INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team distribcom

Distributed Models and Algorithms for the Management of Telecommunication Systems

Rennes - Bretagne Atlantique

THEME COM

Activity Report

2007

Table of contents

1. Team **1**
2. Overall Objectives **1**
2.1. Objectives of the team 1
2.2. Highlights 2
3. Scientific Foundations **2**
3.1. Overview of the needed paradigms 2
3.2. Models of concurrency: nets, scenarios, event structures, graph grammars, and their variants 3
3.2.1. Scenarios. 3
3.2.2. Event structures. 3
3.2.3. Nets and languages of scenarios. 3
3.2.4. Extensions and variants. 4
3.2.5. Handling dynamic changes in the systems: graph grammars. 4
4. Application Domains **4**
4.1. Network and service management in telecommunications 4
4.2. Web services and workflow management 4
5. Software **5**
6. New Results **5**
6.1. User oriented modeling 5
6.2. Fundamentals of modeling 6
6.2.1. Event structures and unfoldings for distributed systems monitoring. 6
6.2.2. Probabilistic event structures and Markov nets. 7
6.2.3. Scenarios 7
6.3. Algorithms: distributed testing and diagnosis 8
6.4. Algorithms for Scenarios 9
6.5. Specific studies: cross-domain services in heterogeneous networks 9
6.6. Specific studies: Web services orchestrations 10
6.7. Specific studies: Active XML documents 10
7. Other Grants and Activities **11**
7.1. ANR Docflow 11
7.2. CREATE ActiveDoc 11
7.3. ANR Dots 11
7.4. Politiques de sécurité: TEST et Analyse par le Test de systèmes en réseau ouvert 12
7.5. Control, Analysis and Synthesis of Distributed Systems 12
8. Dissemination **12**
8.1. Scientific animation 12
8.2. Teaching 13
8.3. Visits and invitations 13
9. Bibliography **14**

1. Team

Head of project–team
Albert Benveniste [DR INRIA, part–time, HdR]
Claude Jard [PROF. ENS CACHAN , HdR]

Administrative assistant
Laurence Dinh

Research scientist (INRIA)
Éric Fabre [CR , HdR]
Stefan Haar [CR, on sabbatical at University of Ottawa]
Loïc Hélouët [CR]

Research scientist (CNRS)
Blaise Genest [CR]

Faculty member (ENS Cachan)
Anne Bouillard [assistant professor]

Technical staff (INRIA)
Deepak Bhatia [Ingénieur Associé, since september 14th]

Guests

Post–doctoral fellow (INRIA)
Shaofa Yang

Ph.D. student
Thomas Gazagnaire [Moniteur Normalien]
Debmalya Biswas [INRIA grant]
Hélia Poullyau [INRIA grant]
Sidney Rosario [INRIA grant]
Didier Devaurs [MENRT fellowship, ending October 1st]
Bartosz Grabiec [INRIA grant]

2. Overall Objectives

2.1. Objectives of the team

Keywords: *Orchestrations*, *Quality of Service*, *Web services*, *distributed algorithms*, *distributed testing*, *fault management*, *self-management*, *telecommunications*.

The DistribCom team addresses models and algorithms for distributed network and service management, and the distributed management of Web services and business processes.

Today, research on network and service management as well as Web Services mainly focuses on issues of software architecture and infrastructure deployment. However, these areas also involve algorithmic problems such as fault diagnosis and alarm correlation, testing, QoS evaluation, negotiation, and monitoring. The DistribCom team develops the foundations supporting such algorithms. Our algorithms are model-based. Our research topics are therefore structured as follows:

1. *Fundamentals of distributed observation and supervision of concurrent systems*: this provides the foundations for deriving models and algorithms for the above mentioned tasks.
2. *Self-modeling*: for obvious reasons of complexity, our models cannot be built by hand. We thus address the new topic of self-modeling, i.e., the automatic construction of models, both structural and behavioral.
3. *Algorithms for distributed management of telecommunications systems and services.*
4. *Web Services orchestrations, functional and QoS aspects.*
5. *Active XML peers for Web scale data and workflow management.*

Our main industrial ties are with Alcatel-Lucent, on the topic of networks and service management.

2.2. Highlights

- Two important events in 2007 have concluded the work of the team in the area of distributed monitoring and event correlation: Eric Fabre got his *Habilitation (HdR)* [10], and co-authored, with Albert Benveniste, an important paper [13] for the special issue of JDEDS journal collecting selected papers from Wodes'06, where Albert Benveniste was plenary speaker.
- Paper [24] on Causal Message Sequence Charts (a model of scenarios) at Concur'07 was an important follow-up of the INRIA associated team established with Prof. P.S. Thiagarajan, from National University of Singapore.

For this year, we closed our joint industrial applications with Alcatel and opened new application areas on Web services. We thus have no highlight to report regarding applications.

3. Scientific Foundations

3.1. Overview of the needed paradigms

Keywords: *asynchronous system*, *distributed system*, *monitoring*, *quality of service*, *testing*.

Management of telecommunications networks and services, and Web services, involves the following algorithmic tasks:

Observing, monitoring, and testing large distributed systems: Alarm or message correlation is one of the five basic tasks in network and service management. It consists in causally relating the various alarms collected throughout the considered infrastructure—be it a network or a service sitting on top of a transport infrastructure. Fault management requires in particular reconstructing the set of all state histories that can explain a given log of observations. Testing amounts to understanding and analyzing the responses of a network or service to a given set of stimuli; stimuli are generally selected according to given test purposes. All these are variants of the general problem of *observing* a network or service. Networks and services are large distributed systems, and we aim at observing them in a distributed way as well, namely: logs are collected in a distributed way and observation is performed by a distributed set of supervising peers.

Quality of Service (QoS) evaluation, negotiation, and monitoring: QoS issues are a well established topic for single domain networks or services, for various protocols — e.g., Diffserv for IP. Performance evaluation techniques are used that follow a "closed world" point of view: the modeling involves the overall traffic, and resource characteristics are assumed known. These approaches extend to some telecommunication services as well, e.g., when considering (G)MPLS over an IP network layer.

However, for higher level applications, including composite Web services (also called *orchestrations*, this approach to QoS is no longer valid. For instance, an orchestration using other Web services has no knowledge of how many users are calling the same Web services. In addition, it has no knowledge of the transport resources it is using. Therefore, the well developed "closed world" approach can no longer be used. *Contract* based approaches are considered instead, in which a given orchestration offers promises to its users on the basis of promises it has from its subcontracting services. In this context, contract composition becomes a central issue. Monitoring is needed to check for possible breaching of the contract. Coutermeasures would consist in reconfigurating the orchestration by replacing the failed subcontracted services by alternative ones.

The DistribCom team focuses on the algorithms supporting the above tasks. Therefore models providing an adequate framework are fundamental. We focus on models of discrete systems, not models of streams or fluid types of models. And we address the distributed and asynchronous nature of the underlying systems by using models involving only local, not global, states, and local, not global, time. These models are reviewed in section 3.2. We use these mathematical models to support our algorithms and we use them also to study and develop formalisms of Web services orchestrations and workflow management in a more general setting.

3.2. Models of concurrency: nets, scenarios, event structures, graph grammars, and their variants

Keywords: *Models of concurrency*, *event structures*, *graph grammars*, *nets*, *scenarios*.

For Finite State Machines (FSM), a large body of theory has been developed to address problems such as: observation (the inference of hidden state trajectories from incomplete observations), control, diagnosis, and learning. These are difficult problems, even for simple models such as FSM's. One of the research tracks of DistribCom consists in extending such theories to distributed systems involving concurrency, i.e., systems in which both time and states are local, not global. For such systems, even very basic concepts such as "trajectories" or "executions" need to be deeply revisited. Computer scientists have for a long time recognized this topic of concurrent and distributed systems as a central one. In this section, we briefly introduce the reader to the models of scenarios, event structures, nets, languages of scenarios, graph grammars, and their variants.

