applications were twofold. In the context of the ModEasy Interreg project we have studied anti-collision radars for cars and in the context of the I-Trans competitiveness pole we collaborated with the INRETS on the model driven test of the ERTMS European railway signalization standard. Today we continue to develop signal and image processing for transportation in the MED 3+3 project.

The automotive industry has specific problems, particularly due to increased safety requirements and legal framework. Some of these electronic systems have the potential to endanger the safety of vehicle occupants or other road users should an error or a mis-operation occur.

In the ModEasy Interreg project we modelled a cruise control connected to the satellite positioning system, GPS. From a UML specification and using classical verification and model checking techniques, we assured the correct behaviour of the system. The codes are generated using FPGA devices. Moreover, as ModEasy is based on metamodels and transformations between metamodels, new algorithms or new FPGAs can rapidly be integrated in the system by the re-use of existing functional blocks.

Today form the Gaspard2 environment , we propose to model Network on Chip dedicated to Image processing. Using transformation chain, we target FPGA connected to intelligent camera. The embedded characteristics of this system can benefit of the SystemC simulation before synthesis to find the good number of processors on the NoC.

4.3. High-performance computing

Using the OpenMP/Fortran code generation chain, we have experimented the generated code in a typical operation in the scientific field: the matrix multiplication. We have compared generated code with optimized BLAS library function. Different algorithms have been generated: row-column multiplication, multiplication by block, multiplication by block using optimized BLAS function for the sequential part. Those algorithms have been compared with the sequential BLAS function and the parallel BLAS function. The results [84] show that the way to use Gaspard2 in the High Performance Computing field is to entrust Gaspard2 to manage parallelism and to use optimized function for the sequential part.

5. Software

5.1. Gaspard2

Keywords: *Eclipse*, *IDE*, *SoC Design*, *Visual Design*.

Participant: Pierre Boulet [contact person].

Gaspard2 is an Integrated Development Environment (IDE) for SoC visual co-modeling. It allows or will allow modeling, simulation, testing and code generation of SoC applications and hardware architectures.

Its purpose is to provide one single environment for all the SoC development processes:

- High level modeling of applications and hardware architectures
- Application and hardware architecture association
- Application refactoring
- Deployment specification
- Model to model transformation (to automatically produce models for several target platforms)
- Code generation
- Simulation
- Reification of any stages of the development

The Gaspard2 tool is based on the Eclipse [65] IDE. A set of plugins provides the different functionalities. Gaspard2 is capable to chain transformations written with the MoMoTE plugin and, at the end, call a code generator written with the MoCodE plugin. Each transformations chain is described in a chain model conform to a chain metamodel. An important part of the core of Gaspard2 is an engine that executes models transformations chains.

Application, hardware architecture, association, deployment and technology models are specified and manipulated by the developer through UML diagrams, and saved by the UML tool in an XMI file format. Gaspard2 manipulates these models through repositories (Java interfaces and implementations) automatically generated thanks to the Ecore specification.

5.2. Papyrus

Keywords: *Eclipse*, *MDE*, *Papyrus*, *Profile*, *UML*, *Visual Tool*.

Participant: Cédric Dumoulin [contact person].

The Papyrus tool is an UML Development Environment fully compliant with the UML standard and providing all UML diagrams. It is in the process of becoming an Eclipse project (http://www.eclipse.org/modeling/mdt/?project=papyrus#papyrus). First release as an Eclipse project is planned for the end of 2008. The Papyrus Tool is developed under an Open source licence in collaboration with CEA, Atos, Obeo, Airbus, Prodevelop, Integranova.

6. New Results

6.1. Co-modeling for HP-SoC design

Participants: Adolf Abdallah, Rabie Ben Atitallah, Pierre Boulet, Jean-Luc Dekeyser, Cédric Dumoulin, Anne Etien, Abdoulaye Gamatié, Frédéric Guyomarc'h, Souha Kamoun, Thomas Legrand, Emmanuel Leguy, César Moura, Alexis Muller, Safouan Taha, Julien Taillard, Abdellatif Tinzefte, Huafeng Yu.

6.1.1. A requirement model for exploration

The optimization of resource usage is one of the basic decisions in the design of embedded systems. The decision depends essentially on the information related to performance, such as execution time, power consumption, cost, memory capacity and limitations on resource usage.

During the last year, this research activity has focused on exploring the simultaneous use of SysML and the MARTE UML profile to model applications, hardware architectures and mappings including design constraints, requirements and time aspects. We have proposed a method that provides models to help the designer to take decisions at higher abstraction levels. Thus, the designer can use these models to support the estimation of the final system physical characteristics directly from models. This method is a necessary step towards the design space exploration by offering the possibility to compare the performance of each design and to obtain architectures that satisfy a set of constraints while establishing enough flexibility to support a given set of software components. SysML and MARTE are complementary: SysML allows to model requirements in the early design phases while MARTE defines concepts to model the timing and the non functional aspects and is more suitable for the later design phases.

6.1.2. A MARTE-compliant metamodel for deployment

In order to generate an entire system from a high level specification, all implementation details of every elementary component have to be precised. Low level details are much better described by using usual programming languages instead of graphical UML models. In the SoC industry, individual components are called IP (*Intellectual Property*). They correspond to one specific implementation of a given functionality, either hardware or software. In SoC design, one functionality can be implemented in different ways. This is necessary for testing the system with different tools, or at different abstraction levels. For instance, different IPs can be provided for a given application component and may correspond to an optimized version for a specific processor or a version compliant with a given language.

Although the notion of deployment is present in UML, the SoC design has special needs, not fulfilled by this notion. Hence, Gaspard extends the MARTE profile to allow deploying elementary components with IPs. For this purpose, we have introduced the concept of VirtualIP to express the behavior of a given elementary component (either software or hardware), independently from the usage context. A VirtualIP is implemented by one or several IPs, each one being used to define a specific implementation at a given abstraction level and in a given language. Finally, the concept of CodeFile is used to specify, for a given IP, the file corresponding to the source code and its required compilation options. The used IP is selected by the SoC designer by linking it to the elementary component through the Implements dependency. Some IPs provided by the SoC industry can be parametrized. These parameters are specified using the Characteristic concept.

6.1.3. Template in models

We have proposed a mechanism defining parameterizable elements in the Gaspard's modeling process, so that Gaspard's elements may be only partially defined in a first step, leaving some "blanks" to be defined in a later time, but still in the modeling phase (in other words, before instantiating these elements). This would enable the creation of component libraries, increasing reuse.

Since UML tools are the front-end interface used in the Gaspard2 modeling process, it is important to provide users with ways of expressing parameterizable models. However, once a parameterizable model has been created using UML tools, the next step is to express the same model in the MARTE metamodel. Consequently, MARTE itself needs to offer support to parameterizable models and so on for the other metamodels. We have developed a generic approach that suggest how to add our generic metamodel on top of any metamodel and to consequently modify the transformation rules according to keep the parameters further in the chain or to instantiate them.

