INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team DaRT

Dataparallelism for Real-Time

Lille - Nord Europe

Theme : Embedded and Real Time Systems

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) until september 2009]
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]
Philippe Marquet [Associate professor, USTL]

External Collaborator

Rabie Ben Atitallah [Associate professor, UVHC, from september 2009]
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 grant, until september 2009]
Souha Kamoun [French Minister grant]
Imran Rafiq Quadri [HEC Pakistan and INRIA grant]
Wendell Rodrigues [VALEO, GPU Tech and Regional grant]
Chiraz Trabelsi [INRIA grant, from september 2009]
Nicolas Wojcik [INRIA grant]

Post-Doctoral Fellow

Jean-Marie Mottu [Post-doctoral fellow, INRIA, until august 2009]
Rosilde Corvino [Post-doctoral fellow, INRIA, since december 2009]

Visiting Scientist

Abou El Hassan Benyamina [Associate Professor, University of Oran, Algeria]
Vlad Rusu [Research scientist, INRIA Rennes-Bretagne Atlantique, HdR]
Laure Gonnord [Associate professor, USTL, from september 2009]

Administrative Assistant

Karine Lewandowski [INRIA]

2. Overall Objectives

2.1. Overall Objectives

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 help to 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 [46], MDE [79] and Eclipse EMF [62] for the modeling and model handling; Array-OL [56], [57], [54], [52] and synchronous languages [50] as computation models with strong semantics for verification; SystemC [77] for the simulation; OpenMP for the shared memory parallel execution; VHDL for the synthesis; and Java to code our prototypes.

3. Scientific Foundations

3.1. Introduction

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. We developed new heuristics to minimize the power consumption. 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

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) [79] 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, 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 [45] for modeling, the MOF (Meta Object Facilities [75]) for metamodel expression and QVT [76] 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 [56], [57], [54], [52] 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 [50] 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 [55] and Signal [71] 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" [63]. 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 modeled 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 behaviors 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.

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:

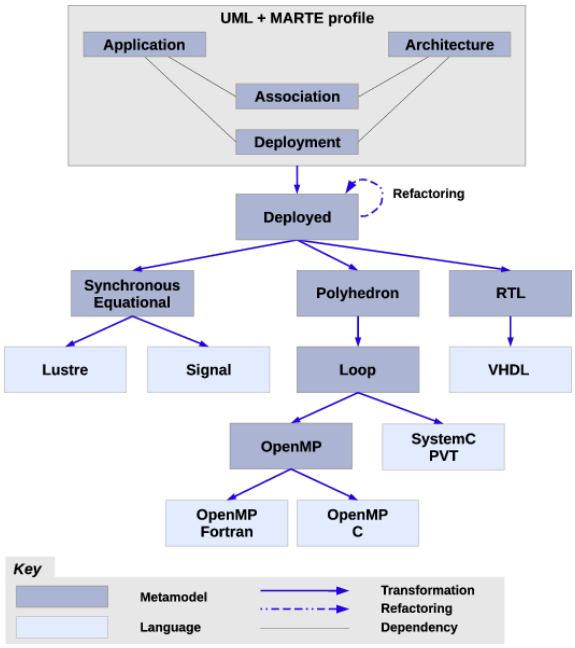

Figure 1. Overview of the design concepts.

- 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 [48]).
- 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 [49]. 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 [78]. 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 [81].
- 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 [43] or Verilog [82] with the corresponding IPs. However, it is precise enough to enable the generation of synthesizable HDL code [70].

- 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 [84].

3.3. Model-based optimization and compilation techniques

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 [76] 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) 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 engines 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 [64]. 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 [83]. 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 [67].

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.

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

3.3.2.1. Data-parallel code transformations

We have studied Array-OL to Array-OL code transformations [54], [80], [59], [58], [60] [66]. 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 [72] 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 [51] while we have proposed a heuristic to deal with the regular part (data parallelism) [73].

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. Another tool MoCode has also been defined, for the model to text transformation (i.e., the code generation). It relies on JET 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

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 and Transaction Level Modeling

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 modeling 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.

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.

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.

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.

3.4.1.4. 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 synchronization primitive calls.

The SoC architecture is directly produced from the architecture model. A module in SystemC simulates the behavior 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. Contributions 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. 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

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 (http://www2.lifl.fr/modeasy) 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 modeled 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 behavior 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 [81] 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

Participants: DaRT members, 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 a single environment for all the SoC development processes:

- High level modeling of applications and hardware architectures
- Application and hardware architecture association (mapping and scheduling)
- 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 [61] IDE. A set of plugins provides the different functionalities. Gaspard2 provides an internal engine to execute transformation chains. This engine is able to run either QVT (OMG standard) or Java transformations. It is also able to run model-to-text transformations based on Acceleo [44]. The Gaspard2 engine is defined to execute models conform to an internal transformation chains meta-model. A GUI has been developed to specify transformation chain models by drawing them.

For the final user, 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. Several transformation chains are provided with Gaspard2 to target, from UML models, several execution or simulation platforms (OpenMP, Pthread, SystemC, VHDL, ...). This input language is based on the MARTE UML profile.

5.2. Papyrus

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 now an Eclipse project (in the incubator state). (http://www.eclipse.org/modeling/mdt/?project=papyrus#papyrus). A first version of Papyrus Eclipse should be released with the Eclipse Helios train (mid 2010). Pre-releases version are also available as well as the fully functional Papyrus I (http://www.papyrusuml.org)

The Papyrus Tool is developed under an Open source license in collaboration with CEA, Atos, Obeo, Airbus, Prodevelop, Integranova.