3.2.1. Scenarios.

The simplest concept related to concurrency is that of a finite execution of a distributed machine. To this end, scenarios have been informally used by telecom engineers for a long time. In scenarios, so-called "instance" exchange asynchronous messages, thus creating events that are totally ordered on a given instance, and onyl partially ordered by causality on different instances (emission and reception of a message are causally related). The formalization of scenarios was introduced by the work done in the framework of ITU and OMG on High-level Message Sequence Charts and on UML Sequence Diagrams in the last ten years, see [66], [75]. This allowed in particular to formally define infinite scenarios, and to enhance them with variables, guards, etc [80], [78], [79]. Today, scenarios are routinely offered by UML and related systems and software modeling tools.

3.2.2. Event structures.

The next step is to model sets of finite executions of a distributed machine. *Event structures* were invented by Glynn Winskel and co-authors in 1980 [74], [81]. Executions are sets of events that are partially ordered by a *causality* relation. Event structres collect all the executions by superimposing shared prefixes. Events not belonging to a same execution are said in *conflict*. Events that are neither causally related nor in conflict are called *concurrent.* Concurrent processes model the "parallel progress" of components. Categories of event structures have been defined, with associated morphisms, products, and co-products, see [82]. Products and co-products formalize the concepts of parallel composition and "union" of event structures, respectively. This provides the needed apparatus for composing and projecting (or abstracting) systems. Event structures have been mostly used to give the semantics of various formalisms or languages, such as Petri nets, CCS, CSP, etc [74], [81]. We in DistribCom make a nonstandard use of these, e.g., we use them as a structure to compute and express the solutions of observation or diagnosis problems, for concurrent systems.

3.2.3. Nets and languages of scenarios.

The next step is to have finite representations of systems having possibly infinite executions. In DistribCom, we use two such formalisms: *Petri nets* [77], [58] and *languages of scenarios* such as High-level Message Sequence Charts (HMSC) [66], [79]. Petri nets are well known, at least in their basic form, we do not introduce them here. We use so-called *safe* Petri Nets, in which markings are boolean (tokens can be either 0 or 1); and we use also variants, see below.

3.2.4. Extensions and variants.

Two extensions of the basic concepts of nets or scenario languages are useful for us. Nets or scenario languages enriched with variables, actions, and guards, are useful to model general concurrent and distributed dynamical systems in which a certain discrete abstraction of the control is represented by means of a net or a scenario language. Manipulating such *symbolic nets* requires using abstraction techniques. Time Petri nets and network of timed automata are particular cases of symbolic nets. Probabilistic Nets or event structures: Whereas a huge literature exists on stochastic Petri nets or stochastic process algebras (in computer science), randomizing *concurrent models,* i.e., with ω's being concurrent trajectories, not sequential ones, has been addressed only since the 21st century. We have contributed to this new area of research.

3.2.5. Handling dynamic changes in the systems: graph grammars.

The last and perhaps most important issue, for our applications, is the handling of dynamic changes in the systems model. This is motivated by the constant use of dynamic reconfigurations in management systems. Extensions of net models have been proposed to capture this, for example the *dynamic nets* of Vladimiro Sassone [57] and *net systems* [59]. For the moment, such models lack a suitable theory of unfoldings. A relevant alternative is the class of *graph grammars* [49], [56]. Graph grammars transform graphs by means of a finite set of rules.

4. Application Domains

4.1. Network and service management in telecommunications

Keywords: *QoS monitoring and negotiation*, *failure diagnosis*, *telecommunication transport network*.

Telecommunications have grown up from a basic technology of networks and transport to a much more complex jungle of networks, services, and applications. This motivates a strong research effort towards "autonomic management systems". While much effort has been devoted to architecture aspects of this topic, less has been done on functions to be performed for this, and supporting algorithms. Related problems share several features. First of all, they involve concurrent systems, *i.e.* systems where several things can happen independently at the same time. Secondly, they are built in a modular way, by combining elementary components into large connected structures. Thirdly, these systems exhibit dynamicity. Reconfigurations, connections/deconnections of new components or clients are part of the normal activity, and should not require that monitoring algorithms be reset or modified each time the system is changed. Finally, the size and heterogeneity of these systems prevents from using a centralized monitoring architecture. This motivates developing distributed and modular approaches.

One example is diagnosis in transport networks, i.e. the low-level layers of networks (physical, transport and network layers). Consider for example circuit oriented networks, such as SDH/WDM or GMPLS. Corresponding management systems assemble hundreds of functions or components, and the failure of one of them generally induces side-effects in many others. This phenomenon is known as "fault propagation;" it results in hundreds of alarms produced by the various components and collected at different locations in the network. Identifying origins of faults from these alarms has now reached a level of complexity that prevents the traditional human analysis of alarms. Due to the size of systems, the automatic diagnosis task cannot be done in a centralized manner, and must be solved by a network of local supervisors that coordinate their work to provide a coherent view of what happened in the network.

Another example of problem is the handling of QoS in cross-domain heterogeneous networks and services. The emphazis is on guaranteeing desired levels of QoS in situations where SLAs have to be negotiated, instantiated, and monitored in a non-local fashion.

4.2. Web services and workflow management

Keywords: *Active XML*, *QoS*, *Web services*, *choreographies*, *orchestrations*.

When considering transaction oriented Web services (such as used in business processes), the quest for flexibility motivates the development of techniques and tools to rapidly assemble Web-services into larger services, called *orchestrations* or *choreographies*. Recently, standard languages for service workflow have even been proposed such as IBM's Web Services Flow Language [50] or Microsoft's XLang [51], which converged to the BPEL4WS proposal [48] and subsequently WSCDL proposal [67] for choreographies. Tools for BPEL are, among others, commercially available from Telelogic (now IBM)[1] and Oracle[2]. The implementation of orchestration and choreography description languages raises a number of difficulties related to efficiency, clean semantics, and reproducibility of executions. In addition, issues of composite QoS associated with orchestrations are not a mature area [83]. We develop studies in these areas.

A serious shortcoming of approaches to Web Service orchestration and choreography is that they mostly abstract data away. Symmetrically, modern approaches to Web data management [64], [84] typically based on XML and Xqueries rely on too simplistic forms of control. We believe that time has come for a convergence of sophistication in terms of control and richness in data, for workflow and data management over the Web. We believe that *active Peer-to-Peer XML-based documents*, as proposed by S. Abiteboul under the name of AXML [3] provide the basis for an adequate infrastructure for this. We cooperate with S. Abiteboul and A. Muscholl at lifting AXML to handle workflow management as well, thus providing a document oriented alternative to Web services orchestrations and choreographies.

5. Software

5.1. SOFAT : a scenario toolbox

Keywords: *partial orders*, *scenarios*.

Participants: Loïc Hélouët, Deepak Bhatia.

The SOFAT toolbox is a scenario manipulation toolbox. Its aim is to implement all known formal manipulations on scenarios. The toolbox implements several formal models such as partial orders, graph grammars, graphs, and algorithm dedicated to these models (Tarjan, cycle detection for graphs, Caucal's normalization for graph grammars, etc.). The SOFAT toolbox is permanently updated to integrate new algorithms. It is freely available form INRIA's website. The last update of SOFAT includes the fibered product operation described in [68]. This year SOFAT has been adapted to serve as a frontend for a toolchain that derives performance models from annotated functional models. D. Bhatia is currently redeveloping SOFAT to improve and distribute it. The obtained software will be a demonstrator and a support for all our proposals in standardization committees at ITU. This involvement is standardization is also the occasion for numerous contacts with MSC users (France Telecom, Nokia, Motorola), but also with tool designer that build case tools handling ITU languages (fromer Telelogic now part of IBM). We hope that when SOFAT becomes robust enough, this situation may creates industrial transfer opportunities.