A hardware architecture can also benefit from parameterizable elements as seen with the specification of parameterizable NoCs. The current UML specifications were modified and extended to introduce the notion of component templates, nested component templates and an alternative template binding notation [30]. Finally a recursive approach related to template components was introduced. These concepts were then used to model the Delta network family of Multistage Interconnection Networks. This approach can be extended to model all families of Interconnection networks having aspects of repetitive regular construction.

6.1.4. Reactive control modeling

Gaspard control mechanism is firstly proposed in [73] based on mode automata. An extension and improvement of this proposition have been developed regarding the following aspects: formal semantics, parallel and hierarchical composition [17]. The improved control mechanism remains generic and high-level because no specific execution model is assumed. For instance, it can be projected onto different execution models, for instance, the synchronous execution model. The graphical implementation of the Gaspard control according to MARTE has been also proposed [17]. This implementation is based on the UML state machines and collaborations. Extensions of Gaspard metamodel and synchronous metamodel and a transformation chain (not yet implemented) from graphical Marte/Uml descriptions into synchronous languages have been also defined. The implementation will be an extension of an existing one [44]. The targeted synchronous languages include Lustre, Lucid Synchrone, Signal.

As a case study, the control part of a multimedia processing functionality of a cellular phone has been modeled with the enhanced control mechanism [17].

6.1.5. Timing aspects modeling

This research action mainly focuses on the validation of Gaspard models specified in a high level specification language. The synchronous approach is still considered to deal with such validation issues. This is in perfect complementarity with our already started activity about design validation in Gaspard.

We were already able to address a certain number of functional properties in Gaspard models, such as single assignment, absence of dependency cycles. This is possible by considering the translation towards the synchronous model. Here, we extend the validation to non functional properties of a Gaspard model. To do so, our synchronous metamodel and its associated transformations are extended so as to explicitly handle temporal properties. From high-level model described in MARTE, including timing aspects specified with the concepts of the Time package, we consider a translation into synchronous programs via the extended synchronous metamodel. The aim is that the synchronous technology provides us with suitable analysis tools, which help to deal with both functional and non functional properties.

6.2. Model-based optimization and compilation techniques

Participants: Vincent Aranega, Abou El Hassan Benyamina, Pierre Boulet, Jean-Luc Dekeyser, Cédric Dumoulin, Anne Etien, Calin Glitia, Frédéric Guyomarc'h, Thomas Legrand, Jean-Marie Mottu, Vlad Rusu, Julien Taillard.

6.2.1. High performance computing

The transformation chain producing OpenMP code has been extended. We deal with the optimization of the produced OpenMP code. The global approach to model and produce scientific computing code has been presented in [34] while the compilation steps have been presented in [35].

One of the important point of the optimization is the use of data in read/write in order to minimize the cache fault which will occur during the execution of the application. Scientific computing libraries often use parameters as input and output. As the Array-OL specification link to a Gaspard model implies a single assignment model, we have proposed a way to specify the use of parameters as input and output to use those libraries and reduce cache fault.

We have apply the Gaspard environment to parallel an application based on the finite elements method. This application, called CARMEL, aims to simulate electromagnetism phenomenons based on the Maxwell's equations. The execution times is taken mostly into two steps, the assembly of the matrix and the resolution of the problems using the Conjugate Gradient method. These two steps have been modeled in the Gaspard environment and OpenMP codes have been produced through the transformations chains. Results have shown that the Gaspard methodology can be successfully used for this kind of application : the acceleration is quasi-linear in the assembly step until 8 processors [36].

6.2.2. Data dependence refactoring

We have proposed an extension of the Array-OL specification model to enable the specification of delays and stateful computations by the way of uniform dependence in the data repetition constructions. After an analysis on the interaction between these dependences and the already available data-parallel transformations, a methodology to connect these dependencies through hierarchy levels has been studied and formalized. The correctness of newly proposed transformations has been mathematically proved. These results represent the theoretical background for the implementation of an extension of the transformation toolbox in order to manage inter-repetition dependencies, work ongoing. Following the passage of the team to the OMG standard MARTE, the toolbox with the Array-OL transformations has been implemented as Papyrus plug-in. This implementation provides an improved user-interface based on the GUI provided by Papyrus.

In addition, in the context of the participation to the Ter@ops project, a converter between Ter@ops application models represented by PIPS XML models and our UML+Marte Profile models has been implemented as a Papyrus import/export plug-in. Finally, an architecture flattening transformation has been added to the transformation toolbox.

6.2.3. Traceability

Our solution relies on two models the Local and the Global Trace metamodels proposed in [70]. The former is used to capture the traces between the input and the output of one transformation. The Global Trace metamodel is used to link Local Traces according to the transformation chain.

We extend the Local Trace Model with new concepts like *Rule*, *Black-Box* to trace rules and black-box used to debug transformation rules and *PrimitivePropertyRef* to trace property values needed for architecture exploration and to debug modeled applications. We also propose a search algorithm in order to exploit the traces generated for a complete transformation chain.

We are currently in the process of changing our transformation language. As the trace generation is independent from the trace exploitation, we are working on a generic solution, to generate the required Local and Global traces, that can be easily adapted whatever the transformation language used.

6.2.4. Multiobjective heuristics for mapping and scheduling

We have proposed a multiobjective static allocation and scheduling heuristics for MPSoCs. By a combination of a genetic algorithm with Dijkstra's shortest path algorithm, we are able to derive schedules for task graphs on MPSoCs with voltage scaling. This algorithm gives a pareto front following the two objectives of latency and power consumption minimizing.

We study the performance of our algorithm over several examples and are in the process of extending this proposal to deal with hierarchy and data-parallel repetitions.

6.3. HP-SoC simulation verification and synthesis

Participants: Adolf Abdallah, Mohamed Abid, Rabie Ben Atitallah, Mouna Baklouti, Hajer Chtioui, Jean-Luc Dekeyser, Abdoulaye Gamatié, Sébastien Le Beux, Philippe Marquet, Samy Meftali, Smaïl Niar, Imran Rafiq Quadri, Wendell Rodrigues, Nicolas Wojcik, Huafeng Yu.

6.3.1. Partial and Dynamic Reconfiguration (PDR) implementations

Currently GASPARD RTL Chain allows the creation of hardware accelerators: using model to model transformations, it is possible to create the HDL code automatically. However, the creation of hardware accelerators is limited as it is only possible to create 1 hardware accelerator and afterwards synthesis is possible using usual synthesis design flows and tools. Some aspects of Partial Dynamic Reconfiguration are being added to the existing chain and the current RTL chain is being heavily modified to create reconfigurable hardware accelerators from high level UML descriptions. The goal is to 1) create part of a reconfiguration controller (basically the state machine part of the controller which is responsible for changing the partial bitstreams) 2) creation of several configurations, with each configuration corresponding to one implementation of the reconfigurable hardware accelerator in the reconfigurable region.

