ERCIM NEWS

Special:

Automated Software Engineering

Contents

ERCIM's 15th Anniversary Celebration

ERCIM celebrated fifteen years (1989-2004) of cooperation in research, technology transfer, innovation and training. More than 150 people, including representatives from ERCIM institutes and invited guests from academic and research institutions, as well as from the European Commission, gathered in Crete, Greece on 26 May 2004 to commemorate the ERCIM accomplishments.

During the 15th ERCIM anniversary event, the President of ERCIM, Stelios Orphanoudakis (Director and Chairman of the Board of FORTH, Greece) gave an introductory presentation, 'ERCIM as a model for collaboration', which stressed the importance of expanding existing collaborations and taking on new challenges. The presentation included a brief history of ERCIM and the opportunities offered to young researchers, such as the Cor Baayen Award and the Fellowship Programme.

Cor Baayen, and Gerhard Seegmueller, two of the three founding fathers of ERCIM,

The ERCIM 15th Anniversary cake was cut during the Gala Dinner. From left: Stelios Orphanoudakis, Gerhard Seegmüller and Cor Baayen.

shared their early steps and experiences towards the implementation of their vision for a European research community, favouring close collaboration and mobility of researchers. Alain Bensoussan, the third founding father had expressed his deep regrets that he was not able to attend.

Invited representatives of the European Commission, Gerard Comyn, Head of Unit, ICT for Health, and Thierry van der Pyl, Head of Unit, Future and Emerging Technologies congratulated ERCIM for its significant achievements and expressed their support for its continuing efforts towards European R&D excellence.

Keith Jeffery (CCLRC, UK), Vice President of ERCIM, presented the achievements of ERCIM and its growth over the 15 year-period, and finally, Andreas Rauber (Vienna University of Technology, Austria) Cor Baayen award winner 2002, referred to the benefits of such an award on both a professional and a personal level.

A New ERCIM Working Group on Rapid Integration of Software Engineering Techniques

by Nicolas Guelfi and RISE's Working Group members

The main goal of the new ERCIM Working Group on Rapid Integration of Software Engineering Techniques (RISE) is to conduct research into new, integrated and practical software engineering approaches. These will be part of a methodological framework and will apply to new and evolving applications, technologies and systems.

The Working Group (WG) defines itself by the following keywords:
Integrated solutions: we envisage the WG developing projects that will promote the integration of software engineering (SE) techniques. The rate of progress in SE research means that many techniques are simultaneously under development and must often be integrated with complementary approaches (for example, research on development process definition and architecture engineering, architectural frameworks and

architecture description languages, reuse and software architecture, reuse and test techniques, modelling languages and semantics). We believe that working on the integration of SE techniques will:
• ease collaborations between ERCIM partners
• raise targeted/applicative/technological research problems for each of the integrated techniques as well as for the novel integrated solution
• allow the joint development of new and powerful solutions covering

several aspects of the software engineering process.

Practical solutions: we want to focus on concrete applications. Research in software engineering should provide solutions that will support the development process, which means the research problems and the real-life situations in which they exist must be well defined. This does not mean that fundamental research is excluded; rather it means that fundamental problems must be raised and

solved in a manner that preserves efficiency and usefulness. Linked with the previous notion of integration, we will undertake research for the mid-term (three to six years). Further fundamental research raised by the WG will be taken up by partners outside the WG.

New solutions: the objective is to identify novel SE techniques, methods and tools before the need for them arises. In other words, due to ICT being such a rapidly evolving sector, we intend RISE to work on solutions to anticipated problems, rather than currently existing ones (we do not want to repeat the so-called software crisis of the 60s).

Methodological guidance: we believe that software engineering solutions should always come with methodological support. RISE will therefore work on the definition of this notion, and any solutions it provides will always be provided with methodological support.

Sub-Domains of Interest

Rather than addressing the entire scope of software engineering, the RISE working group will focus on the following sub-domains:
- software/system architecture
- reuse
- testing
- model transformation/model-driven engineering

- requirement engineering
- lightweight formal methods
- CASE tools.

Specific Domains of Application and Industry Sectors

The RISE working group will limit its research to specific application domains:
- Web systems
- mobility in communication systems
- high-availability systems
- embedded systems.

Further, certain specific industry sectors will be given priority:
- finance and insurance
- telecommunications
- avionics and automotive
- e-Government.

Constraints on Contribution

In order to attract a coherent set of contributions, RISE wishes the following constraints to be applied to proposals:
- a clear statement of the research problems to be addressed must be provided
- at least two approaches for integration should be proposed
- a methodological guideline should be proposed
- tool support should be provided
- a validation example should illustrate the proposal.

These constraints should be considered as final objectives for the proposals

submitted to the WG. An incremental approach will be followed to organise the research work. While the working group will address problems that only partially adhere to these objectives, we want to stress that such problems should be presented with reference to a globally coherent perspective that respects the constraints of the WG.

The current RISE participants are: Juan Bicarregui (CCLRC), Antonia Bertolino (CNR-ISTI), Paul Klint (CWI), Nicolal Guelfi (FNR), Anthony Savidis (FORTH), Marc Born, Eckhard Moeller, Ina Schieferdecker (Fraunhofer FOKUS), Valérie Issarny (INRIA), Finn Arve Aagesen (NTNU), Joe Armstrong (SICS), Ernesto Pimentel (SpaRCIM), László Monostori (SZTAKI) and Jyrki Haajanen (VTT).

Interested scientists and teams from research and industry who would like to participate in the RISE Working Group should contact the WG coordinator Niclas Guelfi.

Link:
http://www.rise.lu

Please contact:
Nicolas Guelfi, RISE WG coordinator
University of Luxembourg
Tel: +352 420 101 251
E-mail : Nicolas.Guelfi@uni.lu

ERCIM PhD Fellowship Programme

ERCIM offers 18-month fellowships in leading European information technology research centres. Fellowships are available for PhD-holders from all over the world.

Conditions
Applicants must:
- have a PhD degree or be in the last year of the thesis work
- be fluent in English
- be discharged or get deferment from military service
- start the grant before October 2005
- have completed their PhD before starting the grant.

Fellowships are usually spent in two ERCIM insitutes. The fellow will receive a monthly allowance which may vary depending on the country. In order to encourage the mobility, a member

institute will not be eligible to host a candidate of the same nationality. ERCIM does not only encourage researchers from academic institutions to apply, but also scientists working in industry. In order to encourage the mobility, a member institution will not be eligible to host a candidate of the same nationality.

Deadline
Next deadline for Applications: 30 September 2004.

More information and application form:
http://www.ercim.org/fellowship/

Information Society Technologies - Future and Emerging Technologies

IST-FET Workshop for Preparing for FP7 — 'New Directions for ICTs in FP7: Grand Challenges for Basic Research'

Brussels, 21-22 April 2004

In preparation of the 7th Framework Programme, the Future and Emerging Technologies Unit of the Directorate General Information Society has organised a 2-day workshop, on the 21st and 22nd of April in Brussels. Due to the importance of this event, ERCIM News publishes the full executive summary of the report.

The aim of this event was to discuss the new grand challenges that lie ahead of us and to identify future key and visionary basic research directions in Information and Communication Technologies (ICTs), as well as in their combination with other disciplines for addressing the new challenges. The event was targeted to a wide spectrum of leading scientists, industrialists and science policy makers and was attended by more than 120 participants.

The workshop comprised a set of plenary sessions and four parallel thematic sessions, covering issues on 'Strategy and policy for ICT-based collaborative basic research', 'Components Research', 'Systems Research', and, 'Intelligence and Cognition'. During the event, the participants exchanged and shared their knowledge, experience and visions and drew a set of recommendations and conclusions on all the above issues. These are highlighted in the next sections.

Thematic Session on Strategy and Policy for ICT-Based Collaborative Basic Research in FP7

The high added value and importance of collaborative basic research in ICTs are now widely recognised in Europe. Defining and promoting future basic research directions for collaborative basic research in ICTs at EU scale, developing strategies for fostering research excellence beyond national boundaries, motivating and enabling effective university-industry collaboration and well integrating the new member states into the EU's basic research and innovation fabric are some

of the new challenges lying ahead of us in the coming FP7. These are the main issues discussed in this session. Its findings are summarised below.

Concerning the definition of future basic research directions, it was emphasised that the boundaries of ICT research are now further expanding and ICTs are increasingly cross-fertilising with many other science and technology fields. In this framework, three mechanisms were discussed for selecting future basic research directions.

First, through identification of grand challenges, ie, visionary themes demanding breakthroughs in basic research and engineering in many key technologies focusing 10+ years in the future. Grand challenges should lead to 'pictures of the future' and lie at the edge of what 'just might be possible', so as to inspire researchers beyond the boundaries of ordinary thinking. Second, through identification of key technological issues of major importance to economic growth. And third, derived from major social and societal drivers.

Irrespective of the mechanism used, a basic research direction selected would have to clearly address the following four major aspects: what is the issue to be addressed, how can it be best addressed, for whom is it addressed, and by whom.

For the development of research directions it was also found beneficial to foresee a 'planning phase' during which relevant stakeholders are brought together to build a consensus on the primary motivation and basis for a new

research programme. Multi-disciplinary representation in this planning phase is a key issue but multi-disciplinarity does not necessarily imply that in a research programme there must be equal participation across all relevant disciplines concerned.

Research excellence beyond national boundaries is often a prerequisite to ensure excellence. New programmes must be designed to attract the best researchers in each of the disciplines concerned and provide methods to train new generations of researchers but also researchers in industry. Education and training mechanisms have to become an integral part of future EU programmes.

Excellent research will often be characterised by co-sponsoring from national funding agencies. Such co-sponsorship is highly desirable and mechanisms for achieving it need to be put in place. Co-sponsored programmes need to remain open to participation from all member states and not limited only to the states whose agencies are co-financing them.

Aiming for excellence in new research programmes calls for a definition of excellence. While a number of methods now exist on how to measure excellence, individual projects need also to define metrics through which their excellence can be measured. Metrics of progress should also be defined at the level of a new programme, from its beginning.

Stronger emphasis needs to be placed on the dissemination and promotion of results achieved in research programmes. Multifaceted mechanisms need to be put in place for going beyond

standard academic publications and conferences by including measures for dissemination of results to industry to ensure take-up and to society at large.

In terms of academia/industry collaboration: strong industrial involvement in new research programmes is aimed. This involvement may take many different forms, from project observers to 'matching of research funds'. Explorative research should primarily be carried out in an open environment. At the same time, industrial involvement can leverage the research directions and the later take-up of results. However, the rules for participation and the associated rules for IPR ownership must be clear from the start and at the level of a new programme.

The ten new member countries have a long tradition of excellent research. There is however a strong need there to consider initiatives for added infrastructure support (eg structural funds, new equipment through research projects, etc) and methods to get researchers more tightly integrated into the EU research community, in particular by better exploiting the existing networks of excellence and co-ordination actions.

Thematic Session on Components Research

In the coming decade, application driven research in components will be guided by ambient intelligence applications focusing on health, comfort and leisure, communication, mobility, safety and security. Opportunity-driven innovations will be increasingly based on novel nanodevice building blocks and related nanofabrication techniques, on new component design and architectures, and likely on bio-inspired approaches and concepts. Such developments will complement the mainstream RTD efforts outlined in the ITRS roadmap and were the subject of discussions held in this thematic session.

With nanoscale devices reaching characteristic dimensions of 10 nanometres (nm) in 2015-2020, new opportunities will emerge to combine ultimate 'top-down' semiconductor platforms with 'bottom-up' developments in materials, physics, chemistry and biology. A set of

grand challenges was identified that summarises the multidisciplinary research themes required for bringing these visions into tomorrow's innovations.

Advances in nano-scale materials science will provide the basis to add significant extra functionality and performance to devices and systems on mainstream silicon platforms. These in turn will bring along and require new system architectures that will underpin new applications in ICTs and the life sciences. Two main issues demand particular attention, namely heat generation and dissipation as dimensions shrink, and architectures. Inorganic nanowires and nanotubes will likely have a prominent position in these developments.

The second grand challenge is to enable the combination and interfacing among a growing diversity of materials, functions, devices and information carriers. These 'information carriers' could be electrons, photons, spins, ions, etc.

New materials, devices and circuits will require cost-effective fabrication techniques for complex systems with deep nanometre scale devices. Nanoscale components must be grown and patterned at scales around and below 10 nm, going far beyond the current limitations of lithography. Self-assembly of nano-objects mediated by (bio)chemical interactions appears as one of the promising routes for a sustainable manufacturing of downscaled nano-components.

Pushing the limits of miniaturisation to the one nanometre scale requires new methods and tools to accurately model, manipulate, fabricate and characterise nano-objects down to the atomic scale. It also requires new paradigms to exchange information with single atoms or molecules.

Denser integration and combination of top-down, bottom up and self-organised devices will vastly increase the complexity of ICT components and architectures. These require methods and tools to master giga-complexity ICT architectures, integrating billions of

devices with nano-scale dimensions and coping with variability, defects and energy-dissipation issues. Inspiration from bio-systems is likely to lead towards innovative and lower cost solutions.

A number of new physical phenomena or properties of matter at the meso-scale have recently been discovered or demonstrated. These should be further investigated and, as appropriate, developed into new functions or technological developments for the ICT. Research for the discovery and further investigation of such new phenomena needs also to be supported.

The increasingly fading boundaries between ICT and other related fields such as materials sciences, physics, chemistry, biochemistry and life sciences were stressed all along the discussions held in this session. Future research is thus expected to become more multidisciplinary and be based on strong and effective integration of excellent researchers coming from all these different disciplines.

Thematic Session on Systems Research

Large scale systems like communication networks, large databases and software systems, the Internet, large distributed control systems, businesses and the global economy, are examples of huge, interdependent open information-processing systems with behaviour that is increasingly difficult to predict and control. Modelling, simulation, design and control of such large scale systems in technology, business and the sciences are major issues to address in the coming years and were the subject of discussions held in this thematic session.

Present and prospective developments were discussed as a basis for understanding how to address the challenges of building systems that are robust, resilient, dependable and secure, exhibit multi-purpose functionality, and guarantee operation in mission critical tasks. It was recognised that interdisciplinary research drawing on results from complex systems research is now essential if we are to establish new paradigms within which to address the challenges

ahead. Five research directions were identified.

The first research direction addresses the need to infer system models even when only inconsistent and incomplete information is available about their functioning and interactions. We need to develop techniques for inferring the dynamics of complex systems, the laws governing their interaction, and ways to describe their behaviour, in order to simulate many systems for which there is at present insufficient direct knowledge. Such systems occur especially in ecology, medicine, molecular biology, certain technological-information processes such as the internet, and systems in management, finance, and economics whose behaviour is very dependent on the human in the loop.

The second research direction addresses the design of human responsive ICT systems that integrate well with humans and adapt well to human needs. It was recognised that today we can no longer treat ICT systems as separate from their human users and others affected by them, and that an ICT system and the context in which it is deployed together form a system. Research is required to establish new design principles that accommodate the changing needs and desires of human participants in complex systems, rather than treating them as outside the system, and rather than presuming to know what they need. The research would need to acquire better understanding of human behaviour - especially regarding group man-machine interactions – from which generative theories should inform the organisation of architectures that support and sustain participation, including participative co-design of the systems themselves.

Three complementary research directions address the need to ensure that systems built have the properties we demand. All of them insist on the need to underpin the formal description of such systems.

The first of this set of research directions is to develop underpinning 'foundations' for software-intensive systems. The aim is to make a fundamental leap in the scientific basis of software engineering

technologies to capture evolution and dynamics, selfish interests of individual entities, various levels of bounded rationality, learning aspects and self-emerging behaviour, in a strict, yet tractable way. Advances required include new algorithmic techniques for distributed systems and property-aware compilation and implementation techniques for non-functional properties such as security, safety, scalability, resource optimisation, quality of service, and efficiency, in order to be able to guarantee these non-functional properties.

The other two research directions in this set, namely 'Control and Integrated Design of Complex Systems' and 'Many level, Many Unit, Many Purpose Systems' are closely related. It was recognised that for systems comprising many interacting elements we currently lack techniques for modelling that enable prediction of the types of structures, the dynamic behaviour, and the properties that are likely to emerge at higher, collective levels of activity. A key characteristic of many such systems is that they involve components with uncertain operation and interactions, which are at least partially unpredictable and often not yet formalised in algorithmic terms. A number of biological, social, management and economic systems exhibit these properties, yet they must continue to function adaptably, malleably and resiliently, in the face of such unpredictability. The ambition is on the one hand to establish engineering guidelines that draw inspiration from complex natural, social, technological and economic systems and on the other to establish an underpinning framework of formal or mathematical techniques. Together, they should enable us to find cost-effective solutions to problems that cannot be solved with current techniques.

Thematic Session
on Intelligence and Cognition

Achieving true machine intelligence remains an illusive challenge for the perception, cognition and AI research communities. Panel discussions in this session focussed on new promising research directions for achieving leap progress in this area through the under-

standing of the processes underpinning intelligence and cognition in living organisms.

The vision that was promoted is the one Toward Natural Cognition, where the goal is to build artificial cognitive systems inspired by biology, in particular neuroscience, under the following two assumptions: Cognition by systems interacting with the real world is depending on and is facilitated by their body. The structure of this body, the environment and the body-environment interaction are inseparable from one another.

Traditional cognitive science, cognitive psychology and AI make no commitment to the form of a cognitive system's implementation. Today, especially in cognitive neuroscience and robotics, the infrastructure (ie embodiment) is considered much more crucial to the understanding of cognition. One obvious difference between IT systems and biological cognition is the extent to which biology is self-programming, has adaptive configuration of sensors and effectors, and has extendable processing able to make analogies and cope with novel percepts.

Toward Natural Cognition is aimed at taking a relatively radical step away from classical AI-based IT approaches to cognition toward research on self-organisation and development as a natural framework for cognition. In this context, cognition is seen as more than just an inferential process. It is a property that results from the interaction of an organism with its environment. Embodiment, as the central notion in this vision, is characterised by a number of attributes, such as: Embodiment is intrinsically developmental and is structured by interaction with the environment. It enables affective interaction, the acquisition of meaning from percepts created through sensory-action integration and the grounding of 'concepts' in the agent's sensory-motor and social interaction (which provides the basis for natural language). It facilitates learning by formation of cross-modal associations through induction and generation of correlations. It includes continuous dynamics with discrete attractor states,

provides the basis for grounding and maintains the distinction between the description of cognition by external observers, and its underlying mechanisms.

The underlying research challenges of this vision include:
- The exploration of non-classical computation and the development of robust scalable self-constructing/repairing architectures
- the exploitation of phylogenetic (evolutionary) and ontogenetic (individual) development and the development of cognitive systems with self-

regulation and self-maintenance ('homeostasis') properties
- the investigation and subsequent exploitation of emergent properties of large-scale structures (hardware and simulation) for cognitive processes
- The attainment of high-level cognition by bottom-up organisation.

For addressing the above challenges, relevant research topics to investigate include: new forms of biology-inspired analogue processing and related tools for implementing such large-scale biology-inspired systems; body construction or development and underlying materials

and morphological issues; exploring 'how much embodiment is sufficient', and the relationship between top-down knowledge, innateness and learning.

Toward Natural Cognition and its underlying challenges would require not only inspiration from biology (in particular, neuroscience) but also exploring the relationship between cognition and architecture. Trans-disciplinarity is also a must, offering a synergistic interaction of IT with neuro- or, more generally, life sciences.

Link:
http://www.cordis.lu/ist/fet/7fp.htm

INTAS Calls 2004

In 2004, the calls of INTAS, the International Association for the promotion of cooperation with scientists from the New Independent States (NIS) of the former Soviet Union, either address specific scientific & interdisciplinary subjects (thematic calls), or encourage international cooperation with industrial or national partners (collaborative calls with Airbus S.A.S., Kazakhstan and Uzbekistan).

In line with its mission and strategic plans, INTAS officially published on 2 April 2004 its Thematic and Collaborative Calls 2004. While teams from all NIS countries remain eligible to participate, particular attention will be given, wherever possible, to scientists from NIS countries weakly represented in the international research programmes: Armenia, Azerbaijan, Georgia, Kazakhstan, Kyrgyzstan, Moldova, Tajikistan, Turkmenistan and Uzbekistan. The web-based submission system is now open for:

- INTAS Thematic Call for Proposals for Research Projects and Networks on Information Technology (indicative budget Euro 1 million)
- INTAS Thematic Call for Proposals for Research Projects and Networks on Position Sensitive Detectors (indicative budget Euro 800,000)
- INTAS Thematic Call for Proposals for Research Projects and Networks on Transforming Societies: East and West (indicative budget Euro 1.5 million)

- INTAS Collaborative Call with Airbus for Proposals for Research Projects in the field of Aeronautics (indicative budget Euro 1 million)
- INTAS Collaborative Call with Kazakhstan for Proposals for Research Projects on 'Management of Man-made Pollution' (indicative budget Euro 1.5 million)
- INTAS Collaborative Call with Uzbekistan for Proposals for Research Projects on Sustainable Development through the Use of Local Natural Resources and New Technologies (indicative budget Euro 1 million).

In parallel to these calls, the INTAS programme for Young NIS Scientist Fellowships (indicative budget Euro 2.5 million) fosters the mobility of researchers in Europe and enables them to either pursue or start their career in Science. Additionally, special support to promote innovation and marketing of innovative results derived from INTAS funded projects is offered through the INTAS Innovation Grants with an indicative budget of € 500,000 for 2004.

Deadline
The deadline for all the above INTAS calls is 3 September 2004, 13H00 Brussels time.

Further Information
The General Rules and a Technical Guide on the electronic submission for each of the actions can be obtained on http://www.intas.be, Section 'Funding Opportunities'. For further questions on the INTAS calls 2004, please send an e-mail to infopack@intas.be

For general information about INTAS, please contact the Public Relations Team at intas@intas.be or by fax: +32-2-549 01 56, or check the INTAS web site.

Link:
http://www.intas.be/

Tim Berners-Lee receives Millennium Technology Prize

Ms. Tarja Halonen, President of the Republic of Finland, presented the first Millennium Technology Prize to W3C Director Tim Berners-Lee on 15 June in Helsinki.

The honor, which is accompanied by one million euros, is bestowed by the Finnish Technology Award Foundation as an international acknowledgement of outstanding technological innovation that directly promotes people's quality of life, is based on humane values, and encourages sustainable economic development.

Photo: Heikki Savolainen

President of the Republic of Finland, Tarja Halonen, presenting the Millennium Technology Prize to Tim Berners-Lee.

"The Web has significantly enhanced many people's ability to obtain information central to their lives," says Pekka Tarjanne, former secretary-general of the International Telecommunication Union and chairman of the International Award Selection Committee. "The Web is encouraging new types of social networks, supporting transparency and democracy, and opening up novel avenues for information management and business development."

Berners-Lee was nominated by two organisations: The Royal Academy of Engineering in London, whose Fellow he is, as well as the Finnish IT center for Science.

Berners-Lee's selection was made unanimously by the board of the Finnish Technology Award Foundation at an April 14 meeting based on the recommendation of the International Award Selection Committee.

Link:
http://www.technologyawards.org/

W3C leads a New European Project: Multimodal Web

Financed by the European Commission's IST Programme and led by the W3C, Mulitimodal Web (MWeb) will support the development and adoption of industry standards ('W3C Recommendations') enabling multimodal Web access using mobile devices. MWeb is a key opportunity for Europe leverage key strengths (mobile technology) in order to improve Web technology developments and increase overall competitiveness. Multimodal Web access will likely be the first widespread practical use of multimodal technology, and will have a similar impact on the adoption as the original Web had on adoption of Internet technology.

The MWeb project includes both European outreach activities on first-generation W3C multimodal Recommendations and technical support required for developing a second generation of specifications with significant European participation.

The MWeb outreach activities will increase awareness and visibility of W3C's multimodal specifications within Europe. W3C will promote the Multimodal Web by taking part in public events through talks and panels, by organizing seminars, and by editing brochures and posters, etc.

The MWeb technical activities will provide European research and industry with competent partners within the W3C. Through W3C workshops, W3C hopes to raise European participation in W3C's multimodal work.

MWeb Milestones
W3C Workshop on Multimodal Interaction
19-20 July 2004, Sophia Antipolis
The goal of this public workshop is to get feedback from multimodal user communities less well represented in the W3C Multimodal work (eg the automotive industry/telematics, the home entertainment industry, healthcare, aerospace and manufacturing), along with feedback and input from the multimodal research community in order to strengthen the W3C Multimodal Interaction (MMI) Framework.

W3C Workshop on Metadata for Content Adaptation
12-13 October 2004, Dublin, Ireland
A document profile is a set of content metadata using XML or RDF vocabularies that will help achieve device independent presentation of any Web content. The scope of this workshop is to describe sharable, transparent and machine understandable explicit semantics at the content level (image, link, etc.), that the Device Independence Working Group (DIWG) will then frame as a Document Profile specification.

Links:
MWeb: http://www.w3.org/2004/MWeb/Overview.html
MMI Workshop: http://www.w3.org/2004/02/mmi-workshop-cfp.html
MMI Framework: http://www.w3.org/TR/mmi-framework/
DP Workshop: http://www.w3.org/2004/06/DocProf-WS/CFP.html
DIWG: http://www.w3.org/2001/di/#wg

New W3C Activities

W3C Math Activity Relaunched

The Math Interest Group is chartered to continue the task of facilitating the use of mathematics on the Web, both for science and technology and for education. This involves:

- maintenance of the recent version 2.0 (Second Edition) of the MathML specification (W3C Recommendation, 21 October 2003)
- encouragement of its wider deployment
- preparation of errata as appropriate
- continued liaison with other Working Groups within the W3C to ensure that the potential of MathML is realized
- and relations with other organizations.

The W3C Math Activity is designed to strengthen the position of MathML and enhance the manipulation of mathematics on the Web. The use of MathML in Web documents is encouraged by the W3C and should significantly contribute to the usefulness of the Web for science, technology and education.

Synchronized Multimedia Activity Relaunched

The Synchronized Multimedia (SYMM) Activity is composed of the Synchronized Multimedia Working Group and the Timed Text Working Group.

The SYMM Working Group will extend the SMIL 2.0 W3C Recommendation which enables simple authoring of interactive audiovisual presentations. This implies:

- revision of the current SMIL 2.0 Recommendation to incorporate known SMIL 2.0 errata
- SMIL 2.1 for mobile devices incorporating features currently in use within the mobile industry, where SMIL is used as the basis of the MMS (Multimedia Messaging Service).
- an enhanced future version of SMIL, addressing extended functionalities for professional multimedia content authoring.

The Timed Text Working Group is developing XML vocabulary and document types for subtitles and captions.

Links:
http://www.w3.org/Math/
http://www.w3.org/AudioVideo/

Latest W3C Recommendations

- **Document Object Model (DOM) Level 3 Core Specification**,
  7 April 2004, Jonathan Robie, Mike Champion, Steve Byrne, Arnaud Le Hors, Philippe Le Hégaret, Lauren Wood, Gavin Nicol
- **Document Object Model (DOM) Level 3 Load and Save Specification**,
  7 April 2004, Johnny Stenback, Andy Heninger.

An exhaustive list of all W3C Technical Reports:
http://www.w3.org/TR/

W3C Open Source Software New Releases

W3C Markup Validator Upgraded

The W3C Markup Validation Service is a free service that checks documents like HTML and XHTML for conformance to W3C Recommendations and other standards. This latest release furthers improvements on the usability of the tool.

W3C Link Checker

The W3C Link Checker verifies the state of all the links in an HTML document, helping Web authors spot broken links or access control issues. In addition to being available as a popular online service, the latest versions are also available as a standalone tool.

Links:
http://validator.w3.org/
http://validator.w3.org/checklink

ERCIM is the European host of W3C.

Automated Software Engineering

Introdu

**by Paul Grünbacher
and Yves Ledru**

Software engineering is concerned with the development and evolution of large and complex software-intensive systems. It covers theories, methods and tools for the specification, architecture, design, testing, and maintenance of software systems. Today's software systems are significantly large, complex and critical, that only through the use of automated approaches can such systems be developed and evolve in an economic and timely manner.

Automated software engineering applies computation to software engineering activities. The goal is to partially or fully automate these activities, thereby significantly increasing both quality and productivity. This includes the study of techniques for constructing, understanding, adapting and modelling both software artefacts and processes. Automatic and collaborative systems are both important areas of automated software engineering, as are computational models of human software engineering activities. Knowledge representations and artificial intelligence techniques applicable in this field are of particular interest, as are formal techniques that support or provide theoretical foundations.

Automated software engineering approaches have been applied in many areas of software engineering. These include requirements definition, specification, architecture, design and synthesis, implementation, modelling, testing and quality assurance, verification and validation, maintenance and evolution, configuration management, deployment, reengineering, reuse and visualisation. Automated software engineering techniques have also been used in a wide range of domains and application areas including industrial software, embedded and real-time systems, aerospace, automotive and medical systems, Web-based systems and computer games.

The ASE community meets annually at the International Conference on Automated Software Engineering, sponsored by IEEE and ACM (visit http://www.ase-conference.org for information on the current conference). The Journal of Automated Software Engineering (Kluwer) is also a natural place to find recent advances in this discipline.

This special issue includes three invited articles and around thirty submitted articles. The invited articles by Motoshi Saeki and Andrew Ireland illustrate the links between ASE and artificial intelligence techniques. Motoshi Saeki's work builds on ontologies to favour the reuse of requirements and the components of requirements. Andrew Ireland's work applies automated reasoning techniques, in particular proof planning, to the static analysis of programs. Wolfgang Emmerich's invited contribution addresses the development of trusted and 'quality of service'-aware distributed systems. This is particularly important as such systems are increasingly deployed across organisational boundaries.