6. New Results

6.1. User oriented modeling

Keywords: *Automatic abstraction*, *Self-modeling*, *learning*, *scenario weaving*, *testing*.

Participants: Claude Jard, Eric Fabre, Loïc Hélouët, Blaise Genest, Thomas Gazagnaire, Shaofa Yang.

[1] http://www.telelogic.com/standards/bpel.cfm
[2] http://www.oracle.com/appserver/bpel_home.html
[3] http://activexml.net

The emerging topic of *self-modeling* addresses the automatic construction of sophisticated behavioral models. This problem is a real challenge for large systems, and an unavoidable and delicate task that directly affects the performances of model-based monitoring tools. We address self-modeling issues in two different ways: by assembling generic model blocs, and by learning methods.

The first approach has been developed in previous RNRT projects, dedicated to alarm correlation techniques in telecommunication networks, and is now part of the team background. The reader is referred to the activity report of year 2006. These last three years, effort has be shifted toward the second approach, by automatically inferring or refining (part of) the model with tests and learning algorithms (see the 2006 activity report for this last topic).

This year, we have investigated several approaches to provide users with languages and tools to build expressive models more easily. The first step in this direction proposed in 2006 [69] was to weave behavioral aspects in scenario models. Aspects are a new approach to separate different concerns of software, and, more recently, models. The work of 2006 defines behavioral aspects in MSCS as a pattern rewriting for scenarios. This year, we have defined a expressive variant of MSCs called Causal Message Sequence Charts [24], [42]. This variant allows for the definition of typical behaviors of distributed systems called sliding windows, that could not be modelled using HMSCs. Furthermore, for this new language, the diagnosis algorithms proposed in [65] remain decidable for causal MSCs. This is an important property, as behaviors defined by Causal HMSCs are almost the traces of implemented systems, the accuracy of model-based diagnosis is increased.

Another research direction consists in helping a user collecting and gathering observations of a system to build a coherent model. This year, we have proposed a new approach to composition [43]. The main idea of this work is to define separate view of a system, and to combine them by synchronizing common events in both views. We have studied the limits of this approach, namely common synchronized events must be located on a single process, and more important, we have shown that when the composed specifications are implementable, it is decidable whether their product is implementable.

6.2. Fundamentals of modeling

Keywords: *category theory*, *concurrent systems*, *distributed algorithms*, *event structures*, *probabilistic event structures*, *unfoldings*.

Participants: Albert Benveniste, Claude Jard, Eric Fabre, Stefan Haar, Loïc Hélouët, Blaise Genest.

In this section we collect our fundamental results regarding the models we use for distributed systems.

6.2.1. Event structures and unfoldings for distributed systems monitoring.

The monitoring algorithms developed in Distribcom heavily rely on an efficient representation of trajectory sets for large concurrent systems. Unfoldings, event structures, or the more recently proposed time-unfoldings are natural candidates for that. A key feature of our approach is the use of the so-called factorization property of these compact trajectory representations. Specifically, when a system can be expressed as a combination of components, its (time-)unfolding can be expressed as well as the combination of the (time-)unfoldings of its components. This property automatically comes from category theory arguments as soon as the “combination” of components can expressed as a limit. Like, for example, the composition by a standard synchronous product, or more interestingly like the composition by a pullback, which expresses that components interact via an interface.

Distributed monitoring algorithms can be derived from a formal analogy between compound dynamic systems and Markov random fields. The factorization property of a dynamic system, inherited by its (time-)unfolding, is the counterpart of the factorization property of the probability distribution of a Markov field. The conditional independence also has an equivalent form for dynamic systems, which allows us to recycle standard Bayesian estimation procedures into distributed inference algorithms for compound dynamic systems. The latter take the form of message-passing algorithms, working at the scale of each component, and based on two simple operations: projections (on a component) and combinations, either by product or by pullback. This formal connection is detailed in E. Fabre’s habilitation [61].

Distributed computations based on event structures, on branching processes (prefixes of unfoldings) or on trellis processes (prefixes of time-unfoldings) are quite technical. This is essentially due to the specific features of the true concurrency semantics. We have done the exercise of re-expressing the theory in a simpler framework, where runs are ordinary sequences of events rather than partial orders. All results can be rederived nicely: one can design distributed monitoring algorithms for networks of automata in the same way as for combinations of Petri nets. Several surprises came up however. In particular, trellis processes enjoy the necessary properties only if they are defined with respect to a local notion of time (time elapses differently in each component), instead of a global notion of time. This strongly suggests that one should avoid using a global clock if distributed computations are desired. And once local clocks are necessary, one is not far from true concurrency semantics [60].

Unfoldings are traditionally used for model checking purposes, under the form of finite and complete prefixes (that contain all reachable states of a Petri net). However, the factorization property of unfoldings was never used in this context, with the aim, for example, to solve reachability problems in a modular manner. In collaboration with Agnes Madalinski (postdoc in 2006), we have shown that it is possible to obtain a complete prefix of a modular system directly in factorized form, i.e. without computing first a complete prefix of the global system and then projecting it on its components. The construction uses again a variant of message passing algorithms. This was presented at the UFO workshop [70], and a more detailed publication of this preliminary work is in preparation.

6.2.2. Probabilistic event structures and Markov nets.

Our work on *true concurrency probabilistic models* is joint work with our former PhD student Samy Abbes, now Maître de Conférences in Mathematics at Paris VI, PPS Laboratory. We have established the foundations for probabilistic models of distributed and concurrent systems, in which traces, not interleaving sequences, are randomized. The work of this year has consisted in finalizing papers and getting one new result. We review our progresses and refer the reader to the 2004-2006 activity reports for the motivations and previous results. Regarding ongoing publications, our long paper on Markov nets, their renewal theory and associated Law of Large Numbers has been accepted for publication in the special issue of *Theoretical Computer Science* devoted to selected papers of FOSSACS'05 [53]. Then, this year, we have got our first result regarding true concurrency products of stochastic processes. Surprisingly, no probabilistic counterpart exists of the very basic synchronous product of automata, where two automata synchronize on their shared actions. We have provided such a construction for two Markov Chains, with the result being a Markov net. This work is submitted to a conference.

6.2.3. Scenarios

A major challenge with scenarios is to model distributed systems while preserving the decidability of some problems. In their simplest version, scenarios can be seen as automata labeled by partial orders that aim at describing non-interleaved execution traces of distributed systems, i.e. families of partial orders. This simple model already embed the expressive power of Mazurkiewicz traces. This results in undecidability of several problems such as vacuity of families intersection, equality, and so on [73]. Fortunately, several subclasses of HMSCs and CHMSCs allowing for the decision of some problems have been identified, and some interesting questions, such as diagnosis remain decidable [65]. Fundamental research on scenarios has followed two main directions. The first direction is the comparison of the expressivity of different models to understand their power. The second direction is to give ways of handeling partial orders, which allows us to understand them better.

A major aspect of our work is to compare different modeling formalisms to know what can and cannot be described with a particular formalism. Last year, we showed in [63] that many existentially bounded scenario formalisms (HMSC, communicating finite state machines, interleaving languages and second order logic), i.e. for which there exists an upper bound on the contents of the message channels within which all scenarios in the language *can* be run are equivalent. That is, they behave like bounded formalisms with respect to expressivity. This is important since if a system is B-bounded, all its executions are B-bounded, and in particular it is existentially B-bounded. That is, when analyzing a system, many executions can be forgotten while giving the same result as if every execution was considered. This year, we studied the question to know when a given system is existentially bounded, and if then obtain a bound [15]. This question can be solved for the same models as in the bounded case, but the proofs are more involved to obtain. However, the algorithms are not more time consuming, and one can often obtain a faster answer due to a smaller bound. A perspective is to use such results in control, and more particularly in quasi static scheduling.