Initial experiments at RTL level were carried out to understand the emerging PDR mechanism and to integrate it in the model to model transformations. Currently, the deployment model is being modified into a controlled deployment model in order to integrate the control aspect of PDR which is an offshoot of the works being done in the synchronous domain in the GASPARD environment [32]. The current deployment level is being modified to allow the concept of configurations: an elementary component can have different implementations (IPs) in different configurations. A notion of a standard interface is being added due to the nature of PDR to implement a heterogeneous SoC in the FPGA. While the initial RTL chain only takes GASPARD based UML abstraction level specifications, our chain has been modified to specify the application at the MARTE abstraction level.

6.3.2. Hierarchical memories

The disparity between processor and memory speed is especially important in parallel and shared memory MPSoC. Thus, the cache has become a basic essential mechanism for reducing latency access memory and energy consumption. By adding caches to these machines, the memory delay is reduced, but the problem of cache coherence is added.

Two fundamental protocols are generally used to maintain cache coherence. The invalidation protocol consists on sending invalidation messages to processors that share this data, while the update protocol decides to send the new data to all other sharers. The invalidation protocol performs well for applications in which accesses to a particular data block are performed mostly by the same processor or when the data block migrates between processors. In another situation, for applications in which a single data is frequently read and written by different processors, the update protocol performs better. However these two situations can be presented in a single parallel application, then using a single protocol can increase traffic and contention of network resources. Hence, the necessity for a new dynamic hybrid protocol that takes advantage of the two protocols and that adapts to the way in which the data is used. It is dynamic because it can be changed during the execution of the application. In this context we have proposed a new dynamic hybrid cache coherence protocol, witch is based on a completely hardware solution for shared memory MPSoC and using a full bit vector directory. It is based on an original architecture which facilitates its implementation. Its advantage is the ability to be adapted easily to data access patterns at run-time. We have evaluated this protocol with the SoCLib platform by the FFT application. The results show that it can significantly reduce traffic in interconnection network and should be integrated in the Gaspardlib as a new Cache IP.

6.3.3. Performance evaluation of MPSoC in SystemC

Using the Gaspard2 environment, we are able to generate automatically a SystemC simulation from a high level MPSoC description. As a first step, we target a transactional abstraction level called Pattern Accurate (PA). The objective at this level is to make profit from the data intensive processing domain and to observe contentions in the interconnection network and bottlenecks to access to shared resources. For this in PA level, application tasks are carried out natively onto the host machine to accelerate simulation. However, memory accesses are simulated with SystemC. In intensive processing application, data transfers have a significant impact to determine the system performances. To implement the PA level, various kinds of component models that have been designed: caches, interconnection network, RAM, DMA controller, etc. At this level, a timing model is defined and plugged in the architectural simulator to approximate the execution time. Experimental results show that our MPSoC modelling gives a high simulation speedup factor of up to 25 with a performance estimation error up to 28% comparing to the CABA level. Now, we start working lower level then PA using Instruction Set Simulator (ISS) to execute application tasks. Certainly, this will improve performance estimation with the cost of lower simulation speedup.

6.3.4. mppSoC massively parallel processing System on Chip

To build IP-based massively parallel architectures (mppSoC), we propose a SIMD architecture composed of a number of processor elements (the PEs), organized in a 2D topology, working in perfect synchronization. A small amount of local and private memory is attached to each PE. Every PE is potentially connected to its neighbours via a regular network. Furthermore, each PE is connected to an entry of mpNoC, a

massively parallel Network on Chip that potentially connects each PE to one another, performing efficient irregular communications. All the system is controlled by an Array Controller Unit (ACU). We propose then a methodology to produce FPGA implementations of the mppSoC architecture and to integrate this IP assembling into the Gaspard deployment model.

An implementation on FPGA, ALTERA StratixII 2s180, is performed in order to proof its feasibility. The architecture consists of general IPs (processor IPs, memory IPs, etc.) and specific IPs supplied with the mppSoC system (control IPs, etc.). Specific IPs must always be used to construct the architecture to assure its right functioning. However, general IPs present a defined interface which must be respected by the designer if it wants to produce its own IP. For this kind of IPs we provide a library to alleviate their design. The architecture designed is configurable and parametric.

To evaluate the proposed design methodology we have implemented different sized architectures with various configurations. We have also tested some examples of data parallel applications such as FIR application, reduction application and matrix multiplication. As a result we have proposed an IP based methodology for the construction of mppSoC system used to make architectural exploration and helping the designer to choose the best configuration for a given application.

6.3.5. Formal validation

While the synchronous model is suitable to express data-parallelism and task parallelism defined in applications, the code generated from this model allows to reason about critical design properties of these applications. In other words, the formal validation and analysis based on the code generated from Gaspard models contributes to the safe design.

Besides the analysis of functional properties of Gaspard models such as single assignment, acyclic data dependency, array initialization, which is already addressed [15], [25], new validation issues concern the model checking of further functional properties, which is intended to deal with the safe control of Gaspard applications. Moreover, non-functional aspects related model checking is studied [17]. We check properties such as invariance and reachability under non-functional constraints of the system: energy, memory usage, processor load, etc. In addition to the validation, discrete controller synthesis has been experimented. It allowed us to generate automatically a safe system controller, which enforces safety properties in the system. All these aspects have been illustrated on a case study about a multimedia processing functionality of a cellular phone [17].

While the above non functional properties are addressed with the Sigali model checker, we also used the Polychrony compiler to deal with some temporal properties. This mainly concerns a synchronizability analysis based on the affine clock systems of the Signal language. An illustration of that is provided on a model of a video streaming application [20].

6.3.6. Hardware accelerator exploration

In the UML-VHDL chain, we have integrated an estimation process that allows one to immediately evaluate the generated hardware accelerators. This accelerates the exploration of the design space which relies on strategies modifying high level models in order to meet the performance requirements in low level models. We have validated the relevance of our high level synthesis flow for the design of a video processing application.

7. Contracts and Grants with Industry

7.1. The OpenEmbedd Project: A RNTL Project

Partners: Airbus, Anyware Technologies, CEA, CS SI, France Telecom, INRIA (AOSTE, DaRT, ESPRESSO), LAAS (CNRS), Thales Aerospace, Thales R&D, Verimag.

The OpenEmbedd project[4] aims to develop an engineering model driven open-source platform for real time and embedded systems. It deals with (1) UML standard for Real Time and Embedded systems, (2) innovating technology for interoperability, (3) mastering methodology chain, (4) real time models for simulation. The tools are evaluated in practical domains, e.g. the aeronautic sector, automobile sector, and telecom sector. The project will succeed in providing a technological core for Model Driven Engineering, by producing a set of tools dealing with different concerns about real time and embedded systems, and by validating an approach in the representative domains, with both applicative and methodological concerns. Software developed will be open-source. Future platform aims to federate academic partners effort and will guarantee a wide diffusion of the software.

Our OpenEmbedd partners are Airbus, Anyware Technologies, CEA, CS SI, France Telecom, INRIA, LAAS (CNRS), Thales Aerospace, Thales R&D, Verimag. The project has link with three competitiveness poles: Minalogic, System@tic, Aerospace Valley.

