Bibliographie :

Innovations et retours d’expériences en Ingénierie dirigée par les modèles.

Classement thématique / chronologique.

Entrée en matière :

Définition de MDA (Model Driven Architecture)
< http://en.wikipedia.org/wiki/Model-driven_architecture >, [page consultée le 29 octobre 2009].
Présentation en une page et en anglais de l’architecture dirigée par les modèles ou MDA (Model Driven Architecture) = « démarche de réalisation de logiciel, proposée et soutenue par l’OMG. C’est une variante particulière de l’ingénierie dirigée par les modèles (IDM, ou MDE pour l’Anglais Model Driven Engineering). »

Définition de « Metamodeling »
< http://en.wikipedia.org/wiki/Metamodelling >, [page consultée le 29 octobre 2009].
Présentation en une page en anglais du terme de métamodélisation, ou ingénierie des modèles.

Object Management Group - OMG
« OMG Task Forces develop enterprise integration standards for a wide range of technologies, including: Real-time, Embedded and Specialized Systems, Analysis & Design, Architecture-Driven Modernization and Middleware and an even wider range of industries, including: Business Modeling and Integration, C4I, Finance, Government, Healthcare, Legal Compliance, Life Sciences Research, Manufacturing Technology, Robotics, Software-Based Communications and Space. OMG’s modeling standards, including the Unified Modeling Language™ (UML®) and Model Driven Architecture® (MDA®), enable powerful visual design, execution and maintenance of software and other processes, including IT Systems Modeling and Business Process Management. OMG’s middleware standards and profiles are based on the Common Object Request Broker Architecture (CORBA®) and support a wide variety of industries. »

Définition de SOA (Service Oriented Architecture)
Présentation en une page et en français de l’architecture orientée services (SOA) = « forme d’architecture de médiation qui est un modèle d’interaction applicative qui met en œuvre des services (composants logiciels) ». 
< http://fr.wikipedia.org/wiki/Architecture_orient%C3%A9e_services >, [page consultée le 29 octobre 2009].

“Software architecture is foundational to the development of large, practical software-intensive applications. This brand-new text covers all facets of software architecture and how it serves as the intellectual centerpiece of software development and evolution. Critically, this text focuses on supporting creation of real implemented systems. Hence the text details not only modeling techniques, but design, implementation, deployment, and system adaptation -- as well as a host of other topics -- putting the elements in context and comparing and contrasting them with one another. Rather than focusing on one method, notation, tool, or process, this new text/reference widely surveys software architecture techniques, enabling the instructor and practitioner to choose the right tool for the job at hand. Software Architecture is intended for upper-division undergraduate and graduate courses in software architecture, software design, component-based software engineering, and distributed systems; the text may also be used in introductory as well as advanced software engineering courses”.


«Nouvelle approche d'ingénierie logicielle élaborée par l'OMG, MDA (Model Driven Architecture) prône l'utilisation systématique de modèles dans toutes les phases du cycle de vie des applications informaticques, de l'analyse et conception à la génération de code et au déploiement. Pour une meilleure productivité et une plus grande pérennité des développements, la logique métier d'une application est modélisée indépendamment de toute plate-forme d'exécution, un mécanisme de transformation de modèles permettant ensuite de produire automatiquement les modèles et le code spécifiques à chaque plate-forme (J2EE, .NET, PHP...). “ [...]” Après une description détaillée des standards sur lesquels repose MDA, tels que UML, MOF, OCL et XMI, l'ouvrage insiste sur les aspects pratiques de sa mise en œuvre.»

Le coin des spécialistes :

MDA:


“When software projects evolve their actual implementation and their intended architecture may drift apart resulting in problems for further maintenance. As a countermeasure it is good software engineering practice to check the implementation against the architectural description for consistency. In this work we check software developed by a Model Driven Software Development (MDSD) process. This allows us to completely automate consistency checking by deducing information from implementation, design documents, and model transformations. We have applied our approach on a Java project and found several inconsistencies hinting at design problems. With our approach we can find inconsistencies early, keep the artifacts of an MDSD process consistent, and, thus, improve the maintainability and understandability of the software.”

Perovich, D; Bastarrica, MC; Rojas, C. Model-Driven Approach to Software Architecture Design. In 4th Workshop on Sharing and Reusing Architectural Knowledge (SHARK 2009), May 16, 2009 Vancouver, CA, pp: 1-8
“Software Architecture (SA) allows for early assessment of and design for quality attributes of a software system, and it plays a critical role in current software development. However there is no consensus on fundamental n°s such as design methods and representation organization and languages, and current proposals lack specificity and preciseness. Thus, it is extremely difficult to build a complete and appropriate software architecture, even though it is recognized as a fundamental artifact. In this paper we define an architecture design method that enables the systematic and assisted construction of the SA of Enterprise Applications, taking into account major quality attributes involved in this family of systems. We apply Model-Driven Engineering techniques to achieve this goal. The architecture is treated as a mega-model (a model composed of related models) and the application of design decisions is encoded in terms of model transformations. The architectural rationale is explicitly registered as the set of tran, formations that yields the complete SA from scratch. We illustrate the application of the approach by designing the SA of a case study from the literature”.