Another aspect of our work was extensions of scenarios. Indeed, HMSCs are not able to model typical behaviors of so-called sliding windows. When a protocol implements a sliding window, distributed executions may have the shape of infinite braids. Partial order automata only allow for the description of recognizable MSC languages [72], i.e., families of partial orders that can be build by concatenation of a finite number orders from a finite order alphabet. To extend the expressivity of order automata, several solutions have been proposed. E. Gunter proposed an extension called compositional MSC, that embeds into order automata the expressive power of communicating automata. Unsurprisingly, several decidable problems that were decidable for order automata (and among them diagnosis) become undecidable for compositional MSCs. The results published this year in [62] propose an extension of order automata called causal MSCs that allows some commutation among composed orders. This new scenario model allows for the modeling of sliding windows protocols, embed the expressive power of order automata, and more interesting, every known subclass of HMSCs and CHMSCs find a counterpart with the same decidable properties in Causal HMSCs. Furthermore, diagnosis remains decidable on this model. Quite surprisingly, most of decidability results that were established for HMSCs or CHMSCs rely on the existence of a bound on the contents of communication channels, but decidable subclasses of causal HMSCs do not necessarily describe existentially bounded families of partial orders.

The last result of the year concerns which executions of a partial orders have to be considered in order to have all the 'relevant' executions. It is known that when a system is analyzed in Breast First Search, unfoldings of partial orders or representative interleavings (as the lexicographically first interleaving representing a scenario) can be used. This year, Esparza et al. showed that unfolding cannot be used as is in Depth First Search without the risk of forgeting an important information. We show a similar result (but with a very different proof) in [19], that is, lexicographic representatives of interleavings cannot be used in Depth First Search. However, we show in [19] that it suffices to expands slightly the set of representatives in order to have all the useful informations. That is, one representative per scenario is not enough when Depth First Search is used, but we do not need all the interleavings either. This result might be useful for understanding better unfoldings.

6.3. Algorithms: distributed testing and diagnosis

Keywords: *Concurrent testers*, *Partial order automata*, *Testing*.

Participants: Claude Jard, Emmanuel Donin, Stefan Haar.

After having developed theory and algorithms for computing time trajectories of distributed models [29] (based on T. Chatain's thesis of last year), we focussed our work in 2007 on testing in distributed environment. We feel that distributed observation, diagnosis and testing are the three facets of the future (self-) management of distributed systems and networks, closing the control loop paradigm. Our work has been performed in three complementary directions, all using partial order representations of time behaviours:

- The first aspect is to be able to automatically generate test scenarios (defined as sequence diagrams or MSCs) from high-level executable models (defined as communicating UML state charts). The method [16] is based on classical model-checking and has been invented and implemented during a collaboration with the Vertecs and Triskell research groups at Irisa.
- The second aspect is a result of a collaboration with the France Telecom research center in Lannion during the developpement of a new testing language called Late, in the context of the Emmanuel Donin's thesis [9]. We proposed a new kind of non-deterministic testing. One testing scenario can describe several different executions and the interpreter tries to find the execution that well fit with the real behavior of the System Under Testing.

- In the last aspect, we proposed an extension of the Finite State Machine framework in testing distributed systems, using *input/output partial order automata (IOPOA)* [28]. In this original model, transitions can be executed non-atomically, reacting to asynchronous inputs on several ports, and producing asynchronous output on those ports. We developed the formal framework for distributed testing in this architecture and compare with the synchronous I/O automaton setting. The advantage of the compact modelling by IOPOA combines with low complexity: the number of tests required for concurrent input in our model is polynomial in the number of inputs. This work has been performed in cooperation with the university of Ottawa.

6.4. Algorithms for Scenarios

Keywords: *Mazurkiewics Traces*, *Scenario*, *Symbolic Unfoldings*, *diagnosis*.

Participants: Loïc Hélouët, Blaise Genest, Thomas Gazagnaire, Shaofa Yang.

Scenario languages enjoy a visual and appealing formalism. In scenarios, behaviors are described by means of a set of events and dependencies and independencies between these events, thus capturing causality. This makes scenarios convenient for diagnosis. On the other hand, using scenarios in algorithms is often non trivial: since a scenario can be decomposed in several ways, finding a particular occurrence of a scenario in a set of behavior is not easy.

Our first result is an algorithm [25] for recovering lost causalities from a partial observation, and from a model of the system. That is, we are able to do event correlation efficiently, without the need for reconstructing all the possible explanations of the observation. To this end, we use boxed pomset as a scenario model in order to be very efficient. The reason is that causalities in a boxed pomset are closed by projection, and partial observation can be seen as a projection. In order to speed up the algorithm, an on-the-fly abstaction of the scenario can be used [26]. More precisely, given an equivalence relation compatible with the causalities of a partial order (as atoms), it is possible to compute on the fly the abstracted relation.

Our last work in the topic of algorithms for Scenarios was to consider basic test on languages of scenarios (graphs of scenarios or set of scenarios which describe the different behaviors of the system), to be included in our tool SOFAT. The first one is an efficient test to check whether there exist a race in a set of MSCs [21]. In practice, a message shall be received as soon at it arrives. Hence two scenarios which differ only by when a message is received should be equivalent, and if the system contains one but not the other one, then we say there is a race. This problem of finding races was known to be undecidable for graphs of scenarios, but we proved that it is decidable for sets of MSCs, and even very efficient when co-regions are not used. This result is important because it is easier to get set of test case instead of a complete model of the system, as a graph of scenario which might exhibit counterintuitive behaviors. The second result [22] concerns the degree of disorder in such a graph of scenarios, in that it provides an efficient algorithm to display its “most counterintuitive” behavior.

6.5. Specific studies: cross-domain services in heterogeneous networks

Keywords: *heterogeneous networks*, *negotiation*, *telecommunications services*.

Participants: Stefan Haar, Eric Fabre, Hélia Pouyllau.

Deploying critical services (e.g. VPN etc.) over the Internet X-domain topology requires guaranteing *end-to-end QoS*. For this, *Service Level Agreements (SLAs)*, also called *QoS contracts*, are committed pairwise between domains. End-to-end QoS provisioning and monitoring for multi-domain services is still an open issue. Key difficulties are heterogeneity, independence and privacy of the individual domains.

Mathematically, the problem is to satisfy an end-to-end QoS budget by a contract chain taking into account: 1/ how the QoS parameters for the different domains are combined along the chain; 2/ the request for domain independence and contract privacy, which forbids from using any centralized solution, and 3/ the optimization of a global cost function. Typical cases reduce to Integer Linear Programming and the general assignment problem, which is NP-Hard. Whereas this can be solved using centralized ILP techniques in principle, domain independence and contract privacy requires fully distributed solutions.

In 2006, we had developed such solutions by using distributed Dynamic Programming [76], [54]. In 2007, we considered the problem of jointly negotiating several chains of contracts in order to reach a global optimum. To this end, we have defined a distributed version of a symmetric Dynamic Programming algorithm [31], [32]. To avoid the burden of performing one negotiation per request, we have considered contract negotiation for a cluster of contract chains associated with possibly different connections [33].

All our algorithms have been integrated into the prototype management platform of the project SWAN.

6.6. Specific studies: Web services orchestrations

Keywords: *Monitoring*, *Orchestrations*, *P2P Systems*, *QoS*, *SLA*, *Web Services*.

Participants: Sidney Rosario, Albert Benveniste, Claude Jard, Stefan Haar, Anne Bouillard.