The activity of the DaRT Project in the OpenEmbedd RNTL project is to normalize models about real time and embedded systems domains, and more precisely the MARTE profile. The objectives are to participate to elaboration of a graphical editor generated from OpenEmbedd tools, and to work on plugins dedicated to simulation and checking. During the OpenEmbedd meetings that have been held in April and November in Grenoble and Toulouse respectively, DaRT presented the compilation chain from UML towards VHDL and the collapse of the hierarchical and repetitive Gaspard2 architecture model.

7.2. The Ter@ops Project: A System@tic Project

Partners: THALES (TRT, TOSA), Thomson, EADS (EADS Astrium, MBDA), Dassault Aviation, Renault, Valeo, Freescale (SAS), M2000, ARTERIS, Esterel Technologies, VirtualLogix, CEA-LIST, INRIA (Alchemy, Caps, DaRT), IEF, ENSTA, PRiSM, CRI (ARMINES / École des Mines), Laboratoire ETIS, RATP.

The Ter@ops project of the System@tic competitiveness pole aims at developing a hardware platform and the associated development framework for computation intensive applications. This project has started in December 2006.

We work at the level of the framework definition where we study the integration of Gaspard2 in a complete compilation and optimization framework for the Ter@ops platform and the compilation of the control proposed during the thesis of Ouassila Labbani in Gaspard2.

7.3. Collaboration within the competitiveness pole I-Trans

I-Trans is the official industrial cluster, which aims at bringing together major French actors in rail technology and innovative transport systems. The DaRT project strongly participates to this initiative through the collaborations with both concerned academic and industrial actors.

In this direction, we already have many discussions with partners (Inrets-Estas, Lagis, Lamih, Utc/Heudiasyc, Alstom and Certifier). These discussions lead to a few project proposals (Geneve, Klifr). The goal of these proposals concerns on the one hand, the decrease of validation and certification costs in the implementation of the new European railway system ERTMS/ETCS and on the other hand, the ease of interoperability through the mutual recognition of ERTMS components between European member countries. This qualification requires costly and long tests. More specifically, in the context of innovate equipments for railways, the objective consists in developing methods, models and tools dedicated to the generation of scenario tests for the validation of ERTMS components.

The contribution of the DaRT project to the definition of solutions to the above issues is twofold: first, its capabilities in the design of architectures and the association between application and architecture, so as to explore impacts on the test of the deployment on embedded architectures; second, its design experience in the automotive domain (see section 8.1).

[4] http://www.openembedd.org

7.4. Collaboration with CEA List

Partners: CEA List, DaRT

Since last year we have started a point to point collaboration with CEA around a UML profile for co-design. This work is done together with a PhD student (CEA funding). The main contribution consists in defining a metamodel for hardware architecture for the future MARTE standard. The results of this research is also integrated in the Gaspard2 tools at INRIA and in the AccordUML environment at CEA.

This collaboration is complementary with the above partnership between CEA and DaRT in the Cortess project.

7.5. Collaboration with Thales

Partners: ENSIETA, INRIA (DaRT), Thales

In order to increase productivity and thus decrease time to market, we propose to apply Model Driven Engineering (MDE) through the use of process components, which encapsulate the main activities of co-design processes. We consider that activities going from requirements analysis to implementation, whatever the chosen life-cycle, can be capitalized through process components. They are several formalisms to describe a process (BPML, CPR, PSL, SPEM, etc.). We propose a modified version of OMG's SPEM profile (Software Process Engineering Modelling) that allows modelling executable processes in order to implement a full MDE process.

We experiment our approach in a co-design process based on the use of the new MARTE profile and we intend to provide a tool that implements it in order to help engineers. In [71], we explain our approach applied to the development of Radio Frequency Transceiver. During our experimentation, we had to face some problems of metamodel formalization using tools. We have then proposed in [72] a framework to define more formalized metamodels.

The next challenging issue concerns the identification of a way our co-design process can be adapted to the Gaspard2 framework for design exploration.

This work is done in the context of a CIFRE PhD contract co-supervised by Joël Champeau from ENSIETA (Brest) and Jean-Luc Dekeyser.

7.6. Collaboration with Valeo - CNRT Futurelec

Partners: Valeo, INRIA (DaRT), L2EP

The objective of this project consists of fitting Gaspard2 for high performance computing and meta-computing. This work is done with a PhD student (Valeo/region funding) and is part of the program 1 of MEDEE (Parallelization of CARMEL). The results of this research are to be considered for integration in the Gaspard tool-set at INRIA and will provide models for finite elements simulations for electric alternator simulator developed by Valeo and L2EP.

7.7. Collaboration with Valeo - GPUTech

With the generalization of the GPGPU Computing (General Purpose GPU), we plan to add a new target for our Gaspard2 Framework: producing optimized code for GPU. This task is financed by Valeo, who plan to run their simulations on GPU, and by GPUTech who will includes their technology in the framework.

8. Other Grants and Activities

8.1. ModEasy Interreg III A Franco-English Cooperation

The ModEasy project[5] develops software tools and techniques in order to facilitate the development of reliable microprocessor-based electronic (embedded) systems using advanced development and verification systems. The defined tools will be evaluated in practical domains such as automotive.

[5] http://www.lifl.fr/modeasy

We have two partners in this project. First, the University of Kent, which works on formal system verification, embedded system development support and hardware integration from research councils and industry. Second, the IEMN (Institut Electronique Microelectronique Nanotechnologies), which has substantial expertise in the safety of land-based transportation systems especially for collision avoidance.

In this context, we have proposed a FPGA prototype that merges a reactive cruise control and an anti-collision radar in a single chip. The results of all partners have been presented during ModEasy Workshop[6] that held in Barcelona in September 2007.

8.2. Co-modeling and model engineering for HPC in electromagnetism software

Partners: IRISA, DaRT, L2EP.

The goal of this project is to use the power of new software engineering tools to design a new version of the electromagnetism solver CARMEL. This project emphasis on the fact that scientist from many different fields (physics, applied mathematics, high performance computing, software engineering) need to collaborate to ensure its success, and deliver a model of CARMEL using Gaspard2 metamodel. Then with the Gaspard2 development environment, we can generate a parallel software to solve the Maxwell equations.

8.3. STIC INRIA - Tunisia program

We have been co-advising two PhD students and several Master students in collaboration with the team of Pr. Mohamed Abid at CES-ENIS in Sfax. This collaboration is supported by the STIC Inria-Tunisia program, which aims at promoting the design of metamodels, transformation tools and techniques for the implementation of reconfigurable systems-on-chip. The resulting co-design environment will be validated on embedded systems dedicated to security in automobile, and more specifically in the design of cruise control systems integrating anti-collision radars.

Several successful student exchanges have been realized since 2006 between DaRT and CES-ENIS.

8.4. International initiatives

8.4.1. ECSI

The European Electronic Chips & Systems design Initiative Missions are to identify, develop and promote efficient methods for electronic system design, with particular regards to the needs of the System-on-Chip and to provide ECSI members with a competitive advantage in this domain for the benefit of the European industry. The list of participants is on:
http://www.ecsi.org.

Our team became an ECSI member in 2004. In this context we organized the ECSI conference in Lille: FDL'04. Pierre Boulet is a member of the executive committee of ECSI and secretary of ECSI.