6. New Results

6.1. Co-modeling for HP-SoC design

Participants: Adolf Abdallah, Pierre Boulet, Jean-Luc Dekeyser, Abdoulaye Gamatié, Souha Kamoun, Vlad Rusu, Nicolas Wojcik.

6.1.1. An operational semantics for RSM

The Repetitive Structure Modeling (RSM) package of the UML MARTE profile is used to describe repetitive computations and topologies (e.g., data-parallel algorithms, grid of processing units) in an embedded system. In Gaspard2, the concepts provided by this package are of prime importance for the specification of data-intensive applications. A formal semantics [53] has been previously defined for the Array-OL language, which is the basis for the definition of RSM. Now, we propose an new formal semantics for RSM, which is operational unlike [53]. Execution semantic descriptions are rarely taken into account in the definition of UML profiles. This raises several serious correctness issues about the manipulation of models defined with these profiles. The aim of our new semantics is to answer this demand by proposing a help for understanding the behavior and execution of models specified with RSM concepts in UML MARTE.

6.1.2. Clock-based modeling of embedded system behavior

This research mainly focuses on the modeling of data-intensive embedded system behaviors within the Gaspard2 framework [22] [39]. Algorithms describing data intensive computations are described in the application specification. In particular, the concepts defined in the RSM package of MARTE allow one to suitably describe the application. In order to add more details about the system functional behavior, logical clocks are associated with components to describe the expected rates at which data should be processed. The Time sub-profile of MARTE is used to model this rate information. It offers a rich expressivity for describing both logical and physical time aspects [47]. The rate constraints are expressed using the CCSL package of MARTE in the form of clock constraints. We refer to this clock constraints as functional clock properties.

In order to implement the functionality of the system, the physical resources that implement the associated algorithms are also specified in an architecture specification phase. In Gaspard2, only the HWLogical sub-package of MARTE is used. For each physical resource, hardware IPs are deployed in order to refine the models towards a specific technology. At this level, we extract information concerning the processors speed represented by its frequency. We synthesize new clocks that represent the periods of the clock cycles for each processor involved in the execution. All clocks are related to an ideal clock. The occurrence of the instants of the ideal clock are fast enough to capture any instant of the processors clocks. We refer to these clock specifications as physical clock properties.

Since application functionality and hardware architecture are modeled independently in Gaspard2, the allocation phase bridges these two different views in order to map functionality on their associated physical resources. In terms of clocks, this allocation is expressed as the mapping of functional clock properties onto physical clock properties, according to a particular mapping algorithm. The result of such allocation is a new set of clocks reflecting the simulation of the temporal behavior of the system during execution. We refer to these clock description as simulation clock properties. They are usable for a very relevant system analysis.

6.1.3. High-level modeling and exploration of non functional properties

During the last year, we have proposed an approach for high-level modeling and exploration of non functional properties. Our work proposes a Model Driven Engineering (MDE)-based approach to integrate non functional requirements for systems on chip and defines metamodels that allow the integration of external optimization tools in the Gaspard2 environment. The designer creates the application and architecture models at a high level. The designer should then take the decision to allocate application functions on hardware components. This decision depends essentially on the non functional properties of both of the software and hardware components. For this reason, it is necessary to express these requirements. The proposed methodology uses models enriched with non-functional properties to drive the optimization of resource allocation. It aims at the integration of two optimization tools: SynDEx and an internal tool developed within the team.

The designer starts creating applications in Gaspard2 by defining application and architecture models using the UML component diagram. Components are stereotyped with MARTE concepts. The next step consists in transforming the obtained models according to the choice of the optimization tool to use. For the SynDEx transformation only the application and architecture model are considered. However, for the Benyamina tool a mapping graph of tasks into hardware components is needed. We have defined a graph metamodel that allows generating graph models that serve as support for optimization heuristics defined in the Benyamina tool.

6.1.4. HPF towards Marte

Concerning the power of expression of the MARTE RSM subprofile that we have defined, we have studied the data and computation distribution capabilities. We have proved that the MARTE «distribute» stereotype is at least as expressive as the well known High Performance Fortran data distribution. The proof is constructive: starting from an ALIGN and a DISTRIBUTE HFP directive, we build a MARTE «distribute».

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, Emmanuel Leguy, Jean-Marie Mottu, Alexis Muller, Wendell Rodrigues, Vlad Rusu.

6.2.1. Transformation

Our investigations tend to make up a complex model transformation from smaller transformations jointly working in order to build a single output model. These transformations involve different parts of the same input metamodel (e.g. the MARTE metamodel); their application field is localized. We propose a new way to define transformations focusing on the impacted concepts. The localization of the transformation is ensured by the definition of the intermediary metamodels as delta. The delta metamodel only contain the few concepts involved in the transformation (i.e. modified, or read). The specification of the transformations only use the concepts of these deltas. We define the Extend operator to build the complete metamodel from the delta and transpose the corresponding transformations. The complete metamodel corresponds to the merge between the delta and the MARTE metamodel or an intermediary metamodel. The transformation then becomes the chaining of metamodel shifts and the localized transformation.

This new way to define the model transformations has been used in the Gaspard2 environment. It allows a better modularity and thus also reusability between the various chains. The maintenance is enhanced since each transformation involves only a few concepts, has a well defined and restrict purpose and contains maximum ten rules (i.e. 250 lines of code). Finally, this approach should ease the evolvability of the chain after the evolution of one metamodel for example to answer new requirements, or to reach a new target.