The articles under this special theme describe research in automating software development activities or providing collaborative support to users engaged in such activities. These include the following:
- articles in the Requirements group address tool support for the scenario-based discovery of requirements, animation and validation of business transactions, or the use of natural language processing to improve the quality of requirements
- several articles deal with Program Understanding and Architecture and present research in software maintenance support via reengineering, automatic support for software reuse and support for designing and validating architectural specification

ction

- a larger group of articles addresses Software Testing, covering aspects such as automatic generation of test cases, testability analysis, or test-suite planning and derivation using the Unified Modelling Language (UML)
- the articles on Software Verification deal with model checking and the verification of groupware protocols
- another group of articles addresses Aspects and Language Technology, with topics including aspect-oriented techniques for software adaptation for pervasive computing, use of language technology for the analysis and transformation of large software systems, or the identification of cross-cutting concerns in embedded C code
- the articles on Configuration Management and Deployment include aspects such as tool environments for software maintenance and adaptation, automation of software release and deployment, or frameworks and associated services for deploying and maintaining networks of collaborative, distributed environments.
- finally, several articles demonstrate the use of Models in software engineering; transformation and verification of models, model transformation languages, and model-driven development of pervasive systems.

Please contact:
Paul Grünbacher
Johannes Kepler University Linz, Austria
E-mail: paul.gruenbacher@jku.at

Yves Ledru, Laboratoire Logiciels Systèmes Réseaux/IMAG – Institut d'Informatique et de Mathematiques Appliqués de Grenoble, France
E-mail: Yves.Ledru@imag.fr

ARTICLES IN THIS SECTION

Ontology-Based Software Development Techniques

by Motoshi Saeki

An on-going research project at the Tokyo Institute of Technology is looking at supporting software-requirements elicitation and composing software from re-usable architectures, frameworks, components and software packages. The relevant techniques are developed through the use of ontology and its reasoning mechanism so as to maintain semantic consistency.

The first steps in software development processes are requirements analysis and design phases. Software engineers can have a great effect during these steps on the quality of the software produced and on its development cost. Recently, it has become less common to develop new software from scratch. Instead, software is developed by adapting and/or combining existing re-usable software architectures, components such as COTS, frameworks such as Struts and Turbine, and software packages such as ERP and CRM (in this article, we describe these as 're-usable parts'). In this situation, software engineers need to have sufficient knowledge and skills to employ these re-usable parts correctly and effectively. In particular, the engineers working for requirements analysis and design must be able to select and adapt existing re-usable software parts according to customers' requirements.

Several techniques have been studied and developed for eliciting requirements from the customers and prioritising the applicable re-usable parts selected from these.. However, these techniques cannot deal with possible ambiguities in customer requirements. Consequently, these applications often elicit different requirements from those the customers truly want, and select re-usable parts that are semantically wrong, even if the requirements could be correctly elicited. This leads to the development of software that does not meet the customer's requirements.

In this research, we focus on semantic processing of requirements and re-usable parts by using ontology techniques. We are developing techniques that enable us to elicit semantically correct requirements and to select the implementation

structures that are semantically suitable for those requirements. We have an ontology system whose concrete structure is the thesaurus of domain-specific words for each problem domain. In our new methodology, requirements are elicited based on the ontology, or more concretely, the ontology system guides the analysts' activities to elicit requirements. As a result, the meaning of the elicited requirements can be represented with a set of relevant words included in the ontology system. Re-usable parts are also semantically related to the words in the ontology system; that is, the meaning

of each re-usable part is provided by the ontology system. Consequently, we have a kind of database of re-usable parts, including their semantic information. The selection of re-usable parts is performed by means of word matching in the ontology system. The re-usable

parts are then adapted so as to satisfy the requirements and are integrated into a final product.

Our ontology system has two layers; one for requirements elicitation and the other for re-usable parts. By establishing relationships between the two layers, the ontology system can play a role in bridging gaps between a requirements specification and an architectural design at a semantic level.

The following scenario and Figure 1 illustrate the basic idea. An analyst

Ontology system.

elicits requirements by exploring and using the relevant vocabularies in the ontology as examples. Suppose that he adopts an ontological concept (vocabulary) A, which relates to concept B that connects to re-usable parts such as Struts. He can also write a specification

document by using a fragment of requirements specifications (such as a sample sentence using vocabulary A). The strength of the relationship to B then suggests that he can compose the final product using Struts.

To (partially) automate the construction of the ontology system for each problem domain, we explore the application of natural language processing for extracting domain-specific words from documents on the Internet. Our research project consists of three sub-projects: (i) constructing the ontology on a problem domain by processing natural language texts describing the domain and by mining them into common vocabularies as domain-specific concepts; (ii)

extending the existing requirements-elicitation techniques such as Goal-Oriented Analysis and Scenario Analysis so that they can be guided by the constructed ontology system; and (iii) developing the techniques to select the semantically suitable re-usable parts and adapt them in a semantically consistent manner. These techniques and their supporting tools allow novice engineers to easily perform requirements elicitation, design and implementation through the adaptation and integration of re-usable parts. Since the meanings of both the requirements and the implementation structures are represented by a unified ontology system, we can measure semantic relationships between them by analysing the relevant ontology structures, thereby

determining the 'achievability' of the requirements based on the selected implementation structure.

This project is currently in its first stage, and we have recently established the techniques to process natural language texts describing business-processing domains, and to extract common vocabularies and their relationships as a thesaurus. We are now tailoring a practical ontology system and developing a reasoning mechanism on the ontology system.

Please contact:
Motoshi Saeki
Tokyo Institute of Technology, Japan
E-mail: saeki@titech.ac.jp

TAPAS — Trusted and Quality-of-Service Aware Provision of Application Services

by Wolfgang Emmerich, Santosh Shrivastava, Fabio Panzieri, Jon Crowcroft and Werner Beckmann

The The TAPAS project focuses on the principles, architectures, methods and tools that are needed for the systematic engineering of quality of service aware distributed systems. To meet this objective, TAPAS brings together researchers and practitioners with a background in software engineering, distributed systems and networking. TAPAS is funded for three years through the IST programme.

TAPAS acknowledges that distributed systems are increasingly deployed across organisational boundaries. Examples of such systems include applications that are hosted by third party service providers, storage resource brokerage or grid-computing. In such settings it becomes important that not just functional correctness of the application is assured, but also that the non-functional qualities of service provision are managed. This management includes the definition of required and provided qualities of service in service level agreements, the monitoring of service level agreements through QoS aware middleware, and fault-tolerant computing through group communication and replication.

TAPAS has defined a language to define service level agreements (SLAng). The semantics of SLAng are defined using the model denotational approach to semantics definition that has been used by the precise UML group. The SLANG semantics definition relates a class model of the syntax of SLANG with a model describing events that can be observed in the environment in which services are provided. The semantics definition then uses OCL constraints over the relation between these models to define the meaning of SLAs.

The main advantage of having a formal SLA language (over the current approach of using carefully crafted lawyer speak) is that SLAs can be reasoned about. We are interested in

several forms of reasoning. We would like to support the automated derivation of SLA validators. Such validators are given a list of events that have been obtained by monitoring service providers and consumers and can then decide whether an SLA has been met. We have built a meta-SLA compiler, that can generate an SLA validation component from the UML/OCL definition of the SLA language semantics. We are also interested in reasoning about various aspects of service composition. Given that services may be provided by different parties and that they can be composed into some higher-level service the question arises as to whether the service quality guaranteed for the component services is sufficiently strong to allow a service provider to enter into a

service level agreement for the higher-level service. To this extent, we have successfully used queuing network models and stochastic process algebras to support quantitative reasoning about service level agreements and their composition.

We found it very difficult to achieve quality of service-aware distributed applications without also modifying the middleware that is used as a run-time environment for the components that provide services. The TAPAS team has therefore made a number of extensions to an existing application server (JBoss). Firstly, the project used the extension mechanisms of JBoss (interceptors) to provide monitoring capabilities about

events of interest, such as the time that it takes to remotely invoke an enterprise java bean or the time it takes to execute a JDBC query. These events can then be fed to an SLA validator component in order to reason about SLAs. Secondly, we are modifying the availability and reliability support of JBoss by improving the clustering mechanisms so that JBoss is capable of running heterogeneous clusters and in response to changes in load can re-adjust the replication strategy.

TAPAS is the first JBoss Academia project. JBoss Academia projects are intended to accelerate the dissemination of research ideas into practice. To this end, we are working with the JBoss open

source community. In the next version of the TAPAS platform, we will generalise our current use of interceptors and use the emerging JBoss aspect oriented programming mechanisms to provide our extensions for QoS-aware service provision.

Please contact:
Wolfgang Emmerich
University College London, UK
E-mail: w.emmerich@cs.ucl.ac.uk

Towards Increased Verification Automation for High Integrity Software Engineering

by Andrew Ireland

The proliferation of safety and security critical software applications represents a key challenge for today's software engineers. Within the context of the SPARK approach to high integrity software development, the NuSPADE project is addressing this challenge through the integration of novel automated reasoning and program analysis techniques.

Formal verification involves the use of mathematical logic and formal reasoning techniques in proving properties of software and hardware systems. Early work on formal verification focused on proving full functional correctness. In the area of software verification, the complexity of software and the lack of effective tools limited the applicability of this approach. The focus has moved away from full functional correctness to property based formal verification. That is, proving properties of software that are of interest to the developers and the customers. This property based approach can be used to verify desired properties as well as undesirable properties, ie show the presence of software bugs. The advantage of the property based approach is that the verification task is much simpler than full functional correctness and therefore increases the level of automation that can be achieved.

Moreover, the properties of interest are typically generic, ie properties that are applicable across a wide range of applications. This means that programmers are not required to hand craft the properties. Notable successes in this area are the SLAM (Microsoft Research) and ESC/Java (HP Labs) projects. The NuSPADE project builds upon another success story for the property based approach, namely the SPARK approach to high integrity software development (Praxis Critical Systems). The SPARK approach advocates 'correctness by construction', where the focus is on bug prevention rather than bug detection. SPARK has been applied successfully across a wide range of applications including railway signalling, smartcard security and avionics systems such as the Lockheed C130J and EuroFighter projects. The approach has been recently (April 2004) recognized by the US

National Cyber Security Partnership as one of only three software development processes that can deliver sufficient assurance for security critical systems.

The SPARK programming language is defined as a subset of Ada which is expressive enough for industrial applications, but restrictive enough to support rigorous analysis early in the development process. In particular, SPARK supports a language of program annotations and associated tools. The annotations allow the programmer to specify the intended behaviour of their programs, while the toolset supports the verification of the specifications. The formal verification capabilities of SPARK are most commonly used for what are known as exception freedom proofs, ie proving that a system is free from run-time errors. Within safety critical applications, run-time errors may
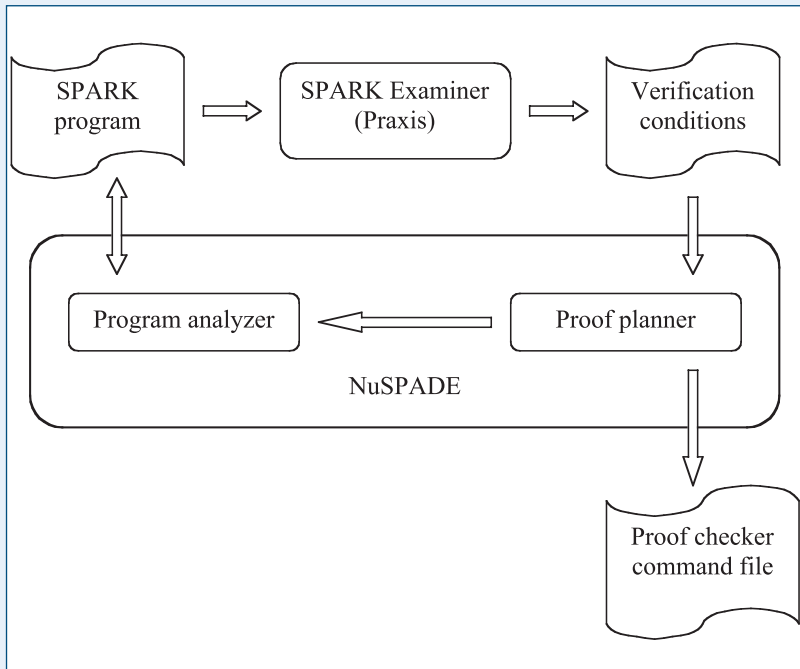
NuSPADE integrates proof planning and program analysis within the context of the SPARK approach to high integrity software development. The SPARK Examiner (Praxis) performs various levels of quality and consistency checks as well as generating verification conditions (proof obligations). For verification conditions that the SPARK toolkit (SPADE Simplifier) fails to prove, a proof planner is used to search for a proof. During this proof search a program analyzer may be used in order to induce auxiliary program annotations (specifications) that are necessary in order for the proof search to succeed. A successful proof planning attempt generates a command file description of a proof which is then checked using the SPARK toolkit (SPADE Proof Checker).

give rise to catastrophic failures, eg an integer overflow run-time error led to the loss of Ariane 5. The same holds true for security critical applications, eg buffer overflows have been the most common form of security vulnerability in the last ten years. The ability to verify that software applications are free of such undesirable behaviour has obvious social and economic benefits. The SPARK approach has had significant success in automating exception freedom proofs through the SPADE Simplifier, a special purpose automated reasoning tool.

The goal of the NuSPADE project has been to build upon this success and increase the level of proof automation in general, and for exception freedom proofs in particular. There are two key problems. Firstly, to increase the proof automation and, secondly, to automate the program annotations that are required in order to support the proof process. Our starting point is proof planning; a computer-based technique for automating the search for proofs. At the core of the technique are high-level proof outlines, known as proof plans. Proof plans encode heuristic knowledge that is used in automating the search for proofs. A key feature of proof planning is that it separates proof search from proof checking. This gives greater flexibility in the strategies that can be used in guiding proof search as compared to conventional proof development envi-

ronments. Proof critics are an example of this greater flexibility. Proof critics support the automatic analysis and patching of proof planning failures. Within the NuSPADE project we have broadened their role, ie we use proof critics to provide a tight integration between proof planning and program analysis (see Figure). For example, certain classes of proof-failures can be attributed to the need for additional knowledge about the program being verified. This additional knowledge is added through SPARK program annotations. Typically these annotations are crafted by the programmer after they have analyzed a proof-failure. Within NuSPADE we have been able to reduce the need for hand crafted annotations, ie we have been able to automate the proof-failure analysis and use the analysis to generate the required program annotations automatically.

NuSPADE is a three year project which is funded by the UK's Engineering and Physical Sciences Research Council (EPSRC) and is in collaboration with Praxis Critical Systems. The project is entering its final evaluation phase which will involve applying the techniques to industrial strength applications. A follow-on project is planned through EPSRC's Research Assistants Industrial Secondments (RAIS) scheme. The aim of the RAIS scheme is to support knowledge transfer where a research project

has had a strong industrial collaborative component. The RAIS project will provide the first step towards technology transfer.

The technique of proof planning originated within the Mathematical Reasoning Group (MRG) at the University of Edinburgh. The NuSPADE project and the development of the proof planning technique within the area of software verification has been pioneered within the Dependable Systems Group (DSG) at Heriot-Watt University. The DSG and the MRG have collaborated closely over nearly ten years in the area of automated software engineering.

An Automatic Tool for Improving the Quality of Software Requirements

by Giuseppe Lami and Gianluca Trentanni

The Quality Analyzer for Requirements Specifications (QuARS) is an innovative tool that enables the user to analyse natural language requirements automatically.

Natural Language (NL) is still the most common way to express software requirements. In practice, there are numerous tools and techniques for managing such requirements; many are oriented towards the definition and provision of configuration management, and control the distribution of requirements. However, there is a scarcity of automatic support for the quality analysis of requirements expressed in NL. In the software industry, ambiguity analysis and consistency and completeness verification of software requirements are performed by humans, generally through a tedious clerical process that consists in reading requirements documents and looking for linguistic errors. For these

Figure 1:
The Natural Language requirements analysis process with QuARS.

reasons, we have developed QuARS (Quality Analyser for Requirements Specifications), an innovative tool that enables the user to analyse requirements expressed in NL automatically.

QuARS performs a lexical and syntactical parsing of software requirements expressed in NL, and provides the following functionalities.

Defective Sentence Identification (Expressiveness Evaluation)

Similar to any other evaluation process, the quality evaluation of NL software requirements must be conducted against a model. The Quality Model we have defined for the Expressiveness property of the requirements produces an evaluation that is quantitative (ie that allows the collection of metrics), corrective (ie that could be helpful in the detection and correction of defects) and reproducible (ie that provides the same output against the same input in every domain).

The Expressiveness quality model consists of three features, to be evaluated by means of indicators. Indicators are linguistic components, directly detectable and measurable in the requirements document, which reveal a defect in requirements. Special dictionaries contain the indicators QuARS needs for its analysis.

The Expressiveness features are:
- non-ambiguity: the property of having a unique interpretation
- understandability: the property of being fully comprehensible both when

used by the software developer and when read by the user in the Requirements Specification Document
- specification completion: the ability of each requirement to uniquely identify its object or subject.

QuARS performs a lexical and syntactic analysis of the input file in order to identify those sentences containing defects according to the quality model.

Requirements Clustering (View Derivation)

QuARS also provides support for consistency and completeness analysis by means of the View derivation functionality. A View is a subset of the input requirements document, composed of those sentences dealing with a particular argument. The availability of Views makes the detection of inconsistencies and incompleteness easier because the reviewer need consider only smaller sets of sentences, in which possible defects can be found with much less effort.

Using QuARS

QuARS provides a systematic and disciplined NL-requirements analysis process (see Figure 1). Input to the tool is composed of the requirements document which is to be analysed. The permitted format of this input file is plain text. This file is given to the syntax parser component, which produces a new file containing the parsed version of its sentences. The tool relies on a set of indicator-related dictionaries; these may contain either terms indicating a defect type according to the quality model, or the domain dictionaries used for the View derivation. The dictionaries are in simple text format. Once the user selects the type of analysis to be performed, the corresponding dictionary is made available and can be tailored, if desired, according to the application domain.

Figure 2: QuARS expressiveness analysis.

Figure 3: View Derivation.

When the Expressiveness analysis is performed, the list of defective sentences is displayed in the GUI's output frame and a log file is created. The defective sentences can be tracked in the input requirements document and corrected if necessary (see Figure 2). Metrics measuring the defect rate and the readability of the requirements document under analysis are calculated and stored.

The output of the View derivation function is a log file containing the necessary clusters of requirements, ready to be used for consistency and completeness analysis purposes. The graphical representation (in an MS Excel graph) of the number of occurrences of these sentences belonging to a View in single sections of the document is also provided (see Figure 3).

Because of its precision and ease of use, the QuARS tool can greatly improve the analysis of NL requirements. Experiments aimed at evaluating the impact of QuARS in the software development process and on the quality of the final product are currently in progress.

Efficient: A Framework for Animating and Validating e-Business Transactions

by Bertrand Grégoire, Christophe Incoul, Sophie Ramel, Michael Schmitt, Laurent Gautheron, Pierre Brimont and Eric Dubois

To support the modelling and validation of new ebXML transactions, the CRP Henri Tudor is developing a tool set within the context of the EFFICIENT project supported by the National Research Fund (FNR) in collaboration with the University of Luxembourg and the CRP Gabriel Lippmann.

ebXML (Electronic Business using eXtensible Markup Language) is becoming the new international standard for the specification and deployment of complex B2B transactions, progressively replacing EDI (Electronic Data Interchange). Such transactions are inherently distributed, involving many players exchanging XML messages according to complex flows and rules. This complexity hampers validation of the correctness of a modelled business transaction by business experts. To alleviate this problem, scientists at CRP Henri Tudor, Luxembourg, have developed a tool set, consisting of a CASE tool for modelling ebXML transactions and an animator for validating the modelled transactions. The animator tool allows business experts to understand and 'play' business transactions before they are implemented.

The tool set consists of an extension of a commercial UML-based CASE tool that supports the modelling of ebXML business transactions, and an animator tool based on a workflow engine that supports the execution of the above UML models. The animator allows business experts to cooperatively validate transaction models at the time they are built, that is, before their implementation has begun. Rather than simulation, we prefer to use the word 'animation' since the validation is done in an interactive way, with each business expert playing a role and participating in the execution of the transaction by receiving messages and sending answers. By doing this, business experts can validate the transaction by playing out a variety of possible

scenarios that include different messages.

In our project, we use a four-layered approach (see Figure 1). The top two layers are extensions that we are investigating regarding the ebXML standard, while the other layers are fully compatible with the proposed standard in terms of UML models and of XML schemas produced.

The value layer provides the global business view, which gives an understanding of the context of the transaction and allows its viability to be evaluated from the benefits it provides to the parties involved. Understanding the business requires knowledge about different elements, including the customer demands a business is willing to address, the products proposed to fulfil these demands, the activities required to produce these products and the infrastructure underlying the production requirements, including partnerships and some risk-management considerations.

The business layer gives a general overview of the business transactions between organisations. In this layer, the global structure of each business transaction is depicted with UML diagrams. A Use Case diagram specifies the global structure of the business processes underlying the business transaction, introducing its main actions. A class diagram then specifies the information manipulated in the business transaction, called the business domain. In addition, business rules can be specified in natural language. At this level, the concept of message does not yet exist.

The specification layer details the message-based structure of a business transaction. This detailed specification is needed to support the B2B automation of business transactions. The flow of the business transaction is specified with a UML activity diagram. The content of each message is specified within a class diagram. The activity diagram refines the Use Case diagram at the business layer. Each class diagram is a particular view of the global class diagram at the business layer. The message may have attached to it a business rule defined at the business layer in order to constrain the message's content, as well as new rules relating the messages to each other. All these rules are expressed in a formal way.

At the technical layer, the business transaction is executed using the animator. The infrastructure used at the technical layer (workflow engine, interfaces, XML messages, rules checker) is automatically configured and fully generated from the UML models developed at the specification layer.

The animator architecture is distributed and Internet-based, allowing business experts from different sites to participate in the animation of the same business transactions using a simple Web browser. The animator automatically checks the business rules defined upon the message content.

Our technological choices have followed two principles: choose open-source tools wherever possible, and follow standards when they exist. As a result, we use:
- WMFOpen, an open-source workflow engine, to animate the transactions (configured using the XML Process Definition Language (XPDL) standard definitions)
- SOAP, as the core messaging protocol
- XML:DB API (http://www.xmldb.org/), to access the eXist XML database (http://exist.sourceforge.net)
- W3C XMLSchema, for the description of messages
- Chiba, based upon the XForms recommendation of the W3C, for the display of the messages.

This choice to respect standards and use open-source technology allows us a greater flexibility and interoperability.

In conclusion, the animator, tightly integrated with state-of-the-art XML technology, shows the feasibility of business transactions, possibly implemented in ebXML compliant infrastructures.
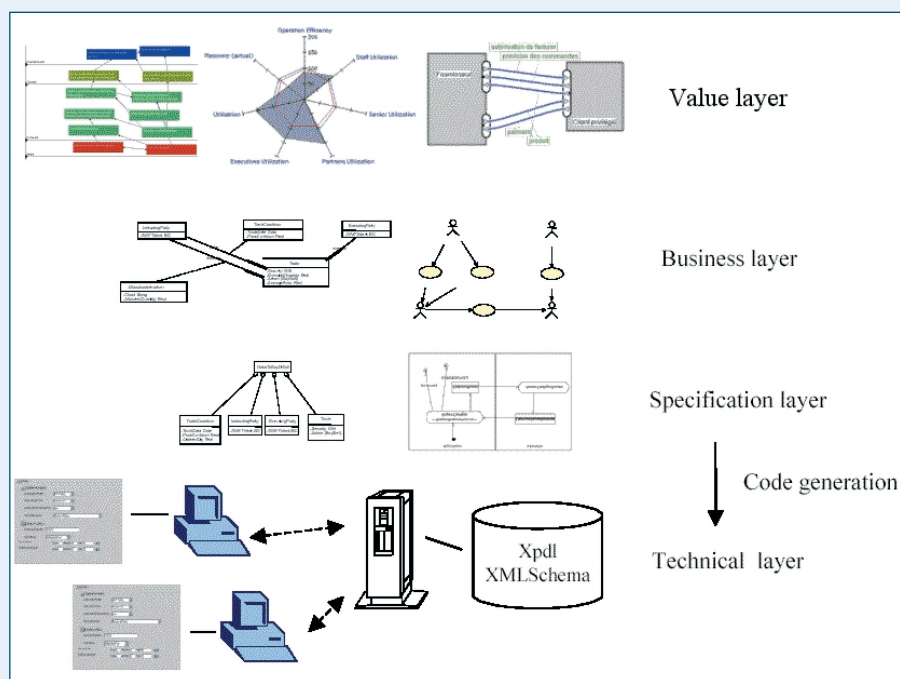
The four-layered model of a B2B transaction.

Discovering Requirements with Scenarios: The ART-SCENE Solution

by Neil Maiden

Discovering requirements such as what people want from new computer systems remains a challenging problem for most European organisations. ART-SCENE is an innovative new solution to this problem. We already know that scenarios – stories that describe how the future system might work – help us to discover these requirements. ART-SCENE takes this a stage further by enabling organisations to generate scenarios automatically and use them systematically in discovering people's requirements for new systems.

Although scenarios are commonplace in systems development, developers rarely know how many scenarios to produce, what their content and structure should be, and how they should use them to discover new requirements from users. As a result, most software developers use scenarios in an ad hoc, non-optimal way. What they need are tools that are clear and simple –to –use, which allow them to generate and use scenarios more systematically, and hence effectively.

ART-SCENE is one solution to these problems. It is a process with Web-enabled tool support that organisations can use to generate and walk through scenarios, and thus discover the complete and correct requirements for new computer systems. It enhances current Rational Unified Processes and Use Case approaches to systems development.

ART-SCENE was developed by City University's Centre for HCI Design in London. Its origins were in the EU-funded Framework IV 21903 'CREWS' long-term research project. Since then ART-SCENE has been evaluated and extended in the UK EPSRC-funded SIMP project and bi-lateral projects, primarily with Eurocontrol and the UK's National Air Traffic Services.

Important ART-SCENE features include:
- simple-to-use tools for specifying and parsing Use Cases
- automatic generation of normal and alternative course scenarios from Use Case specifications
- guided scenario walk-throughs using Web-enabled tools
- multimedia representation of scenarios to improve discovery of requirements
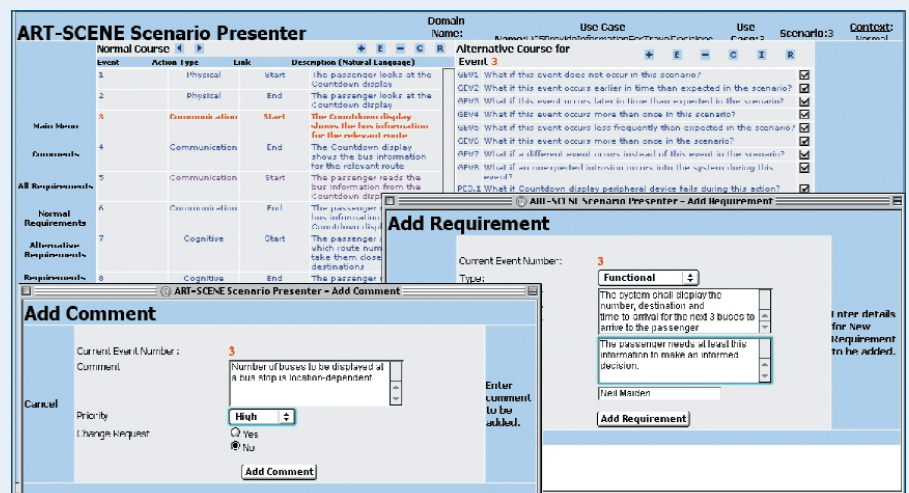
ART-SCENE example.

- scenario walk-throughs on PDAs, to enable the walk-throughs to occur in the user's workplace
- support for standard requirements processes such as VOLERE
- specialisation of ART-SCENE to application domains
- process and facilitation guidelines for preparing and running ART-SCENE scenario workshops.

Typically, a project team using ART-SCENE writes Use Case specifications using ART-SCENE structured templates, style and content guidelines. This specification is then parameterised and parsed to enable ART-SCENE's two-step scenario generation algorithm to generate one or more scenarios.

In the first step, the algorithm generates normal course scenarios from the action-ordering rules and generation parameters in the Use Case specification. Each different possible ordering of normal course events is a different scenario. In the second step, the algorithm generates candidate alternative courses, which are expressed as 'what-if' questions for each normal course event, by querying a database containing over fifty different classes of abnormal behaviour and state in socio-technical systems. These classes are based on error taxonomies in the cognitive science, human-computer interaction and safety-critical disciplines. The algorithm generates, for example, alternative courses that describe events not occurring, actions not being completed, and human-agent mistakes, machine failures and interaction failures occurring. Examples are shown in the ART-SCENE example.

ART-SCENE delivers the generated scenarios with innovative Web-enabled tools such as the Scenario Presenter shown. The Scenario Presenter presents a scenario in four parts. The left-side menu provides different functions for viewing the scenario and the requirements generated for it. The top-line buttons offer

walk-through functions (eg next or previous event) and functions to add, edit or delete events, comments and requirements. The left-hand main section describes the normal course event sequence for the scenario. Each event describes the start or end of an action, thus enabling a scenario to describe concurrent actions in this text-list form. The right-hand main section describes generated alternative courses for each normal course event, presented in the form of 'what-if' questions. Different alternative courses are presented for different normal course events.

Some of the most important features are accessed using the top-line buttons. Each major feature is available either for the selected event in the normal course (accessible above the normal course event sequence) or the selected alternative course (accessible above the alternative course list). The most important features are the add comment [C] and add requirement [R] features. A user can enter a comment or requirement associated with any normal or alternative course event at any time during a walk-through. To add a requirement, the Scenario Presenter allows the user to specify the type, description, rationale and source of the requirement – a subset of the VOLERE requirements shell.