8.4.2. University of California - Irvine

We started collaboration with the Center of Embedded Computer Systems (CECS) of the University of California at Irvine around networks on-chip design. Our objective consists mainly in modelling high performance networks as multistage Delta networks as cycle accurate, SystemC, reusable IPs [77]. We plan to continue this in this topic our collaboration and move in the future to FPGA implementations of these networks. In fact, aspects of dynamic reconfiguration of FPGAs make them very adapted to 'intelligent' NoC implementation.

[6] http://www.lifl.fr/modeasy/workshop.html

8.4.3. University of Montreal

The collaboration with University of Montreal laboratory continued this year. The interested laboratory is the LASSO (Laboratoire d'Analyse et de Synthèse des Systèmes Ordinés[7]), from DIRO department (Département d'Informatique et de Recherche Opérationnelle[8]). Student Frédéric Bastien came in the team during two months, and worked on the VHDL description of a massively parallel machine on FPGA.

8.4.4. University of Oran

A collaboration has started with the university of Oran, Algeria. Abou El Hassan Benyamina has been invited for one year (mid 2006 to mid 2007) and then three more months in 2008 to initiate the collaboration. He is working with Pierre Boulet on scheduling and mapping algorithms for SoC.

8.4.5. Collaboration with Spain

We initiate a collaboration with IUMA, University of Las Palmas (Spain) with Antonio Nunez. Some visits and student exchanges are scheduled for the next months. Our collaboration will be focused on NoC simulation, verification and design. Euromed 3+3 will be one of the projects supporting this work.

8.5. National initiatives

We are members of the ASR[9] and SoC-SiP[10] GDRs (research groups from the CNRS).

9. Dissemination

9.1. Scientific Community

Pierre Boulet was in the steering committee and the program committee of FDL[11] since 2005. He is the UML/UMES topic program chair of FDL'07, FDL'08 and FDL'09. He has initiated the MARTE user group in 2008 (first meeting in September 2008). He was a member of the INRIA evaluation committee from September 2005 to September 2008. He was in several PhD thesis committees in 2008. He is vice-director of the LIFL starting January 1st, 2008.

Jean-Luc Dekeyser has been member of different program committees: ECMDA-FA 2008, recosoc08, IS-PAN'08, IDT'08; invited speaker in FETCH'08 Montréal, ICESCA'08 and IDT'08, referee for some journals and conferences in MDE and SoC design. He was in about ten PhD and HdR thesis committees. As director of the Ph.D. program at LIFL, Inria and Doctoral School SPI, he was involved in the belgium/france relationship concerning PhD program.

Anne Etien is member of the program committee of the revue objet, special issue on MDE.

Frédéric Guyomarc'h is member of the program committee of PMAA'08.

Smail Niar is member of the "European Network of Excellence on High Performance and Embedded Architecture and Compilation" (Hipeac) European network of excellence (NoE). He has been member of scientific committees and selection committees of several international conferences and journals in the field of embedded system design (DATE, HPCA, Journal of system architectures, etc.).

Cedric Dumoulin is member of the programm committee of IDM08. He is committer of the eclipse-Papyrus project.

[7] http://www.iro.umontreal.ca/~lablasso/lablasso/
[8] http://www.iro.umontreal.ca
[9] http://asr.cnrs.fr/
[10] http://www.lirmm.fr/soc_sip/
[11] http://www.ecsi.org/fdl

Samy Meftali has been member of scientific committees and selection committees of several international conferences and journals in design automation (DATE, ACM, ..). He is managing the MVAR Europed project (starting 2009).

The team has presented a half-day tutorial about Gaspard2 at FDL'08. Several members of the team have visited ENIS Sfax and EμE laboratory of Monastir, Tunisia.

9.2. Teaching

As the DaRT team is mostly composed of professors and associate professors, we have a very large teaching activity. The more directly related to the research themes of the team are the master-level courses "System-on-Chip design" (Pierre Boulet, Jean-Luc Dekeyser, Samy Meftali, Abdoulaye Gamatié, Anne Etien), "Introduction to real-time operating systems" (Philippe Marquet), "Simulation of Systems and Architectures" (Philippe Marquet and Samy Meftali), "Distributed Systems and Infrastructures" (Pierre Boulet), "Advanced Computer Architecture" (Jean-Luc Dekeyser, Pierre Boulet, Calin Glitia) and "Model Driven Engineering" (Anne Etien). Smail Niar is in charge of several courses for master students in relationship of embedded system design, particularly "introduction to embedded system design" and "hardware/software co-design for high performance embedded systems".

The following internships were advised in the team:

- Asma Charfi, M2 Computer Science, ENIS Université de Sfax (Tunisia)
- Flori Glitia, M2 Computer Science, Université des Sciences et Technologies de Lille
- Nicolas Wojcik, M2 Computer Science, Université des Sciences et Technologies de Lille
- Ahmed Mekki, M2 Computer Science, École Centrale de Lille
- Amira Hasnaoui, Master, facultï¿½ des sciences de Tunis, Tunisie.
- Chiraz Trabelsi, Master, INSAT, Tunis
- Amen Souissi, M1 Computer Science, Université des Sciences et Technologies de Lille
- Romuald Vaz Madera, M1 Computer Science, Université des Sciences et Technologies de Lille

10. Bibliography

Major publications by the team in recent years

[1] A. C. ALJUNDI, J.-L. DEKEYSER, M. T. KECHADI, I. D. SCHERSON. *A universal performance factor for multi-criteria evaluation of multistage interconnection networks*, in "Future Generation Comp. Syst.", vol. 22, n^{O} 7, 2006, p. 794-804.

[2] R. BEN ATITALLAH, É. PIEL, S. NIAR, P. MARQUET, J.-L. DEKEYSER. *Multilevel MPSoC simulation using an MDE approach*, in "IEEE International SoC Conference (SoCC 2007), Hsinchu, Taiwan", September 2007.

[3] P. BOULET, C. DUMOULIN, A. HONORÉ. *6*, in "Model Driven Engineering for System-on-Chip Design", ISBN:1-905209-59-0, Sébastien Gérard, Jean-Philippe Babeau and Joël Champeau, editors, ISTE, September 2006, http://iste.co.uk/index.php?f=a&ACTION=View&id=147.

[4] A. CUCCURU, J.-L. DEKEYSER, P. MARQUET, P. BOULET. *Towards UML 2 extensions for compact modeling of regular complex topologies*, in "MODELS/UML 2005, ACM/IEEE 8th international conference on model driven engineering languages and systems, Montego Bay, Jamaica", October 2005.

[5] A. GAMATIÉ, É. RUTTEN, H. YU, P. BOULET, J.-L. DEKEYSER. *Synchronous Modeling and Analysis of Data Intensive Applications*, in "EURASIP Journal on Embedded Systems,eurasip", Article ID 561863, vol. 2008, july 2008.