6.2.2. Compilation for GPU

The GPU has a particular memory hierarchy. In order to model the memory details, we purpose an approach to extend the MARTE metamodel to describe low level characteristics of the memory. The model described in UML with Marte profile model is chained in several inout transformations that adds and/or transforms elements in the model. For adding memory allocation concepts to the model, a QVT transformation based on

«Memory Allocation Metamodel» provides information to facilitate and optimize the code generation. Then a model to text transformation allows to generate the C code for GPU architecture. Before the standard releases, Acceleo is appropriate to get many aspects from the application and architecture model and transform it in CUDA (.cu, .cpp, .c, .h, Makefile) and OpenCL (.cl, .cpp, .c, .h, Makefile) files. For the code generation, it's required to take into account intrinsic characteristics of the GPUs like data distribution, contiguous memory allocation, kernels and host programs, blocks of threads, barriers and atomic functions. For the moment, we work in both sides (model and generated code) to elaborate a chain of transformations.

6.2.3. Data dependence refactoring

The paper on the formalism of Array-OL with delays ("Array-OL with delays, a domain specific specification language for multidimensional intensive signal processing") was accepted and published in the journal "Multidimensional Systems and Signal Processing".

The study on the interaction between the high-level data-parallel transformations and the inter-repetition dependences (allowing the specification of uniform dependences) was accepted and presented at DASIP'09 conference. Because the ODT formalism behind the Array-OL transformations cannot express dependences between the elements of the same multidimensional space, in order to take into account the uniform dependences we proposed and proved an algorithm that, starting from the hierarchical distribution of repetition before and after a transformation, is capable to compute the new uniform dependences that express the same exact dependences as before the transformations. It all comes down to solving an (in)equations system, interpreting the solutions and translating them into new uniform dependences.

The algorithm was implemented and integrated into the refactoring toolbox and enables the use of the transformations on models containing inter-repetition dependences.

In order to validate the theoretical work around the high-level Array-OL refactoring based on the data-parallel transformations, together with Eric Lenormand and Michel Barreteau from THALES Research & Technology we worked on a study on optimization techniques in the context of an industrial radar application, work partially part of the Ter@ops project. We have proposed a strategy to use the refactoring toolbox to help explore the design space, illustrated on the radar application modeled using the Modeling and Analysis of Real-time and Embedded systems (MARTE) UML profile.

6.2.4. Traceability

Our traceability solution relies on two models the Local and the Global Trace metamodels proposed in [65]. The former is used to capture the traces between the inputs and the outputs of one transformation. The Global Trace metamodel is used to link Local Traces according to the transformation chain.

The huge modifications of the transformation chain engine involved to consequently adapt the existing trace navigation/creation algorithms. Furthermore, based on our trace metamodels, we develop new algorithms to ease the model transformation debug. Based on the trace, the localization of an error is eased by reducing the search field to the sequence of the transformation rule calls [23].

We start to automate the mutation analysis process dedicated to model transformations. This technique aims to qualify a test model set. If the test model set is not enough qualified new models have to be added in order to raise the set quality [74]. The local trace, coupled to a mutation matrix, helps the tester to create adequate new test models and thus improve the test data set. The first obtained results are really promising and we are currently working on the test data set improvement full automation.

6.2.5. Model transformation towards Pthreads

The strategy in previous version of the Gaspard framework imposes a global synchronization mechanism between all the tasks of the application. This mechanism does not allow one to reach an optimal execution. We have investigated a new strategy to overcome this problem, based on fine grain synchronizations between the different tasks of the modeled application. For this new strategy, we use the pthread API. Each task of the UML application model is transformed into a thread. The data exchanges between the tasks are ensured by a buffer-based strategy. The best compromise between the memory used and the performance can be reached by

adjusting the size of each buffer. Moreover, we have developed this strategy to facilitate its use in simulation targets such as SystemC-PA.

6.2.6. Verifying conformance and semantics-preserving model transformations

We give formal executable semantics to the notions of *conformance* and of *semantics-preserving model transformations* in the model-driven engineering framework [36]. Our approach consists in translating models and meta-models (possibly enriched with OCL invariants) into specifications in *Membership Equational Logic*, an expressive logic implemented in the Maude tool. Conformance between a model and a meta-model is represented by the validity of a certain *theory interpretation*, of the specification representing the meta-model, in the specification representing the model. Model transformations between origin and destination meta-models are mappings between the sets of models that conform to the those meta-models, respectively, and can be represented by rewrite rules in *Rewriting Logic*, a superset of Membership Equational Logic also implemented in Maude. When the meta-models involved in a transformation are endowed with dynamic semantics, the transformations between them are also typically required to preserve those semantical aspects. We propose to represent the notion of dynamic semantics preservation by means of *algebraic simulations* expressed in Membership Equational Logic. Maude can then be used for automatically verifying conformance, and for automatically verifying dynamic semantics preservation up to a bounded steps of the dynamic semantics. These works will eventually be incorporated in the Gaspard2 environment and will hopefully lead to better understood meta-models and models, and to model transformations containing fewer errors.

6.2.7. Scheduling data-parallel tasks with inter-repetition dependences

The introduction of uniform inter-repetition dependences in the data-parallel tasks of Gaspard2 has had several consequences. Aside the modification of the refactoring (see section 6.2.3), we have studied the compilation of such tasks. This compilation involves the scheduling of such repetitions on repetitive grids of processors and the code generation. This scheduling problem is NP-complete and we have proposed a heuristic based on the automatic parallelization techniques to compute a good (efficient both in time and code size) schedule in the case when all loop bounds and processor array shapes are known.

6.3. HP-SoC simulation verification and synthesis

Participants: Adolf Abdallah, Rabie Ben Atitallah, Mouna Baklouti, Hajer Chtioui, Rosilde Corvino, Pierre Boulet, Jean-Luc Dekeyser, Abdoulaye Gamatié, Laure Gonnord, Thomas Legrand, Philippe Marquet, Samy Meftali, Smaïl Niar, Imran Rafiq Quadri.