Disponible sous forme électronique sur l’intranet IST INRIA


“Context information constitutes an essential aspect of service development and provision in mobile computing in the attempt to provide users with personalized services. The problem of handling context in these environments, as well as the development of context-aware services, have become quite challenging research tasks in the last years. In this paper, the ongoing work towards context handling of web services is presented along with a model-driven methodology for context-aware service engineering for web applications built on web services. The solution focuses on decoupling the context management mechanism from the core service logic in all development stages.”

Disponible sous forme électronique sur l’intranet IST INRIA


“Model-Driven Architecture (MDA) promotes the development of software systems through successive building and generation of models, improving the reusability of models. Applying the same principles to the area of Agent-Oriented Software Engineering (AOSE) advances the ideas behind MDA even more significantly, due to the inherent adaptivity of software agents We describe an appropriate set of models originating from requirements specification and transformable to models understandable and executable by agents thus demonstrating an Agent-oriented Model-Driven Architecture (AMDA) approach. In AMDA, agents use hierarchical business knowledge models with business process rules at the top, business rules to control policy and logic in the middle and a base layer defining business concepts. Being externalised, knowledge is easily configurable by human beings and applied by software agents. A real case study is used to illustrate the process. The main advances over the object-oriented MDA are (i) the addition of component dynamics (ii) the use of agent-executable rule-based business models and (iii) a proposed higher level of abstraction with the direct representation of business requirements. “

Disponible sous forme électronique sur l’intranet IST INRIA


“The fourth edition of the European Conference on Model-Driven Architecture - Foundations and Applications (ECMDA-FA 2008) was dedicated to furthering the state of knowledge and fostering the industrialization of the model-driven architecture (MDA) methodology. MDA is an initiative proposed by the Object Management Group (OMG) for platform-generic software development. It promotes the use of models in the specification, design, analysis, synthesis, deployment, and evolution of complex software systems. “[...]” ECMDA-FA addresses various MDA areas including model management, executable models, concrete syntaxes, aspects and concerns, validation and testing, model-based systems engineering,
model-driven development and service oriented architectures, and the application of model-driven development.”

Disponible sous forme électronique sur l’intranet IST INRIA

**MDA et ingénierie documentaire :**


“Defining a formal domain ontology is generally considered a useful, not to say necessary step in almost every software project. This is because software deals with ideas rather than with self-evident physical artefacts. However, this development step is hardly ever done, as ontologies rely on well-defined and semantically powerful AI concepts such as description logics or rule-based systems, and most software engineers are largely unfamiliar with these. Gašević and his co-authors try to fill this gap by covering the subject of MDA application for ontology development on the Semantic Web. Part I of their book describes existing technologies, tools, and standards like XML, RDF, OWL, MDA, and UML. Part II presents the first detailed description of OMG’s new ODM (Ontology Definition Metamodel) initiative, a specification which is expected to be in the form of an OMG language like UML. Finally, Part III is dedicated to applications and practical aspects of developing ontologies using MDA-based languages. The book is supported by a website showing many ontologies, UML and other MDA-based models, and the transformations between them”.

Disponible en Prêt-Entre-Bibliothèque auprès du réseau IST INRIA

**SOA : (Service Oriented Architecture):**


“The Web is now evolving into a medium for providing a wide array of e-commerce and other information based services that are solely based on distributed software interactions for collaborative work. Service-oriented and event-driven architectures (SOA and EDA) are the complementary paradigms for building such distributed information systems based on software interactions as their characteristics of modularity, loose-couplings, and flexibility. However, given the rapid change of the web environment (including the dynamic behavior of the users), these paradigms are not yet able to empower a built system adapting its behavior according to the changes in its environment or in parts of the system itself. This paper presents an architecture of adaptive integration of EDA and SOA based on the novel concept of value-centric processing and communication of events grounded on value theory. This architecture has also been used on a case study of distributed care service delivery. In the architecture, value models are devised to represent events consumed and develop a value-centric mechanism for this adaptive integration. We have also implemented the system and simulated certain representative scenarios in order to justify the claimed values of this adaptive integration on the case study. “

Disponible sous forme électronique sur l’intranet IST INRIA


“In large-scale software projects that increasingly adopt service-oriented software architecture and technologies, availability of sound systems engineering principles, methodology and tools for service-oriented applications is mission-critical for project success. However, engineering service-oriented applications poses specific requirements that differ from traditional software engineering and the discipline of software service engineering (SSE) is not yet established. Consequently, there is an urgent need for research community and industry practitioners to develop comprehensive engineering principles, methodologies and tool support for the entire software development lifecycle of service-
oriented applications. “[...]”These papers represent a rich variety of topics revolving around principles, methods and application domains of SSE with a special focus on analysis and design”.