Web services *orchestrations* and *choreographies* refer to the composition of several Web services to perform a co-ordinated, typically more complex task. Several standardization efforts have been made to model and execute such compositions, including BPEL [55] and the less mature proposal WS-CDL [67] for choreographies. We decided to base our study on a simpler and much cleaner formalism for WS orchestrations, namely the ORC formalism proposed by Jayadev Misra and co-workers [71]. Orc has been successfully used to model standard workflow patterns over the internet [52].

Jointly with the inventors of ORC, namely William Cook and Jayadev Misra, from University of Texas at Austin, we have developed direct translations of ORC into partial order models (more precisely, into so-called Asymmetric Event Structures) [35]. Claude Jard has developed a small tool written in Prolog which implements the original sequential semantics of ORC, as well as its partial order semantics, thus dramatically reducing the size of the executions for storing [34].

Most studies of WS service compositions concentrate of the functional aspects of a composition and ignore the non-functional aspects completely. Main challenges related to QoS include: 1/ To model and quantify the QoS of a service. 2/ To establish a relation between the QoS of queried Web services and that of the orchestration (contract composition); 3/ To monitor and detect the breaching of a QoS contract, possibly leading to a reconfiguration of the orchestration. The QoS of a service is modeled by a *contract* between the provider and consumer of a given service. To account for variability, we have proposed soft probabilistic contracts specified as probabilistic distributions involving the different QoS parameters. We have studied *contract composition* for such contracts [34].

Based on the above translation of ORC to partial order models, a tool named *TOrQuE* (***T****ool for* ***Or****chestration simulation and* ***Qu****ality of service* ***E****valuation*) has been built by Sidney Rosario. *TOrQuE* is a tool running QoS-enhanced Monte-Carlo simulations of the orchestration described in Orc, and deriving empirical estimates for the end-to-end QoS parameters of the orchestration from the QoS parameters of the called sites. Using *TOrQuE* we have shown the possibility of performing significant overbooking in orchestrations, thus improving efficiency of the orchestration in terms of QoS.

6.7. Specific studies: Active XML documents

Keywords: *Active XML*, *P2P Systems*, *XML*.

Participants: Stefan Haar, Albert Benveniste, Eric Fabre, Loïc Hélouët, Blaise Genest, Debmalya Biswas [PhD], Il-Gon Kim [Post-Doc].

The language *Active XML* or *AXML* is an extension of XML which allows to enrich documents with *service calls* or sc's for short. These sc's point to web services that, when triggered, access other documents; this materialization of sc's produces in turn AXML code that is included in the calling document. One therefore speaks of dynamic or intentional documents; note in particular that materialization can be *total* (inserting data in XML format) or *partial* (inserting AXML code containing further sc's). AXML has been developed by the GEMO team at INRIA Futurs, headed by Serge Abiteboul; it allows to set up P2P systems around repositories of AXML documents (one repository per peer).

We are cooperating with the GEMO team (Serge Abiteboul) and the LABRI laboratory in Bordeaux (Anca Muscholl) to explore the behavioral semantics of AXML in the framework of the former ASAX INRIA-ARC (see the 2006 activity report), and to analyze such systems in the frameword of the Docflow and Activedoc projects, see 7.1, 7.2 below. We succeeded into giving a precise semantics to AXML systems [36] that was precise enough to serve as a starting point to develop models following that semantics.

New issues arise in AXML context, such as the discovery [17] of new services, because the visibility of each service is not total. We developed algorithms [37] to exchange local knowledge between the services of a composite system. Qualitative properties of composite web services become harder to study, e.g., atomicity. It raises non trivial issues regarding *compensations,* i.e., the operation of back-tracking a distributed operation in case of interruption, which have been partly addressed in [18].

Our current objective regarding AXML is to be able to check whether some basic properties are decidable, even for documents beyond the so-called "positive" class [2]. One example of such property of interest is confluence. This problem is difficult due to unbounded data, distribution and asynchrony. We succeeded into defining restrictions that are weaker than positiveness, and not too intrusive in order to model real systems, but not permissive enough such that the basic problem we want to solve stay decidable [44].

7. Other Grants and Activities

7.1. ANR Docflow

Participants: Albert Benveniste, Eric Fabre, Loïc Hélouët, Blaise Genest, Hélia Pouyllau, Debmalya Biswas.

Contract INRIA *ANR-06-MDCA-005 January 2007 - December 2009*

Docflow (http://www.labri.fr/perso/anca/docflow/main.html) is a national research project where Distribcom cooperates with INRIA's GEMO team, and the LABRI/Bordeaux. It is a follow up of the ARC Asax (see below), started in January 2007 and is scheduled to end in December 2009. The aim of the project is to model, analyze and monitor real life composite services, as tour operators (Opodo) or supply chain (DELL). It builds on the understanding between the Database community (data centric views) and the Discrete Event community (control centric), brought by the past ASAX meetings. The main tool is Active XML, see URL http://activexml.net on Active XML and Web services. So far, only a fragment of AXML was considered. It is called "positive AXML", and have simplistic control (no move or deletion of data, only copy of nodes are possible at some given nodes, and every copy is possible in parallel). We try to develop a model where control can simulate worflow, and structured data (XML) can be used in the same formalism. This starting point will allow us to develop algorithm to analyse, monitor and optimize worflows with rich data.

7.2. CREATE ActiveDoc

Participants: Albert Benveniste, Eric Fabre, Loïc Hélouët, Blaise Genest, Hélia Pouyllau, Il-Gon Kim, Debmalya Biswas, Sidney Rosario.

Contract INRIA *CREATE February 2007 - August 2011*

Activedoc is funded by Region Bretagne, supporting the ANR Docflow project. It started in February 2007, for 18 months, and can be extended twice for 18 months. In addition to the Docflow program, it grants funding to study composite web services in a quantitative way. The fundamental models proposed in Docflow will be a starting point. For instance, developing methods to compose the Quality of Service of different web services is a difficult problem if one wants realistic values which are not too imprecise. Methods to elaborate and use contracts between heterogeneous services would thus be simplified.

7.3. ANR Dots

Participants: Claude Jard, Loïc Hélouët, Blaise Genest.

Contract INRIA *ANR-06-SETI January 2007 - December 2010*

Dots (http://www.lsv.ens-cachan.fr/anr-dots/) is a national research project where Distribcom cooperates with the LSV/ENS Cachan, the LABRI/Bordeaux, the LAMSADE/Paris Dauphine and the IRCYNN/Nantes. It started in January 2007 and is scheduled to end in December 2010. The ambitious goal of the project is to consider open systems (that is interacting with other undefined systems) which are distributed and require timing information, in order to analyze concrete systems without abstracting one of these aspects. For instance, the interference between several systems require a combination of opened, distributed and timed information. Distribcom is in charge of the interaction of distributed systems with timing aspect (as timed Petri nets) or openness (as distributed controllers and distributed games).

7.4. Politiques de sécurité: TEST et Analyse par le Test de systèmes en réseau ouvert

Participant: Loïc Hélouët.

ACI Sécurité — september 2004 - september 2007

The purpose of the Potestat ACI (http://www-lsr.imag.fr/POTESTAT/) is to study security policies in networks, and to analyze the security of such networks with test techniques. The partners involved in this ACI are : LSR/IMAG - INPG (Vasco team), VERIMAG (DCS team), INRIA Rennes (Vertecs, Lande, and DistribCom teams)

7.5. Control, Analysis and Synthesis of Distributed Systems

Participants: Loïc Hélouët, Blaise Genest, Thomas Gazagnaire, Shaofa Yang, Anne Bouillard.

Associated Team INRIA-NUS — 2006-2007

This associated team is a collaboration with the National University of Singapore. The main research theme is the control and diagnosis of distributed communicating systems. Two application areas are targeted: Real-time embedded systems and telecommunications systems and services. Although very different in nature, both areas make fundamental use of models of concurrency. Several types of formal models are considered: scenario languages, communicating automata and Petri-nets. More specifically, we work together on the following problems:

- An extension of scenario models for distributed systems diagnosis.
- Distributed control synthesis, with applications to the quasi-static scheduling problem.