So far, ART-SCENE has been successfully applied to specify several major air-traffic management systems, including Eurocontrol's Conflict Resolution Assistant and Departure Management systems, the latter in conjunction with the UK's National Air Traffic Services.

We have also extended ART-SCENE through collaborations, in particular with the RWTH-Aachen in Germany to add multi-media functionality, and with Johannes Kepler University Linz in Austria to develop a PDA version of ART-SCENE. An ART-SCENE support team has been established at City University. New ART-SCENE development projects and roll-outs are currently being planned, and we welcome further opportunities to develop, evaluate and use ART-SCENE in requirements processes across Europe.

Link:
www.soi.city.ac.uk/artscene

Please contact:
Neil Maiden, City University, London, UK
Tel: +44 20 7040 8412
E-mail: N.A.M.Maiden@city.ac.uk

Automated Support for Agile Software Reuse

by Mel Ó Cinnéide, Nicholas Kushmerick and Tony Veale

Agile software development methodologies tend towards sparse up-front design and minimal commenting of source code. Researchers in the Department of Computer Science, University College Dublin are investigating ways of providing automated support for software reuse in this code-centric context.

In recent years there has been a trend towards lightweight, flexible software development methodologies, collectively known as Agile Processes. In these approaches the focus is on the source code produced rather than design documentation or methodological considerations. This presents a new challenge to the age-old problem of software reuse, specifically the task of recommending software components. Approaches that involve design-level reuse (eg, based on UML descriptions) are no longer valid as this design documentation is typically not produced. Approaches that rely on the programmer adding extra information to the code are also bound to fail in a context where minimal commenting is the norm.

One of the novel approaches taken by UCD researchers has been motivated by the existence of large open-source repositories such as SourceForge. Consider a programmer building an application using a framework such as Swing. SourceForge contains a lot of information on how best to develop Swing applications, but it is expressed as raw source code and it is hard to determine which particular components are of benefit in any particular programming effort. We have applied Collaborative Filtering (CF) to the problem. CF is based on the premise that a group of users who share preferences for certain items are likely also to agree on future items. In this context, the user is the class the programmer is currently developing and the items in question are existing components such as classes and methods.

During the development of the software, we use the partial class written by the programmer so far in order to find other similar classes. The classes and methods used by these similar classes are then suggested to the programmer as suitable candidates for reuse. This approach initially seems naïve, but a study performed on the Swing classes in SourceForge proved extremely promising. For example, when a class was only half specified by a programmer, our approach can predict with over 50% accuracy the remaining components the programmer will need to use in order to complete the class.

While Agile Processes reduce the potential for reuse as outlined above, they provide added possibilities as well. Programmers are encouraged to reduce commenting but to improve the comprehensibility of the code itself. Because of this, names of classes, methods and fields tend to be longer and more expressive. This has led UCD researchers to start examining the meanings of these names in an effort to improve reuse possibilities.

WordNet is an on-line ontology of the English language that organises the lexicon into sets of synonyms and defines various relationships between these synonym sets, including inheritance and

composition relationships. This enables components to be suggested to a programmer based on the names already used in the program. As a simple example, if a programmer introduces a class named Student, the system can suggest Person as a suitable superclass, based on the fact that Person is a superclass of Student in the WordNet ontology. If the programmer subsequently introduces a Lecturer class, the system can predict that a university application is being developed and provide either a nascent set of stub classes that model a university's structure, or suggest classes to reuse from a repository. This approach

to software reuse is predicated on user-defined names in current software being English words. Our study of one large application from SourceForge demonstrated that over 60% of class and method names are fully defined in WordNet and a further 25% are partially defined. This lends strong credence to our notion of lexically driven software component retrieval.

The goal of both research strands is similar: to propose components for reuse based on the code written by the developer so far, without requiring any further input from the programmer and making

only minimal and realistic assumptions about a repository of existing code. Even in this constrained set of circumstances, we have demonstrated the potential for automated support for software reuse. We are currently experimenting further with these ideas and plan to develop a prototype tool in the form of an Eclipse plug-in.

Links:
http://www.cs.ucd.ie/projects/ASR
http://www.cs.ucd.ie/projects/RASCAL

Please contact:
Mel Ó Cinnéide
University College Dublin, Ireland
E-mail: mel.ocinneide@ucd.ie

CHARMY: A Framework for Designing and Validating Architectural Specifications

by Mauro Caporuscio, Paola Inverardi, Henry Muccini and Patrizio Pelliccione

CHARMY is a tool-supported framework, initially proposed in 2001 to check the consistency between architectural models. Since then the approach has evolved and can now be employed in a number of ways: to check the validity of architectural properties with respect to a software architecture specification; to incrementally create an architectural prototype; to use a compositional approach for the verification of middleware-based applications; and to verify architectural patterns.

Software Architectures (SAs) emerged in the nineties and were used to structure complex software systems, exploiting commonalities in the organisation of specific domains and providing a high-level system description. Nowadays SA is an independent discipline focusing on the overall organisation of a large software system and using abstractions to express the logical coordination structure of complex distributed systems. The emphasis in SA specification is on capturing the system structure (ie the architecture topology) by identifying architectural components and connectors, and the system behaviour (ie the architecture dynamics) by identifying how components and connectors interact.

To specify SAs, informal box-and-line notations are now replaced or complemented by formal and rigorous Architecture Description Languages (ADLs). Many methods and tools, developed on the basis of these ADLs, have been proposed for SA-level testing, analysis and model checking, deadlock analysis, performance analysis and so on. In other words, many studies have proved the suitability of these formal languages for analysis purposes.

However, we note that formal languages are not commonly used in industrial applications, which tend to prefer model-based, semi-formal notations. In fact, in current industrial practice, state-based machines and scenarios are the tools most commonly used to model behavioural aspects, even at the architectural level: state diagrams describe component behaviour while scenarios (eg Message Sequence Charts or Sequence diagrams) identify their interaction. The introduction of the Unified Modelling Language (UML) as the de-facto standard for modelling software systems has increased the use of modelling notations to describe

software systems. Furthermore, the introduction of UML extensions to model SAs makes UML diagrams more suitable for SA modelling.

Introduced in the early stages of the software development process, the CHARMY framework will assist the software architect in the design and validation phases. CHARMY enables the formulation of the SA through model-based specifications (extensively used in industrial projects), which are automatically translated into a formal prototype and validated against selected properties.

In particular, the state diagrams used to specify how the architectural components should behave are automatically interpreted in order to synthesise a formal prototype (see Figure). Properties to be validated are modelled using scenarios. Through the CHARMY engine, the model checker SPIN is used

The CHARMY framework.

formal architectural prototype and eventually uses SPIN for its verification. With the aim of enabling the verification of the architecture without the need for close familiarity with model-checking processes, we are also developing a plug-in component which will integrate SPIN into CHARMY, thus hiding the complexity of the model checker. Another interesting idea for future work is to use CHARMY to manage the counter-example output produced by SPIN. SPIN is already able to represent counter examples graphically, but the visualised counter example often has an abstraction level which differs from the system specification. However, we are not restricted to the use of SPIN as our model checker. We are also investigating the possibility of using the SMV or Bogor model checkers. For example, it could be interesting to exploit the plug-in structure of Bogor to define a customised algorithm search.

to check the conformance of the formal prototype with respect to the behavioural properties.

In order to make CHARMY useful in an industrial context, we have hidden the complexity of the approach, providing the software engineer with an automated easy-to-use tool that takes the architectural model as input, creates a prototype and automatically analyses it, thereby reducing human intervention to the minimum. The tool is a plug-in system and is structured to allow a fast, simple evolution and integration with other tools.

By using CHARMY, the software architect can save time and improve space efficiency. CHARMY provides guidelines on how to model the system and automatically generates an optimised system prototype.

In the future, we are planning several interesting extensions. Currently, the user runs CHARMY to produce the

Link:
http://www.di.univaq.it/charmy

Please contact:
Paola Inverardi, Project Advisor
University of L'Aquila, Italy
E-mail: inverard@di.univaq.it

Moose – A Language-Independent Reengineering Environment

by Oscar Nierstrasz and Stéphane Ducasse

Aging software systems are difficult and expensive to maintain. Moose is a language-independent environment that supports a wide range of tools to visualise, analyse and manipulate complex software systems.

Successful software systems are continually being adapted to meet evolving requirements. Unfortunately, the consequent software changes inevitably lead to a degradation in software quality, as the new requirements are divergent from the original design. As software systems age, they consequently become more complex, and thus harder to understand and maintain. A legacy software system is a system that has an undeniable business value, but has become difficult to extend and maintain.

Reengineering is the process of transforming an existing software system into a new form that can then be adapted to new purposes. The reengineering process generally entails the activities of analysing the system to extract and understand its design, assessing the problems, and migrating the system to new or transformed designs and implementations. This process is iterative, and can benefit greatly from tools to support the software engineer in manipulating, navigating and analysing models of software.

Moose is a language-independent environment for representing software models. Software source code written in different programming languages can be parsed using various third-party tools, and imported into Moose by means of the CDIF or XMI exchange formats. The Moose repository then represents these models using a language-independent source-code meta-model called FAMIX. Moose offers a set of services for navigating and manipulating multiple software model versions, thus offering an infrastructure for reengineering tools to communicate and exchange information and services. Individual tools are free to extend the meta-model for their own purposes. We will briefly describe three such tools: CodeCrawler, ConAn and Van.
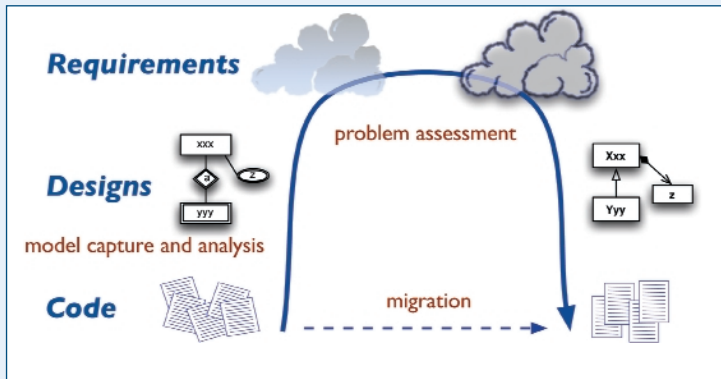
Figure 1: The reengineering process.

CodeCrawler is one of the key tools supported by Moose. One of the biggest problems a software engineer faces when initially encountering a complex software system is building an impression of the state of the software and important entities in the system. CodeCrawler addresses this problem by providing *polymetric views* — simple graphical presentations of various direct software metrics. Either coarse or fine-grained views may be of interest, depending on which stage the reengineering process is being undertaken. The system complexity view, for example, displays a class hierarchy in which the shape and shading of the class boxes indicates the numbers of attributes, methods and lines of code (see Figure 3). This is especially useful for identifying unusual classes in the system, such as those with too few or too many responsibilities.

ConAn is another tool that applies Formal Concept Analysis to identify sets of entities with common properties. These sets help the software engineer to understand the source code at different levels of granularity. The entities to be considered are chosen by the software engineer. For example, when the entities represent relationships between attributes and methods within a class, then the identified concepts highlight phenomena such as splittable classes (whose methods consistently access a partition of the state), or core methods (which access the entire state). When entities are chosen to represent behavioural dependencies between classes in a hierarchy, then concepts highlight behavioural patterns, such as hook methods or repetitive code constructs that could be refactored. And when entities represent structural dependencies, then concepts tend to correspond to common coding idioms, design patterns and reuse contracts.

Van focuses on understanding software evolution by implementing a meta-model centred around the notion of history and by defining various measurements which summarise the evolution of software entities. Applications of Van include: (i) understanding which parts of the system are likely to change in the future, (ii) using time information to improve the detection of design flaws, and (iii) clustering entities that change together to detect hidden dependencies or 'bad smells' in the code.

Other tools being developed address: (i) detection of duplicated code, (ii) clustering techniques to reduce coupling and improve cohesion, (iii) analysis of maintainability to determine the ease with which a system can evolve, by detecting and tracking features and cross-cutting concerns.

Project Details

The first version of Moose was developed during the FAMOOS project, 'A Framework-Based Approach for Mastering Object-Oriented Software Evolution', ESPRIT Project 21975/ Swiss BBW No. 96.0015 (Sept 1996-Dec 1999).

Current development is funded by the Swiss National Science Foundation under projects, 'Tools and Techniques for Decomposing and Composing Software', Project No. 2000-067855.02 and 'RECAST: Evolution of Object-Oriented Applications', Project No. 620-066077.

Figure 2: The Moose environment.

Please contact:
Oscar Nierstrasz, Stéphane Ducasse
University of Bern/SARIT, Switzerland
E-mail: {nierstrasz,ducasse}@iam.unibe.ch
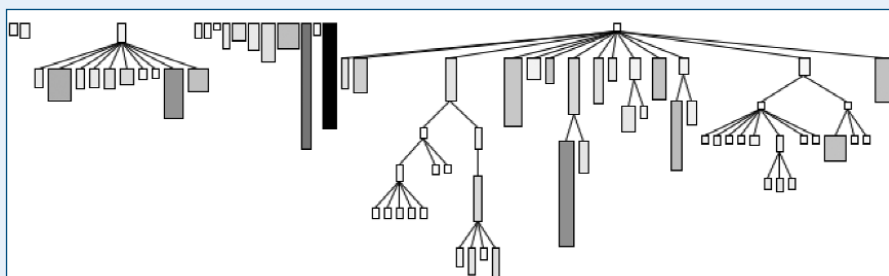
Figure 3: Code Crawler provides graphical presentations of software metrics.

MaC: A Testability Analysis Tool for Reactive Real-Time Systems

by Huy Vu Do, Michel Delaunay and Chantal Robach

The MaC tool has been developed for testability analysis of graphical designs of real-time systems. This tool helps the designer to detect parts of a system that are difficult to test, and identifies the minimum set of test objectives that are useful during the test phase.

Today, reactive real-time systems are widely used in many safety-critical domains: automotive, aerospace, nuclear and so on. These systems require a very high level of confidence, since a failure in any one of them could be a disaster. In the development of these systems, validation plays an important part, since it ensures confidence in these systems. The validation process is divided into two activities: the proof of a part of the system, and the testing phase that reveals faults in the system. However, testing of reactive real-time systems is expensive in terms of complexity and cost. This is where a testability analysis can help, by appraising the ease with which testing can be undertaken.

In the development of reactive real-time systems, the designer can use a graphical

However, in complex, large-scale systems, it is difficult to analyse the weaknesses of the architecture of real-time systems. A system can be hierarchical, and potential faults can be hidden in subsystems. To solve this difficulty, we propose an automatic method of analysing the testability of graphically described systems. Our method, which is derived from the SATAN technology (System's Automatic Testability Analysis), consists in analysing the data flow of systems.

The SATAN Technology

The SATAN technology was developed for analysing the testability of data-flow systems. With it, elementary functions of the system can be identified, test objectives determined, and the testability values of each component in the system computed.

This technology uses an Information Transfer Graph (ITG) to model a data-flow system. From the ITG, elementary functions of the system are identified. Test objectives can then be obtained by applying one of two strategies: the progressive structural strategy (Start-Small) that is suitable for the progressive detection of faults during the validation process; and the cross-checking strategy (Multiple-Clue) that is suitable for diagnosis during maintenance. These strategies allow the number of test objectives to be reduced while ensuring that all components of the system are covered.

In this technology, the testability is defined as a combination of two values: controllability and observability. The controllability expresses the easewith which the input values of an internal

component can be controlled through the input values of the system. The observability expresses the easewith which the results of an internal component can be observed at the outputs of the system (see Figure 1). These values are normalised between 0 and 1.

If a component has a low value of controllability or observability, it is a weak point in the system in terms of testing. This allows the designer to then modify the algorithm in line with these values.

The MaC tool

Through a collaboration between the MBDA company, the Free & Alter Soft company and the LCIS laboratory, we have developed the MaC tool based on the SATAN technology. This tool allows

Figure 1: Testability values of a component.

environment (eg Scade, Sildex, Simulink) to describe algorithms. In this method, a system can be hierarchically viewed as a diagram of operators or subsystems. A subsystem is also a composition of operators. The designer can simulate the system and analyse the results. When a design is justified, it can be converted into instrumented code for prototyping. Hence, the overall design process is reduced.

Figure 2: The graphical user interface of the MaC tool.

the testability of components in a system to be analysed from graphical descriptions of systems developed with Scade, Sildex, or Simulink. Inputs of this tool are graphically described algorithms of reactive real-time systems. Since we know that a system can be hierarchical, the system description must be flattened before applying the SATAN technology. Users can choose a level for the flattening. Different functionalities of the SATAN technology, such as strategies, can be activated via the graphical user interface (see Figure 2). The testability analysis can be activated step by step. With a detail log and a status report, users can follow the execution of the MaC tool. After the analysis, a report in HTML format is created. This report contains a hierarchical table of the testability measurements and labelled operator diagrams. Each labelled operator diagram corresponds to an operator diagram of the system. Users can easily find the measurements for each component.

An example of application of the MaC tool is shown in Figure 3. In this example, we apply the MaC tool to analyse the testability of a Simulink
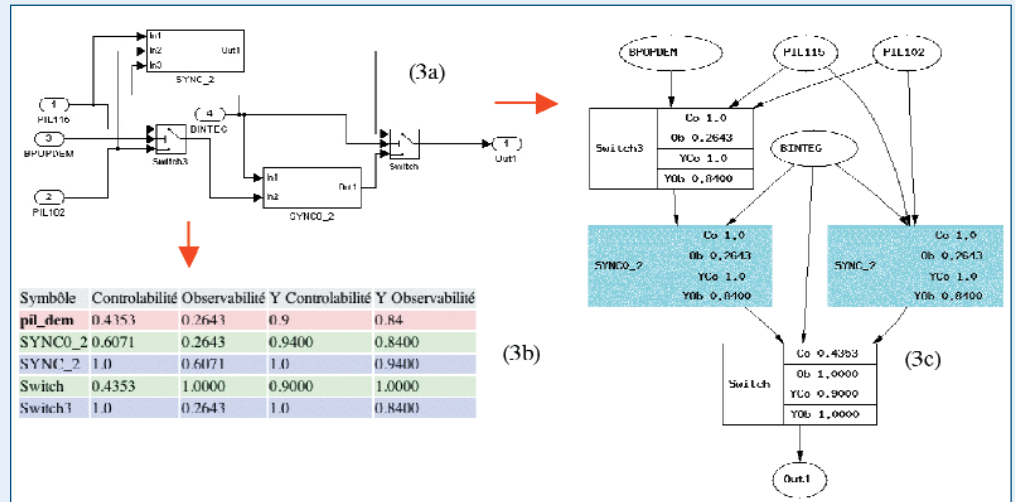
Figure 3: An example of applying the MaC tool on a Simulink model.

model (see Figure 3a). We obtain a table of values (see Figure 3b) and a labelled operator diagram (see Figure 3c).

The MaC tool has been validated on industrial examples provided by the MBDA company. The results show that the testability analysis gives useful information for the design and the validation of these examples: operators with low testability are identified and the minimum set of test objectives is determined.

Using the MaC tool helps the designer to take testing into consideration very early in the design phase. It therefore helps in

reducing the cost of validation and in increasing the reliability of the system.

In future work, we intend to improve the ergonomics of this tool, and extend this tool to other graphical development environments.

Link:
http://www.esisar.inpg.fr/LCIS/valsys/
Les.Groupes/valsys/outils.html

Please contact:
Huy Vu Do, Institut National Polytechnique Grenoble, France
E-mail: Huy-Vu.Do@esisar.inpg.fr

Automated Software Testing with Metaheuristic Techniques

by Eugenia Díaz, Raquel Blanco and Javier Tuya

Software testing is an expensive, difficult and time-consuming process. The use of techniques for automating the generation of software test cases is very important as it can reduce the time and cost of this process. We have used two methods for automatic generation of test cases: Tabu Search and Scatter Search. These methods had not been previously used for this task.

Software testing is the process of executing a program in order to find errors in its source code. It has been estimated that software testing involves more than 50 per cent of software development. This cost can be significantly reduced with the use of automated generators.

Previous works in automatic structural testing used the metaheuristic techniques Simulated Annealing and mainly Genetic Algorithms, but there exist other metaheuristic techniques that have been used successfully in other engineering fields. In our work, we have automated the test case generation by means of two metaheuristic search techniques called

Tabu Search and Scatter Search, which had not been used for this kind of problems yet. We have developed two different algorithms that explore the program control flow graph (whose nodes represent statements and whose edges represent the flow of control between statements) to determine the test cases that attain the desired code
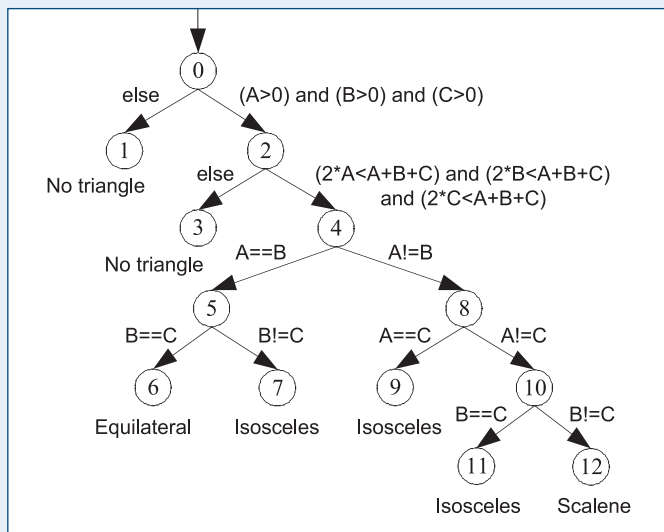
Figure 1: Triangle control flow graph.
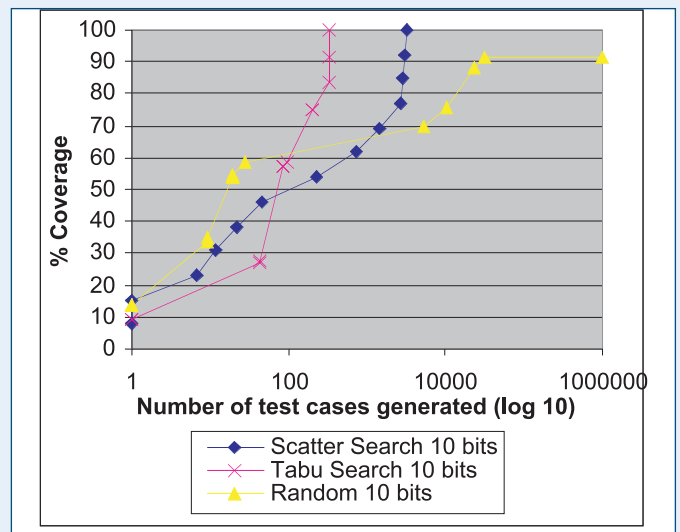
Figure 2: accumulated coverage in per cent versus generated test cases.

coverage. Both source code instrumentation and the test case generation are fully automated.

Test Case Generation using Tabu Search

Tabu Search is a metaheuristic search technique based on that of the next k neighbours, while maintaining a tabu list (memory) that avoids repeating the search in the same area of the solution space.

The algorithm use the control flow graph, which stores relevant information (for example the best tests and their costs). The goal is to automatically obtain branch coverage.

One of the main characteristics of Tabu Search is that it has short-term memory and long-term memory, along with their corresponding handling strategies. In our approach, short-term memory stores the tests that have been the best for the goal node and long-term memory stores the worst tests during the search process. In our approach, short-term memory stores the tests that have been the best for the goal node and long-term memory stores the worst tests during the search process.

In each iteration, the goal of the algorithm is to cover a node that has not been covered yet. Once we have established the sub-goal node, n neighbours are generated starting from the best-known test for its parent node. Once the tests candidates are generated, we verify

whether these are tabu tests, in which case they are rejected. If a candidate is not tabu (not in memory), it is executed with the program under test and if it has been the best test known for some of the nodes that it has reached, it is stored as the best test for the node.

Test Case Generation using Scatter Search

Scatter Search is an evolutionary method which works on a solutions set, called Reference Set. The solutions in this set are combined in order to obtain better new solutions than the original ones. The reference set stores the better solutions that have been generated so far. To determine if a solution is good, its quality and its diversity are considered.

The test case generator based on Scatter Search use the control flow graph in order to determine the covered branches. Each node has a solution set and the algorithm will try to make the sets as diverse as possible, using a diversity function to generate solutions that can cover different branches of the program.

The goal of the algorithm is to obtain the maximum branch coverage, ie, they must be solutions that allow covering all the nodes of the control flow graph. Since these solutions are stored in the nodes, our goal is, therefore, that all the nodes have at least one element in their solutions set.

Results
We present the results obtained with the well known classify triangle benchmark, whose control flow graph can be seen in Figure 1.

The graph in Figure 2 represents the number of test cases generated for out two algorithms (Tabu Search and Scatter Search) in comparison with a random algorithm. The range used for input variables is 10 bits. The vertical axis represents the accumulative percentage of branch coverage and the horizontal axis the number of test cases generated in logarithmic scale.

The random algorithm obtains the worst result and it is not able to achieve full coverage. Both Tabu Search and Scattter Search algorithms reach 100 % coverage and they generate less test cases than random algorithm. The Tabu Search algorithm gives the best result.

Please contact:
Eugenia Díaz, Raquel Blanco, Javier Tuya
University of Oviedo, Spain
Tel: +34 985 18 24 97
E-mail: {eugenia, rblanco, tuya}@lsi.uniovi.es

GATeL: Automatic Test Generation from Lustre Descriptions

by Benjamin Blanc

GATeL is a tool developed by the French Nuclear Research Agency (CEA). Its main objective is to automate the testing phase of critical control applications. The key point here is that automation is not used as a means for generating numerous test cases, but rather helps to exercise truly problematic cases. GATe has been implemented GATeL is currently in use by the French Nuclear Certification Agency, and in avionics and automotive research centres.

Lustre is a formal modelling language belonging to the synchronous data-flow family. It describes programs which cyclically receive input from sensors and send output to actuators during the same cycle. Given a control program and a partial description of its behaviour as a Lustre model, the main role of GATeL is to automatically generate test sequences according to user-oriented test cases. These test sequences (representing an evolution of input data flows over time) can then be submitted to the program being tested. GATel also provides a basis for an automatic oracle, whether or not the program has been automatically generated from the model, depending on the development process followed.

In addition to the Lustre model, GATeL requires two more user inputs: a model of some aspects of the environment in which the program will be run, and a declarative definition of desired test cases. The model of the environment, also written in Lustre, is intended to filter out from all possible behaviours those corresponding to realistic reactions, thereby decreasing the state space to be explored. Unrealistic behaviours could include incompatible values for two input flows, or for an input flow where the occurrence of a past event depends on an output flow. Each filtering expression is stated by an assert directive that must be true for each cycle of the generated sequences.

Requested test cases can then be finely characterised in order to exercise meaningful situations. Test case selection is a crucial part of the testing process. Several approaches have been proposed to automate it, but none is universally recognised. For instance, one can try to perform a statistical coverage of the input domains through a forward simulation, but meaningful test cases may have a very low probability of occurrence (since they require the evolution of a particular input flow over hundreds of cycles). Another approach is to define a structural coverage criterion of the Lustre model, and to then generate test sequences until a given percentage is reached. However, relying only on a predefined coverage criterion assumes that the fault model encoded within this criterion will discover all possible bugs, which may not be sufficient for highly critical systems.

For all these reasons, we prefer to allow users to define their own selection strategies. The first step in this direction is the definition of a test objective. This states a number of important expected properties of the program under the test to be checked. These can be either invariant properties or reachability properties, both expressed in Lustre. Invariant properties are stated with assert directives. The properties that must be satisfied in at
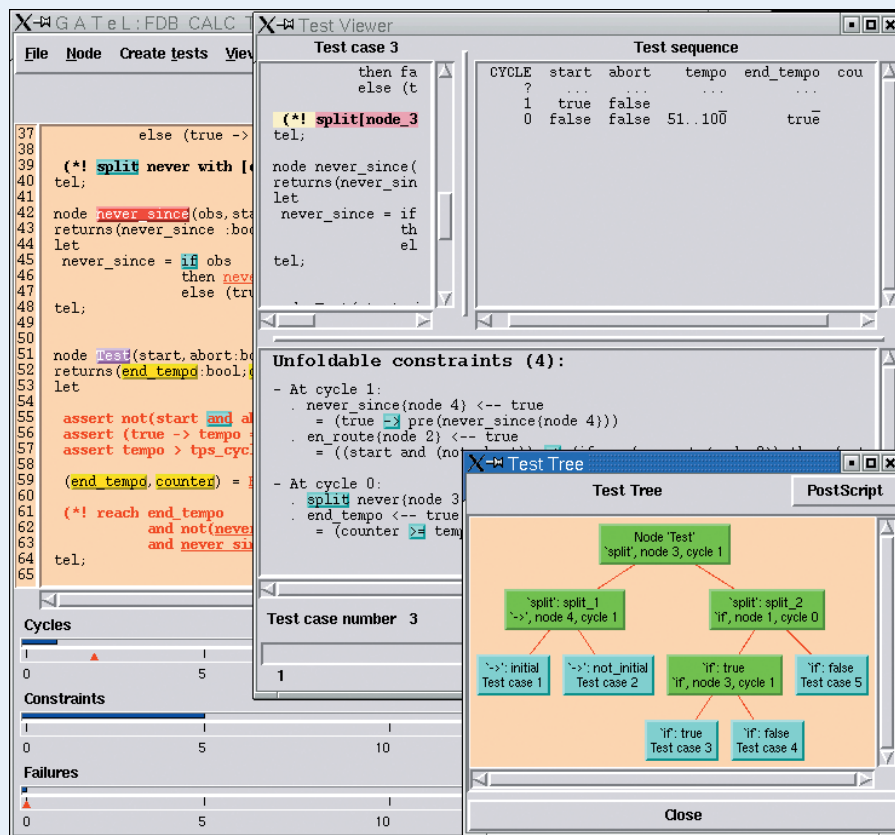
Left: The GATeL main window in which the Lustre model, the description of the environment and the test objectives are displayed.