Disponible sous forme électronique sur l’intranet IST INRIA


“Endorsed by all major vendors (Microsoft, Oracle, IBM, and SAP), SOA has quickly become the industry standard for building next-generation software; this practical guide shows readers how to achieve the many benefits of SOA. Begins with a look at the architectural principles needed to create successful applications and then goes on to examine the process for designing services and SOA implementations. Each stage of the design process has an accompanying chapter that walks readers through the details and provides helpful tips, techniques, and examples. The author team of SOA practitioners also provides two unique, comprehensive, end-to-end case studies illustrating the architectural and design techniques presented in the book”.

Ouvrage en commande par le service IST Nancy Grand-Est

Métamodélisation:


“The book first presents the theoretical basis of metamodeling for method engineering, discussing information modeling, the potential of metamodeling for software systems development, and the introduction of the metamodeling tool ConceptBase. The second, and larger, portion of the book reports on applications of the metamodeling approach to method engineering. These detailed case studies range from telecommunication service specification, hypermedia design, and data warehousing to cooperative requirements engineering, chemical device modeling, and design of new abstraction principles of modeling languages. Although these chapters can stand alone as case studies, they also relate to the earlier theoretical chapters. The metamodeling approach described in the book is based on the Telos metamodeling language implemented by the ConceptBase system.”

Disponible en Prêt-Entre-Bibliothèque auprès du réseau IST INRIA


“A basic premise of model-driven development (MDD) is to capture all important design information in a set of formal or semiformal models, which are then automatically kept consistent by tools. The concept, however, is still relatively immature and there is little by way of empirically validated guidelines. In this paper, we report on the use of MDD on a significant real-world project over several years. Our research found the MDD approach to be deficient in terms of modeling architectural design rules. Furthermore, the current body of literature does not offer a satisfactory solution as to how architectural design rules should be modeled. As a result developers have to rely on time-consuming and error-prone manual practices to keep a system consistent with its architecture. To realize the full benefits of MDD, it is important to find ways of formalizing architectural design rules, which then allow automatic enforcement of the architecture on the system. model. Without this, architectural enforcement will remain a bottleneck in large MDD projects.”

Disponible sous forme électronique sur l’intranet IST INRIA

Nous proposons dans cette thèse une démarche permettant de décrire un DSML (Domain Specific Modeling Language) et les outils nécessaires à l’exécution, la vérification et la validation des modèles. La démarche que nous proposons offre une architecture générique de la syntaxe abstraite du DSML pour capturer les informations nécessaires à l’exécution d’un modèle et définir les propriétés temporelles qui doivent être vérifiées. Nous nous appuyons sur cette architecture pour expliciter la sémantique de référence et l’implanter. Plus particulièrement, nous étudions les moyens : - d’exprimer et de valider la définition d’une traduction vers un domaine formel dans le but de réutiliser des outils de model-checking. - de compléter la syntaxe abstraite par le comportement ; et profiter d’outils génériques pour pouvoir simuler les modèles construits. Enfin, de manière à valider les différentes sémantiques implantées vis-à-vis de la sémantique de référence, nous proposons un cadre formel de métamodélisation. 

Disponible sous forme électronique sur internet


“A model is a simplified representation of an aspect of the world for a specific purpose. In complex systems, many aspects are to be handled, from architectural aspects to dynamic behavior, functionalities, user-interface, and extra-functional concerns (such as security, reliability, timeliness, etc.). For software systems, the design process can then be characterized as the weaving of all these aspects into a detailed design model. Model Driven Design aims at automating this weaving process, that is automatically deriving software systems from theirs models. This paper explores the relationship between modeling and aspect weaving. It points out some of the challenges related to such automatic model weaving, illustrating them with the example of a weaving process for behavioral models represented as scenarios”.

Disponible sous forme électronique sur l’intranet IST INRIA


“L'Ingénierie dirigée par les modèles est une approche pour la construction de systèmes fondées sur l'utilisation de modèles. Malgré une recherche très active sur les technologies d'IDM et leur application, il existe relativement peu de travaux sur la réutilisation et la composition fiable des éléments qui manipulent des modèles. Cette thèse présente une approche fondée sur les idées et formalismes du domaine des systèmes de types, pour définir des notions claires de modèles et de types de modèles, ainsi qu'une relation pour la substitution des types de modèles. Ces principes du typage de modèles ont été validés par leur application dans le langage de méta-modélisation Kermeta. La thèse revisite le problème de l'extensibilité des langages sous l'angle de l'ingénierie dirigée par les modèles, et elle détaille comment les types de modèles en résolvent certains aspects.”