6.3.1. Partial and Dynamic Reconfiguration (PDR) implementations

Current Gaspard Model transformation chain to Register Transfer Level (RTL) allows to generate two key aspects of a partial dynamically reconfigurable system: namely the dynamically reconfigurable region and the code for the reconfiguration manager that carries out the switch between the different configurations of this dynamic region. For this, the MARTE metamodel has been extended to integrate concepts of UML state machines and collaborations, which help in creation of mode automata semantics at the high abstraction levels. Integration of these concepts in the extended MARTE metamodel helps in the respective model-to-model transformations.

Moreover, the high level application model has several building blocks: the elementary components, each associated to several available intellectual properties (IPs). The current deployment level has been also extended to integrate the notion of “configurations” [32], which are unique global implementations of the application functionality, with each configuration comprised of different combinations of IPs related to the elementary components. Using a combination of the deployment level and the introduced control semantics, it is possible for a designer to change the configuration related to an application, resulting in different results such as consumed FPGA resources, reconfiguration times, etc. We incorporate two model-to-model transformations in our flow, first the UML2MARTE transformation, with integrated state machine and configuration concepts. This transformation results in an intermediate MARTE model, which is converted into an RTL model by the MARTE2RTL transformation. The application model is converted into several implementations of a dynamically reconfigurable hardware accelerator, along with the source code for the configuration switch.

Finally, the design flow has been validated in the construction of a dynamically reconfigurable delay estimation correlation module [34] that is part of a complex anti-collision radar detection system in collaboration with IEMN Valenciennes [31]. The simulation results from the different configurations correspond to an initial MATLAB result, validating the different configurations. Additionally change of IPs related to a key elementary component in the module resulted in different reconfiguration times proving methodology.

6.3.2. Gaspardlib extensions

The different chains available within Gaspard produce simulation code. Among them, the SystemC code generation allows simulations at the TLM-PA level. Regarding the architecture design, the process acts at as a connector between existing SystemC modules. They correspond to basic components such as memories, processors, caches. They are gathered in the Gaspardlib to be included or linked at the code compilation step. On one hand, both application and architecture IPs have been modeled using UML to easily drag-and-drop the available components inside the user's model. On another hand, we aimed at providing the most flexible design for the SystemC architecture. Each SystemC module is composed of an interface and behavioral sections; the interfaces have been updated to the new OSCI standard: TLM2 (http://www.systemc.org/apps/group_public/workgroup.php?wg_abbrev=tlmwg). This update has allowed a high interoperability for our SystemC components with any other SystemC-TLM architecture. Consequently, additional SystemC modules have been integrated to extend the Gaspardlib. They come from other free simulation environments: ReSP (http://www.resp-sim.org), SocLib (https://www.soclib.fr/trac/dev/wiki), Unisim (https://www.soclib.fr/trac/dev/wiki).

Similarly, several developments have been undertaken to provide ISS-based (Instruction Set Simulator) components for low-level simulations. A set of ISS have been extracted from the environments cited above and reorganized to be usable within Gaspard. An Eclipse standalone have been created to easily handle the whole Gaspardlib. It helps to build advanced architectures relying on SystemC. This environment is a preliminary development step as it allows one to realize ISS-simulations that may later be automatically generated by the Gaspard chains.

6.3.3. Clock-based analysis of embedded system behavior

Starting from the simulation clock properties of an embedded system (as described previously), we can now start an analysis of the system behavior. On the one hand, we verify whether or not the functional clock constraints specified by the designer in the application specification are met during the system execution on considered physical resources. When these constraints are not met, the simulation clock traces can be used to reason and find the solutions to satisfy the constraints. For instance, this may amount to decrease the speed of processors that compute data very fast or to increase the speed of processors that compute data very slowly. The modification of the processors speed by increasing or decreasing the speed should always respect the functional constraints imposed by the designer. It appears in the simulation clock traces by determining new physical clock properties from the suitable processor frequencies. Another example of solution may consist in delaying the first activation of a faster processor until an adequate time to begin the execution. Such an activation delay could be seen as minimizing the voltage/frequency.

On the other hand, we can reduce the power dissipation of the system which is the combination of dynamic and static power. The dynamic dissipated power, which represents 80-85% of the total power dissipation, can be lowered by minimizing the processors frequency which leads to an automatic reduction of the voltage. The static power dissipation can be reduced by cutting off the power supply for the processor if it is not used for a long period. These aspects can be addressed by analyzing the simulation clock traces.

6.3.4. IP based configurable massively parallel processing SoC

A methodology and a tool chain to design and build IP-based configurable massively parallel architectures is proposed. The defined architecture is named mppSoC, massively parallel processing System on Chip. It is a SIMD architecture composed of a number of processor elements (the PEs) working in perfect synchronization. A small amount of local and private memory is attached to each PE. Every PE is potentially connected to its neighbors 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). Our objective is to propose then a methodology to produce FPGA implementations of the mppSoC architecture.

The whole mppSoC architecture with its various components is implemented following an IP based design methodology. An implementation on FPGA, ALTERA StratixII 2s180, is proposed as a proof of 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 are used as a glue to build the architecture. 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 designed architecture is configurable and parametric. In fact, to construct a mppSoC system, we assemble IPs to generate a FPGA configuration. The designer has to make different choices. He has to determine the different components in his architecture, for example if it contains an irregular communication network with a defined interconnection router [38] or a neighborhood one or both. Since we propose a parametric architecture, he has to choose also some architectural parameters such as the number of PEs, the memory size and the topology of the neighborhood network [24] if it exists. After fixing the architecture, the designer will choose then the basic IPs which will be used such as processor IP, interconnection network IP, etc. By this way, the user can choose the most appropriate mppSoC configuration satisfying his needs. 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, reduction, matrix multiplication, image rotation and 2D convolution. Through simulation results we can choose the most appropriate mppSoC configuration with the optimal performance metrics: execution time, FPGA resources and energy consumption. As a result we have proposed an IP based methodology for the construction of mppSoC system helping the designer to choose the best configuration for a given application. It is a first step towards a mppSoC architecture exploration.