A joint paper has been published in a top conference on sliding windows, and works is still ongoing on quasi static scheduling (see *highlights* of the year). Shaofa Yang was a PhD from NUS and is now postdoc in the team. The associated team is planned to be extended in 2008, both geographically (to Chennai, India, team of Madhavan Mukind) and in subject (process calculi). Blaise Genest and Thiagarajan (NUS) will go to Chennai in November to start the new projects.

8. Dissemination

8.1. Scientific animation

A. Benveniste is associated editor at large (AEAL) for the journal *IEEE Trans. on Automatic Control*. He is member of the Strategic Advisory Council of the Institute for Systems Research, Univ. of Maryland, College Park, USA. He is in charge of assisting François Baccelli in the management of the INRIA side of the Alcatel external Research Programme (ARP).[4]

[4] Only facts related to the activities of DistribCom team are mentioned. Other roles or duties concern the S4 or Sisthem teams, to which A. Benveniste also belongs.

E. Fabre has co-organized with Victor Khomenko (Newcastle) the UFO workshop, a satellite event of ATPN'07. He has been member of the Program Committee of DX'07.

C. Jard has been in 2007 member of the Program Committee of the following conferences: FORTE, NOTERE, ICAPTN, UFO, MSR, and has been invited for 2008 to the Program Committee of ICAPTN, ROGICS, FORTE and NOTERE. He has served as an expert in several programs of the ANR. He is also member of the editorial board of the *Annales des Télécommunications* and the steering committee of MSR series of conferences. C. Jard supervises a CNRS national transverse program on formal approaches for embedded systems (AFSEC). C. Jard is a member of the board of directors of the ENS Cachan and is the director of the research of its Brittany extension. He participated to the scientific evaluation committee of the french lab LIFC in Besançon. In 2007, C. Jard was member of the PhD Committees of S. Baarir (University of Paris 6) and E. Donin (University of Rennes 1), president of the PhD committees of PA.Reinier (ENS Cachan), JB. Raclet (Univ. Rennes 1), and G. Le Guernic (University of Rennes 1), reviewer of the Habilitation of E. Encrenaz (University of Paris 6), reviewer of the Habilitation of I. Parissis (Univ. Grenoble), and president of the Habilitation commitees of JC. Attiogbé and F. Cassez (University of Nantes).

Stefan Haar is member of the working group for evaluation of international activities with the COST committee of INRIA; he also served on the IFSIC's "commission de spécialistes" section 27 until summer of 2006. He is an associate editor of *IEEE Transactions on Automatic Control*.

Loïc Hélouët is co-rapporteur at ITU for the question 17 on MSC language. He has proposed a document [45] that summarizes all undecidability results in the application of Message Sequence Charts published this last decade. This proposal was debated during the ITU rapporteur's meeing in September 2007, and should be acceoted as an appendix or as a suplement to standard Z.120. He was also invited to participate in the program committee of SDL 2007. Loïc also animates (with S. Pinchinat (S4) and Th. Genet (Lande)) the 68NQRT, a weekly seminar of IRISA on software, theory of computing, discrete mathematics in relation to computer science and artificial intelligence.

8.2. Teaching

E. Fabre teaches "information theory and coding" at Ecole Normale Supérieure de Cachan, Ker Lann campus, in the computer science and telecommunications Master program. He also teaches "numerical and combinatorial optimization," and "distributed algorithms and systems" in the computer science Master program at the University of Rennes 1.

Until Summer of 2006, S. Haar taught queuing systems (models of telecommunication networks) at Ecole Normale Supérieure de Cachan, Ker Lann campus, in the computer science and telecommunications magistère program. He also participated in the 'module MAS' (with C. Jard and L. Helouet) of master M2RI, that is dedicated to models and algorithms for large systems supervision.

L. Hélouët has participated in module MAS (with C.Jard and S.Haar) of master M2RI, that is dedicated to models and algorithms for large systems supervision until summer 2007.

C. Jard is a full-time professor at the ENS Cachan and teaches mainly at the Master level, in Computer Science and Telecom, and in Maths. He manages the Info-Telecom track of the Master-Recherche-STS of the Rennes 1 university. He is also in charge of the competitive examination for the entry of new students in computer science in the French ENS schools.

Anne Bouillard is an Assistant Professor at the ENS Cachan and teaches at the last year Bachelor and Master level in computer science. She is also involved in the preparation of students to the Agrégation of mathematics (highest competitive examination for teachers in France).

8.3. Visits and invitations

C. Jard was invited to give a tutorial on supervision of distributed systems at the NOTERE conference in Marrakech in June 2007.

Doron Peled was invited as a Professor for one month in march 2007, funded by ENS Cachan. He gave a lecture in ENS Cachan and one in IRISA, interacted with the team on several subjects, including Unfoldings, Partial Order Reduction (a follow up of [19] was discussed), and algorithms for scenarios (the paper [22] was written in this period).

B. Genest will spend two weeks in novembre 2007 in Chennai,India to Discuss with P.S. Thiagarajan and M. Mukund about the new associated team (joint with Chennai and NUS, Singapore).

9. Bibliography

Major publications by the team in recent years

[1] S. ABBES, A. BENVENISTE. *Probabilistic true-concurrency models: branching cells and distributed probabilities*, in "Information and Computation", february 2006, p. 231-274.

[2] S. ABITEBOUL, Z. ABRAMS, S. HAAR, T. MILO. *Diagnosis of asynchronous discrete event systems: datalog to the rescue!*, in "PODS '05: Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, New York, NY, USA", ACM Press, 2005, p. 358–367.

[3] A. BENVENISTE, E. FABRE, C. JARD, S. HAAR. *Diagnosis of asynchronous discrete event systems, a net unfolding approach*, in "IEEE Transactions on Automatic Control", vol. 48, n^o 5, May 2003, p. 714–727.

[4] E. FABRE, A. BENVENISTE, S. HAAR, C. JARD. *Distributed Monitoring of Concurrent and Asynchronous Systems*, in "Journal of Discrete Event Systems, special issue", 5 2005, p. 33–84.

[5] B. GAUJAL, S. HAAR, J. MAIRESSE. *Blocking a transition in a free choice net and what it tells about its throughput*, in "J. Comput. Syst. Sci.", vol. 66, n^o 3, 2003, p. 515-548.

[6] T. GAZAGNAIRE, B. GENEST, L. HÉLOUËT, P. THIAGARAJAN, S. YANG. *Causal Message Sequence Charts*, in "Proceedings of CONCUR 2007", 2007.

[7] B. GENEST, A. MUSCHOLL, D. PELED. *Message Sequence Charts.*, in "Lectures on Concurrency and Petri Nets", vol. LNCS 3098, 2003, p. 537-558.

[8] L. HÉLOUËT, C. JARD, B. CAILLAUD. *An Event Structure Semantics for Message Sequence Charts*, in "Mathematical Structures in Computer Science (MSCS) journal", vol. 12, n^o 4, 2002, p. 377–403.

Year Publications

Doctoral dissertations and Habilitation theses

[9] E. DONIN DE ROSIÈRE. *Un langage non déterministe pour l'écriture de scénarios de test*, Ph. D. Thesis, Université de Rennes 1, september 2007.

[10] E. FABRE. *Bayesian Networks of Dynamic Systems*, Ph. D. Thesis, Université de Rennes 1, 2007.

Articles in refereed journals and book chapters

[11] S. ABBES, A. BENVENISTE. *Probabilistic true-concurrency models: Markov nets and a law of large numbers*, in "TCS special issue: FOSSACS 2005", to appear, 2007.