Disponible sous forme électronique sur internet


“L'IDM, généralisation du MDA de l'OMG, est une nouvelle approche qui met les modèles, et non pas les programmes, au centre de la démarche en Génie Logiciel. Les avantages annoncés de l'IDM sont nombreux : indépendance vis à vis des évolutions technologiques, meilleure maîtrise de la complexité, meilleure réutilisation etc. [...].”Cet ouvrage propose un discours argumenté sur les points forts et les points faibles de l'IDM, en essayant d'identifier les fondements de cette approche plutôt que d'en rester aux techniques du moment. L'ouvrage s'attache à montrer les emprunts et les synergies entre l'IDM et des disciplines plus anciennes. Il donne une vision vaste et critique des diverses facettes, des tendances, des succès et des difficultés que rencontre aujourd'hui l'IDM.”

Disponible en Prêt-Entre-Bibliothèque auprès du réseau IST INRIA

“Modeling languages that aim to capture PIM level behavior are still a challenge. We propose a high level behavioral formalism based on the Abstract State Machines (ASMs) for the specification and validation of software systems at PIM level. An ASM-based extension of the UML and its Action Semantics is here presented for the construction of executable class models at PIM level and also a model weaving process which makes the execution of such models possible. Our approach is illustrated using an Invoice Order System taken from the literature”.

Disponible sous forme électronique sur l’intranet IST INRIA

Outils et applications

EMF et Eclipse :


“The Eclipse Modeling Framework enables developers to rapidly construct robust applications based on surprisingly simple models. Now, in this thoroughly revised Second Edition, the project’s developers offer expert guidance, insight, and examples for solving real-world problems with EMF, accelerating development processes, and improving software quality. [...]” The authors illuminate the key concepts and techniques of EMF modeling, analyze EMF’s most important framework classes and generator patterns, guide you through choosing optimal designs, and introduce powerful framework customizations and programming techniques. (Defining models with Java, UML, XML Schema, and Ecore, using extended Ecore modeling to fully unify XML with UML and Java, Leveraging the latest EMF features, including extended metadata, feature maps, EStore, cross-reference adapters, copiers, and content types, Chapters on change recording, validation, and utilizing EMF in stand-alone and Eclipse RCP applications, Modeling generics with Ecore and generating Java 5 cod.”

Disponible en Prêt-Entre-Bibliothèque auprès du réseau IST INRIA

Eclipse website :
The Eclipse Foundation. « Eclipse Corner Articles : these following articles have been written by members of the development team and other members of the eclipse community. ”
Articles traitant de « Modeling » et de « EMF ».
< http://www.eclipse.org/articles/?sort=date&category=Modeling >, [page consultée le 29 octobre 2009].
< http://www.eclipse.org/articles/?sort=date&category=EMF >, [page consultée le 29 octobre 2009].

Eclipse Modeling Project
“The Eclipse Modeling Project focuses on the evolution and promotion of model-based development technologies within the Eclipse community by providing a unified set of modeling frameworks, tooling, and standards implementations.”
< http://www.eclipse.org/modeling/ >, [page consultée le 29 octobre 2009].

Equipes de recherche et sociétés commerciales:

Plateforme Galaxy INRIA :
http://galaxy.gforge.inria.fr

Equipe Tuvalu (Galaxy SOA) :
“The open source SOA/BPM development team TUVALU mainly represents the INRIA galaxy R&D collaborative action.”
http://tuvalu.inrialpes.fr/presentation.en.html

Equipe Triskell:
L'équipe-projet Triskell est une équipe de recherche au sein de l'IRISA (unité mixte de recherche regroupant le CNRS, l'Université Rennes 1, l'INRIA et l'INSA à Rennes), leader mondial de la recherche en Ingénierie des Modèles (IDM, ou MDE en anglais).
http://www.irisa.fr/triskell/home_html

Société Obeo (Obeo designer):
http://www.obeo.fr/pages/obeo-designer/fr


Société Itemis : http://www.itemis.com/

Groupe de recherche « Document Structure » de Xerox:

Bibliographie réalisée par le service IST INRIA Grenoble - Rhône-Alpes/ Octobre 2009

Pour tout renseignement, contactez isabelle.rey@inria.fr / La version web de cette bibliographie est accessible sur le site des séminaires Intech’ (INRIA Grenoble Rhône-Alpes)