Ongoing work aims at integrating the mppSoC in a real application such a video processing framework. Future work will aim at improving the proposed IP assembling methodology to construct mppSoC systems. Our ultimate goal is to provide a completely tool to generate a mppSoC configuration in order to help the designer in a semi-automatic architecture exploration for a given application.

6.3.5. Caches in MPSoCs

In Multi-Processor System-on-Chip (MPSoC) architectures using shared-memory, caches plays an important impact on performance and energy consumption levels.

When the executed application depicts a high degree of reference locality, caches may reduce the amount of shared-memory accesses and data transfers on the interconnection network. Hence, execution time and energy consumption can be greatly optimized. However, caches in MPSoC architectures put forward the data coherency problem. In this context, most of the existing solutions are based either on data invalidation or data update protocols. These protocols do not consider the change in the application behavior. This paper presents a new hybrid cache-coherency protocol that is able to dynamically adapt its functioning mode according to the application needs.

An original architecture which facilitates this protocol's implementation in Network-On-Chip based MPSoC architectures has been proposed in [25]. Performances, in terms of speed up factor and energy reduction gain of the proposed protocol, have been evaluated using a Cycle Accurate Bit Accurate (CABA) simulation platform. Experimental results in comparison with other existing solutions show that this protocol may give significant reductions in execution time and energy consumption can be achieved.

6.4. Application : electrical engines simulations

Participants: Jean-Luc Dekeyser, Frédéric Guyomarc'h, Abdellatif Tinzefte, Wendell Rodrigues.

Within a collaboration with the L2EP, Abdellatif Tinzefte has started to model and program a parallel version of a Maxwell equations solver based on a Finite Element Method (FEM) formulation, and the first step of this work is defining a parallel preconditioner for the Krylov solver. The idea here is to use the Finite Integration Technique (FIT) in order to compute the electromagnetic phenomena as like the FEM but on a coarser grid. The F.I.T is efficient if the mesh is generated by hexahedron elements. With this regular mesh the obtained system is large, sparse and have some properties of regularity which can be used for the parallel computing. Thus this FIT parallel solver will be used to approach the FEM operator. The first numerical results show an excellent precision and a good acceleration for the FEM solver. Another part of this work, is to was implemented in CUDA and OpenCL as well a solver for Maxwell's equations (Finite-Element Method and a conjugate gradient). The aim is to accelerate and verify the parallel code on GPUs. The first results showed a speedup around 6 times against sequential code on CPU. Another approach uses an algorithm that explores the sparse matrix storage format (by rows and by columns). This one did not increase the speedup but it allows to evaluate the impact of different accesses to the memory.

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.

Finished: june 2009
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.

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

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. Thier needs in computation power are huge and thus the DaRT project strongly participates to this initiative through the collaborations with both concerned academic and industrial actors, to design a new GPU-based server which aims to reach the Petaflops performance.

In this direction, we already have many discussions with the leading partner of this project : GPUTech. These discussions lead to the build of a smaller pre-prototype for this machine, and also the design of a set of applications which should be ported to this new plateform. This is were the DaRT project strongly contribute with its experience in massively parallel architectures.

7.4. Collaboration with CEA List

Partners: CEA List, DaRT

A PhD thesis (Asma Charfi) is co-advised between our team and the CEA List on optimized code generation from MARTE models. The idea is that some information is lost when the code is generated from a high level model to code. The compiler then tries to find back this lost information to optimize the code. If these optimisations were taken care of at model level, the compiler would have a simplified task to do and we could expect improved performance.

DaRT and the CEA List also collaborate on the Papyrus UML 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 (e.g., BPML, CPR, SPEM). We propose a modified version of OMG's SPEM profile (Software Process Engineering Modeling) 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 [68], 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 [69] 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 within the MEDEE research pole

Partners: Valeo, INRIA (DaRT), L2EP

MEDEE (stands for "Maîtrise énergétique des entrainements électriques") is a research pole involving many well-known industrial and academic partners. The recent propositions aim to convert it to a competive cluster. The DaRT project is especially involved in the first operation in the MEDEE programm about simulations of these electrical engines. In fact, due to the large computation volume needed to run FEM simulations, most of the tracks lead to the use of parallel computers to solve these problems always faster.

7.7. Collaboration with GPUTech and Valeo

Partners: INRIA (DaRT), L2EP, GPUTech, Valeo

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 partially financed by Valeo, who plan to run their simulations on GPU, and by GPUTech who will include their technology in the framework.

7.8. The ANR Open-PEOPLE project

The Open-PEOPLE (Open Power and Energy Optimization PLatform and Estimator) project is a national project funded by the ANR (Agence Nationale de la Recherche), the French National Research Agency. The objective of Open-PEOPLE is to provide a platform for estimating and optimizing the power and energy consumptions. Users will be able to estimate the consumption of an application deployed on a hardware architecture chosen in a set of parametric reference architectures. The components used in the targeted architecture will be chosen in a library of hardware and software components. Some of these components will be parametric (such as reconfigurable processors or ASIP) to further enlarge the design space for exploration. The library will be extensible; users will have the possibility to add new components, according to the evolution of both applications and technology. Open-PEOPLE is definitely an open project. The software platform for conducting estimation and optimization, will be accessible through an Internet portal. This software platform will be coupled to an automated hardware platform for physical measurements. The measurements needed to build models for new components to be added in the library will be remotely controlled through the software platform. A library of benchmarks will be proposed, to help building models for new components and architectures.