Top right: the test sequences generated.

Bottom: a tree representing a splitting process and the current test cases obtained.

least one cycle (in fact, in the last cycle of sequences built by GATeL) are stated by reach directives. To build a sequence reaching the test objective according to the Lustre model of the program and its environment, these three elements are automatically translated into a constraint system. A resolution procedure then solves this system through alternate propagation and labelling phases. Propagation checks the local coherence of the system, while labelling aims at incrementally eliminating the constraints by the choice of a variable and a value within its authorised domain.

The random aspect of this resolution procedure implies that the input domains are not fairly covered, meaning quite distinct sequences may be generated for the same objective (for instance, different ways to raise an alarm). A second step in the definition of a selection strategy is to help GATeL to distinguish these sequences. This can be achieved by splitting the constraint system so that each sub-system characterises a particular class of behaviour reaching the objective. This splitting can be processed interactively either by applying predefined decompositions of boolean/integer/temporal operators in the Lustre expressions corresponding to the current constraint system, or by declaratively stating through a dedicated split directive the various behaviours one wants to observe. Notice that a systematic unfolding of different classes of operators would lead to the usual structural coverage criteria.

Finally, test submission consists in reading input sequences generated with GATeL, computing program outputs, and then comparing these values to the expected ones evaluated during the generation procedure. When the program has not been automatically generated from the Lustre model, this gives an automatic oracle. For the alternative case, the truth value of the test objective can play the role of a partial oracle. Methodological and efficiency aspects of GATeL are still under development. However, it has been successfully applied implemented in on industrial case studies, and has generated sequences of a thousand cycles, after the resolution of several thousand constraints over more than forty input flows.

Please contact:
Bruno Marre, Benjamin Blanc
Commisariat à l'Energie Atomique, France
Tel: +33 1 6908 5416
E-mail: benjamin.blanc@cea.fr

Cow_Suite: A UML-Based Tool for Test-Suite Planning and Derivation

by Francesca Basanieri, Antonia , Gaetano Lombardi, Giovanni Nucera, Eda Marchetti and Alberto Ribolini

The Cow_Suite tool provides an integrated, practical and tool-supported approach to the strategic generation and planning of Unified Modeling Language (UML)-based test suites for large industrial software systems. The tool can be adopted from the early stages of system analysis and modelling, and uses the UML diagrams developed for analysis and design without requiring any additional formalism or specific ad-hoc effort for testing purposes.

The Cow_Suite (COWtest pluS UIT Environment) approach supports the early generation of high-level strategic test plans, which can be outlined during the early phases of software life cycles and continuously refined and updated throughout development as the design evolves. Cow_Suite combines two original components: a method to derive test cases, called UIT (Use Interaction Test), and a strategy for test prioritization and selection, called Cowtest (Cost-Weighted Test Strategy). These two components work in conjunction as Cowtest helps decide which and how many test cases should be planned within the universe of test cases that UIT could derive for the entire system under consideration.

The tool was developed at ISTI-CNR in collaboration with Ericsson Lab Italy (ERI, Rome), within the framework of the PISATEL inititative.

Cow_Suite Features

With respect to the many UML-based test approaches already proposed, the main innovative features of Cow_Suite can be summarised as:
- usability: there is no need to augment the UML specifications with specific annotations to facilitate test derivation, nor to translate the UML diagrams into an intermediate notation that the testing methods can process. Cow_Suite adapts to the modelling notations and procedures in use, and not vice versa.

- timeliness: while other methods require a complete and fairly detailed set of UML diagrams, Cow_Suite can begin to outline a test plan from the early stages of software development. Naturally the plan will be as abstract as the diagrams being processed.
- incrementality: Cow_Suite has been conceived for system and integration testing, typically conducted in an incremental fashion, considering progressively larger parts of the system and addressing, at each step, the relevant functionalities and interactions for that level.
- scale: Cow_Suite trades thoroughness for comprehensiveness: it addresses UML-based testing of real-world systems in a practical, efficient way.
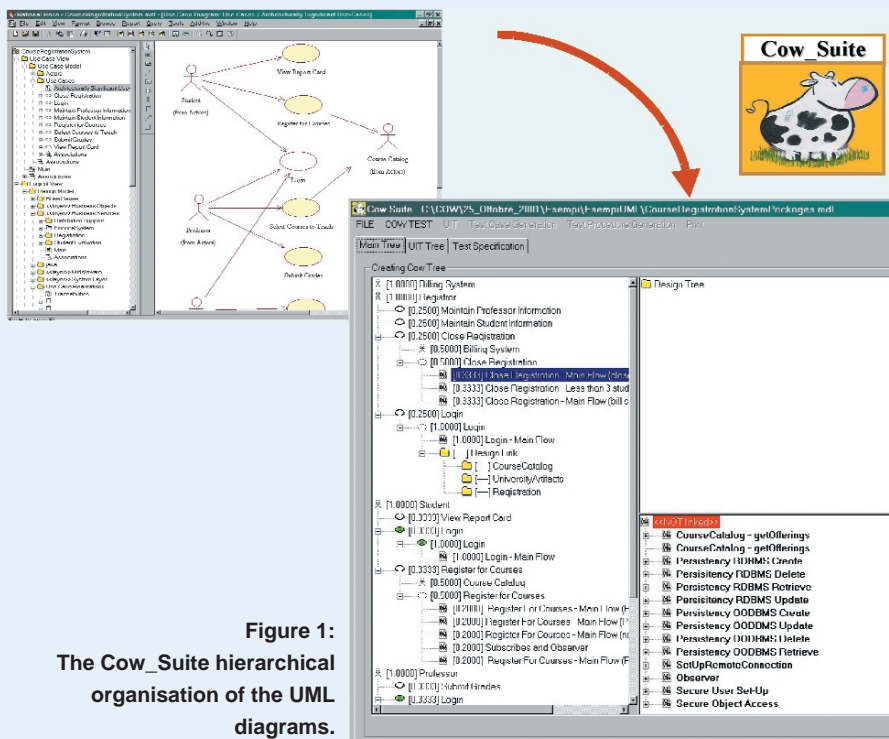
**Figure 1:
The Cow_Suite hierarchical
organisation of the UML
diagrams.**

The combined usage of Cowtest and UIT makes it possible to derive a feasible number of test cases while keeping the coverage of functional areas as wide as possible.

Cow_Suite Usage

The Cow_Suite tool has been designed to be compatible with the Rational Rose tool from which it retrieves the required information. In particular, it employs the Use Case, Sequence, Communication and Class diagrams.

The tool execution starts by importing information on the UML design elements and organising it in a sort of hierarchy, whose root is represented by an Actor and leaves by Sequence or Communication Diagrams (see Figure 1). This hierarchy provides the user with a complete view of the status of the functionality specification and up-to-date documentation on Use Cases and their Realisation, Sequence and Communication Diagrams associated with each specification level, the reused nodes and those elements not linked with the other parts of the design.

At this point, users can annotate each node with a specific weight, representing the relative 'importance' of this node with respect to the other nodes at the same level, and choose between two supported test strategies: a fixed number of test cases, or fixed functional coverage.

Users can then decide the integration level at which the test suite should be derived (or which of the elements of the UML model should be tested), by simply highlighting a portion of the hierarchy.

The UIT component will then automatically derive a list of test cases on the basis of the UML diagrams corresponding to the chosen integration level. These will be specified with a granularity corresponding to the degree of detail at which the considered diagrams are modelled. In Figure 2 we show a schema of the process adopted by the tool.

Case Study

The Cow_Suite tool has already been applied to several case studies. In one industrial case study we compared the UIT-derived test plan with an existing test plan (called ERI). The ERI test plan had been developed manually, following the standard in-house procedures at Ericsson, based mainly on the testers' skill and their knowledge of the system. The UIT test plan was instead derived automatically at ISTI-CNR, using only the available UML design diagrams. The purpose of the comparison was to evaluate the main advantages of the UIT test plan in terms of cost, schedule and test strategy selection. The following advantages emerged from experimentation with the tool:

- derivation of a detailed test plan in advance with respect to the testing stage; this plan can be used as a baseline for deciding the most appropriate test selection strategy
- a realistic evaluation of the requirements and functional coverage that can be reached (timely implementation of
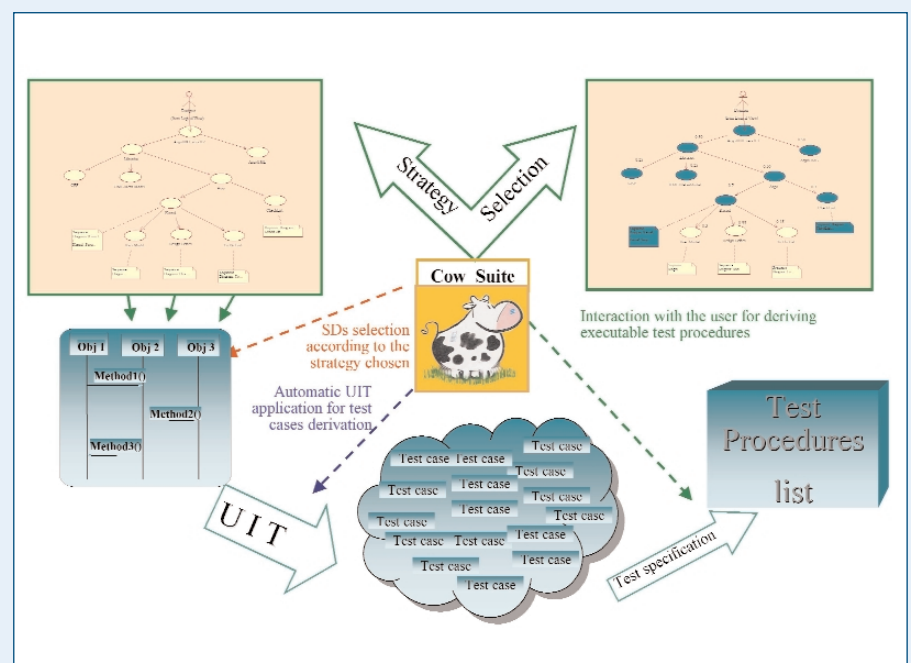
Figure 2: The Cow_Suite usage schema.

corrective actions or different choices for the test strategy are possible)

- reduction of the time necessary for test plan derivation (the same level of requirement coverage was obtained in a quarter of the time).

On the negative side we observed that the automatic derivation of test cases failed to include exceptional test cases, that is, test cases to handle abnormal system behaviour. This is reasonable and suggests that as good practice an expert should check the automated test plan before deployment to cover special situations.

Further experimentation is of course desirable, and Cow_Suite is freely available in a prototype version for public usage.

Three Countries' Offensive towards Testing with Advanced Languages

by Wan Fokkink, Matti Kärki, Jaco van de Pol, Axel Rennoch, Ina Schieferdecker and Markus Sihvonen

The need for appropriate means to test systems and software is still alive! Even though a huge number of test tools are on the market, series of conferences on testing continue to be held, and an international standard on testing methodology has existed for more than ten years. Most industries are still looking for ways to make their testing process more effective, efficient and understandable, to strengthen confidence in their products and services. Three European countries, Finland, Germany and the Netherlands, have started a common offensive to establish a basis for the industrial application of tests and testing methodologies with advanced languages. Independent national funding sources have been brought together for the TT-Medal project under the supervision of the Information Technology for European Advancement (ITEA) association.

The goal of TT-Medal is to develop a test architecture that covers the whole life cycle of a product, starting from the initial specifications and ending up with regression testing during the maintenance phase. Obviously this goal can only be reached by a testing methodology that addresses tests at both an abstract and an executable level.

In TT-Medal key roles are assigned to international standards, the Testing and Test Control Notation (TTCN-3) by ETSI, the Universal Modeling Language (UML2.0) and its testing profile by the OMG. The TT-Medal strategy covers a systematic testing methodology including the production of TTCN-3 based tests from UML models and the development of a generic test infrastructure and test architectures for TTCN-3 with open interfaces for test execution and management. Special emphasis is given to the reuse of test cases between testing phases and for different products in the same domain. Three ERCIM

research institutes, CWI, Fraunhofer FOKUS and VTT Electronics, are the driving force behind these innovations.

An important aim of TT-Medal is to develop a test platform based on TTCN-3 for major European industries such as automotive, railways, telecommunication and embedded systems. Four industrial partners in the project, DaimlerChrysler (Ge), ProRail (NL), Nokia (Fi,Ge) and LogicaCMG (NL), will apply the project results with the objective of unifying the various test environments in use at the different companies, and making seamless the transition from one environment to another.

Furthermore, tool support will be developed for automatic test-case generation, validation and deployment, and to support the reuse of test cases. Three tool developers, Conformiq (Fi), NetHawk (Fi) and Testing Technologies (Ge), are responsible for this part of the project.

Finally, a complete training package will be developed by Improve QS (NL), and will enable European industries to integrate the new improved test practices into their development processes.

The strong application of the standardised and formalised language constructs of TTCN-3 and UML2.0 in mainstream industry provides a special challenge to the customers and providers of test products and services. Coupling the development of system models and tests helps to optimise the rapid implementation of test solutions. Combining the potential of UML2.0, the maturity of TTCN-3 and the advice of industrial manufacturers is an exciting process.

The project work is organised into five work packages, which address issues on methodology, test development tools, test execution tools, industrial case studies, and dissemination and standardisation work. There are strong relationships such as major requirements from
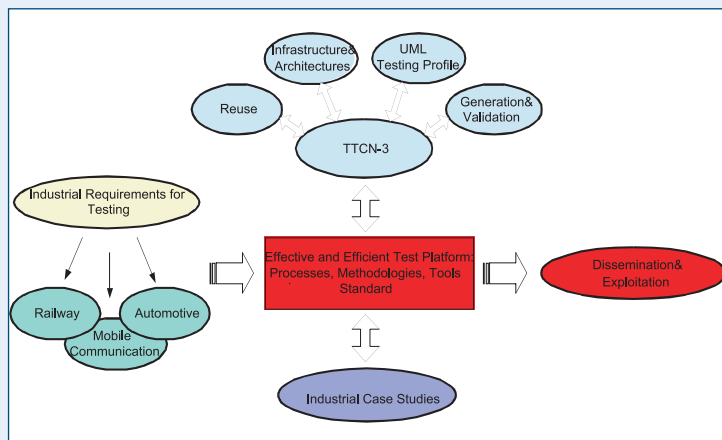
Topics and goals within TT-Medal.

the industrial cases and theories from methodology that are used as input for the tool-development work packages. Since several partners are actively involved in standardisation work at ETSI and OMG, there is also an exchange of ideas with the originators of TTCN-3 and UML2.0.

The major outcome of the TT-Medal project is an effective and efficient test platform. The platform combines the work of the five different work packages and provides necessary processes,

methodologies, tools and standards to improve testing. The effectiveness of the platform is indicated by better product quality, and efficiency ensures that this quality is achieved within a given time frame and resource limits. The TT-Medal project will also show how these new technologies should be used by providing training and organising workshops, as well as demonstrating where and why TT-Medal technologies should be used by gathering industrial experience. The figure summarises the

work and the major elements of the TT-Medal project.

The TT-Medal project, which runs until end of 2005, may be followed in various ways: participate in our upcoming national and international workshops and exhibitions, visit our project Web site for the latest news and publications, or contact a partner or national representative from the project for further information.

Automated Verification of Groupware Protocols

by Maurice ter Beek, Mieke Massink, Diego Latella, Stefania Gnesi, Alessandro Forghieri and Maurizio Sebastianis

Recently, researchers from the Formal Methods and Tools (FM&&T) group of ISTI-CNR teamed up with researchers from think3, a global provider of integrated product development solutions. The goal was to apply formal modelling and verification techniques to enhance think3's Product Data Management (PDM) application, 'thinkteam' (a registered trademark of think3 Inc) with a publish/subscribe notification service.

Computer-Supported Cooperative Work (CSCW) is an interdisciplinary research field which deals with understanding how people work together and finding ways in which computer technology can assist. This technology mostly consists of multi-user computer systems called groupware (systems). Many concepts and techniques from computer science, like concurrency control and data consistency, need to be rethought in the groupware domain. This has led to the devel-

opment of new formal models like team automata, which were introduced explicitly for the description and analysis of groupware systems (see [TA] for details). Groupware systems are typically classified according to two dichotomies, namely (i) whether the users are working together at the same time (synchronous) or at different times (asynchronous) and (ii) whether they are working together in the same place (co-located) or in different places

(dispersed). Examples of synchronous groupware include video conferencing and multi-user games, while electronic mail and the version-control systems which are often used in software engineering to coordinate changes made by multiple programmers to the same program are examples of asynchronous groupware.

An additional difficulty that arises during the design of synchronous group-
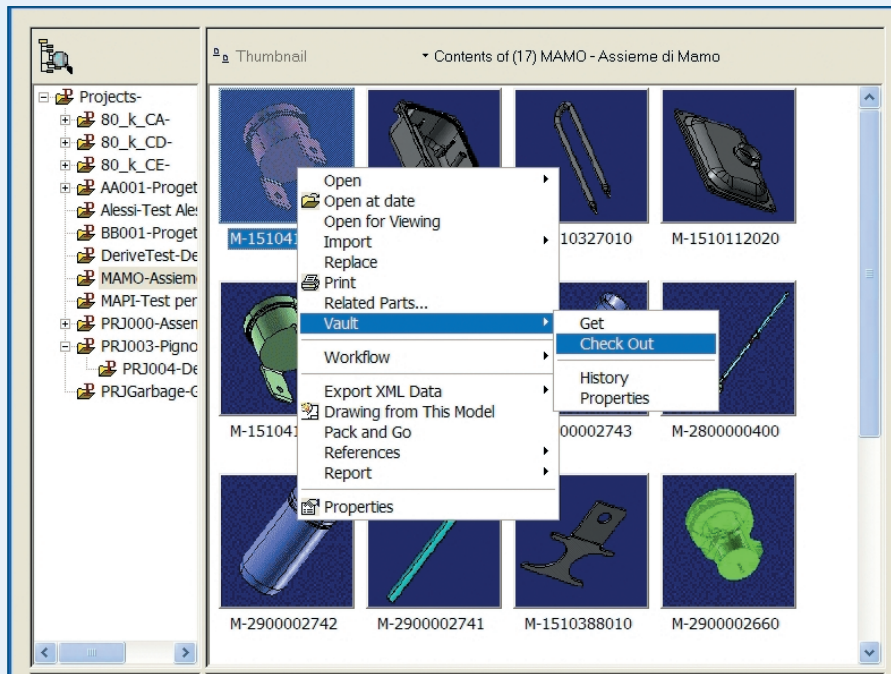
Figure 1: A thinkteam user checks out a document from the vault.

ware is the inherently distributed nature of such systems, which makes it necessary to address issues like network communication, concurrency control and distributed notification. This has led to the development of groupware toolkits that aid developers with programming abstractions aimed at simplifying the development of group-ware applications.

We are currently modelling and veri-fying the addition of a publish/subscribe notification service to thinkteam, think3's PDM application catering to the product/document management needs for design processes in the manufac-turing industry. The main strengths of this service are a rapid deployment and start-up cycle, flexibility, and a seamless integration with thinkdesign - think3's CAD solution - as well as with other third party products. Thinkteam allows enterprises to capture, organise, auto-mate, and share engineering product information in an efficient way. The controlled storage and retrieval of docu-ment data in PDM applications is tradi-tionally called vaulting, where the vault is a file-system-like repository. Its two main functions are (i) to provide a single, secure and controlled storage environ-ment in which the documents controlled by the PDM application are managed, and (ii) to prevent inconsistent updates

or changes to the document base while still allowing the maximal level of access compatible with the business rules. While the first function is integrated in the lower layers of the vaulting system, the second function is implemented in thinkteam's underlying groupware protocol by a standard set of operations on the vault, namely get (extract a read-only copy of a document), import (insert an external document), checkout (extract an exclusive copy of a document with the intent of modifying it), checkin (replace an edited and previously checked-out document), checkinout (replace an edited document in the vault, while at the same time retaining it as

checked out), and uncheckout (cancel the effects of a checkout). Figure 1 shows a screen shot of thinkteam.

Thinkteam is thus an example of an asynchronous and dispersed groupware system. In order to develop the publish/subscribe notification service we first had to define an abstract specifica-tion (model) of the groupware protocol underlying thinkteam. The idea behind this type of service is to more widely inform the users of an application by intelligent data sharing. For instance, whenever a user publishes a file by sending it to a centralised repository, all users who subscribe to that file are noti-fied automatically via a multicast communication. The automatic verifica-tion of publish/subscribe notification services is currently attracting a lot of attention. The addition of such a service to thinkteam should allow us to solve a problem that commonly arises in connection with the use of composite documents and which is a variant of the classic 'lost update' phenomenon, depicted in Figure 2. This phenomenon arises when a user performs a checkout/modify/checkin cycle on a document that may be used as reference copy by other users.

Our research shows that with relatively simple models we can verify highly rele-vant properties of groupware protocols with freely available verification tools. Model checking is an automatic tech-nique to verify whether a system design satisfies its specifications. The verifica-tion is exhaustive, ie all possible input combinations and states are taken into account. One of the best-known and most successful model checkers is Spin, which was developed at Bell Labs during the last two decades. It offers a spectrum of verification tech-niques, is freely avail-able, and is very well documented. Publish/subscribe systems require specific prop-erties, including data consistency through

Figure 2: The 'lost update' phenomenon.

distributed notification, view consistency, absence of (user) starvation and key issues related to concurrency control [BMLG]. The properties we verify are mostly formalised as formulae of a Linear Temporal Logic (LTL). This has the advantage that they are close to the 'scenarios' design. LTL formulae reflect properties of typical behaviour (or uses) of the groupware system. This makes it possible to evaluate design alternatives before their implementation and validation and not only afterwards. Design errors constitute up to 40% of software errors and are among the most expensive

to resolve when discovered only after the implementation phase. Our system allows these to be detected early on in development, leading to considerable reductions in cost and improvements in quality. The specification (model) of thinkteam's underlying groupware protocols has been developed using this methodology and in close collaboration between members of FM&&T and think3.

This research is conducted within the framework of a broader national research project on global computing,

funded by the Italian Ministry of Research and Education (MIUR/SP4).

Please contact:
Maurice ter Beek, Mieke Massink
ISTI-CNR, Italy
E-mail: {terbeek,massink}@isti.cnr.it

Parallel Model-Checking

by Luboš Brim

With the rapid increase in computer system complexity, it has become very important to develop formal methods to ensure their quality. Several novel parallel and distributed techniques for enumerative model-checking of safety and liveness properties expressed in a simple temporal logic have been developed in the Parallel and Distributed Systems laboratory at the Masaryk University Brno under the project 'Automated Verification of Parallel and Distributed Systems'.

Early detection of errors requires application of advanced analysis, verification and validation techniques to test modelling resources, temporal properties, data-type invariants, and security properties. Various techniques for automated and semi-automated analysis and verification of computer systems have been proposed. In particular, model checking has become a very practical technique due to its push-button character. The basic principle behind model checking is to build a finite model of the system under consideration together with a formal description of the verified property in a suitable temporal logic. The model-checking algorithm is a decision procedure, which in addition to the yes/no answer returns a trace of a faulty behaviour in cases where the checked property is not satisfied by the model. One of the additional advantages of this approach is that verification can be performed against partial specifications, by considering only a subset of all specification requirements. This allows for increased efficiency by checking correctness with respect to only the most relevant requirements.

Although model checking has been applied fairly successfully for verification of quite a few real-life systems, its applicability to a wider class of practical systems has been hampered by the so-called state explosion problem (ie the enormous increase in the size of the state space). For large industrial models, the state space does not completely fit into the main memory of a computer. Consequently, when the memory becomes exhausted and the system starts swapping, the model-checking algorithm becomes very slow.

A typical approach to dealing with these practical limitations is to increase the computational power (especially random-access memory) by building a powerful parallel computer as a network (cluster) of workstations. Individual workstations communicate through a message-passing interface such as MPI. From outside, a cluster appears as a single parallel computer with high computing power and a huge amount of memory. In recent years a lot of effort has been put into using parallel and distributed environments to solve the

computational and space complexity bottlenecks in model-checking.

The main question is whether we can (re)design verification techniques in such a way that they could be implemented on parallel computer architectures. In other words, we would like to find techniques for decomposing complex verification problems into 'smaller' independent sub-tasks, which could then be either further decomposed or directly solved. Efficient parallel solution of many problems often requires the invention of original, novel approaches that are radically different from those used to solve the same problems sequentially. Several methods for parallel model-checking have been accepted and implemented in industrial tools. Performance results on either parallel machines or on a cluster of workstations show significant improvements over sequential techniques, both in extension of the size of the problem and in computational times, along with adequate scalability with the number of processors.

We have developed several novel parallel and distributed techniques for

enumerative model-checking of safety and liveness properties expressed in a Linear Temporal Logic (LTL). The model-checking problem being considered can be reformulated as a cycle detection problem in an oriented graph, and the basic principles behind our algorithms rely on efficient solutions to detecting cycles in a distributed environment. In particular, we employ specific structural properties of the underlying graphs (often computable in advance from the given system specification), use additional data structures to divide the problem into independent sub-problems, or translate the model-checking problem into a different problem that admits efficient parallel solution. Several of our algorithms are based on sequentially less efficient but parallelisable breadth-first exploration of the graph or on placing bounds that limit the size of the graph to be explored.

Development of a tool that supports the distributed verification of systems is one of our recent projects. The goal is to build an environment for easy implementation of parallel and distributed

The cluster PC SGI 1200 is being used for experimental evaluation of the parallel algorithms.

verification algorithms on clusters of workstations, followed by experimental evaluations and comparisons. The main characteristics are the support for the

distributed generation of the state space, dynamic load balancing, distributed generation of counter-examples, fault-tolerance and re-partitioning. The distributed environment quite naturally allows for the integration and cooperation of methods and algorithms as well.

Other research performed in the laboratory is focussed on the development of original methods and techniques for the automated verification of large-scale industrial critical systems, with an emphasis on the practical aspects of their application. Further, it looks at applying these and other known methods and techniques to real-life systems, optimising them to improve their efficiency, and providing software support to use them.

Aspect-Oriented Software Evolution

by Tom Mens, Kim Mens and Tom Tourwé

All software systems are subject to evolution. This poses great challenges for software engineers. Aspect-oriented software development (AOSD), in addition to separating the different concerns during software development, can be seen as a way of overcoming many of the problems related to software evolution. At the same time, AOSD may benefit from tools and techniques that automate software evolution. ERCIM members are therefore exploring the cross-fertilisation between these two active research domains.

Real-world software needs to evolve continually in order to cope with ever-changing software requirements. Manny Lehman identified this characteristic in his so-called first law of software evolution, which addresses continuing change: a program that is used must be continually adapted else it becomes progressively less satisfactory.

This need for software to evolve continuously poses important challenges for software engineers. Advanced automated software engineering techniques and tools are needed to improve software

evolution support. An ERCIM Working Group on Software Evolution is currently being formed to address this need. Two important techniques that will be investigated are software restructuring and aspect-oriented software development.

Software Restructuring
Software restructuring should be an essential activity in software engineering, according to Lehman's second law of software evolution, which addresses increasing complexity: as a program evolves its complexity increases unless work is done to maintain or reduce it.

In program transformation research, two different restructuring approaches can be distinguished. The term rephrasing is used to refer to techniques that improve the structure of the software without changing the implementation language. A typical example is software refactoring, which tries to improve the internal structure of a program without changing its external behaviour.

The term 'translation' refers to techniques that restructure the software across programming languages. A typical example is migration of legacy
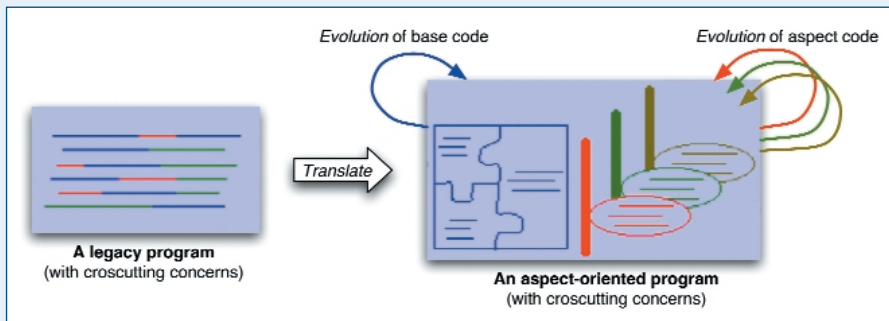
Figure 1: Cross-fertilisation of software evolution and AOSD.

code to an object-oriented equivalent (eg COBOL to Java).

Aspect-Oriented Software Development

An essential problem with software development is the tyranny of the dominant decomposition. No matter how carefully a software system is decomposed into modular units, there will always be concerns (typically non-functional ones) that cut across the chosen decomposition. The code of these cross-cutting concerns will necessarily be spread over different modules, which has a negative impact on the software quality in terms of comprehensibility, adaptability and evolvability.

Aspect-oriented software development (AOSD) has been proposed as a solution to this problem. In order to capture cross-cutting concerns in a localised way, a new abstraction mechanism (called an aspect) is added to existing programming languages (eg AspectJ for Java). As a result, cross-cutting concerns are no longer distributed over different modules. This means the software is easier to maintain, evolve and understand.