[6] J. KHAN, S. NIAR, A. MENHAJ, Y. EL HILALI, J.-L. DEKEYSER. *An MPSoC architecture for the Multiple Target Tracking application in driver assistant system*, in "Application-Specific Systems, Architectures and Processors, 2008. ASAP 2008. International Conference on", July 2008, p. 126-131.

[7] O. LABBANI, J.-L. DEKEYSER, É. RUTTEN. *Safe Design Methodology for an Intelligent Cruise Control System with GPS*, in "VTC'2006 Fall: 64th IEEE Vehicular Technology Conference, Montréal, Québec, Canada", (Also appeared as an IEEE ITSS Newsletter in December 2006), September 2006.

[8] S. LE BEUX, P. MARQUET, J.-L. DEKEYSER. *A Model Driven Co-Design Approach for High Perforamnce Embedded Systems Dedicated to Transport*, in "Studies in Informatics and Control Journal", vol. 2008, n^{o} 4, 2008.

[9] P. MARQUET, S. DUQUENNOY, S. LE BEUX, S. MEFTALI, J.-L. DEKEYSER. *Massively Parallel Processing on a Chip*, in "ACM Int'l Conf. on Computing Frontiers, Ischia, Italy", May 2007.

[10] H. SBEYTI, S. NIAR, L. EECKHOUT. *Pattern-Driven Prefetching for Multimedia Applications on Embedded Processors*, in "Journal of System Architecture", vol. 52, n^{o} 4, April 2006, p. 199-212.

[11] J. TAILLARD, F. GUYOMARC'H, J.-L. DEKEYSER. *A Graphical Framework for High Performance Computing using an MDE Approach*, in "16th Euromicro International Conference on Parallel, Distributed and network-based Processing, Toulouse, France", February 2008.

Year Publications

Doctoral Dissertations and Habilitation Theses

[12] R. BEN ATITALLAH. *Modèles et simulation des systèmes sur puce multiprocesseurs - Estimation des performances et de la consommation d'énergie*, in french, Thèse de doctorat (PhD Thesis), Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, Lille, France, March 2008.

[13] S. TAHA. *Modélisation conjointe logiciel/matériel de systèmes temps réel*, in french, Thèse de doctorat (PhD Thesis), Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, Lille, France, february 2008.

[14] H. YU. *A MARTE-Based Reactive Model for Data-Parallel Intensive Processing: Transformation Toward the Synchronous Model*, in french, Thèse de doctorat (PhD Thesis), Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, Lille, France, November 2008.

Articles in International Peer-Reviewed Journal

[15] A. GAMATIÉ, É. RUTTEN, H. YU, P. BOULET, J.-L. DEKEYSER. *Synchronous Modeling and Analysis of Data Intensive Applications*, in "EURASIP Journal on Embedded Systems,eurasip", Article ID 561863, vol. 2008, july 2008.

[16] S. LE BEUX, P. MARQUET, J.-L. DEKEYSER. *A Model Driven Co-Design Approach for High Perforamnce Embedded Systems Dedicated to Transport*, in "Studies in Informatics and Control Journal", vol. 2008, n^{o} 4, 2008.

[17] H. YU, A. GAMATIÉ, É. RUTTEN, J.-L. DEKEYSER. *Safe Design of High-Performance Embedded Systems in an MDE framework*, in "Innovations in Systems and Software Engineering (ISSE)", vol. 4, n^{o} 3, 2008, http://tinyurl.com/3qr8jz.

Invited Conferences

[18] J.-L. DEKEYSER, A. GAMATIÉ, A. ETIEN, R. BEN ATITALLAH, P. BOULET. *Using the UML Profile for MARTE to MPSoC Co-Design*, in "First International Conference on Embedded Systems & Critical Applications (ICESCA'08), Tunis, Tunisia", May 2008.

[19] É. PIEL, R. BEN ATITALLAH, P. MARQUET, S. MEFTALI, S. NIAR, A. ETIEN, J.-L. DEKEYSER, P. BOULET. *Gaspard2: from MARTE to SystemC Simulation*, in "Proceeedings of the DATE'08 workshop on Modeling and Analyzis of Real-Time and Embedded Systems with the MARTE UML profile", March 2008.

International Peer-Reviewed Conference/Proceedings

[20] A. ABDALLAH, A. GAMATIÉ, J.-L. DEKEYSER. *MARTE-based Design of a Multimedia Application and Formal Analysis*, in "Forum on specification and design languages (FDL'08), Stuttgart, Germany", September 2008.

[21] M. BAKLOUTI, P. MARQUET, M. ABID, J.-L. DEKEYSER. *A Design and an implementation of a parallel based SIMD architecture for SoC on FPGA*, in "Conference on Design and Architectures for Signal and Image Processing (DASIP 2008), Bruxelles, Belgium", November 2008.

[22] M. BAKLOUTI, S. LE BEUX, P. MARQUET, M. ABID, J.-L. DEKEYSER. *Implementation of a Simple Massively Parallel Processor System on FPGA: Case Study*, in "First International Conference on Embedded Systems & Critical Applications (ICESCA 2008), Tunis, Tunisia", May 2008.

[23] H. CHTIOUI, S. NIAR, M. ABID. *Performances evaluation of multi-modules caches memories for embedded systems*, in "First International Conference on Embedded Systems & Critical Applications (ICESCA 2008), Tunis, Tunisia", May 2008.

[24] A. GAMATIÉ, T. GAUTIER, L. BESNARD. *An Interval-Based Solution for Static Analysis in the SIGNAL Language*, in "15th Annual IEEE International Conference and Workshop on Engineering of Computer Based Systems (ECBS 2008), Belfast, Northern Ireland", IEEE Computer Society, april 2008, p. 182-190.

[25] A. GAMATIÉ, É. RUTTEN, H. YU, J.-L. DEKEYSER. *Modeling and Formal Validation of High-Performance Embedded Systems*, in "7th International Symposium on Parallel and Distributed Computing (ISPDC 2008), Krakow, Poland", July 2008.

[26] C. GLITIA, P. BOULET. *High Level Loop Transformations for Multidimensional Signal Processing Embedded Applications*, in "International Symposium on Systems, Architectures, MOdeling, and Simulation (SAMOS VIII), Samos, Greece", July 2008.

[27] F. GLITIA, A. ETIEN, C. DUMOULIN. *Traceability for an MDE Approach of Embedded System Conception*, in "Fourth ECMDA Tracibility Workshop, Berlin, Germany", June 2008.

[28] J. KHAN, S. NIAR, A. R. MENHAJ, Y. EL HILALI. *Multiple Target Tracking System Design for Driver Assistance Application*, in "Design and Architectures for Signal and Image processing DASIP'08, Brussells, Belgium", November 2008.

[29] J. KHAN, S. NIAR, Y. EL HILALI, J.-L. DEKEYSER. *An MPSoC Architecture for the Multiple Target Tracking Application in Driver Assistant System*, in "19th IEEE International Conference Application-specific Systems, Architectures and Processors (ASAP'08), Leuven, Belgium", July 2008.