[12] A. BOUILLARD, E. THIERRY. *An Algorithmic Toolbox for Network Calculus*, in "Journal of Discrete Event Dynamical Systems", to appear, 2007.

[13] E. FABRE, A. BENVENISTE. *Partial Order Techniques for Distributed Discrete Event Systems: why you can't avoid using them*, in "Journal of Discrete Events Dynamical Systems", to appear, 2007.

[14] E. FABRE. *Trellis Processes: a Compact Representation for Runs of Concurrent Systems*, in "Journal of Discrete Event Dynamical Systems", vol. 17, n^{o} 3, 2007, p. 267-306.

[15] B. GENEST, D. KUSKE, A. MUSCHOLL. *On Communicating Automata with Bounded Channels.*, in "Fundamenta Informaticae, IOS Press.", vol. 80, 2007, p. 1-21.

[16] S. PICKIN, C. JARD, T. JÉRON, J.-M. JÉZÉQUEL, Y. LE TRAON. *Test Synthesis form UML Models of Distributed Software*, in "IEEE Transactions on Software Engineering", vol. 33, n^{o} 4, April 2007, p. 252–268.

Publications in Conferences and Workshops

[17] D. BISWAS. *Web Services Discovery and Constraints Composition*, in "RR", vol. LNCS 4524, 2007, p. 73-87.

[18] D. BISWAS, I.-G. KIM. *Atomicity for P2P based XML Repositories*, in "2nd IEEE International Workshop on Services Engineering (SEIW)", IEEE Computer Society, 2007.

[19] D. BOSNACKI, E. ELKIND, B. GENEST, D. PELED. *On Commutativity Based Edge Lean Search.*, in "ICALP", vol. LNCS 4596, 2007, p. 158-170.

[20] A. BOUILLARD, B. GAUJAL, S. LAGRANGE, E. THIERRY. *Optimal routing for end-to-end guarantees: the price of multiplexing*, in "VALUETOOLS", 2007.

[21] E. ELKIND, B. GENEST, D. PELED. *Detecting Races in Ensembles of Message Sequence Charts.*, in "TACAS", vol. LNCS 4424, 2007, p. 420-434.

[22] E. ELKIND, B. GENEST, D. PELED, P. SPOLETINI. *Quantifying the Discord: Order Discrepancies in Message Sequence Charts.*, in "ATVA", LNCS, n^{o} 4762, 2007, p. 378-393.

[23] E. FABRE. *Modular Processings Based on Unfoldings*, in "UFO, Workshop on Unfoldings and Partial Order Techniques", invited talk, Sieldce (Poland), June 26 2007.

[24] T. GAZAGNAIRE, B. GENEST, L. HÉLOUËT, P. THIAGARAJAN, S. YANG. *Causal Message Sequence Charts*, in "Proceedings of CONCUR 2007", 2007.

[25] T. GAZAGNAIRE, L. HÉLOUËT. *Reconstructing Causal Ordering with boxed pomsets*, in "Proceedings of FORTE 2007", 2007.

[26] T. GAZAGNAIRE, C. JARD. *Abstraire à la volée les événements d'un système réparti*, in "Proceedings of NOTERE 2007", 2007.

[27] S. HAAR. *Unfold and Cover: Qualitative Diagnosability for Petri Nets*, in "Proceedings CDC", 2007.

[28] S. HAAR, C. JARD, G.-V. JOURDAN. *Testing Input/Output Partial Order Automata*, in "Testcom/Fates 07, 19th IFIP International Conference on Testing of Communicating Systems and 7th International Workshop on Formal Approaches to Testing of Software", Lecture Notes in Computer Science, n^{o} 4581, Springer, June 2007, p. 171-185.

[29] C. JARD. *Concurrent Operational Semantics of Safe Time Petri Nets*, in "UFO, Workshop on Unfoldings and Partial Order Techniques", invited talk, Sieldce (Poland), June 26 2007, p. 19-31.

[30] A. MADALINSKI, E. FABRE. *Modular Construction of Finite and Complete Prefixes*, in "UFO, Workshop on Unfoldings and Partial Order Techniques", Sieldce (Poland), June 26 2007, p. 69-84.

[31] H. POUYLLAU, S. HAAR. *A protocol for QoS contract negotiation and its implementation using Web Services*, in "IEEE International Conference on Web Services (ICWS 2007) Salt Lake City, Utah, USA", IEEE Computer Society, 2007, p. 168-175.

[32] H. POUYLLAU, S. HAAR. *Distributed End-to-End QoS Contract Negotiation*, in "Inter-Domain Management, First International Conference on Autonomous Infrastructure, Management and Security, AIMS 2007, Oslo, Norway", Lecture Notes in Computer Science, Springer, 2007, p. 180-183.

[33] H. POUYLLAU, S. HAAR. *End-to-end QoS of X-Domain pipes*, in "IEEE International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), 2007, Vancouver, Canada", 2007.

[34] S. ROSARIO, A. BENVENISTE, S. HAAR, C. JARD. *Probabilistic QoS and soft contracts for transaction based Web services*, in "ICWS", 2007, p. 126-133.

[35] S. ROSARIO, D. KITCHIN, A. BENVENISTE, W. COOK, S. HAAR, C. JARD. *Event Structure Semantics of ORC*, in "4th International Workshop on Web Services and Formal Methods (WS-FM 2007), Brisbane, Australia", Oct 2007.

Internal Reports

[36] ASAX GROUP. *Semantics and Unfolding of AXML*, Technical report, Asax Project, 2007.

[37] D. BISWAS, K. VIDYASANKAR. *Formalizing Visibility Characteristics in Hierarchical Systems*, Technical report, n^{o} RR-6225, INRIA, 2007, http://hal.inria.fr/inria-00156525.

[38] A. BOUILLARD, C. CHANG. *An Explicit Control Algorithm for Optical FIFO Queues*, Technical report, INRIA Research Report RR-6097, 2007, http://hal.inria.fr/inria-00123901.

[39] A. BOUILLARD, E. THIERRY. *An Algorithmic Toolbox for Network Calculus*, Technical report, INRIA Research Report RR-6094, 2007, http://hal.inria.fr/inria-00123643.

[40] A. BOUILLARD, E. THIERRY. *Some examples and counterexamples for (min,+) filtering operations*, Technical report, INRIA Research Report RR-6095, 2007, http://hal.inria.fr/inria-00123709.

[41] M. BOZGA, L. HÉLOUËT. *Définition de la sémantique des diagrammes d'activités à l'aide de Réseaux de Petri*, Technical report, Délivrable Projet Persiform, 2007.

[42] T. GAZAGNAIRE, B. GENEST, L. HÉLOUËT, P. THIAGARAJAN, S. YANG. *Causal Message Sequence Charts*, INRIA Research Report, n° RR-6301, INRIA, 2007, http://hal.inria.fr/inria-00173529.

[43] B. GENEST, P. DARONDEAU, L. HÉLOUËT. *Products of Message Sequence Charts*, INRIA Research report, n° 6258, INRIA, 2007, http://hal.inria.fr/inria-00156035.

[44] B. GENEST, A. MUSCHOLL, O. SERRE, M. ZEITOUN. *Well Structured Control of Data Tree and the Decidability of AXML*, Research report, 2007.

[45] L. HÉLOUËT. *Application of Message Sequence Charts : a proposal for a new appendix to Z.120*, Technical report, Draft proposal for ITU SG17-Q12, 2007.

[46] C. JARD, O. CONSTANT. *Validation des formalismes proposés sur des applications concrètes*, Technical report, Délivrable Projet Persiform, 2007.

[47] S. ROSARIO, D. KITCHIN, A. BENVENISTE, W. COOK, S. HAAR, C. JARD. *Event Structure Semantics of Orc*, Technical report, n° 1853, IRISA, june. 2007.