Cross-fertilisation

In order for AOSD to become truly successful, existing software systems need to be translated into their aspect-oriented equivalents and rephrased continuously (see Figure 1). Given the size and complexity of industrial software systems, this must be achieved with as much automated support as possible. More specifically, automated support is needed for three essential activities:
- aspect mining - techniques should be used to identify the relevant concerns in the source code

- aspect introduction - techniques are needed to define the appropriate aspects for any of the identified concerns, in order to translate the software into an aspect-oriented equivalent
- aspect evolution - techniques are required in order to evolve aspect-oriented software.

Our research investigates how formal techniques, successful in supporting traditional software evolution, can support these three new activities. For example, in a recent experiment, we used formal concept analysis to mine for cross-cutting concerns in object-oriented source code. With this approach we detected interesting features corresponding to cross-cutting functional or non-functional concerns, as well as occurrences of design patterns like the Visitor design pattern (see Figure 2).
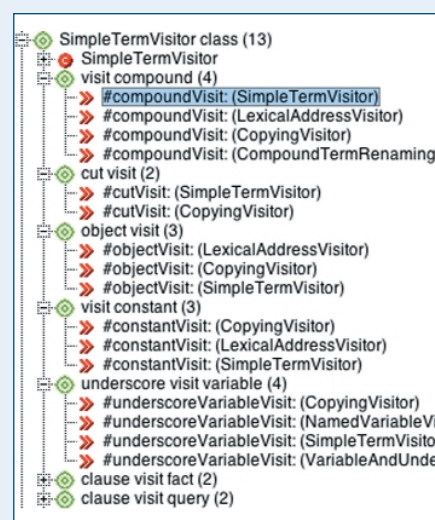
Figure 2: An occurrence of the Visitor design pattern detected by formal concept analysis. The results are classified hierarchically into groups of related software entities (in this case, classes and methods).

Typically, the implementation of such a pattern spans multiple classes and methods, and grouping these in a single hierarchical view allows the developer to understand and manipulate the pattern more easily.

In another experiment we dealt with aspect introduction and aspect evolution. More specifically, we investigated how evolution of the base code affects the definition of aspects that work on it. In order to address this issue, we proposed a more sophisticated aspect-oriented programming language, accompanied by an advanced development environment. The environment helps a developer to define aspects as logic rules, and is able to assess the impact of evolution on these aspects automatically. To this extent, the environment relies on a machine learning algorithm called inductive logic programming.

European Collaboration

The authors are involved in many successful international research activities that have been initiated in the domains discussed above:
- a Belgian FWO-funded scientific research network on 'Foundations of Software Evolution' (http://prog.vub.ac.be/FFSE/network.html)
- a Belgian IWT-funded inter-university research project on 'Architectural Resources for the Restructuring and Integration of Business Applications' (http://arriba.vub.ac.be)
- a Belgian FNRS-funded 'Research Center on Structural Software Improvement'
- a European ESF-funded scientific network on 'Research Links to Explore and Advance Software Evolution' (http://labmol.di.fc.ul.pt/projects/release/)
- a European EU-funded network of excellence on 'Aspect-Oriented Software Development'
- an ERCIM working group on 'Software Evolution'.

Please contact:
Tom Mens
Université de Mons-Hainaut, Belgium
Tel: +32 65 37 3453
E-mail: tom.mens@umh.ac.be

Generic Language Technology: Basics for Automated Software Engineering

by Mark van den Brand

Research in the Generic Language Technology project at CWI is focusing on the development of fundamental techniques for language processing. These techniques are applied in the development of high-quality tools for the analysis and transformation of large software systems written in languages including C, Java, Cobol and PL/I.

The focus of the research in the Generic Language Technology (GLT) project at CWI is on the development of fundamental techniques for analysis and transformation of (programming) languages. Besides the development of formalisms for describing the syntax and semantics of programming languages, tools for processing languages and programs are also being developed. The scope of research includes exploring new fundamental concepts, varying from incremental techniques, efficient term-rewriting engines, advanced parsing technology, new analysis techniques, and powerful code generators.

The roots of this project are in the mid-1980s. Together with INRIA (Sophia-Antipolis) and various other partners, the Esprit project 'Generation of Interactive Programming Environments' (GIPE) was started, and later continued as GIPE-II. Research focused on the development of a framework for programming environment generators, with CWI in Amsterdam looking at the development of a Meta-Environment - an environment for developing programming language descriptions, based on incremental technology for scanning, parsing, and rewriting.

Application areas for this technology include the design and implementation of domain-specific languages, software renovation, and advanced code generators. In cooperation with a Dutch software house and a Dutch bank, a domain-specific languagefor describing financial products, RISLA, was developed. RISLA and a prototype of RISLA to Cobol compiler were then implemented, and are still being used by several Dutch banks. Various projects in the field of software renovation and reverse and reengineering have been carried out over the last few years. A powerful generic parsing technology allowed us to tackle both the problem of handling various dialects of Cobol and that of embedded languages in Cobol, such as SQL, assembler, and CICS. Projects in various other industries (eg transportation, networking, telecommunication) have also been carried out.

The application area of software renovation triggered development of new scalable language-processing technology.

Figure 1: Software engineering tasks as document transformations.

The Meta-Environment developed within the GIPE projects was entirely redesigned and based on new modern component-based ideas and technology. The focus of our research moved from incremental techniques to scalability, flexibility, re-usability and efficiency of tools. The specific topics we are tackling in our current research are:
- exploring the benefits of integrating scannerless generalised LR parsing with advanced declarative disambiguation mechanisms
- developing new and efficient term-rewriting technology such as rewriting with annotations, compilation of rewriting rules to high-performance C-code, and the integration of rewriting technology with relation calculators
- exploring the possibilities of using rewriting technology to obtain more powerful, semantics-directed, disambiguation mechanisms.

All these technologies are immediately applied in various industrial projects.

Generic language technology research has been receiving an increasing amount of attention. There are various reasons for this. The first is that as computers become more powerful, more powerful algorithms can be applied to bigger problems, for example, in the field of software renovation. The second reason is that software in general has an ever-increasing life-cycle and it is crucial for (financial and other) industries to keep software that was written in the 60s and 70s operational. In order to analyse and transform this software automatically, advanced generic technology is needed (see Figure 1). A third reason is related to the refactoring techniques developed for Smalltalk and Java. These are currently spreading towards other languages as well, which means re-implementing these refactorings for each separate language. Expressing these in a

language-independent way and applying them to programs written in other languages has proven be an interesting challenge. A fourth reason is the development of programming-environment frameworks like Eclipse (see http://www.eclipse.org). Although initially developed for Java and designed to be open for other programming languages as well, reality has proven to be more complex. More and more people are working on Eclipse plug-ins for their favourite language, whereas a more generic approach would increase the flexibility of the Eclipse framework.

The concepts of generic language technology are realised in the framework of the ASF+SDF Meta-Environment. This is an interactive development environment for the automatic generation of interactive systems for constructing language definitions and generating tools for them. A language definition typically includes such features as syntax, prettyprinting, typechecking, and execution of programs in the target language. The ASF+SDF Meta-Environment offers openness, reuse, extensibility, and in particular the possibility of generating complete stand-alone environments for user-defined languages. ASF+SDF allows the definition of syntactic as well as semantic aspects. It can be used for the definition
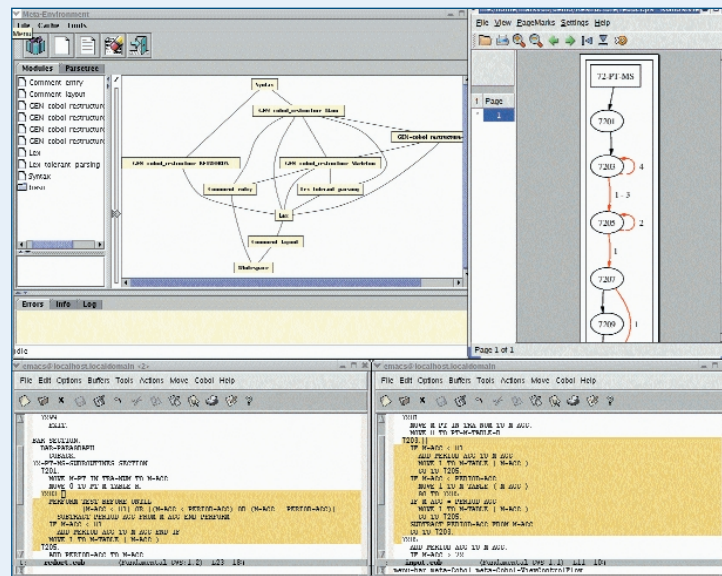
Figure 2:
The ASF+SDF
Meta-
Environment
as Cobol
restructuring
tool.

of languages (for programming, writing specifications, querying databases, text processing, or other applications). In addition, it can be used for the formal specification of a wide variety of analysis and transformation problems (see Figure 2).

The Generic Language Technology project is currently working in close cooperation with the Protheo group at INRIA/LORIA in Nancy and with the group of Peter D. Mosses at BRICS in Aarhus. We also cooperate with Utrecht University and the Vrije Universiteit in Amsterdam. The ASF+SDF technology

is used by various research groups for various purposes, including mechanical modelling (Mechanical Engineering group of the Technical University of Eindhoven, Netherlands), module composition (DIMAp of UFRN in Natal, Brasil), typesetting mathematics (LORIA, France) and algebraic specification (Univ. Bremen, Germany).

Link:
http://www.cwi.nl/projects/MetaEnv

Please contact:
Mark van den Brand, CWI, The Netherlands
Tel: +31 20 592 4213
E-mail: Mark.van.den.Brand@cwi.nl

Identifying Cross-Cutting Concerns in Embedded C Code

by Magiel Bruntink, Arie van Deursen and Tom Tourwé

The identification and refactoring of cross-cutting concerns is the topic of Ideals, a four-year research project conducted in cooperation with ASML, the world market leader in lithography systems based in Veldhoven, The Netherlands. The research partners in this project are CWI, TU/e, UT, and the Embedded Systems Institute (ESI) based in Eindhoven. The project has commenced in September 2003.

Large-scale industrial software applications are inherently complex, and a good separation of concerns within the application is therefore indispensable. Unfortunately, recent insight reveals that the current means for separation of concerns, namely functional decomposition or object-oriented programming, are insufficient. No matter how well large

applications are decomposed using current means, some functionality, typically called 'cross-cutting concerns', will not fit the chosen decomposition. As a result, implementations of such cross-cutting concerns will be scattered across the entire system, and become entangled with other code. In this case, the consequences for maintenance of the system

and its future evolution, are obviously dire.

Aspect-oriented software development (AOSD) has been proposed as an improved means for separation of concerns. Aspect-oriented programming languages add an abstraction mechanism (called an 'aspect') to existing (object-

Figure 1: Cross-cutting concerns in a 19 KLOC component.

oriented) programming languages. This mechanism allows a developer to capture cross-cutting concerns in a modular way. In order to use this new feature and make the code easier to maintain, existing applications written in ordinary programming languages should be evolved into aspect-oriented applications. To that end, scattered and tangled code implementing cross-cutting concerns should be identified, and subsequently refactored into aspects.

Identifying Cross-Cutting Concerns

Source code implementing cross-cutting concerns (CCCs) tends to involve a great deal of duplication. First of all, since such code cannot be captured cleanly inside a single abstraction, it cannot be reused. Therefore, developers are forced to write the same code over and over again, and are tempted to simply copy and paste the code and adapt it slightly to the context. Alternatively, they may use particular coding conventions and idioms, which also exhibit similar code. We hypothesise from this observation that clone detection techniques might be ideal candidates for identifying CCC code, since they automatically detect duplicated code in the source code of an application.

Case Study

Our experiment was performed on a software component consisting of 19 000 lines of C code, which formed part of the larger code base (comprising over 10 million lines of code) of ASML. Developers at ASML currently use an idiomatic approach to the implementation of CCCs. Consequently, similar pieces of code are scattered throughout the application, as can be seen in Figure 1, where each column represents a module and each colour represents one of four CCCs. For the moment we focused on CCCs dealing with tracing (green), pre- and post-condition checking (yellow), memory-error handling (blue), and general-error handling (red). All together, the CCCs we considered comprised roughly 31% of the code. For the experiment, an ASML developer manually annotated the source code lines of the 19 KLOC component, indicating for each source code line the CCC to which it belongs.
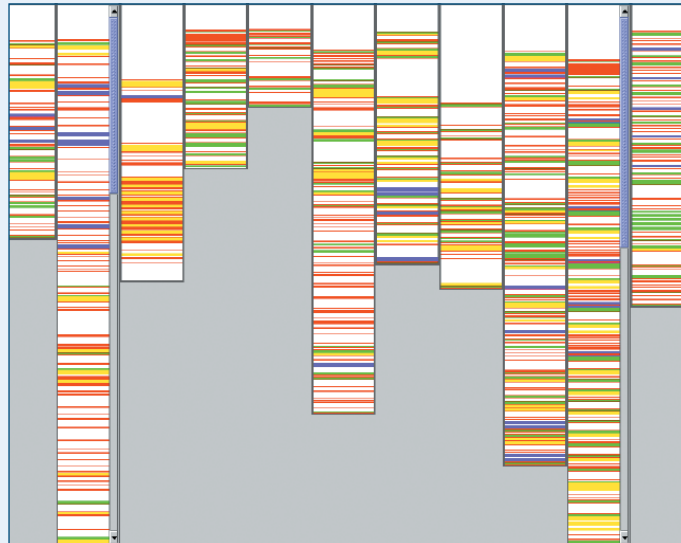
As a result, each CCC is defined by a set of source code lines. Subsequently, we used a clone detector to obtain 'clone classes', ie sets of code fragments that are cloned (similar).

In order to evaluate to what extent the clone detector meets our goal of identifying CCC code, we investigated the level of 'concern coverage' met by the clone classes. Concern coverage is the fraction of source code lines of a concern that occur in the clone classes, ie the source code lines that occur as part of one of the cloned code fragments. Conversely, we evaluate the precision obtained by the clone classes. 'Precision' is defined as the ratio of concern lines to total lines occurring in the clone classes.

We are primarily interested in achieving sufficient coverage without losing too much precision. We therefore focus on the number of clone classes needed to cover most of a concern, where we

consider 80% to be a sufficient coverage level.

Results

Figures 2a and 2b show the coverage and precision levels obtained by the clone classes. The horizontal axis represents those sets of n clone classes that each obtain the highest possible coverage for a CCC. Vertically, we have the level of coverage in Figure 2a and the level of precision in Figure 2b. In this paper we only present the results of a clone detector that implements Baxter's AST-based clone detection algorithm.

Observe that as the number of clone classes increases, the coverage displays a monotonic growth, whereas the precision tends to decrease. The highest coverage is less than 100% in all cases: the remaining percentage corresponds to concern code that is coded in such a unique way that it does not occur in any clone class.

(a) Concern coverage of the clone classes.

(b) Precision of the clone classes.
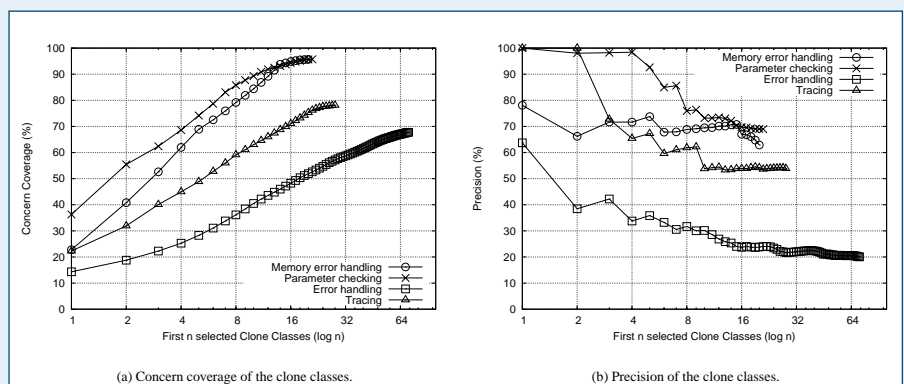
Figure 2: Clone detection results.

The parameter checking and memory-error handling concerns are covered sufficiently well (80%) using a limited number of clone classes. Furthermore, the level of precision obtained by the clone classes in the parameter-checking case is very high: the first four selected clone classes obtain a precision as high as 98%. In contrast, the tracing and error-handling concerns do not show the same results. Both concerns are not sufficiently covered, although the coverage of tracing is almost sufficient (78%). Additionally, the obtained precision is low. An exception is the precision obtained for the tracing concern by the first two clone classes: they cover 32% of the tracing code at 100% precision. However, if more clone classes are considered, the precision degrades quickly.

Conclusion

Evaluating the clone classes in terms of concern coverage and precision identified many clone classes that will be of value during our research within the context of the ASML code base. The component we considered is a small but representative example of the components that exist within this code base. It is likely therefore, that the clone classes we found to cover large portions of CCC code will be of great use for identification of CCC code in the other components.

In a more general sense, we intend to work toward a fully automatic 'concern mining' tool. Based on the results, we can conclude that clone detection techniques are capable of characterising large portions of the code of certain CCCs. However, in our experiment we included an important manual step: the knowledge of an expert is used to identify the CCC code. Further experiments will be needed to show that clone detection techniques can also find CCC code using a more limited amount of expert knowledge.

PROSE: Automated Software Adaptation for Pervasive Computing

by Gustavo Alonso

Adaptation is one of the biggest challenges for software engineering. This is particularly the case since mobile and pervasive computing have turned adaptation from the slow process of software evolution into a highly dynamic run-time procedure that needs to occur as devices and applications move from network to network. To face this challenge, researchers at ETH Zürich have developed PROSE, an infrastructure that supports automated software adaptation by extending applications at run time.

Mobile and pervasive computing poses a significant challenge in terms of software development. Applications running on devices that are mobile and continuously moving from one computing environment to another need to be able to adapt to these changes. For instance, as a device moves from one building to another, it might have to face different access policies from the available services. Perhaps in one building, communications can be sent in plain text and in another they need to be encrypted. The question is whether the ability to change from one mode of operation to another needs to be a dynamically acquired trait or something that is pre-programmed into the application. This is not an idle or insignificant question, since the ability to adapt is critical to that particular application remaining functional. If the adaptation is built into the original code of the application, that adaptation will necessarily be limited in scope and flexibility. First, the adaptation will only apply to situations foreseen by developers: as soon as new networks or new conditions appear, the application ceases to be operational. Second, the adaptation will have to be triggered by the application itself, thereby requiring valuable resources (CPU, memory, battery) to be devoted to it. These problems suggest that the adaptation process needs to be external to the application.

As part of ongoing work in a variety of new applications involving wireless networks and mobile computing settings (robots, flexible production environments, simultaneous utilisation of different network interfaces, etc), we have developed PROSE, a platform for dynamic adaptation.

PROSE is a versatile infrastructure designed to implement adaptation externally to the application. In this way, an application can be extended in a wide variety of ways. For instance, an application can be extended by a base station as soon as the corresponding device enters the domain of that base station. Two devices that meet spontaneously can exchange software extensions to ensure that their applications can interact properly. As another example, software residing in a complex sensor network can be upgraded and debugged remotely through extensions that are dynamically distributed from a central location or from some of the sensors themselves.
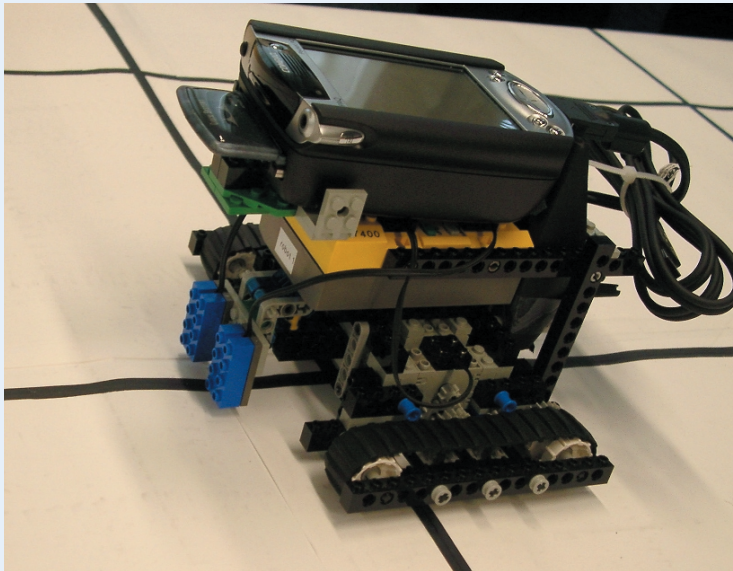
A robot with a PDA and wireless card running PROSE for dynamic adaptation

The process of adaptation can even take place automatically. For instance, a robot moving through a building can be adapted to different situations as it encounters different networks. These adaptations can be made during run time, without stopping the application. Thus,

PROSE provides an excellent middleware platform offering a great deal of versatility in pervasive computing environments.

PROSE uses dynamic Aspect-Oriented Programming techniques to implement

the code of the application and make it amenable to adaptation. Currently several versions of PROSE are available (see links below) using a variety of mechanisms. One is based on the debugger interface of the Java Virtual Machine (JVM). Another is based on the Just-in-Time compiler of the JVM. More recently, versions of PROSE have been developed using dynamic-class loading and recompiling. There is also a version for the .NET platform. Overall, the overhead involved is small (7% slowdown in selected benchmarks) compared with the gains in flexibility and the reduction of core code necessary for a working application.

Links:
http://www.iks.ethz.ch
http://prose.ethz.ch
http://www.inf.ethz.ch/~andfrei/jadabs

Please contact:
Gustavo Alonso
ETH Zürich/SARIT, Switzerland
Tel: +41 1 632 7306
E-mail: alonso@inf.ethz.ch

Deliver: A Fresh Look at Software Release and Deployment

by Gerco Ballintijn, Remy Jansen and Tijs van der Storm

The goal of the Deliver project is to fully automate the processes of software release and deployment by explicitly storing and using knowledge about the software. To achieve this goal, Deliver performs research in the areas of configuration management, software product lines and software deployment.

The release and deployment of enterprise application software is a complex task for a software vendor due to the enormous scale of the undertaking. There are many customers for the vendor to serve, all of whom might require their own version or variant of the application. Furthermore, the application itself will consist of many (software) components that depend on each other to function correctly. On top of that, these components will evolve over time to meet the changing needs of the customers. As a consequence, the release and deployment of these applications takes a significant amount of effort and is frequently prone to errors.

The goal of the Deliver project is to ease the software release and deployment effort and reduce the risks associated with it. This will be achieved by explicitly managing all the knowledge about the application, thereby allowing software vendors to improve the software upgrade process. For instance, the creation of incremental upgrades can be simplified and consistency requirements can be enforced. Furthermore, the explicit management of software knowledge also enables the evaluation of 'what if' scenarios; for example, what will happen to the software configuration of customer X if she upgrades application component Y? These evaluations improve the risk assessment of the

deployment process, and in turn these assessments improve vendor-customer interaction.

Managing software knowledge is, however, only part of the story. The software must still be delivered to customers. The Deliver project aims to support dynamic delivery of software via the Internet, in the form of both full packages and upgrades. Fortunately, the explicitly managed software knowledge can also be used to enhance this process (see Figure 1). Specifically, it can be used to compute the difference between a customer's existing software configuration and the desired new configuration. This difference can then be used to
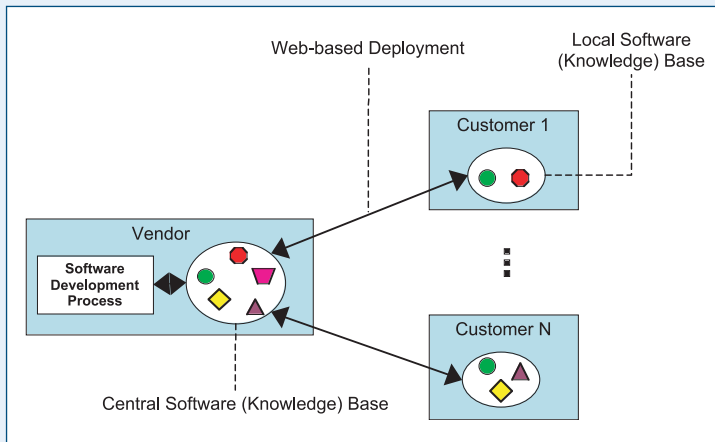
Figure 1: Intelligent software knowledge base drives Web-based delivery.
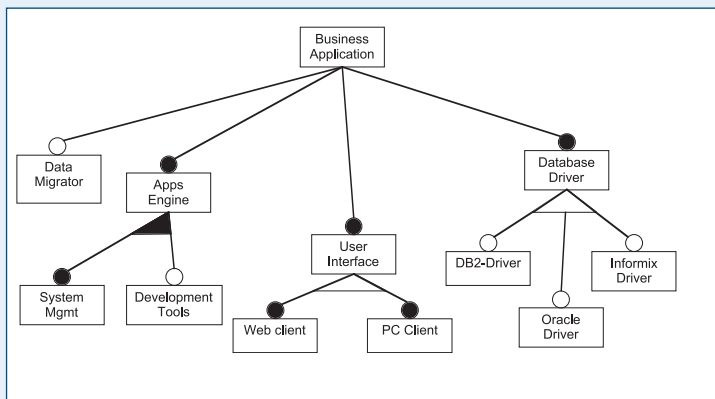
Figure 2: Hierarchical description of software variability.

teristics is the use of case studies that examine the practices of software companies. These case studies supply the Deliver project with knowledge of the specific real-life issues of releasing and deploying large-scale software and of the contexts in which they occur.

Additionally, the case studies offer the companies involved an external assessment of their own release and deployment activities. An important part of this assessment is a comparison of the company's activities with both the 'state of the practice' and the 'state of the art'. Based on these assessments, the Deliver project is able to propose improvements and extensions to the current practices of the companies.

Recently, a case study was completed into the release and delivery processes of the Dutch software vendor Exact Software. Exact Software is a leading manufacturer of software for accounting and office automation, with offices in 26 countries and over 160,000 customers. During the case study a comparison was made between the release and deployment tools used by Exact Software and the features of the ISKB. This comparison lead to several improvement proposals, including the tracking of dependencies between components and the introduction of configuration management techniques, such as branching and change sets.

Link:
http://www.cwi.nl/projects/deliver/

Please contact:
Gerco Ballintijn, CWI, The Netherlands
Tel: +31 20 592 4266
E-mail: Gerco.Ballintijn@cwi.nl

Sjaak Brinkkemper
Utrecht University, The Netherlands
Tel: +31 30 253 3175
E-mail: s.brinkkemper@cs.uu.nl

create the required upgrades. To deliver the software upgrade to the customer, the Deliver project must develop specialised delivery methods and protocols.

Central to the release and deployment activities envisioned by the Deliver project is the Intelligent Software Knowledge Base (ISKB). This knowledge base can be seen as an extension of a version control system that stores all information about all the artefacts that are part of the application's life cycle. The ISKB consists of two parts: a global knowledge base at the vendor's side that stores information on all available applications in all available versions, and a local knowledge base at the customer's side that stores information about the installed applications. An important part of the Deliver research is the design and implementation of the ISKB.

As a first step toward the ISKB, the Deliver project has investigated how Feature Diagrams can be interpreted as configuration interfaces for software components. Feature Diagrams are used for analysing commonality and variability of software products or compo-

nents (see Figure 2). To ensure that components are configured and composed correctly before they are delivered to a customer, a technique has been developed to store these Feature Diagrams in a prototype relational knowledge base.

The Deliver project is based at the Centre for Mathematics and Computer Science (CWI) in Amsterdam and is funded by the Jacquard program of the Netherlands Organisation for Scientific Research (NWO). To further its research goals, the project cooperates with both the Utrecht University and the Dutch Platform for Product Software. The Deliver project started in the summer of 2003 and is intended to run for four years. The research will be performed by five researchers and will result in scientific publications and prototype tools.

Critical to the success of the Deliver research is close collaboration with industrial partners. Such collaboration is necessary to examine the specific characteristics of the release and deployment phase of enterprise application software. One method for examining these charac-

Libresource : An Open-Source Versatile Collaborative Distributed Environment

by Pascal Molli and François Charoy

Current Collaborative Distributed Environments (CDE) have several drawbacks that oblige users to rely on a small number of installations like SourceForge with all the risks that that implies. The goal of the Libresource project is to provide a framework and associated services that will allow to networks of CDE to be deployed and maintained, in order to support a variety of projects, particularly development projects. Libresource is RNTL-funded project. It is developed on the ObjectWeb Jonas application server.

In recent years, a number of platforms have emerged that support cooperation between people over the Internet. In general, these environments all provide the same kind of functionalities or services. Community-oriented platforms (Zope/Plone, PhpNuke) provide communication services (forum, mail, notification) and content-management services (publication, Wiki, file management). Production-oriented platforms (Sourceforge and its descendants Gforge and Savana) provide file-sharing services (configuration management, workspace management), and communication and coordination services (task management, trackers).

One major problem with these environments is the perennial nature of their service. If SourceForge closes today, the community loses access to 80 000 projects and all their data. This centralised approach to managing communities and projects naturally presents a major drawback. The Libresource project intends to provide an alternative to these environments in order to avoid this problem. Instead of installing a single, complex community and project support site, we propose to support federation and pooling. If one node is forced to close for some reason, resources hosted on this site can be moved to another site. With this vision, the infrastructure needed is much simpler and the visibility of the nodes is maintained. Several conditions are required for the success of Libresource: easy project migration, easy installation and maintenance, and appropriation by communities.

Project Migration

Allowing migration of a project from one site to another implies that it can be serialised to an external representation on site and then deserialised on any target site. Current CDEs are built by gluing together existing tools like CVS, majordomo, Telnet and FTP, and their serialisation is complex. In Libresource, every component is serialisable as a Java object and can be transferred on another site. Libresource is built around a generic kernel (la courgette) that

The So6 tool directly supports data process. Red circles represent workspaces and green ellipses represent repositories.

provides basic services: a naming service, an event service and a security service. Two specific components have been integrated: these are So6, a novel configuration management tool and Bonita, a workflow engine.