[30] IMRAN RAFIQ. QUADRI, P. BOULET, S. MEFTALI, J.-L. DEKEYSER. *Using An MDE Approach for Modeling of Interconnection networks*, in "The International Symposium on Parallel Architectures, Algorithms and Networks Conference (ISPAN 08), Sydney, Australia", May 2008.

[31] IMRAN RAFIQ. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *High Level Modeling of Partially Dynamically Reconfigurable FPGAs based on MDE and MARTE*, in "Reconfigurable Communication-centric SoCs (ReCoSoC'08), Barcelona, Spain", July 2008.

[32] IMRAN RAFIQ. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *MARTE based modeling approach for Partial Dynamic Reconfigurable FPGAs*, in "Sixth IEEE Workshop on Embedded Systems for Real-time Multimedia (ESTIMedia 2008), Atlanta, USA", October 2008.

[33] S. REVOL, S. TAHA, F. TERRIER, A. CLOUARD, S. GÉRARD, A. RADERMACHER, J.-L. DEKEYSER. *Unifying HW Analysis and SoC Design Flows by Bridging Two Key Standards: UML and IP-XACT*, in "Distributed Embedded Systems: Design, Middleware and Resources, IFIP 20th World Computer Congress, TC10 Working Conference on Distributed and Parallel Embedded Systems (DIPES 2008), September 7-10, 2008, Milano, Italy", 2008, p. 69-78.

[34] J. TAILLARD, F. GUYOMARC'H, J.-L. DEKEYSER. *A Graphical Framework for High Performance Computing using an MDE Approach*, in "16th Euromicro International Conference on Parallel, Distributed and network-based Processing, Toulouse, France", February 2008.

[35] J. TAILLARD, F. GUYOMARC'H, J.-L. DEKEYSER. *OpenMP code generation based on an Model Driven Engineering approach*, in "The 2008 High Performance Computing & Simulation Conference (HPCS 2008), Nicosia, Cyprus", June 2008.

[36] J. TAILLARD, F. GUYOMARC'H, Y. LE MENACH, F. PIRIOU, J.-L. DEKEYSER. *Modélisation d'une matrice issue de la méthode des éléments finis*, in "Les méthodes numériques en Electromagnétisme (NUMELEC 2008), Liège, Belgium", December 2008.

[37] M. TAWK, K. IBRAHIM, S. NIAR. *Multi-granularity sampling for simulating concurrent heterogeneous applications*, in "International conference on compilers, architectures and synthesis for embedded systems (CASES 2008), Atlanta, USA", October 2008.

[38] H. YU, A. GAMATIÉ, É. RUTTEN, J.-L. DEKEYSER. *Safe Design of High-Performance Embedded Systems in a MDE Framework*, in "First IEEE International workshop UML and Formal Methods (UML&FM'08), Kitakyushu, Japan", October 2008.

National Peer-Reviewed Conference/Proceedings

[39] A. CHARFI, A. GAMATIÉ, A. HONORÉ, J.-L. DEKEYSER, M. ABID. *Validation de modèles dans un cadre d'IDM dédié à la conception de systèmes sur puce*, in "4èmes Jounées sur l'Ingénierie Dirigée par les Modèles (IDM 08), Mulhouse, France", June 2008.

[40] H. CHTIOUI, R. BEN ATITALLAH, S. NIAR, M. ABID, J.-L. DEKEYSER. *Gestion de la cohérence des caches dans les architectures MPSoC utilisant des NoC complexes*, in "Rencontres francophones du Parallélisme (RenPar'18) / Symposium en Architecture de machines (SympA '2008) / Conférence Française sur les Systèmes d'Exploitation (CFSE), Fribourg, Switzerland", February 2008.

[41] IMRAN RAFIQ. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *A MDE design flow for implementing Partially Dynamically Reconfigurable FPGAs*, in "2nd Colloque Nationale of GDR SOC-SIP, Paris, France", June 2008.

[42] C. DE MOURA, J. TAILLARD, F. GUYOMARC'H, C. DUMOULIN. *Adaptation des Templates UML pour la modélisation de composants paramétrables : application à Gaspard2* , in "4èmes Jounées sur l'Ingénierie Dirigée par les Modèles (IDM 08), Mulhouse, France", June 2008.

Scientific Books (or Scientific Book chapters)

[43] É. PIEL, P. MARQUET, J.-L. DEKEYSER. *13*, in "Model Transformations for the Compilation of Multi-processor Systems-on-Chip", Springer Verlag, 2008, p. 459-473.

[44] H. YU, A. GAMATIÉ, É. RUTTEN, J.-L. DEKEYSER. *13*, in "Model Transformations from a Data Parallel Formalism towards Synchronous Languages", Springer Verlag, 2008.

Research Reports

[45] P. BOULET. *Formal Semantics of Array-OL, a Domain Specific Language for Intensive Multidimensional Signal Processing*, Research Report, n^{o} RR-6467, INRIA, March 2008, http://hal.inria.fr/inria-00261178/en/.

[46] A. GAMATIÉ, É. RUTTEN, H. YU. *A Model for the Mixed-Design of Data-Intensive and Control-Oriented Embedded Systems*, Research Report, n^{o} 6589, INRIA, France, July 2008, http://hal.inria.fr/inria-00293909/en.

[47] A. GAMATIÉ, S. LE BEUX, É. PIEL, A. ETIEN, R. BEN ATITALLAH, P. MARQUET, J.-L. DEKEYSER. *A Model Driven Design Framework for High Performance Embedded Systems*, Research Report, n^{o} 6614, INRIA, August 2008, http://hal.inria.fr/inria-00311115/en.

[48] S. LE BEUX, P. MARQUET, J.-L. DEKEYSER. *Model Driven Engineering Benefits for High Level Synthesis*, Research Report, n^{o} 6615, INRIA, 2008, http://hal.inria.fr/inria-00311300/en/.

[49] C. DE MOURA, A. ETIEN, J. TAILLARD, C. DUMOULIN, F. GUYOMARC'H. *Component-based Models Going Generic : the MARTE Case-Study*, Research Report, n^{o} 6632, INRIA, 2008, http://hal.inria.fr/docs/00/31/97/16/PDF/RR-6632.pdf.

References in notes

[50] IEEE (editor). *Std 1076-1993 IEEE Standard VHDL Language, Reference Manual - Description*, Inst of Elect & Electronic, 1994.

[51] OBJECT MANAGEMENT GROUP, INC. (editor). *U2 Partners' (UML 2.0): Superstructure, 2nd revised submission*, January 2003, http://www.omg.org/cgi-bin/doc?ptc/03-01-02.

[52] OBJECT MANAGEMENT GROUP, INC. (editor). *(UML 2.0): Superstructure Draft Adopted Specification*, July 2003, http://www.omg.org/cgi-bin/doc?ptc/03-07-06.

[53] C. BASTOUL. *Code Generation in the Polyhedral Model Is Easier Than You Think*, in "PACT'13 IEEE International Conference on Parallel Architecture and Compilation Techniques, Juan-les-Pins", September 2004, p. 7–16.