7.9. The ANR FAMOUS project

FAMOUS is a research project with an immediate industrial impact. Actually, it will make reconfigurable systems design easier and faster. The obtained tool in this project is expected to be used by both companies designers and academic researchers, especially for modern applications system specific design as smart camera, image and video processing ..etc. FAMOUS tools will be based on well established standards in design community. In fact, modeling will start from very high abstraction level using an extended version of MARTE. Simulation and synthesizable models will be obtained by automatic model to model transformations, using MDE approach. These techniques will contribute to shorten drastically time-to-market. FAMOUS is a basic research project. In fact, most of partners are academic (DaRT team of INRIA Lille Nord Europe, INRIA Rhone Alpes, Université de Bretagne Sud, Université de Bourgogne, Sodius), and its main objective is to explore novel design methodologies and target modern embedded systems architectures.

7.10. ID-TLM

In the framework of the collaboration between ST Micorelectronics and the INRIA, we are developping a collaboration (DGE ID TLM-NANO 2012, allocation n°3573) called ID-TLM between the team of Laurent Maillet-Contoz at ST Microelectronics and the AOSTE and DaRT INRIA teams.

The goal is to explore the contributions of the model driven engineering to the modeling and analysis of transaction level (TLM) models of computation (MoC). This collaboration has started on december 2008 and will last 4 years.

8. Other Grants and Activities

8.1. ARC INRIA - Triade

Partners: Aoste, DaRT, Espresso.

The goal of this project is to use formal models with structuring programmatic constructs as means to translate programs and descriptions written in formalisms widely used in Embedded System and SoC design (http://www.irisa.fr/espresso/Triade). Triade also aims to provide a seamless flow of increasingly time-defined and time-accurate models, so as to progressively obtain the final mapped implementation through provably correct steps from the early description elements.

8.2. 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.

8.3. Euromed 3+3 INRIA program

A collaboration supported by the Euromed 3+3 program started in 2009. It permits to initiate especially collaboration with the University of Monastir (Tunisia) with the team of Pr. Rached Torki on network on chip (NoC) modeling, evaluation and implementation. A co-advised PhD thesis is started on these issues between DaRT and U. Monastir.

8.4. EGID Ulysse Dublin program

Our collaboration with Pr. Tahar Kechadi (UC Dublin in Ireland) is supported by the EGID Ullyss program. It concerns mainly distributed systems communication, modeling and application specific multiprocessor architectures. Two master students have been co-advised on these topics for the moment.

8.5. International initiatives

8.5.1. 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 has been an ECSI member since 2004. Pierre Boulet is a member of the executive committee of ECSI and secretary of ECSI.

8.5.2. University of Oran

A collaboration has started with the university of Oran, Algeria. This collaboration with Abou El Hassan and his team has been active since 2006. We are working together on scheduling and mapping algorithms for SoC. An Egide-Tassili project has been proposed to strengthen this collaboration.

8.5.3. 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 focuses on NoC simulation, verification and design. Euromed 3+3 is one of the projects supporting this work.

We also initiate a collaboration with the IWT2 group of the University of Seville (Spain). Maria Jose Escalona and a student visited us. Our collaboration will be focused on testing model transformation using traceability.

8.5.4. Marte user group

Pierre Boulet has initiated the creation of the MARTE Users' Group (http://www.marte-ug.org/).

The MARTE Users' Group is a group of persons and institutions using, evaluating or developing tools for the Modeling and Analysis of Real Time and Embedded systems UML profile. This profile is a standard of the OMG but the MARTE Users' Group is independent of any organizations, including the OMG. It aims at providing a forum for discussing the usages of MARTE, identifying and disseminating the good practices of MARTE tools, influencing the standard and providing resources to MARTE users.

The first action of this group is to organise a workshop in 2010, the 1st Workshop on Model Based Engineering for Embedded Systems Design, March 12, 2010 - Dresden, Germany, in coordination with the DATE'2010 conference.

8.6. National initiatives

We are members of the ASR[5] and SoC-SiP[6] GDRs (research groups from the CNRS).

We are members of the Action IDM: a transverse group gathering teams of the ASR, GPL[7] and I3[8] GDRs.

9. Dissemination

9.1. Scientific Community

Pierre Boulet has been in the steering committee and the program committee of FDL since 2005. He is the UML/UMES topic program chair of FDL'07 to FDL'10. He has been in the program committee of the first workshop on the definition, evaluation, and exploitation of modeling and computing standards for Real-Time Embedded Systems (STANDRTS'09, Dublin, Ireland). He is in the program committee of the DATE'2010 conference. He has initiated the MARTE user group and is co-organizer of the first workshop on Model Based Engineering for Embedded Systems in 2010 (co-located with DATE'2010). He was in several PhD thesis committees in 2009. He has been deputy director of the LIFL since January 2008.

Jean-Luc Dekeyser was in the program committee of the following conferences and workshops: DSD'2009, SoC'2009, ECMDA'2009, AICCSA'2009, ReCoSoC'2009 and IDT'2009. He has invited speaker in FETCH'2009 and JFMMA'2009. He was in several PhD thesis committees in 2009. 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'10.

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".

[5] http://asr.cnrs.fr/
[6] http://www.lirmm.fr/soc_sip/
[7] http://gdr-gpl.cnrs.fr/
[8] http://www.cnrs.fr/ins2i/

The following internships were advised in the team:

- Amira Hasnaoui, Master, faculté des sciences de Tunis, Tunisie.
- Chiraz Trabelsi, Master, INSAT - Tunis, Tunisie.
- Amen Souissi, Université des Sciences et Technologies de Lille
- Hoa Nguyen Viet
- Térence Mulaja
- Lamia Souissi, Master, Mohammadia School - Rabat, Marroco
- Arnaud Ysma, Université des Sciences et Technologies de Lille
- Lahbib Beddou
- Sana Cheri
- Vlad Ureche

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 IE US .