So6 is a content management (CM) tool based on the transformational approach. It can be considered as a unified framework for building robust merge tools and verifying their security. It also provides a unique mechanism of synchronisation networks that replace the classical and cumbersome branching of CM tools. Thus, data process can be directly supported. This is represented in the Figure, where red circles represent workspace and green ellipses represent repositories.

The Libresource process engine based on Bonita allows two kinds of processes (described as administrative and cooperative) to be defined, controlled and executed. An activity in the workflow can interact with all existing resources of any instantiation of the platform.

Site Installation

The existing CDE is an aggregation of existing tools that need to be installed and maintained independently. However, this is unacceptable if we require a large number of LibreSource nodes to be deployed. Libresource has been developed on top of the Jonas J2EE application server. The kernel and component services are archive files that can be deployed separately. The Web application is built on top of these services. The installation of Libresource is simple. The components are deployed on Jonas and the service is ready to go. Site maintenance is limited to traditional administration activities of a database and of an application server. It is intended that Libresource will eventually be provided as a bundle with Jonas.

Site Appropriation

Sourceforge and its siblings have rather rigid user interfaces that do not allow easy adaptation. The Libresource interface has been designed to be simple and flexible. Navigation is based on the Libresource resource tree and on links in Wiki-like pages. Each menu can be customised for each node as a Wiki page. The resource view in the middle of the interface depends on the resource type. It can be customised by changing its JSP pages.

The organisation of the site and projects, the resources associated with projects and the access rights can all be adapted to new installations and to some extent to each project. Users are therefore not limited to the rigid structure of projects supported by current systems.

Perspectives

We have deliberately employed a very open approach that does not try to bind users to a given installation. Users can install and maintain their own servers or benefit from servers maintained by others without having to worry about their data. Networks of servers can be developed and projects can easily migrate between them. Using our own installation of Libresource, we are now developing new services and enhancing existing ones to cover all the needs of current cooperative projects.

The consortium managing Libresource is composed of the Artenum company, the ECOO research team from LORIA, and the University Paris 7 Denis Diderot. It is supported by the RNTL (the French National Network of Research and Innovation in Software Technologies).

MECASP – An Environment for Software Maintenance and Adaptation

by Elaine Isnard, Enrique Perez and Alexandra Galatescu

MECASP is a development environment dedicated to the maintenance, adaptation and follow-up of upgrades for heterogeneous software and data structures (built with Java, relational database management systems, tools for building graphical interfaces, etc). It can further integrate and manage other types of applications and structures, including C++, VB, CORBA, EJB, .NET etc. It enables these composite applications to be diffused more easily in ASP mode, but its use is not limited to the ASP market.

MECASP (Maintenance and improvement of component-based applications diffused in ASP mode) resulted from an IST European project, which was completed in December 2003. The IST consortium that developed MECASP was composed of Prologue Software (Paris, France), Virtual Desk (Madrid, Spain) and the National Institute for R&D in Informatics (Bucharest, Romania). MECASP relies on an open meta-model and uses new technologies (XML). It is a portable environment (fully written in Java) with a client-server architecture, relying on a free open-source infrastructure (composed of Apache server, MySQL, Castor, Slide, XML:DB API, Xalan etc). The client interface is through a specific browser.

MECASP overcomes the limits of the existing tools for the integration and maintenance of software/data structures, ie version management tools like CVS. The most important limits of these tools are:
- they deal with the versioning of text files only
- they rely only on a diff-like comparison of the source-file versions
- they have a primitive mechanism for tracking and merging the changes.

A database, for instance, is treated as a binary file, without dealing with its schema and the changes on it from one database version to another.

Three features of MECASP differentiate it from existing tools for software maintenance:
- configuration and maintenance of heterogeneous software, based on the XML modelling of the software and data structures
- (semi)-automatic and semantic merging of versions for heterogeneous non-text objects and applications, relying on rule-based decisions to solve conflicts and inconsistencies
- installation of new versions of the software by installing the changes only.

Instead of merging the versions of the physical application (as other tools do), MECASP implements an algorithm for merging the XML models that define these versions. This algorithm relies on the XML customised definitions of the changed actions, existing in each version of the application. These change actions include both standard actions (like 'create', 'delete', 'update' and 'move' objects or properties) and non-standard

actions (like 'compile' or 'search and replace').

The MECASP merger overcomes the limits of the existing tools, by the semantic interpretation of change actions. In the delta-like files of the existing version managers (files that contain the differences between two versions of the same application), any change is tracked by a combination of 'delete' and/or 'append' operations. In the case a populated database, these two operations are not appropriate to switch two columns, for example, because the existing data will be lost during 'delete'. In MECASP, a 'move' operation is implemented and semantically interpreted.

Using a (semi)-automatic rule-based decision mechanism for conflict resolution, the list of change actions is simplified and the change operations of several users are chronologically interleaved.

Special types of change operations (eg compile, search and replace etc), also tracked in deltas, are treated by specific merge rules.

The (semi)-automatic decision on the merge result (including fully automatic merge, merge after user decision, refused merge, recommended merge) relies on predefined rules that depend on the type of change actions, the type of objects they act upon and the role of the version in the merge operation (donor or receptor).

With MECASP, every time a merge is performed, further merges become easier. During the merge process, the user sees not a rough diff-like text comparison, but rather the real changes with their semantics explained.

MECASP addresses the needs of software development companies, providing them with new maintenance and upgrade facilities in order to increase productivity, reduce costs and improve distribution and installation of new software version.

A first MECASP demo has been set up for a CRM application (SellWin) on the SourceForge site. It includes Java source code (Netbeans), database schema (MySQL) and graphical forms (Swing/Netbeans).

Starting in 2004, MECASP will be distributed in open source from http://mecasp.free.fr and the SourceForge site.

Please contact:
Alexandra Galatescu, National Institute for R&D in Informatics, Romania
E-mail: agal@ici.ro

Elaine Isnard, Prologue Software, France
E-mail: eisnard@prologue-software.fr

MTL and Umlaut NG: Engine and Framework for Model Transformation

by Didier Vojtisek and Jean-Marc Jézéquel

Model Tranformation Language (MTL) is an open-source object- and view-oriented imperative language dedicated to model transformations, developed in the context of the OMG Query/View/Transformation (QVT) proposal. MTL is the stepping stone towards defining a transformation framework that we call Umlaut NG. It enables a variety of advanced model transformations, such as application of design patterns, weaving aspects at design level, synthesis test cases from UML models and derivation of products in product lines.

Model-Driven Engineering (MDE) is an approach to application modelling and generation that has recently received a lot of attention. This is a software development approach in which the concept of the model (rather than the code) is central to the development cycle. It allows assistance and/or automation for model creation from early developmental stages to code generation. This is a logical evolution of UML (Unified Modeling Language), supporting the following ideas:
• a model expressed in a formally defined notation is a key to system understanding

• system building can be organised around a set of models by defining a series of transformations between models, and organising the layers and transformations into an architectural framework.

MDE provides a way of working at the most appropriate abstraction level for a given problem.

For example, it may be used to apply an MDA approach (Model-Driven Architecture), formalising and automating the use of PIM (Platform-Independent Model) and PSM (Platform-Specific Model). The resulting conception lifecycle creates abstract models, which are successively refined into more concrete models, and more and more platform dependents.

Along with CEA-List and Thales, INRIA is participating in the Carroll common research program, which is dedicated to studying the model transformation aspect of MDE (see http://www.carroll-research.org). INRIA's Triskell team has developed MTL (Model Transformation Language) in order to tackle this. MTL allows the manipulation of models from any meta-
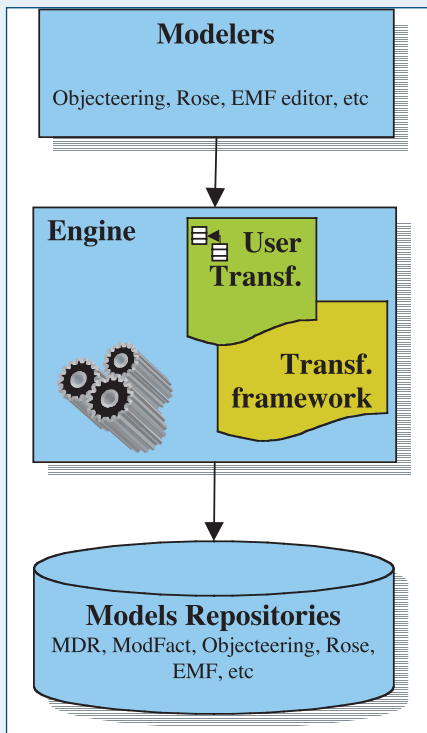
Figure 1: Three-tier architecture separating the engine from the user part (the modeller) and the place where models are stored (the repository).

model, including the various versions of UML. In order to build up a community around model transformation, our project is distributed as open-source software. Its user interface is based on a dedicated plug-in within Eclipse.

The language was designed in keeping with three main goals:
- For a greater adaptability, the transformation engine is completely tool independent. We end up with a three-tier architecture that separates the modeller, the transformation engine and the model repository roles (see Figure 1). Consequently, a transformation can be run on models of any repository, provided the models have compatible meta-models. Currently MTL supports the following repositories: MDR (MetaData Repository, from Sun), ModFact (from Lip6) and EMF (Eclipse Modeling Framework, from IBM).
- Since model transformation may become quite complex, the transformation developers need to reuse popular know-how and best practices in software engineering. In other

words, the transformation must be designed, modelled, tested and so forth. The language therefore uses an object-oriented style similar to popular languages like Java and C#. One of the special features is that elements of the models and classes of the language are manipulated in a consistent way. There is no difference between navigating or modifying a model and using transformation classes (see Figure 2); for instance, both use the concept of association. Best practices obviously include the ability to apply the MDE approach to the transformations themselves. This is done within MTL with a bootstrap process: the components of the transformation engine are written using the engine itself.
- Many transformations already exist, defined using their own approaches. It would be pointless to rewrite them all. Nevertheless, they must be integrated into a common structure. As a central tool, MTL helps investigate and federate different research areas and tools related to model transformation as they share common concepts: models and meta-models. The organisation of all transformations is a real challenge and will constitute a framework dedicated to model transformation. We call it Umlaut NG (Next Generation), by analogy to Triskell's previous transformation framework. For example, the Triskell team has already used MTL in the UML context to demonstrate several concepts related to model transforma-

tion usages. This includes, but isn't limited to, application of design patterns, support of the Design by Contract approach, weaving of modelling aspects, derivation of products from product lines, code generation, simulation of functional and extra-functional features of a system, derivation of state charts from HMSC (High-Level Message Sequence Chart, the basis of UML2.0 sequence diagrams) or synthesising test cases from UML models.

It is currently our intention to organise existing transformations and produce a working framework for this language. It could then be used as a demonstration and, at the same time, as the starting point for practical research in this field. In parallel, we are actively involved in the QVT process to ensure compatibility between MTL and the future normalised OMG language for model transformation.
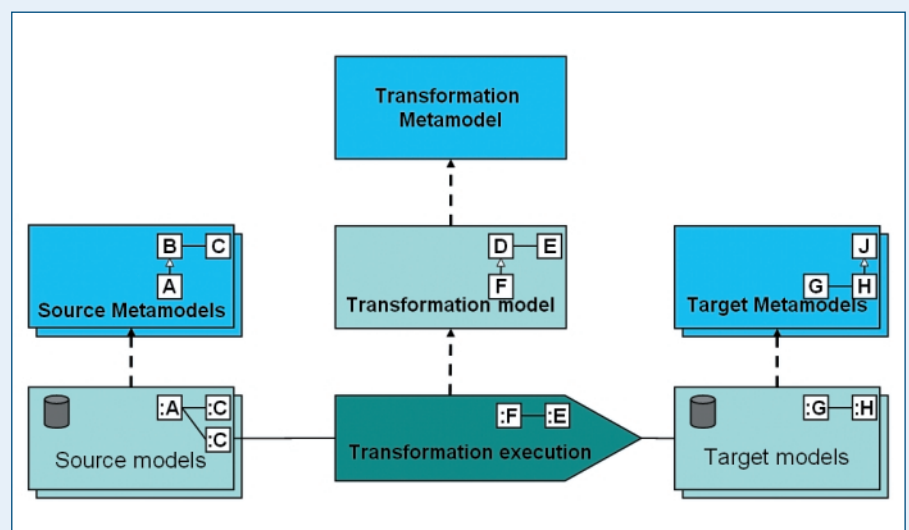
Figure 2: Analogy between model elements and objects of the language.

AriadneTool: Automating the Development of Hypermedia and Web Applications

by Paloma Díaz, Susana Montero and Ignacio Aedo

Because of the increasing size and complexity of hypermedia and Web applications, it is acutely necessary to use design models and methods whenever quality, usability, maintainability or re-usability are critical. AriadneTool is a software environment devoted to assisting hypermedia and Web developers by providing them with tools to specify structural, navigation, presentation, interaction and access requirements in a flexible and integrated manner.

AriadneTool is based upon the Ariadne Development Method (ADM), both of which were created at DEI Laboratory at University Carlos III of Madrid. They constitute the primary result of a three-year national project funded by the Ministerio de Ciencia y Tecnología (project no. TIC2000-0402). This work it is also strongly related to the project 'System to Develop Hypermedia/Web Applications Based on the Use of Design Patterns', funded by CAM and FSE (project no. 07T/0024/2003 1).

ADM proposes a detailed and flexible process to help hypermedia and Web developers produce design models of their applications. Thus, developers can discuss design issues whilst specifying, at different levels of abstraction, how their system is structured, what navigation capabilities will be offered (including links and other navigation aids such as guided tours or maps), how the user will interact with the system components, which functional requirements are identified, what presentation features the system components have and which access rules will apply in order to provide a useful and safe environment. The method is based on the following premises:

- To describe a flexible and user-centred development process: ADM establishes a systematic, iterative and user-centred development process that relies on three phases - Conceptual and Detailed Design which address the hypermedia development from two different abstraction levels, and the Evaluation phase based chiefly on prototypes assessment. Each phase is decomposed into a number of activities, each of which produces certain artefacts (diagrams, specifications, documents and catalogues). Moreover, the method is flexible; developers can establish a process model fitting their needs, so that both bottom-up and top-down development approaches are supported.

- To assume a reference model to describe the real world and transfer it to a physical system: the ADM notation used for the design is based on low-level entities belonging to the Labyrinth hypermedia reference model. These entities represent the core components of any hypermedia or Web system (eg nodes, contents, links, anchors, attributes, events, roles), as well as basic constructs (eg location function, composition mechanism, anchoring system, time and space-based constraints) offering a common language valid for different implementation platforms.

- To provide design artefacts for specifying different kinds of system requirements: ADM integrates several design views within a unified framework, including navigation, presentation, structure, behaviour, function and access. These views enable designers or developers to specify characteristics of the desired system in a progressive but integrated way.

- To support validation and integrity rules among design phases and products: ADM provides a number of inter- and intra-validation rules with which to check completeness, consistency and integrity among the various design artefacts created at each phase of the development process.

- To offer software support tools assisting in the development process of hypermedia and Web systems: ADM is supported by a design toolkit named AriadneTool.
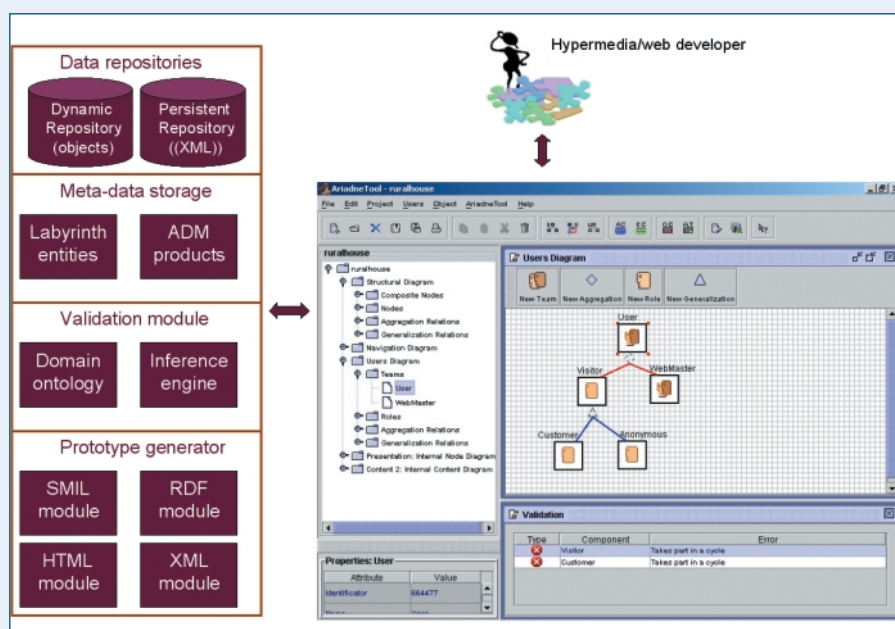
Architecture of the AriadneTool.

For each hypermedia or Web system, an AriadneTool project must be created. The project will hold all the ADM products and documentation concerning such a design. Moreover, AriadneTool users will be able to validate their products, checking the completeness, correctness and integrity of the design as well as generating templates in SMIL, HTML, XML and RDF formats. The architecture of this tool is shown in the Figure.

The AriadneTool front-end is the graphical interface used by developers to design their projects. Due to its highly interactive nature, the toolkit is being developed using an iterative design process, during which the evaluation of AriadneTool is considered vital to improving its utility and usability. Data Repositories hold specific projects created with AriadneTool. The Meta-Data Storage contains all ADM modelling constructs, including core entities and the design artefacts made up by such entities. The Validation Module validates the completeness and correctness of the design artefacts, noting any mistake or warning to the designer. In order to implement this module, ontologies are integrated within the tool. Ontologies help to describe the model in a complete, comprehensible and formal manner and make it possible to check the application semantics. The Prototype Generator dynamically produces the application prototype by querying the Dynamic Repository. Because all design views of an application have been specified declaratively, prototypes can be generated for different delivery languages. All our efforts have been aimed at mark-up languages proposed by W3C, such as HTML, XML, SMIL and RDF/RDFS.

AriadneTool is implemented using JDK1.4, which allows us to obtain an independent operation platform.

The SE2C Experience on Automating the Design, Transformation and Verification of Models

by Paris Avgeriou, Nicolas Guelfi and Paul Sterges

The Software Engineering Competence Centre (SE2C) at the University of Luxembourg conducts cutting-edge research in the area of automated software engineering. Ongoing research is focused on partially automating the design of applications through architectural frameworks, the transformations of UML models, the reconciliation of architectural models and formal verification of models.

Automated software engineering is at the centre of SE2C's research activities. We are interested in providing development teams with tools that will automate, at least partly, the design of applications. We are particularly interested in model-driven software engineering related to the Unified Modeling Language (UML), and in automating the transition between the various models through transformations. We are also working in architecture-centric software development, where the design and transformation of architectural models play a key role in the development process. Finally we are aim at providing formal verification of models in application domains, such as e-business transactions. The current work of the SE2C includes a UML model transformation tool, an architectural framework, transformations of architectural models, and verification tools.

MEDAL (uMl gEneric moDel trAnsformer tooL) is a CASE Tool that supports an approach named Visual Model Transformation (VMT), and that aims at providing generic transformations of UML models. The VMT approach is based on graph transformation theory: each transformation rule defines a graphical pattern to be matched in the source model (known as 'left-hand side' (LHS)) and another graphical pattern depicting the resulting target model ('right-hand side' (RHS)). Both the LHS and RHS are defined using a visual notation plus additional properties that are expressed with a textual notation. Composite transformations can be synthesised from primitive transformations in order to provide a modular mechanism for defining reusable transformations of desirable complexity. The MEDAL Tool is an extension of IBM-Rational's XDE, an environment that supports round-trip engineering, and builds upon XDE's model server, OCL engine, and pattern engine.

The FIDJI architectural framework aims to automate the application of FIDJI: a model-driven, architecture-centric, framework-based software development process. The FIDJI architectural framework is comprised of a component-based application framework for the development of J2EE-based applications, and a CASE tool that assists the extension and customisation of the framework for specific applications. In this sense, development teams of individual projects can reuse the application framework, customize it and extend it so as to satisfy their own functional and quality requirements. The CASE tool assists them in designing the final application and in transforming this design model into a prototype implementation consisting of a UML implementation

model, Java source code, configuration files, and deployment descriptors.

The DIAS project deals with the task of keeping the requirements and the implementation of a software system in sync. This is done by transforming two architectural models - one of which is based on requirements and the other on implementation - into a common model. It is commonplace that the gap between the requirements and the implementation of a software system cannot be bridged directly, but demands intermediate representations. From an architecture-centric viewpoint, these representations take the form of an ideal architectural model that conforms to the requirements, and a realistic architectural model that conforms to the implementation. Semi-automated support in this case is provided through a transformation that combines the two architectural models into a new model that represents a trade-off between the desired requirements

and implementation constraints. This form of architectural reconciliation takes advantage of the aforementioned VMT approach and MEDAL tool. In the context of this project, SE2C is organising a workshop for architectural design and UML (see http://uml2004.uni.lu/), hosted at the UML 2004 conference.

In the E-fficient project, SE2C is working on verification tools for e-business transaction descriptions. The descriptions, which are made in UML by business experts, mainly consist of activity diagrams showing a decomposition of transactions into a number of activities, and of class diagrams indicating the structure of the messages that are exchanged during the transactions by the participants. We are working on an automated verification tool that consists of plug-ins implemented inside a modelling tool. This tool verifies that the set of UML diagrams of a given transaction fulfils a number of predefined static,

dynamic and real-time properties. In running this verification, our tool calls the following model checkers: USE, SPIN and Kronos. The properties to be verified are written in the various languages supported by the tools: OCL for USE, PROMELA for SPIN and TCTL for Kronos. This approach requires translating the UML-based description into formal descriptions for the different tools. The use of different tools allows a wide range of constraints to be specified, ranging from syntactical constraints to real-time constraints.

Links:
SE2C homepage: http://se2c.uni.lu

FIDJI Project: http://se2c.uni.lu/projects/fidji

DIAS Project: http://se2c.uni.lu/projects/dias

E-fficient Project:
http://se2c.uni.lu/projects/efficient

Please contact:
Nicolas Guelfi, University of Luxembourg
Tel: +352 420 101 251
E-mail: nicolas.guelfi@uni.lu

Model Driven Development of Pervasive Systems

by Javier Muñoz, Vicente Pelechano and Joan Fons

Computing-based systems growth is arriving to all environments of our daily life. This situation requires solid engineering methods in order to develop robust systems. Model-driven architecture guidelines application to pervasive systems development can help to build better systems in an easy way.

Current trends in software engineering claim that models must play a more important role in the development of robust systems. These approaches suggest that models must be the most important artefacts in the development cycle; therefore systems developers build and transform systems models in order to achieve automatic code generation. The Object Management Group (OMG) has reflected these trends in its Model Driven Architecture (MDA) proposal, which is being adopted as a new industrial strategy. MDA guidelines application to pervasive systems development can help to build better systems in an easy way.

The MDA proposal defines the building blocks for constructing model driven methods, but it does not specify concrete techniques. In order to make MDA more useful, a MDA method should provide techniques for each MDA building block. Applying the MDA approach (see Figure 1) we propose the following techniques:
- *A precise language for building Platform Independent Models (PIMs).* System developers use this language for precisely describing the system with high-level constructs. We propose to use the Pervasive Modelling Language (PervML).
- *One or many modelling languages for building Platform Specific Models (PSMs).* The constructs of these

languages must be direct rep-resentations of constructs of the technology they model. Currently, we are working in the development of a language for modelling an OSGi system. OSGi is a Java middleware initially created for hosting software of residential gateways.
- *PIM to PSM transformations.* These transformations define how a PIM can be converted to a PSM. Currently, model transformations are a hot research topic. We apply graph grammars for defining the transformations from PervML to OSGi.
- *PSM to source code transformations.* Finally, the code generation from the PSMs is the last step of the development method. Due to the fact that
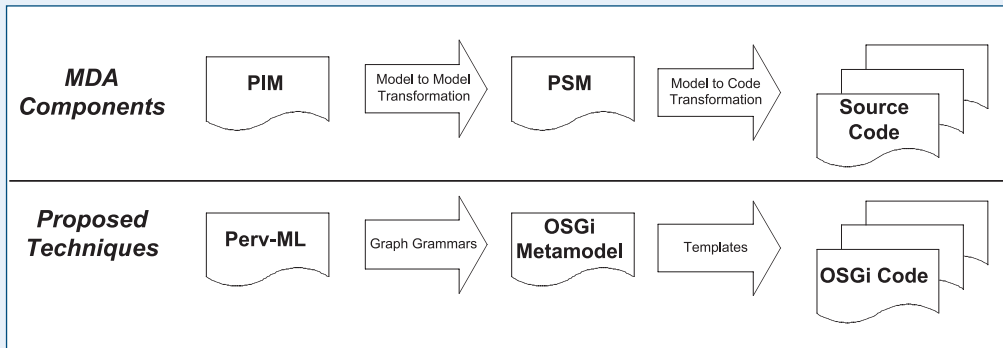
Figure 1:
MDA building blocks and our proposed techniques for pervasive computing.

PSMs are expressed using technological terms, transformation to source code is immediate. We are applying templates to the elements of models in order to obtain the source code.

Pervasive Modelling Language (PervML) is a language designed with the aim of providing the system analyst with a set of constructs that allow to precisely describing the pervasive system. PervML promotes the separation of roles where developers can be categorized as analysts and architects. Systems analysts capture system requirements taking as an input the users needs and they describe the system at a high level of abstraction using three diagrams (models) that constitute what we call the Analyst View. On the other hand, system architects specify what COTS (devices and/or software systems) realize system services building other three specification models that constitute what we call the Architect View. Figure 2 shows the language organization. The dashed arrow of Figure 2 defines the construc-

tion order of the conceptual models that our approach proposes.

We define the graphical notation of PervML by using the extension capabilities that UML provides.

In order to integrate OSGi in our development method, we have to be able to create models that are built using OSGi constructs. We have developed an OSGi metamodel for defining these constructs and their relationship. The models (PSMs) are represented using an UML profile.

To define transformations between PIM and PSM involve jumping a wide gap between abstraction levels. Currently standards for transformation definition do not exist. We use graph transformations and graph grammars as the model transformation engine. From a mathematical point of view, a model can be seen as a graph where model elements are labelled nodes and the relationships between model elements are edges. In this way we apply the existing knowledge for defining graph transforma-

tions in order to achieve model transformations in the MDA context.

Our current research focuses on developing tools supporting this method. These tools guide system developers to systematically develop pervasive systems. They allow the developers to build conceptual models (PIMs) that constitute the source of an automatic code generator which applies model transformation rules to obtain a fully functional pervasive system.

Link:
http://oomethod.dsic.upv.es

Please contact:
Javier Muñoz Ferrara
Universidad Politécnica de Valencia/
SpaRCIM, Spain
E-mail: jmunoz@dsic.upv.es

Figure 2: PervML is structured in six models.

Articles in this Section

Image Analysis on the Archimedes Palimpsest

by Anna Tonazzini, Luigi Bedini and Emanuele Salerno

Image-processing procedures developed at ISTI-CNR have been applied to recover some of the hidden writing in the Archimedes palimpsest, an ancient manuscript in which faint remnants of several treatises by the great philosopher and mathematician are partially hidden under a more recent text.

Digital image analysis techniques developed at ISTI-CNR in the framework of the European CRAFT project IsyReaDeT were presented at the Archimedes Symposium in Baltimore, 1-4 April. This important initiative was sponsored by the owner of the Archimedes palimpsest. Since this is the earliest copy available and could reveal many as yet uninvestigated aspects of Archimedes' thinking, the goal is to develop techniques to recover as much of the ancient text as possible.

The manuscript has an incredibly tortuous history: it was first used in Costantinople during the tenth century to copy the works of Archimedes. It was then erased and reused for a Byzantine lithurgic book (an Euchologion) during the twelfth century, probably in southern Italy. This book then travelled through the Holy Land before returning to Costantinople, where it was rediscovered in 1846, in 1899 and again in 1906 when the well-known philologist Johan Heiberg published what he was able to read of the Archimedes text. After the First World War, the palimpsest disappeared, only to emerge yet again in 1998, when an anonymous collector bought it at a Christie's auction for USD 2,000,000. One of the treatises contained in the document is the only existing copy of The Method of Mechanical Theorems, in which Archimedes gives an account of how a philosopher can reach a result with the help of physical intuition, before formulating a rigorous proof. Many points of this work are now seen as anticipating modern calculus, as developed by Newton and Leibniz almost twenty centuries later. Another important component is the Stomachion, which had long been considered lost, and whose content was not known before 1906 (only a fragment of an Arabic translation was available). It now seems that this treatise deals with combinatorics, another largely anticipatory subject for mathematics in the third century B.C.

The palimpsest is now at the Walters Art Museum in Baltimore, which is in charge of its conservation and study. The book has been taken apart, its pages have been partially restored and all the material is now being imaged using various

Top: One of the available multi-spectral images of the palimpsest.
Middle and bottom: the separated Archimedes text and overwritten text, obtained by our decorrelation techniques.
Copyright: the Owner of the Archimedes Palimpsest.
Image capture: Rochester Institute of Technology and Johns Hopkins University.

techniques aimed at extracting as much information as possible from the original text. Digital image analysis techniques can play a particularly important role in enhancing the possibility of reading the text. The techniques developed in the IsyReaDeT project have proven particularly useful for this task.

IsyReaDeT began in January 2003 with funding for two years from the European Commission. Its objective was to develop a low-cost system with which to digitize, restore and archive ancient degraded texts. This system is based on a multi-spectral camera with dedicated restoration software. The project consortium includes three research institutions and six small enterprises active in the conservation and restoration of works of art and historical documents.