References in notes

[48] *Business Process Execution Language for Web Services Version 1.1*, 2005, http://www.ibm.com/developerworks/library/ws-bpel/.

[49] H. EHRIG, H.-J. KREOWSKI, U. MONTANARI, G. ROZENBERG (editors). *Handbook of graph grammars and computing by graph transformations: vol. 3, concurrency, parallelism, and distribution*, World Scientific, 1999.

[50] *Web Services Flow Language (WSFL 1.0)*, 2001, http://www.ibm.com/.

[51] *XLANG, Web Services for Business Process Design*, 2001, http://www.gotdotnet.com/team/xml_wsspecs/xlang-c.

[52] W. M. P. V. D. AALST, A. H. M. T. HOFSTEDE, B. KIEPUSZEWSKI, A. P. BARROS. *Workflow Patterns*, in "Distrib. Parallel Databases", vol. 14, n° 1, 2003, p. 5–51.

[53] S. ABBES, A. BENVENISTE. *True-concurrency Probabilistic Models: Markov Nets and a Law of Large Numbers*, in "Theoretical Computer Science", to appear, 2007.

[54] A. AGHASARYAN, S. PIEKAREC, H. POUYLLAU, S. HAAR, E. FABRE, L. CIARLETTA, N. MBAREK, E. MOREAU. *Multi-Domain Self Aware Management: Negotiation and Monitoring*, in "13th Int. Conf. on Telecommunications, Funchal, Madeira", May 2006.

[55] T. ANDREW, F. CURBERA, H. Y. GOLAND, J. KLEIN, F. LEYMANN, K. LIU, D. ROLLER, D. SMITH, S. THATTE, I. TRICKOVIC, S. WEERAWARANA. *Business Process Execution Language for Web Services. [BPEL4WS.], version 1.1.*, May 2003, http://xml.coverpages.org/BPELv11-May052003Final.pdf.

[56] P. BALDAN, A. CORRADINI, U. MONTANARI. *Unfolding and event structure semantics for graph grammars*, in "Proc. of FOSSACS 1999", LNCS, vol. 1578, Springer Verlag, 1999, p. 73–89.

[57] M. BUSCEMI, V. SASSONE. *High-Level Petri Nets as Type Theories in the Join Calculus*, in "FOSSACS'2001", 2001.

[58] C. CASSANDRAS, S. LAFORTUNE. *Introduction to discrete event systems*, Kluwer Academic Publishers, 1999.

[59] R. DEVILLERS, H. KLAUDEL. *Solving Petri net recursions through finite representation*, in "IASTED International Conference on Advances in Computer Science and Technology, ACST'2004", ISBN 0-88986-497-3, ACTA Press, 2004, p. 145-150.

[60] E. FABRE, A. BENVENISTE. *Partial Order Techniques for Distributed Discrete Event Systems: why you can't avoid using them*, in "Journal of Discrete Events Dynamical Systems", to appear, 2007.

[61] E. FABRE. *Bayesian Networks of Dynamic Systems*, Ph. D. Thesis, Université de Rennes 1, 2007.

[62] T. GAZAGNAIRE, B. GENEST, L. HÉLOUËT, P. THIAGARAJAN, S. YANG. *Causal Message Sequence Charts*, in "Proceedings of CONCUR 2007", 2007.

[63] B. GENEST, D. KUSKE, A. MUSCHOLL. *A Kleene Theorem and Model Checking for a Class of Communicating Automata.*, in "Information and Computation.", vol. 204, n^{o} 6, 2006, p. 920-956.

[64] A. GUPTA. *Integration of Information Systems: Bridging Heterogeneous Databases*, IEEE Press, 1989.

[65] L. HÉLOUËT, T. GAZAGNAIRE, B. GENEST. *Diagnosis from Scenarios*, in "proc. of WODES'06", 2006, p. 307-312.

[66] ITU-TS. *ITU-TS Recommendation Z.120: Message Sequence Chart (MSC)*, ITU-TS, Geneva, September 1999.

[67] N. KAVANTZAS, D. BURDETT, G. RITZINGER, T. FLETCHER, Y. LAFON. *Web Services Choreography Description Language – WS-CDL, version 1.0.*, 2004, http://www.w3.org/2002/ws/chor/edcopies/cdl/cdl.html.

[68] J. KLEIN, B. CAILLAUD, L. HÉLOUËT. *Merging Scenarios*, in "Proc. of FMICS'04 (Formal Methods in Industrial and Critical Systems", Sep. 2004.

[69] J. KLEIN, L. HÉLOUËT, J. JÉZÉQUEL. *Semantic-based Weaving of Scenarios*, in "Proc. of the 5th International Conference on Aspect-Oriented Software Development (AOSD'06), Bonn, Germany", March 2006.

[70] A. MADALINSKI, E. FABRE. *Modular Construction of Finite and Complete Prefixes*, in "UFO, Workshop on Unfoldings and Partial Order Techniques", Sieldce (Poland), June 26 2007, p. 69-84.

[71] J. MISRA. *A Programming Model for the Orchestration of Web Services.*, in "SEFM", 2004, p. 2-11.

[72] R. MORIN. *Recognizable sets of Message Sequence Charts*, in "Proc. of STACS 2002, 19th Annual Symposium on Theoretical Aspects of Computer Science", H. ALT, A. FERREIRA (editors), LNCS, Springer Verlag, march 2002, p. 523-534.

[73] A. MUSCHOLL, D. PELED, Z. SU. *Deciding Properties for Message Sequence Charts*, in "Lecture Notes in Computer Science", vol. 1378, 1998, p. 226-242, http://citeseer.ist.psu.edu/muscholl98deciding.html.

[74] M. NIELSEN, G. PLOTKIN, G. WINSKEL. *Petri nets, event structures and domains, part 1*, in "T.C.S.", 1981.

[75] OBJECT MANAGEMENT GROUP. *Unified Modeling Language Specification version 2.0: Superstructure*, Technical report, n° pct/03-08-02, OMG, 2003.

[76] H. POUYLLAU, L. CIARLETTA, A. AGHASARYAN, S. HAAR. *X-domain QoS budget negotiation using Dynamic Programming*, in "IEEE Advanced International Conference on Telecommunications (AICT)", 2006.

[77] W. REISIG. *Petri nets*, Springer Verlag, 1985.

[78] M. RENIERS, S. MAUW. *High-level Message Sequence Charts*, in "SDL97: Time for Testing - SDL, MSC and Trends, Evry, France", A. CAVALLI, A. SARMA (editors), Proceedings of the Eighth SDL Forum, September 1997, p. 291-306.

[79] M. RENIERS. *Message Sequence Charts: Syntax and Semantics*, Ph. D. Thesis, Eindhoven University of Technology, 1998.

[80] E. RUDOLPH, P. GRAUBMAN, J. GRABOWSKI. *Tutorial On Message Sequence Charts*, in "Computer Networks and ISDN Systems", vol. 28, 1996, p. 1629-1641.

[81] G. WINSKEL. *Event structures semantics in CCS and related languages*, in "LNCS", vol. 140, Springer Verlag, 1982.

[82] G. WINSKEL. *Categories of models for concurrency*, in "Seminar on Concurrency, Carnegie-Mellon Univ. (July 1984), LNCS 197", 1985, p. 246-267.

[83] R. D. VAN DER MEI, H. B. MEEUWISSEN. *Modelling end-to-end Quality-of-Service for Transaction-Based Services in Multi-Domain Environments*, in "IEEE International Conference on Web Services, ICWS", september 2006, p. 453-462.

[84] T. ÖZSU, P. VALDURIEZ. *Principles of Distributed Database Systems, 2nd Edition*, Prentice-Hall, 1999.