[54] R. BEN ATITALLAH, É. PIEL, S. NIAR, P. MARQUET, J.-L. DEKEYSER. *Multilevel MPSoC simulation using an MDE approach*, in "IEEE International SoC Conference (SoCC 2007), Hsinchu, Taiwan", September 2007, http://www2.lifl.fr/west/publi/BPT+07.pdf.

[55] A. BENVENISTE, P. CASPI, S. EDWARDS, N. HALBWACHS, P. LE GUERNIC, R. DE SIMONE. *The Synchronous Languages Twelve Years Later*, in "Proceedings of the IEEE", vol. 91, n^{o} 1, January 2003, p. 64-83.

[56] A. E. H. BENYAMINA, P. BOULET. *Multi-objective Mapping for NoC Architecture*, in "Journal of Digital Information Management", vol. 5, n^{o} 6, December 2007, p. 378–384.

[57] P. BOULET. *Array-OL Revisited, Multidimensional Intensive Signal Processing Specification*, Research Report, n^{o} RR-6113, INRIA, February 2007, http://hal.inria.fr/inria-00128840/en.

[58] P. BOULET, J.-L. DEKEYSER, J.-L. LEVAIRE, P. MARQUET, J. SOULA, A. DEMEURE. *Visual Data-parallel Programming for Signal Processing Applications*, in "9th Euromicro Workshop on Parallel and Distributed Processing, PDP 2001, Mantova, Italy", February 2001, p. 105–112.

[59] P. CASPI, D. PILAUD, N. HALBWACHS, J.A. PLAICE. *Lustre: a declarative language for real-time programming*, in "Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages (POPL'87)", ACM Press, 1987, p. 178-188.

[60] A. DEMEURE, A. LAFAGE, E. BOUTILLON, D. ROZZONELLI, J.-C. DUFOURD, J.-L. MARRO. *Array-OL : Proposition d'un Formalisme Tableau pour le Traitement de Signal Multi-Dimensionnel*, in "Gretsi, Juan-Les-Pins, France", September 1995.

[61] A. DEMEURE, Y. DEL GALLO. *An Array Approach for Signal Processing Design*, in "Sophia-Antipolis conference on Micro-Electronics (SAME 98), France", October 1998.

[62] P. DUMONT, P. BOULET. *Transformations de code Array-OL : implémentation de la fusion de deux tâches*, Technical report, Laboratoire d'Informatique fondamentale de Lille et Thales Communications, October 2003.

[63] P. DUMONT. *Étude des Transformations d'un Code Array-OL dans Gaspard*, Research Report, n^{o} 02-11, Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, France, September 2002, http://www.lifl.fr/LIFL1/publications/2002-11.ps.

[64] P. DUMONT. *Spécification multidimensionnelle pour le traitement du signal systématique*, (In French), Thèse de doctorat (PhD Thesis), Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, 2005.

[65] ECLIPSE CONSORTIUM. *The Eclipse Project*, 2003, http://www.eclipse.org.

[66] ECLIPSE CONSORTIUM. *EMF Eclipse Modeling Framework*, 2007, http://www.eclipse.org/emf.

[67] D. D. GAJSKI, R. KUHN. *Guest Editor Introduction: New VLSI-Tools*, in "IEEE Computer", vol. 16, n^{o} 12, December 1983, p. 11-14.

[68] I. GALVÃO, A. GOKNIL. *Survey of Traceability Approaches in Model-Driven Engineering*, in "IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007)", October 2007, p. 313-326.

[69] C. GLITIA. *Code transformations for systematic signal processing and memory size optimizations*, Master recherche, Université des sciences et technologies de Lille, 2006.

[70] F. JOUAULT. *Loosely Coupled Traceability for ATL*, in "European Conference on Model Driven Architecture (ECMDA) workshop on traceability", 2005, p. 29–37.

[71] A. KOUDRI, D. AULAGNIER, J. CHAMPEAU. *FPGA design based on UML/MDA Approach: Application to an RF Tranceiver development*, in "4th workshop on tUML for SoC Design", June 2007.

[72] A. KOUDRI, D. AULAGNIER, J. CHAMPEAU. *Une sémantique opérationnelle pour une meilleure métamodélisation*, in "Atelier sur la Sémantique des Modèles", march 2007.

[73] O. LABBANI. *Modélisation de haut niveau du contrôle dans des applications de traitement systématique à parallélisme massif*, Doctorat en Informatique, Université des Sciences et Technologies de Lille, November 2006.

[74] P. LE GUERNIC, J.-P. TALPIN, J.-C. LE LANN. *Polychrony for System Design*, in "Journal for Circuits, Systems and Computers", vol. 12, n^{o} 3, April 2003, p. 261–304.

[75] A. MEENA, P. BOULET. *Model Driven Scheduling Framework for Multiprocessor SoC Design*, in "Workshop on Scheduling for Parallel Computing, SPC 2005, Poznan, Poland", September 2005, http://www.cs.put.poznan.pl/mdrozdowski/spc-ppam05/.

[76] A. MEENA. *Allocation, Assignation et ordonnancement pour les systèmes sur puce multi-processeurs*, Doctorat en Informatique, Université des Sciences et Technologies de Lille, December 2006.

[77] S. MEFTALI, J.-L. DEKEYSER, ISAAC D. SCHERSON. *Scalable Multistage Networks for Multiprocessor System-on-Chip Design*, in "International Symposium on Parallel Architectures, Algorithms, and Networks, Las Vegas, Nevada, USA", December 2005, http://sigact.acm.org/ispan05/.

[78] OBJECT MANAGEMENT GROUP, INC.. *Meta Object Facility (MOF) Core Specification, Version 2.0*, January 2006, http://www.omg.org/docs/formal/06-01-01.pdf.

[79] OBJECT MANAGEMENT GROUP, INC.. *MOF Query / Views / Transformations*, OMG paper, July 2007, http://www.omg.org/docs/ptc/07-07-07.pdf.

[80] OPEN SYSTEMC INITIATIVE. *SystemC*, 2002, http://www.systemc.org/.

[81] OPENMP ARCHITECTURE REVIEW BOARD. *OpenMP Application Programme Interface*, May 2005, http://www.openmp.org/drupal/mp-documents/spec25.pdf.

[82] D. SCHMIDT. *Model-Driven Engineering*, in "IEEE Computer", vol. 39, n^{o} 2, February 2006, p. 41-47.

[83] J. SOULA. *Principe de Compilation d'un Langage de Traitement de Signal*, (In French), Thèse de doctorat (PhD Thesis), Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, December 2001.

[84] J. TAILLARD, F. GUYOMARC'H, J.-L. DEKEYSER. *A Graphical Framework for High Performance Computing using an MDE Approach*, in "16th Euromicro International Conference on Parallel, Distributed and network-based Processing, Toulouse, France", to appear, February 2008.

[85] DONALD E. THOMAS, PHILIP R. MOORBY. *The Verilog Hardware Description Language*, Fourth, Kluwer Academic Publishers, May 1998.

[86] Y. VELEGRAKIS, R. J. MILLER, J. MYLOPOULOS. *Representing and Quering Data transformations*, in "International conference on Data Engineering (ICDE)", April 2005, p. 81-92.