There is a considerable risk that in the future we may lose many governmental, historical and commercial documents through gradual decay. The digital imaging of important documents can ensure their conservation, enhance their readability and make it possible to acquire new information using non-visible wavelengths and digital image processing techniques. OCR processing for automatic transcription and indexing can facilitate access to digital archives and the retrieval of information.

Within IsyReaDeT, ISTI in particular is working on a mathematical description of the typical forms of degradation that affect documents, and the study of restoration techniques exploiting multi-spectral views. The forms of degradation considered are caused by complex background textures, bleed-through, show-through and spots. From multiple views of the document (colour or multi-spectral image data), statistical approaches can be used to separate, extract and classify the different patterns. The data
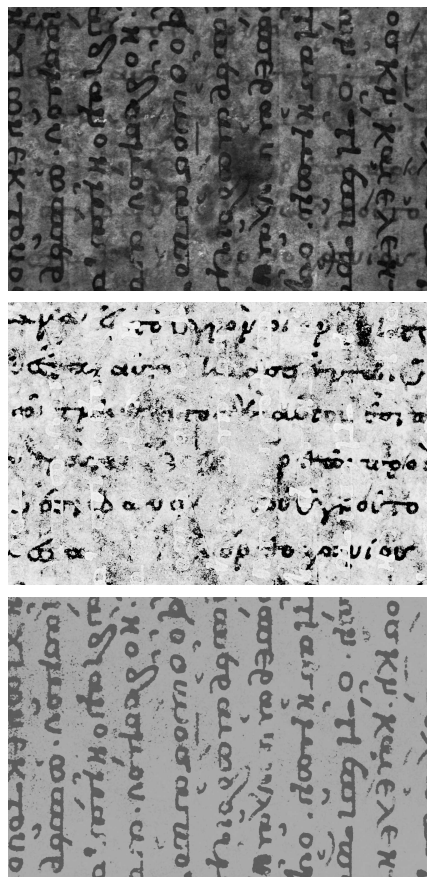
model we assume is a linear mixture of different classes, each characterized by a different reflectivity spectrum. Each class is considered to have an average value in each component (channel) of the data, and these values form an unknown 'mixing' matrix. Each pixel in an image contains the contribution from the local intensities of all the classes, multiplied by the relevant mixing elements. Since the mixing matrix is unknown, we must rely on 'blind' techniques to recover the

different classes. The Principal Component Analysis approach exploits the eigenvalue decomposition of the data covariance matrix to produce mutually orthogonal outputs characterized by maximum variance in each principal direction. A further step is the so-called Independent Component Analysis technique (ICA), in which, besides decorrelation, mutual independence between the output channels is required. The possibility of extracting interfering patterns

from the data can be particularly useful in enhancing barely visible classes, as in the case of underwritten text in palimpsests. IsyReaDeT was thus invited to try these techniques on multi-spectral images of the Archimedes palimpsest.

Links:
http://www.isyreadet.net/home.htm

Please contact:
Anna Tonazzini, ISTI-CNR, Italy
E-mail: anna.tonazzini@isti.cnr.it

Managing Medical Images

by César J. Acuña, Esperanza Marcos, Valeria de Castro and J. A. Hernández

Due to the increasing number of digital images used in medical diagnosis, health centres need new and better applications with which to effectively manage such information. It is also necessary to develop an infrastructure for integrating and sharing that information between medical image repositories. Researchers from the Rey Juan Carlos University of Madrid are working on this problem.

Recently, the use of digital images for medical diagnosis has increased considerably. New and better applications are therefore needed in order to effectively manage large amounts of medical information.

DICOM is currently the standard for Digital Imaging and COmmunications in Medicine. However, despite DICOM being the most widely accepted standard, it can only handle interchange and communications among medical imaging equipment and other specific software tools called PACS (Picture Archive and Communications Systems). PACS tools are developed for several providers that use proprietary formats to store and manage medical information. Drawbacks such as these make the integration and use of DICOM difficult in a wider context such as the Web. In addition, another image storage format, known as Analyze, is widely used. This is a proprietary format part of the Analyze Software.

As a further complication, the standard for information exchange and data transportation between multiple applications is XML. For these reasons, the Database and Web Engineering Kybele Research Group and the Electronic Technology,

Bioengineering and Medical Image Research Group at the Rey Juan Carlos University, together with some private and public medical centres in Madrid (Spain) are currently working on two main research topics. These are aimed at making digital image information more easily exchangeable in a wide context.
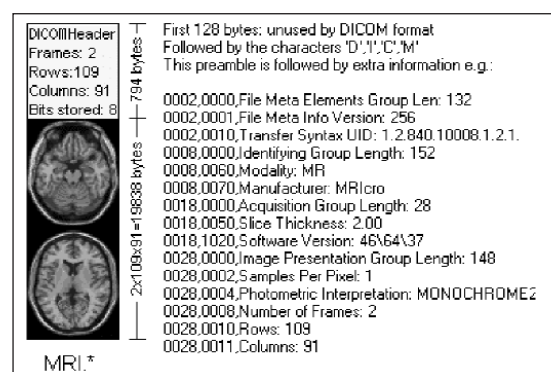
Figure 1: A DICOM file, composed of the data header and the image.

The first of these is the design and development of a Web Information System (WIS) with two main aims. The first of these is that DICOM files will be represented using XML, which will improve the interoperability and integration of medical information in a broader context like the Web. On the other hand,

it will facilitate the integrated organisation, query and retrieval of DICOM and Analyze files by means of an XML database.

The second topic is the design and implementation of a Web Integration Architecture that will integrate different instances of the WIS introduced in the previous topic. The main aim of this architecture is to assist retrieval of integrated information from medical image repositories located in geographically diverse medical centres.

The WIS is currently under development. Taking as a reference point the architectures for Web applications development proposed by .NET and J2EE, the WIS architecture was structured in three layers as shown in Figure 2. These are as follows:
- *Presentation Layer:* low-cost image delivery and display are necessary in most networked hospitals, so a Web-based user interface is the most effective solution in this case. To develop the Web-based user interface we have chosen ASP.Net.

- *Behavioural Layer:* the main components of the WIS are located in the behavioural layer. These are an XML-generator component developed in Java, which allows information to be transformed from DICOM and Analyze files into XML documents; an XSL Transformation component whose main task is to translate the XML documents generated by the first component into another XML document conforming to the database XML Schema; and finally, a Query Processor component responsible for building user queries in order to execute them on the database management system.
- *Persistence Layer:* XML databases are emerging as the best option for storing XML documents, because they provide better queryability, optimised updates, and stronger validation. They also avoid loss of semantics between the XML document structure and the relational database schema that can occur when XML information is stored in a conventional database. We have
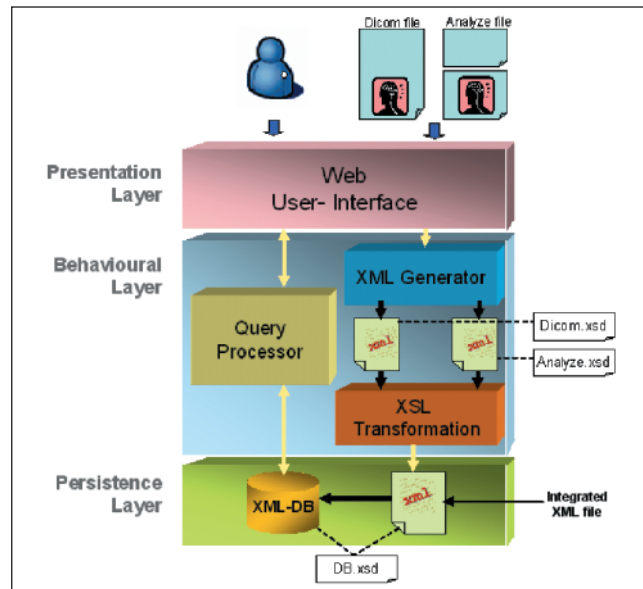
**Figure 2:
Web Information
System architecture.**

chosen the Oracle XML DB solution to implement the persistence layer, based on XML-database comparative studies and taking into account the previous experience of our group.

Our Web Integration Architecture is also in development, and is based on ontologies and Web services in order to develop a Web portal for integrated access to medical image repositories.

Please contact:
César J. Acuña
Rey Juan Carlos University / SpaRCIM, Spain
Tel: +34 91 488 8140
E-mail: cjacunia@escet.urjc.es

NICHE: Natural Interaction in Computerised Home Environments

by Francesco Furfari, Claudia Soria, Vito Pirrelli, Oreste Signore and Rolando Bianchi Bandinelli

Future technologies will provide users with increasing control over surrounding devices embedded in a common home environment. Somewhat paradoxically, this could result in an increase rather than a reduction in complexity if support for high-level interfacing is not introduced. This concern prompted the launching of a medium-term project aimed at promoting natural user-home interaction along the lines of the Ambient Intelligence vision.

NICHE (Natural Interaction in Computerised Home Environment) is a curiosity-driven project recently started in Pisa by ISTI-CNR. It was established on the basis of a collaboration between the Domotics Lab of the Institute, the Dylan Lab in the Institute for Computational Linguistics (ILC-CNR, also located in the Pisa CNR Research Campus), and the W3C Italian Office. The main research objective of NICHE is to address the interoperability and integration of home appliances with next generation smart devices (see Figure 1),

taking into account the impact of new emerging technologies (AmI) on our daily habits.

A key tenet of our approach to the problem is the central role played by human language in mediating user interaction with the home environment. Through language, a user should be able to control, query and program devices by using his/her own most immediate and congenial communication medium. With recent improvements in computer and language technology, natural language

systems begin to appear feasible. However, when it comes to human interaction with electronic devices and system agents, we must squarely face the challenging task of redesigning established practices and anticipating future user reactions and needs. Significant technical issues still need to be addressed before natural-language-driven interfaces become truly natural and conversational. For example, considerable care must be taken to avoid an intelligent system being perceived as an 'intrusive' or annoying presence (see ISTAG
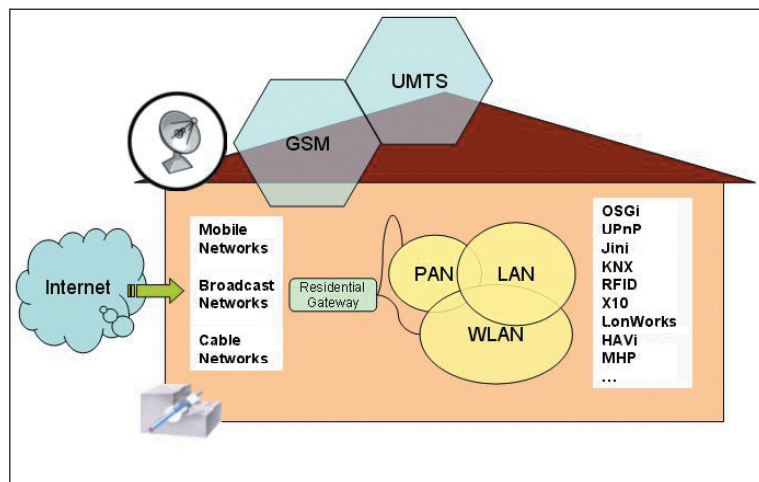
Figure 1: Home Networking Trends.
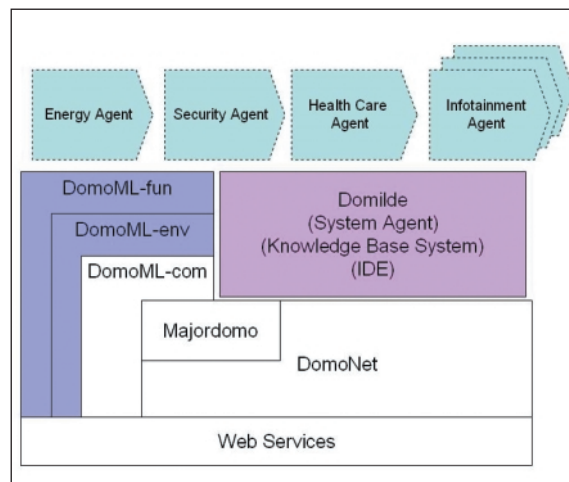
Figure 2: NICHE software architecture.

reports). Furthermore, we need a better understanding of the way humans and machines can communicate in the home environment. The development of a natural language-driven interface thus requires the preliminary definition of a cognitive model of user-home interaction. There are several goals in doing this: firstly, to precisely characterize the roles of those involved and the complexity of the tasks; secondly, to identify a taxonomy of intended devices; and thirdly, to identify those aspects that may affect the specifications of a 'dialogue manager', the 'intelligent' component that regiments the pragmatics of communication.

A second key aspect of our investigation is the ability to capture the unlimited variety of home environments in an incremental way, that is, in their dynamically evolving aspects. Ideally, the user should be able to manage new devices, automatically discovered and integrated in the home system, and interact through the implicit context of each specific device. For example, when using a TV we would like to be able to use concepts such as 'film', 'actor' and 'talk show'. This would allow us to take full advantage of next-generation distributed smart services (Semantic web, MAS) and their potential for proactively offering user-tailored services at home. To date, the use of a semantic layer in the domain of networked home devices mainly concerns the specification of executed commands, the categorisation of available devices, and the mapping between different home protocols. We believe

that recent developments in the Semantic Web research area should also be capitalised on to tackle similar problems within the semantic home intranet. This step is expected to offer the further bonus of making the home environment directly compatible with remote services.

Figure 2 outlines the overall NICHE software architecture. White boxes such as DomoNet (an XML-based middleware defining an abstract level whereby individual devices are represented and described as an overall integrated system) are background resources of the Domotics Lab, while coloured boxes represent the specific focus of the project work. In particular, we intend to release DomoML, a mark-up language intended to define a semantic layer on which novel value-added services (agents) can easily be built.

From a functional perspective, the Domilde box is comparable to an IDE (Integrated Development Environment), by means of which a home environment can be designed and configured. In this respect, Domilde is also reminiscent of networks based on X10, LonWorks, EHS, or UPnP, all of which need different kinds of logical and physical user intervention. From a less procedural viewpoint, Domilde defines a knowledge representation layer. Having natural interaction in mind, we need to come up with a formal description of our surrounding world that is interpretable by the system. For this purpose, we must avail ourselves of a formalism that

describes both pieces of factual knowledge such as 'the lamp is on the desk', and the meaning of those concepts (spatial and temporal relations, pieces of furniture, devices etc). The user will thus use Domilde to create an instance of his/her home environment based on a Web ontology expressed through the DomoML-env(ironment).

Finally, the home environment description is augmented with procedural knowledge added either by the user (by means of scripts) or by intelligent agents. DomoML-fun(ctional) is the formalism intended to express this kind of knowledge and share it with other agents or services. From this point of view, Domilde is equivalent to a system agent (sygent) whose role is to coordinate available services and cooperate with other agents.

Please contact:
Francesco Furfari, ISTI-CNR, Italy
E-mail: francesco.furfari@isti.cnr.it
or Claudia Soria, ILC-CNR, Italy
E-mail: claudia.soria@ilc.cnr.it

Towards Self-Managing Systems

by Joakim Eriksson, Niclas Finne and Sverker Janson

Scientists at SICS are addressing the problem of ensuring dependability and performance of networked systems in the face of uncertain, and often largely unknown, environments. In a first experiment, they compute automatic configuration rules for sensor networks, maximising sensor coverage and minimising radio communication and network latency.

As computing systems become more complex, management problems and costs tend to increase. IBM's Autonomic Computing initiative proposes to address this challenge by making computing systems more self-managing: self-configuring, self-healing, self-optimising and self-protecting, from individual components to complete environments.

An increasingly popular approach to self-management is to apply concepts and techniques from automatic control. Traditional linear control methods are well known, well understood, easy to implement and have a wide range of applications. A simple illustrative example is regulating the CPU load and memory use of a Web server using MIMO linear control.

Controlling an entire networked system is much harder. The system components all have local observation and control points and an incomplete knowledge of the system state. The effect of control is uncertain, interacts with that of other components, and can be much delayed. Optimal decentralised control is known to be computationally intractable, but this is essentially the problem facing system managers and self-management mechanisms.

Sensor network management is a case in point. The advent of low-cost sensor units (sound, light, vibration, temperature etc), which include limited but useful computing and wireless communication support, enables high-resolution monitoring by quickly deployed and cost-efficient sensor networks. Sensor units are distributed over an area, forming a wireless network, and data is collected and routed to users tapping into the network. The technology has a wide range of potential scientific, civilian, and military applications.

Ideally, sensor networks should be plug-and-play. Following deployment, the sensor units should automatically configure themselves for the best possible performance based on locally available information, eg on the perceived local density of sensors and on network throughput. If units then malfunction or are moved for example, or if changes in the environment or

Simulator and policy optimization tool for a sensor network monitoring the perimeter security zone of an airport.

power supply alter the connectivity, the network should automatically reconfigure.

We are investigating methods for computing task-specific self-configuration policies that can be downloaded to sensor units prior to (or during) a mission. A policy is a set of rules that assigns configuration actions to sensor unit states. Self-configuration is achieved as the sensor unit repeatedly checks if the current state matches any rule, and if so performs the corresponding configuration action. Such policy execution can be done efficiently with the limited computing resources of the sensor unit.

Performance goals and assumptions regarding the sensor network environment and sensor unit placement are encoded in a scenario simulator. The simulator runs episodes, where sensor units are randomly placed and a sensor network is automatically configured using a given policy. To reach a configured network, each node performs a number of configuration actions, ideally as few as possible. A policy optimisation method evaluates policies and refines them incrementally, converging to a (locally) optimal policy in a few thousand episodes.

Figure 1 shows the simulation and optimisation tool that has been developed.

The scenario illustrated is to monitor the perimeter security zone of an airport using a combination of sensor units and more powerful link and fusion nodes connected by a wireless network. The information is collected to the command and control centre.

Each sensor unit has limited local awareness of its number of neighbours, routing choices and the degree of network congestion. Based on this information, the sensor units and link and fusion nodes automatically configure scan mode, routing, fusion etc, for the best-expected sensor network performance.

Our solution uses a Monte Carlo policy evaluation and value-function-based policy refinement method, an instance of the large family of approximate dynamic programming methods. Dynamic programming guarantees convergence to optimality for centralised control when the state is fully observable. Approximate dynamic programming offers tools for larger and more complex control problems, in many cases offering theoretical guarantees. However, in the case of partial observability and decentralised control, fewer guarantees are available or even possible. On the research frontier, tractable special cases are being identified.

In our future work, we aim to extend the model with policy-controlled intra-network communication for increasing and optimising system self-awareness. We will also carry out experiments using specific hardware units (eg ESB 430/1), physical environments and missions.

Links:
IBM Autonomic Computing Initiative.
http://www.ibm.com/autonomic/
Autonomic Networked Systems at SICS.
http://www.sics.se/isl/autonomic

Please contact:
Sverker Janson, SICS, Sweden
E-mail: sverker@sics.se

From Regression Testing to Regression Benchmarking

by Petr Tuma

Automated testing is widely used to detect regressions in software functionality during development. This is in contrast with the impromptu handling of regressions in software performance. The Distributed Systems Research Group in the School of Computer Science, Charles University, Prague, is working on extending regression testing to cover performance as well as functionality.

Regression testing is a proven and popular approach to meeting the demand for quality assurance during software development. In this process, a suite of tests is developed alongside the software so that the software can be regularly tested, and regressions in its functionality discovered and fixed. This, however, does not extend to regressions in performance, which are often orthogonal to functionality and consequently missed by regression testing.

Our current research focuses on remedying the existing neglect of performance in regression testing. We are building on the results of our past middleware benchmarking and comparison projects with industrial partners such as Borland International and IONA Technologies. We have designed a regression-benchmarking environment capable of providing extensive and repetitive testing for regressions in performance, dubbed regression benchmarking.
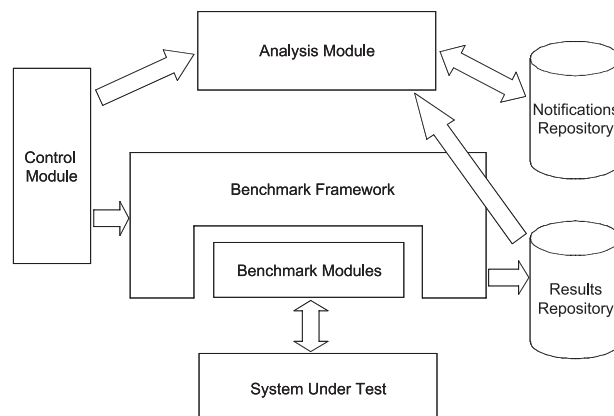
Figure 1: Regression benchmarking environment architecture.

Distinguishing Traits

Regression benchmarking is a special application of benchmarking that is tightly integrated with the development process and is fully automated. Alongside the modules and supporting framework found in most benchmarks, the regression-benchmarking environment contains a results repository that keeps a history of results, and an analysis module that examines the history and detects regressions in performance. The architecture of such an environment is outlined in Figure 1.

Foremost in the features that make regression benchmarking different from benchmarking in general is the requirement that the former be fully automated. The automation requirement concerns both the benchmark execution and the results analysis.
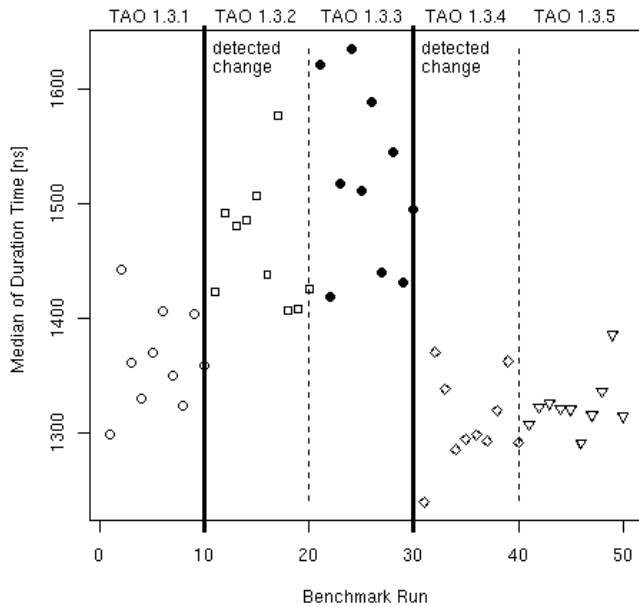
Figure 2: Change detection in simple benchmark results.

Figure 3: Clusters in complex benchmark results.

benchmarks provide little space for interference and so yield precise results with straightforward interpretation, while complex benchmarks exercise multiple functions of the software concurrently and therefore provide room for results to be influenced by complex interactions among the functions.

With simple benchmarks, we primarily strive to minimise variation among results. We design the benchmarks to measure short operations and express the results using robust estimators that are not affected by a small number of exceptional observations. This minimises the probability of any interference that would increase the variation of the results. While the variation is reduced, it is not altogether removed. This prevents performance regression being detected through a direct comparison of the results. We continue by interpreting the results as a sequence of independent identically distributed observations of a random variable, and compare the results using common statistical tests for comparing samples from two populations, as illustrated in Figure 2.

For complex benchmarks, we cannot easily minimise variation of the results, which is significantly larger than with the simple benchmarks. This prevents not only a direct comparison of the results, but also the use of the common statistical tests, which provide only weak results. We tackle this problem by interpreting the results as a union of clusters that can be compared one–by–one, using the traditional iterative clustering algorithms to separate the results into clusters (see Figure 3). Further work on the algorithms is required to avoid the need for manual hints and other input, which is incompatible with the regression benchmarking.

The automated benchmark execution proves to be relatively simple, with the existing remote-access and scripting mechanisms being well up to the task. One problem associated with this is whether the execution time is short enough to allow the execution to run frequently. This puts demands on the ability of the environment to recognise, without undue delay, when the benchmark is producing stable data (as opposed to data distorted during warm-up) and when sufficient amount of data has been collected.

The automated results analysis has proven difficult, especially where precision is concerned. To minimise the cost of finding and fixing a source of a regression in performance, the environment must detect the regression as early as possible. This implies a need to identify minuscule regressions in performance that consist of a sequence of individually negligible changes over a long period of time. Such changes are hard to detect in typical benchmark results, which tend to have a variation in the order of several percent.

Developed Techniques

Our work on regression benchmarking follows two directions, which differ in the complexity of the distinct groups of benchmarks they consider. One looks at the group of simple benchmarks that test an isolated feature of the software under artificial workload, while the other considers the group of complex benchmarks that test a set of software features under a real-world workload. The distinction is important because simple

Agile Engineering of B2B Automation Systems

by Kuldar Taveter

The Plug-and-Trade (PaT) B2B Electronic Commerce project was carried out at VTT Information Technology between March 2003 and April 2004. The project aimed to develop rapid and flexible automation of business-to-business (B2B) processes. In the automation system created in the project, a graphically described model of a business process type is transformed into XML-based representations, which are interpreted by software agents representing the parties in the process.

Because of the development of the Internet and the increasing need to improve the efficiency of inter-organisational business processes, business process automation is gaining momentum. Business processes can be automated based on existing standards for e-commerce, such as RosettaNet. These define a business process type in terms of the types and structure of messages exchanged, as well as the order of their exchange. Since new business process types emerge and old ones frequently change, specifications of B2B automation systems are in constant flux, which complicates their development. The PaT project took a step towards the agile design and implementation of B2B automation systems, in which software agents interpret XML-based descriptions of business process types generated by the graphical editor.

The overall architecture of the solution worked out in the PaT project is represented in Figure 1. In the architecture shown in Figure 1, software agents, shown as circles, represent companies. Agents communicate with each other using messages in an agent communica-

tion language (ACL) based on speech acts. Each agent has a virtual knowledge base (VKB), which represents the internal information systems of the company. In Figure 1, Company B and Bank X have 'outsourced' their agents to the Operator to be held in the Operator's agent pool. The 'outsourced' agents communicate with each other and with the agents outside through the Operator's own agent.

In the prototype application (see Figure 2), inter-organisational business process types, the interfaces between them, and the internal information systems of the companies involved, are described by means of the graphical Agent-Object-Relationship Modelling Language (AORML) which has been extended by the author. The modelling is done using the Integrated Business Process Editor. The latter has been developed within the project as an extension to the CONE Ontology Editor of VTT Information Technology. A business process type is modelled from the perspectives of the parties involved. For example, the business process types 'quoting' and 'ordering' are modelled from the

perspectives of Seller and Buyer. Figure 3 shows a snapshot of the model of the RosettaNet-based business process type 'quoting'. It has been modelled from the perspective of Seller. The model specifies that after receiving a request for a quote message, the corresponding business activity of the type 'process quote request' commences with the request for a quote as a parameter. This activity includes a loop that checks in the company's product database for the availability of each product item requested, and accordingly changes the status of the corresponding quote line item to either isBid or isNoBid.

The Integrated Business Process Editor enables graphical descriptions of business process types expressed in the extended AORML to be transformed into their representations in XML. The representations are based on the XML-schema that was created in the project. Even though the XML-based representations of a business process type are generated automatically, some manual tailoring of process-specific interfaces must be performed as a rule. This occurs between the Business Process Interpreter
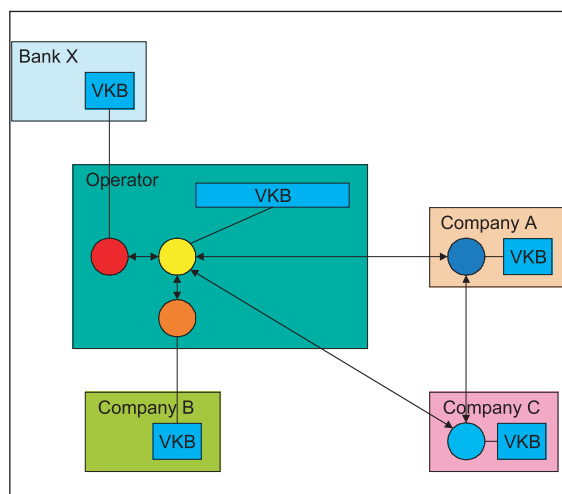

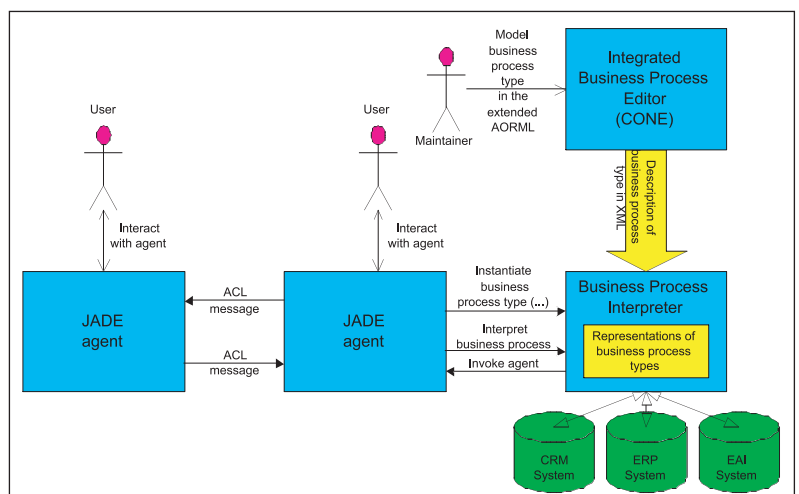**Figure 1: The architecture of the PaT solution.**

Figure 2: The business process automation system.
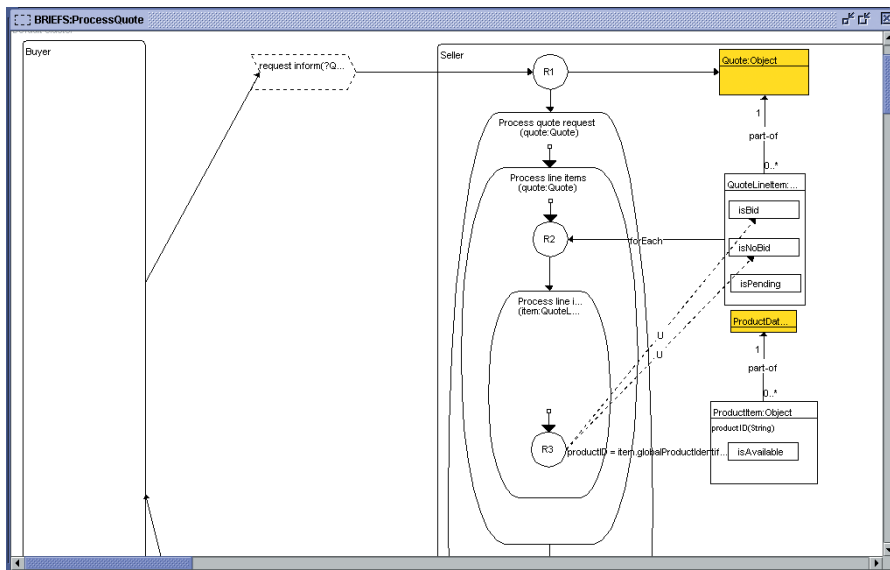
Figure 3: The business process type of quoting.

and internal information systems of the company. Internal information systems can include the Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), and Enterprise Application Integration (EAI) systems shown in Figure 2. After the generation and tailoring have been accomplished, a business process type is ready to be interpreted by the Business Process Interpreter, which works in cooperation with the software agent representing the corresponding party. The latter has been

implemented using the JADE (Java Agent DEvelopment Framework) agent platform. As Figure 2 illustrates, an agent representing a party firstly invokes the Business Process Interpreter to read the description of the business process type, as requested by the agent's human user, and to create its internal representation of the business process type. Thereafter, when the agent receives a message or 'perceives' an input by a human user through the GUI, the agent invokes the Business Process Interpreter

to act according to the process type description. When the Business Process Interpreter acts, it, in turn, invokes the JADE agent and displays messages through the agent's GUI.