[2] P. BOULET, C. DUMOULIN, A. HONORÉ. *Model Driven Engineering for System-on-Chip Design*, in "From MDD concepts to experiments and illustrations", S. GÉRARD, J.-P. BABEAU, J. CHAMPEAU (editors), n^{o} 6, ISTE, September 2006, http://iste.co.uk/index.php?f=a&ACTION=View&id=147, ISBN:1-905209-59-0.

[3] 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.

[4] 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", vol. 2008, july 2008, Article ID 561863.

[5] C. GLITIA, P. DUMONT, P. BOULET. *Array-OL with delays, a domain specific specification language for multidimensional intensive signal processing*, in "Multidimensional Systems and Signal Processing", 2009, http://springerlink.com/content/w382176038114432/?p=fc0a4428f2f4468a9d630d2a434a6f69&pi=0.

[6] J. KHAN, S. NIAR, A. MENHAJ, Y. EL-HILLALI, 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", September 2006, (Also appeared as an IEEE ITSS Newsletter in December 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 BE .

[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] C. GLITIA. *Optimisation des applications de traitement systématique intensives sur System-on-Chip*, Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, Lille, France, November 2009, in french, Thèse de doctorat (PhD Thesis).

[13] J. TAILLARD. *Une approche orientée modèle pour la parallélisation d'un code de calcul éléments finis*, Laboratoire d'informatique fondamentale de Lille, Université des sciences et technologies de Lille, Lille, France, February 2009, in french, Thèse de doctorat (PhD Thesis).

Articles in International Peer-Reviewed Journal

[14] C. BRUNETTE, J.-P. TALPIN, A. GAMATIÉ, T. GAUTIER. *A Metamodel for the Design of Polychronous Systems*, in "Journal of Logic and Algebraic Programming - JLAP, Special Issue on Applying Concurrency Research to Industry", January 2009.

[15] R. GAIGNAIRE, F. GUYOMARC'H, O. MOREAU, S. CLENET, B. SUDRET. *Speeding Up SSFEM Computation Using Kronecker Tensor Products*, in "IEEE TRANSACTIONS ON MAGNETICS", vol. 45, n^{o} 3, March 2009, p. 1432–1435.

[16] A. GAMATIÉ, É. RUTTEN, H. YU, P. BOULET, J.-L. DEKEYSER. *Model-Driven Engineering and Formal Validation of High-Performance Embedded Systems*, in "Scalable Computing: Practice and Experience (SCPE)", vol. 10, n^{o} 2, 2009.

[17] C. GLITIA, P. DUMONT, P. BOULET. *Array-OL with delays, a domain specific specification language for multidimensional intensive signal processing*, in "Multidimensional Systems and Signal Processing", 2009, http://springerlink.com/content/w382176038114432/?p=fc0a4428f2f4468a9d630d2a434a6f69&pi=0.

[18] K. IBRAHIM, S. NIAR. *Power-aware Bus Coscheduling for Periodic Realtime Applications Running on MPSoC*, in "Transactions on High-Performance Embedded Architecture and Compilation HiPeac", vol. 2, April 2009, p. 286-306.

[19] J. KHAN, S. NIAR, M. SAGHIR, Y. EL-HILLALI, A. RIVENQ. *Trade-off Exploration for Target Tracking Application in a Customized Multiprocessor Architecture*, in "EURASIP Journal on Embedded Systems", vol. 2009, November 2009.

[20] I. R. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *A Model based design flow for Dynamic Reconfigurable FPGAs*, in "International Journal of Reconfigurable Computing", 2009.

Articles in National Peer-Reviewed Journal

[21] A. ETIEN. *Business process/information system co-evolution*, in "Ingénierie des Systèmes d'Information (ISI) Special issue on Evolution des systèmes d'information", vol. 14, n^{o} 6, 2009.

International Peer-Reviewed Conference/Proceedings

[22] A. ABDALLAH, A. GAMATIÉ, J.-L. DEKEYSER. *Model-Driven Design of Embedded Multimedia Applications on SoCs*, in "12th Euromicro Conference on Digital System Design (DSD2009), Patras, Greece", August 2009.

[23] V. ARANEGA, J.-M. MOTTU, A. ETIEN, J.-L. DEKEYSER. *Traceability Mechanism for Error Localization in Model Transformations*, in "4th International Conference on Software and Data Technologies (ICSOFT'2009), Sofia, Bulgaria", July 2009.

[24] M. BAKLOUTI, P. MARQUET, M. ABID, J.-L. DEKEYSER. *Study and Integration of a Parametric Neighbouring Interconnection Network in a Massively Parallel Architecture on FPGA*, in "The 7th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-2009), Rabat, Morocco", may 2009 TN .

[25] H. CHTIOUI, R. BEN ATITALLAH, S. NIAR, M. ABID, J.-L. DEKEYSER. *A Dynamic Hybrid Cache Coherency Protocol for Shared-Memory MPSoCs*, in "12th Euromicro Conference on Digital System Design (DSD2009), Patras, Greece", August 2009 TN .

[26] J.-L. DEKEYSER, I. R. QUADRI, A. GAMATIÉ. *Tutorial: Using the UML profile for MARTE to MPSoC codesign dedicated to signal processing*, in "Colloque International Télécom 2009 et 6èmes Journées JFMMA, Agadir, Morocco", March 2009.

[27] A. GAMATIÉ, H. YU, G. DELAVAL, É. RUTTEN. *A Case Study on Controller Synthesis for Data-Intensive Embedded Systems*, in "6th IEEE Int. Conference on Embedded Systems and Software (ICESS'09), Hangzhou, China", May 2009.

[28] C. GLITIA, P. BOULET. *Interaction between inter-repetition dependences and high-level transformations in Array-OL*, in "Conference on Design and Architectures for Signal and Image Processing (DASIP 2009), Sophia Antipolis, France", September 2009.

[29] N. HARB, S. NIAR, J. KHAN, M. SAGHIR. *A Reconfigurable Platform Architecture for an Automotive Multiple-Target Tracking System*, in "Proc. 2nd Workshop on Adaptive and Reconfigurable Embedded Systems (APRES'2009), in conjunction with Esweek (CASES'09, CODES+ISSS'09, EMSOFT'09), Grenoble, France", 2009.

[30] J. KHAN, S. NIAR, A. MENHAJ, Y. EL-HILLALI. *Radar Based Collision Avoidance System Implementation in a Reconfigurable MPSoC*, in "Proc. 9th International Conference on ITS Telecommunications (ITST), Lille, France", 2009.

[31] I. R. QUADRI, Y. EL-HILLALI, S. MEFTALI, J.-L. DEKEYSER. *Model based design flow for implementing an Anti-Collision Radar system*, in "9th International IEEE Conference on ITS Telecommunications (ITS-T 2009), Lille, France", October 2009.

[32] I. R. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *Integrating Mode Automata Control Models in SoC Co-Design for Dynamically Reconfigurable FPGAs*, in "International Conference on Design and Architectures for Signal and Image Processing (DASIP 09), Nice, France", September 2009.

[33] I. R. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *MARTE based design approach for targeting Reconfigurable Architectures*, in "2nd International Conference on Embedded Systems (ESC'09), Alger, Algeria", May 2009, Invited Paper.

[34] I. R. QUADRI, A. MULLER, S. MEFTALI, J.-L. DEKEYSER. *MARTE based design flow for Partially Reconfigurable Systems-on-Chips*, in "17th IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC 09), Florianapolis, Brazil", October 2009.

[35] W. RODRIGUES, F. GUYOMARC'H, J.-L. DEKEYSER, Y. LE MENACH. *Parallel Sparse Matrix Solver on the GPU Applied to Simulation of Electrical Machines*, in "Compumag 2009", november 2009.

[36] V. RUSU. *Formal Executable Semantics for Conformance in the MDE Framework*, in "Proc. 2nd UML & FM Workshop", Artist NoE, 2009.

[37] A. TINZEFTE, Y. LE MENACH, J. KORECKI, F. GUYOMARC'H. *Parallel Direct Solver For The Finite Integration Technique in Electrokinetic Problems*, in "Compumag, Florianapolis, Brazil", November 2009.

[38] R. TLIGUE, Y. AYDI, M. BAKLOUTI, M. ABID, J.-L. DEKEYSER. *The design methodology and the implementation of MPSOC based on Delta MINs on FPGA*, in "The 21th IEEE International Conference on Microelectronics (ICM'2009), arrakech, Morocco", 2009 TN .

National Peer-Reviewed Conference/Proceedings

[39] A. ABDALLAH, A. GAMATIÉ, J.-L. DEKEYSER. *Modélisation UML/MARTE de SoC et analyse temporelle basée sur l'approche synchrone*, in "SYMPosium en Architecture de machines (SympA'13), Toulouse, France", September 2009.

Scientific Books (or Scientific Book chapters)

[40] A. GAMATIÉ. *Designing Embedded Systems with the SIGNAL Programming Language: Synchronous, Reactive Specification*, Springer Verlag, New York, 2009.

Research Reports

[41] C. GLITIA, P. BOULET, É. LENORMAND, M. BARRETEAU. *Repetitive Model Refactoring Strategy for the Design Space Exploration of Intensive Signal Processing Applications*, n^{o} number not yet available, INRIA, September 2009, extended version, Technical report.

[42] I. R. QUADRI, S. MEFTALI, J.-L. DEKEYSER. *From MARTE to dynamically reconfigurable FPGAs : Introduction of a control extension in a model based design flow*, n^{o} 6862, INRIA, France, March 2009, http://hal.inria.fr/inria-00365061/en/, Research Report.

References in notes

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

[44] *Acceleo*, 2009, http://www.acceleo.org.

[45] 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.

[46] 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.

[47] C. ANDRÉ, F. MALLET, R. DE SIMONE. *Time Modeling in MARTE*, in "ECSI Forum on specification & Design Languages (FDL), Barcelona Espagne", ECSI, 2007, p. 268-273, http://hal.inria.fr/inria-00204481/en/.

[48] 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.

[49] 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.

[50] 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.

[51] 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.

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

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

[54] 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.

[55] 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.

[56] 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.

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

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

[59] P. DUMONT. *Étude des Transformations d'un Code Array-OL dans Gaspard*, 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, Research Report.

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

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

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

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

[64] 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.

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

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

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

[68] 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.

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

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

[71] 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.

[72] 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/.

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

[74] J.-M. MOTTU, B. BAUDRY, Y. LE TRAON. *Mutation Analysis Testing for Model Transformations*, in "proceedings of the European Conference on Model Driven Architecture (ECMDA 06), Bilbao, Spain", July 2006.

[75] OBJECT MANAGEMENT GROUP, INC.. *Meta Object Facility (MOF) Core Specification, Version 2.0*, January 2006, http://www.omg.org.

[76] OBJECT MANAGEMENT GROUP, INC.. *MOF Query / Views / Transformations*, July 2007, http://www.omg.org, OMG paper.

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

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

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

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

[81] 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, to appear.

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

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

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