There are plans to propose that the XML schema, which was developed in the project for describing business process types, would be adopted as part of the Rule Markup Initiative (RuleML) standard draft. The prototype system described in this paper is just the first step towards a more flexible approach/system with a looser integration between models of business process types and actual business process instances carried out between agents. This would enable a software agent, possibly assisted by a human, to select from numerous alternatives at each step of a business process the most appropriate actions to be performed.

Links:
http://www.vtt.fi/tte/agents/
http://aor.rezearch.info/
http://www.ruleml.org/

Please contact:
Kuldar Taveter, VTT
E-mail: kuldar.taveter@vtt.fi

Automated User Interface Software Engineering with a Pattern Reflecting Programming Language

by Anthony Savidis and Constantine Stephanidis

I-GET is a programming language that encompasses language constructs that automate the implementation of commonly recurring programming patterns in user interface development. It has been developed in the context of the TIDE-TP-1001 ACCESS Project.

Historically, a typical pathway towards the genesis and evolution of programming languages is pattern reflection. It is often the case that once complex recurring software patterns appear, reflecting higher levels of maturity in the programming domain, programming languages and libraries evolve to accommodate those patterns as built-in constructs (see

Figure 1). For instance, object-oriented programming (OOP) languages appeared as a means to provide built-in support for the effective crafting of structured, re-usable and modular software components. Effectively, while OOP languages are imperative in nature, they provide explicit software organisational elements such as classes, access

qualification, inheritance and polymorphism, which require complex programming patterns to be accommodated in non-OOP imperative languages.

In this context, domain-specific languages such as the I-GET User Interface programming language are primarily designed to be domain-opti-

mised, and therefore better suited to addressing domain-related software development problems in comparison to general programming languages. This does not imply that a domain-specific language is computationally more powerful than generic languages. Rather, the domain-specific language automates the resolution of demanding programming patterns leading to source code that is more compact, cleaner, more robust, and more easily re-usable. In Figure 2, the list of the most representative program-ming patterns for user interface imple-mentation is supplied.

Reflecting most of the previously identi-fied user-interface programming patterns, the I-GET language supports built-in constructs for interaction object classes, input events and toolkit func-tions. This allows strong type checking regarding the deployment of those elements for more type-safe source code, when compared to typical libraries of interaction-object classes for general-purpose languages, which are semanti-cally indistinguishable for the compiler from other object classes.
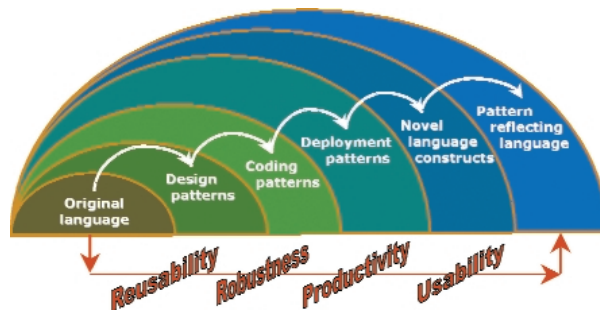

Figure 1: The steps engaged in generating a programming language through pattern reflection, generation and maturity, aiming to accommodate key development requirements.

Figure 3 encompasses an excerpt from the definition of the MFC API. As it is shown, the API elements are organised into domain-specific classes, such as lexical objects, input events and output events, while the deployment of those elements is strongly checked by the compiler. For instance, in Figure 4, a callback code is supplied for the 'quit' object instance of the 'Button' class, to be automatically called when the 'Pressed' method is triggered. In this case, the compiler checks that a method named 'Pressed' is issued in the API specification of the 'Button' object class. Additionally, in Figure 4, the definition

of an agent dialogue component class named 'HelloWorld' is shown, encompassing various interaction object instances. This class is instantiated automati-cally once the 'create if' precon-dition becomes true. Agents are domain-specific classes that constitute the interface-compo-nent programming model in the I-GET language. They are instantiated during interaction either with a declarative style, relying upon creation precondi-tions, or with an imperative style, via explicit creation calls. Agents may encompass various definitions such as declarations of inter-action object instances, implementation of methods, user-defined data types, local functions, variable declarations, and definitions of embedded agent classes.

Figure 5 depicts constraints and moni-tors programming in I-GET. Monitors are code fragments associated with one or more monitored program variables, which are automatically called by the run-time system immediately after any of those monitored variables is modified. Constraints are unidirectional equality relationships that associate constrained

---

1.	**Interaction objects** (e.g. windows, menus buttons, etc.)
∞	Toolkits (i.e. software libraries implementing collections of interaction objects),
∞	Parental relationships,
∞	Object notifications (e.g. pressed, selected), also called *methods*,
∞	Object attributes (e.g. x, y, width, foreground, etc.).
2.	**Composite interface components** (e.g. a dialogue box, a toolbar, etc.)
∞	Consisting of interaction objects,
∞	Activated based on conditions,
∞	Hierarchically organized,
∞	Having mutual control relationships.
3.	**Toolkit events** (e.g. mouse move, key press, window closed, etc.)
∞	Typed event content,
∞	Programmable event handlers (i.e. reaction logic for occurring events).
4.	**Toolkit requests** (e.g. draw line, minimize window, play audio, etc.)
∞	Typical remote procedure calls (RPC), with well-defined formal arguments and returned values.
5.	**Multiple information views** (i.e. alternative views of data sources)
∞	Content monitoring,
∞	Update notifications,
∞	Refreshing display structures.
6.	**Interface dependencies and constraints**
∞	Data elements (e.g. a variable constrained by another),
∞	Interface components (e.g. an object geometrically constrained from another),
∞	Dialogue control variables (e.g. the "save" option is disabled until a documented "is opened").

Figure 2: The list of the frequently occurring user-interface programming patterns, which constituted the design target for the I-GET language

---

Interaction objects, input events, output events

```
lexical Button (MFC) [           inputevent KeyPress(MFC) [
    public:                          string key;
    method Pressed;                  word repeatCount;
    word x, y;                   ]
    string label;
    constructor [...]            outputevent DrawLine (MFC) [
    destructor [...]                 out:
]                                    word gid, x1,y1,x2,y2;
                                     in:
                                 ]
```

Figure 3: Excerpt from the API specification of MFC toolkit elements.

Agent classes and preconditions

```
bool helloWorld = true;

agent HelloWorld
create if (helloWorld == true) destroy if (helloWorld == false)
[
    lexical (MFC) FrameWindow win;
    lexical (MFC) StaticControl msg: perent = {win};
    lexical (MFC) Button quti: parent = {win};

    method {quit}.Pressed [ helloWorld = false; ]

    constructor  [ /* Set appearance attributes of local
objects. */ ]
    destructor [ ... ]
]
```

Figure 4: An example of agent classes hosting instances of interaction objects, supporting precon-dition-based declarative instan-tiation.

Monitors and constraints

```
extern string command;
command : [ Dispatch(command); ] // A data-variable monitor.

extern real a, b;
real x, y;
y := a * x + b; // A uni-directional constraint.
```

Figure 5: Defining monitors and constraints for variables.

---

variables to a constraining expression. Syntactically, constraints are similar to conventional assignments. Semantically, those assignments must be automatically performed during run-time every time a variable directly engaged in the constraining expression is modified. The deployment of monitors and constraints allows multiple data-views, complex interface inter-dependencies and interrelated conditions to be coped with.

Domain-specific languages such as I-GET are domain-optimised, and in comparison with general programming languages are therefore better suited for addressing domain-related software development. We believe that such languages, when they reflect an in-depth insight and automate the implementation of critical domain-specific programming patterns, can accelerate development and lead to higher quality software.

Towards a Library of Problem-Solving Methods on the Internet

by Alvaro Arenas and Brian Matthews

Modern knowledge-engineering frameworks include problem-solving methods. They are reusable software components that incorporate domain knowledge to solve users' tasks. Researchers at CCLRC are developing a library of problem-solving methods implemented as web services available on the Internet.

Problem-Solving Methods (PSMs) are software components that can be assembled with domain knowledge bases to create application systems. The knowledge-engineering community has identified and developed PSMs for specific high-level tasks such as diagnosis, assessment, planning, etc, and develop systems using them as components which are combined and instantiated into complete systems.

Researchers at CCLRC are developing a PSMs library, where each method is expressed as a web service described in the OWL-S language. The library is part of a wider research objective that is applying knowledge-based methodologies for developing intelligent systems for the Web; indeed this research objective was identified in the well-known EU-NSF report on Research Challenges and Perspectives of the Semantic Web in 2002 (http://www.ercim.org/EU-NSF/semweb.html).

In particular, we are utilising the library with the CommonKADS methodology. By contrast to traditional web services discovery, tasks can be determined when modelling a system based on the involved knowledge as well as associated to particular PSMs.

The aim of method is to produce an application which can be deployed on the Web using the emerging web technologies of the Semantic Web and Web Services. Thus the knowledge model in CommonKADS is transformed into an OWL Ontology. The PSM which corresponds to a design pattern is transformed into a class of Web Service. This Web service is represented via a Web service description in the proposed Web Service Description language OWL-S, which uses Semantic Web ontologies to describe the functionality of a web service. When a PSM is invoked to provide a particular instance of the PSM for a particular implementation, a particular instance of the OWL-S class is derived.

Future developments would be using existing OWL-S features to derive a WSDL description of the service. Also, a mechanism of deriving combinations of web-services is required to deliver complete systems deployed across the Web. Further, there is a clear need to be able to express logical rules in the PSMs which is beyond the scope of the current version of OWL-S.

Thus we can use good software and knowledge engineering principles to develop web-based applications. One further advantage of this method is that existing verification methods can be used to develop web applications, leading to more reliable web applications. Consequently, we propose to explore methods such as the B model-based formal method to similarly develop web-based applications. In this case, we use formal verification tools to ensure the correctness of the web application.

The project is developed within the framework of the Semantic Web Advance Development for Europe initiative (SWAD-E).

Cork Constraint Computation Centre applies Artificial Intelligence to Constraint Programming

by Eugene C. Freuder

The Cork Constraint Computation Centre at University College Cork is applying artificial intelligence methods to shift the burden of constraint programming from user to computer.

Difficult problems can offer too many choices, many of which are incompatible, few of which are optimal. Constraint programming makes it easier for computers to help us make these choices. For example, the problem of scheduling employees to shifts is a constraint satisfaction problem. Constraints may range from governmental regulations on working hours to specialized skills of individual workers. Constraints can also take into account costs, preferences, priorities and uncertainties. For example, a worker may prefer a certain shift, an employer may seek to provide sufficient coverage at minimal cost.

Constraints arise in design and configuration, planning and scheduling, diagnosis and testing, and in many other contexts. Constraint programming can solve problems in telecommunications, internet commerce, bioinformatics, transportation, network management, manufacturing, supply chain management, finance, and many other fields. Constraint programming already has wide commercial application, but much remains to be done to fully explore and exploit the technology.

The major scientific challenge and commercial bottleneck at present is the need for further automation. At present difficult constraint programming problems require Ph.D. level expertise. The Cork Constraint Computation Centre (4C) seeks to apply advances in artificial intelligence and other disciplines to make constraint programming easier to use and more useful. More specifically there is a need to assist users in:
- *Acquiring:* Acquiring the constraints from users or databases
- *Modelling:* Representing the constraints in a manner that supports efficient computation
- *Solving:* Determining the computational methods best suited for individual applications

- *Interacting:* Making choices, and understanding options when problems are overconstrained.

The provision of automated assistance in each of these areas is in its infancy. Europe is currently a world leader in constraint programming, academically and commercially, and 4C is one of the leading academic research laboratories in constraint programming. However, to maintain and build on this leadership we must make progress on automation and ease of use.

Artificial intelligence methods are well suited to support such progress. Among the projects 4C is pursuing, with support from Science Foundation Ireland and Enterprise Ireland:

ACE: The Automated Constraint Engine (ACE) harnesses the cognitively-oriented FORR architecture to learn to solve CSPs, and also provides a test-bed for empirical investigation on combinations of heuristics. ACE can select heuristics tailored to specific problem classes. ACE can learn new, efficient heuristics that can be readily used in other programming environments. ACE employs a variety of learning and reasoning methods to structure a decision making algorithm for a given class. ACE is coordinated at 4C by Rick Wallace in collaboration with Susan Epstein's FORR Study Group at The City University of New York.

ACOMME: Constraint programming is a proven, key technology in the optimisation of the activities of manufacturing enterprises. A large bottleneck, however, is the need for highly skilled, experienced consultants to build an optimisation solution for each enterprise. Customisation requires expertise in constraint programming as well as time, effort, and experimentation. This project directly addresses the bottleneck by automating the process of

converting an enterprise model into a constraint programming model. This will allow reductions in time, effort, and expense as well as a significant increase in the uptake of constraint programming by small and medium sized manufacturing enterprises. ACOMME is coordinated at 4C by Chris Beck in a collaboration with Con Sheahan's group at the University of Limerick.

CAUSE: The Constraint Acquisition and Useful Explanation (CAUSE) project studies the critical user interaction that facilitates acquisition of the problem from the user, and assists the user in understanding the solution process. The latter is particularly important when a problem is initially overconstrained, and the user needs guidance on how to relax the problem to reach an acceptable conclusion. Machine learning techniques, eg version spaces, are being applied to constraint acquisition. CAUSE is coordinated at 4C by Barry O'Sullivan in collaboration with a group at LIRMM-CNRS in France led by Christian Bessiere.

O'CASEY: The objective of the O'CASEY project is to facilitate broader access to constraint programming by reducing the knowledge needed to write effective constraint programs. The O'CASEY project uses Case-Based Reasoning to encapsulate constraint programming knowledge in order to assist non-experts to write more effective constraint programs. O'CASEY is coordinated at 4C by Derek Bridge and Cormac Gebruers in collaboration with a group at the University of Bologna led by Michela Milano.

Image Annotation with Presence-Vector Classifiers

by Bertrand Le Saux and Giuseppe Amato

The interpretation of natural scenes, generally so obvious and effortless for humans, still remains a challenge in computer vision. Efficient image recognition software would lead to technological advances in many areas, from effective security to intelligent information retrieval. We propose a system that can classify new images on the basis of information learnt from a training set of previously annotated images.

An emerging research direction in the digital library field is the definition of services for discovering and manipulating knowledge. These include services to search for multimedia documents whose content is described by meta-data. The meta-data can be generated either manually (which is time-consuming but accurate) or automatically (with possibly lower costs but more questionable precision). Procedures for automatic meta-data generation can exploit techniques for image processing and visual feature extraction. Images could be automatically annotated through a visual-feature learning process. To do this, classifiers that can recognise semantic patterns associated with given categories of images are needed.

At ISTI-CNR, we have designed a system that can automatically label images by analysing their visual content. Principally, this involved the definition of an image representation appropriate for scene description and the design of a classification scheme to separate data in the image descriptor space.

Since the human description of visual content is often specific to an object or a given part of the image, image regions can be used to provide more semantic information than the usual global image features. We have developed an image description scheme that can identify which region types are present and which are not. First, the images from a training data-set are segmented into regions. These regions are then grouped into clusters according to their visual similarity, and the discrete range of possible region types that occur in the data-set is thus defined. For each image, a presence vector (indicating which region types are present) is built toprovide information on composition of the image. It is probable that a 'countryside' image would correspond to green foliage and dark ground regions.

The classifiers are built according to a two-step process. First, feature selection is used to estimate which region types are important to discriminate a given scene from others. This is done by mutual information maximisation. A simple implementation of a support-vector machine is then used to solve the margin-maximisation problem on the reduced presence vectors and consequently compute a decision rule to classify the images.

This approach has two advantages. First, it prevents over-fitting, by eliminating the less informative region types that can co-occur randomly with the concept being learnt and thus introduce noise into the classification. The second advantage is that the classifiers can learn the pattern of the scene composition more accurately when the useless image regions are not taken into account. This makes it possible to improve the performance of the predictor. The recognition rates reach more than 90% correctness, depending on the keyword being predicted.
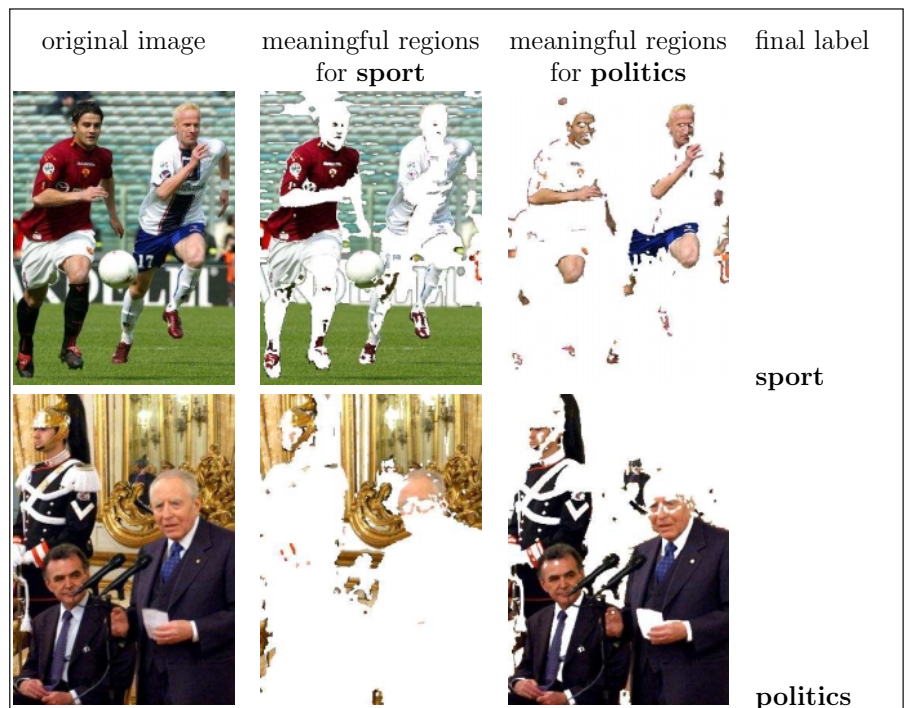
Meaningful regions used to recognize and discriminate between news topics. The original image is on the left, the following pictures show which image regions are considered as positive clues to classify the image according to the keywords 'sport' (the playground and the bright sport jerseys) and 'politics' (skin and dark-suit regions).

We have tested the efficiency of our techniques on various image data-sets: generic ones for the recognition of natural scenes, and the databases of a press agency for topic-based labelling. In this second application, some news topics have been learnt and recognised, such as 'politics' or 'sport' (see Figure).

The tools we have designed for automated image classification have been integrated into the MILOS multimedia content management system developed at ISTI-CNR, where they enhance its meta-data management and document retrieval capabilities. This research was supported by the Enhanced Content Delivery (ECD) project, funded by the Italian Government, and the Delos Network of Excellence, funded by the European Commission under the VI framework program. Future work will focus on improving the accuracy of the classifiers through a more precise description of the regions and the use of information about their spatial arrangement.

Please contact:
Bertrand Le Saux
ERCIM fellow at ISTI-CNR, Italy
Tel:+39 050 315 3139
E-mail: bertrand.lesaux@isti.cnr.it

A Project-Oriented Decision Support System for Production Planning in Make-to-Order Manufacturing

by Tamás Kis, Gábor Erdős, András Márkus, József Váncza

It has become quite common for make-to-order manufacturing industries to use the project-oriented approach for planning production runs on various time scales. In this approach a project consists of tasks, each of which represents a manufacturing process. The connections between the tasks are captured by precedence constraints. Due to the physical characteristics of these processes, the 'intensity' of performing the tasks varies over time. At SZTAKI, new planning software has been developed which is able to adequately model the varying intensity of tasks and the flow of information and material between them.

In project-oriented production management each customer order becomes a project that must be carried out from the ordering of material, through product design, to production and delivery to the customer. Standard planning software packages model the tasks by rectangles, assuming uninterrupted execution and a constant rate of resource usage. Moreover, the precedence constraints between the tasks express only temporal relationships, while the information and material flow between pairs of overlapping tasks are neglected. This makes it very difficult to realise the generated production plan.

In order to create realistic production plans, a new decision support system has been developed in one of the thematic clusters of the three-year 'Digital Factories' R&D project supported by the National Research and Development Project in Hungary. The project started in July 2001 and finished in March 2004. The work was carried out by the Engineering and Management Intelligence Laboratory of SZTAKI in collaboration with the Miskolc University of Technology and the Machinery Plant of General Electric Consumer & Industrial in Hungary.

Our software supports planning with variable-intensity tasks, where the flow of information and material is captured by 'feeding-precedence' constraints between the tasks (defined below). Its graphical user interface helps human planners develop a number of plan variants, organised in a tree hierarchy. The variants may differ in the time windows of the projects, the set of projects selected for planning and the resource capacity (Figure 1). The software obtains all project and resource data from the information system of the plant through file exchange. It was an important requirement to keep the response time short, thereby allowing the users to work interactively with the system.

Modelling with variable-intensity tasks and feeding-precedence constraints

A variable-intensity task is one in which the amount of work spent on the task may vary during the execution. In actual production, tasks usually start at a low intensity during preparations which then gradually increases to a maximum. The resources needed to complete a task are consumed proportionally to the varying
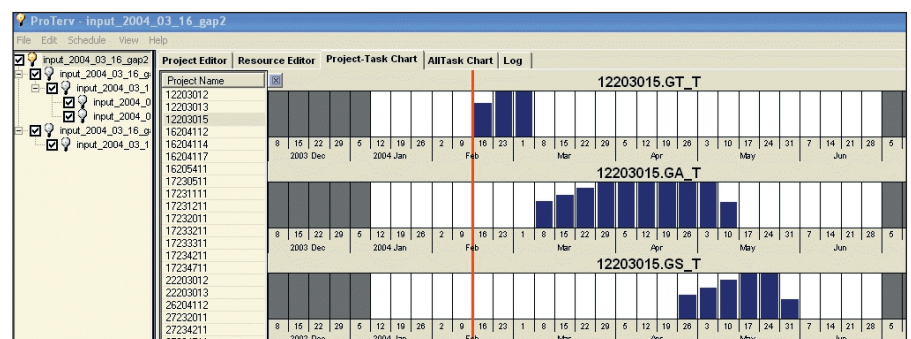

Figure 1: Screen of the planning software.

intensity. The output of the task may be used in downstream production prior to the completion of the whole task. To model this situation, we introduced the notion of a feeding-precedence constraint between a pair of tasks A and B, with parameter p specifying that p percent of task A must be completed before task B may be started and that task B cannot advance faster than task A (see Figure 2).

Techniques employed

The planning problem has been modelled and solved by mathematical programming techniques using cutting planes. The planning module (PM) is based on a commercial integer-programming solver. The PM can optimise various objective functions that occur in practice, such as minimising the extra capacity needed to complete all projects, or the weighted tardiness. The short response time of the system on real-life problems makes it suitable for interactive decision support.

The PM has been embedded into a software package which assists the work of planners by offering features such as displaying the structure (tasks and precedence) and the status (work done on each task) of the projects. The package also provides editing functions, including
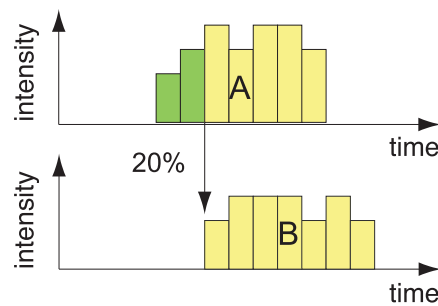

Figure 2: Feeding precedence constraint from task A to task B with p = 20%.

modification of project time windows or resource capacities, and supports the systematic elaboration of plan variants organised in a tree hierarchy. This allows decision makers to choose the final plan from a set of alternatives.

Application areas

The system is suitable for solving production-planning problems in make-to-order manufacturing where the tasks of a project represent aggregated activities. Another domain of application exists when only the production must be scheduled, but on different time horizons. A long-term plan (thirty weeks) can be generated by the planning module, and based on this, the weekly schedules can be obtained by an appro-

priate short-term scheduler. We have developed a pilot system for each of these application domains.

There are a number of key benefits of our project-oriented production-planning system. Planning with variable intensity tasks and feeding precedence constraints results in more accurate production plans. This, along with the combination of project execution and resource capacity planning leads to a better due-date observance and to a better use of resources. As a result, more customer orders can be accepted and production costs are reduced.

The techniques employed in the planning module can be applied in various settings. Current work is focusing on improving the robustness of the planning module and making the user interface capable of communicating with standard, widely used ERP systems.

Links:
http://wwwnew.sztaki.hu/proj_publ/projects/digital_factory/index.en.html

Please contact:
Gábor Erd s, Tamás Kis, András Márkus, József Váncza, SZTAKI, Hungary
Tel: +36 1 279 6156, +36 1 279 6299
E-mail: gabor.erdos@sztaki.hu, tamas.kis@sztaki.hu, markus@sztaki.hu, vancza@sztaki.hu

Next-Generation Metropolitan Area Networks

by Kari Seppänen, Sami Lallukka and Riikka Lemminkäinen

Because of the emergence of new services, the requirements for metropolitan area networks (MANs) have increased and diversified. Consequently, they are harder to meet - at least with the existing SONET/SDH networks - and new solutions are needed. VTT Information Technology has been researching next-generation SONET/SDH (Synchronous Optical Network/Synchronous Digital Hierarchy) networks, which could meet the diverse requirements of MAN while utilising the existing SONET/SDH network infrastructure.

Ideally, network operators would only have to manage a single type of network. One multi-purpose MAN (Figure 1) would connect all the various access networks and provide everything from real-time services to traditional data-transfer services. The network would also provide quality of service (QoS) and handle any kind of traffic, from constant bit-rate traffic to packet- or cell-based traffic. While such a multi-service

network would minimise overall operating costs, the existing SONET/SDH infrastructure is unfortunately unable to meet these requirements. The worst-case scenario is that the operators must maintain their legacy networks for legacy services while the new services need overlapping network infrastructure.

In recent years, SONET/SDH-based transport networks have come to be

considered as too inflexible, inefficient, and overly complex for the purposes of data communication. As the importance of data communication has increased, a search has begun for a replacement for SONET/SDH. However, developing such a replacement is neither easy nor straightforward. On the contrary, many useful functions provided by the SONET/SDH appear to be too complicated to reinvent in other way.

Furthermore, long-established telecommunications companies have already invested billions of euros in their SDH networks, and would therefore prefer to utilise the existing infrastructure.

Fortunately, a new solution based on proven SDH/SONET technology is evolving, and promises to turn SONET/SDH into an efficient multi-service transport network that is easy to manage and provision. 'Data over SONET/SDH' (DoS) is based on three new features in SONET/SDH networks: VC (Virtual Concatenation), LCAS (Link Capacity Adjustment Scheme), and GFP (Generic Framing Procedure). VC and LCAS together enable fine-grained capacity allocation and management. Efficient framing and link-layer statistical multiplexing is achieved using GFP, which provides a unified method to multiplex packets from diverse sources. Furthermore, DoS-capable equipment can be mixed with older, inflexible SDH equipment to provide a reasonable evolutionary upgrade path for 'traditional' network operators.

The ability to provide optical connections rapidly and dynamically while making optimal use of network resources is also important. This can be achieved by adding intelligence to a traditional optical transport network and updating it to an Automatic Switched Transport Network (ASTN).

Optical Access Networking Project

The concept of the next-generation SDH network was studied in the Optical Access Networking (OAN) project, a three-year project which commenced early in 2001. Project collaborators included VTT Information Technology, VTT Microelectronics and Networking laboratory of HUT (Helsinki University of Technology). The project was funded by the National Technology Agency of Finland (TEKES) and supported by two industrial parties - the Nokia Research Center and Elisa.

Through the OAN project, a network evaluation platform was developed. The objective was to design and implement the electrical parts of the feeder network, including all the networking activities starting from the link layer.
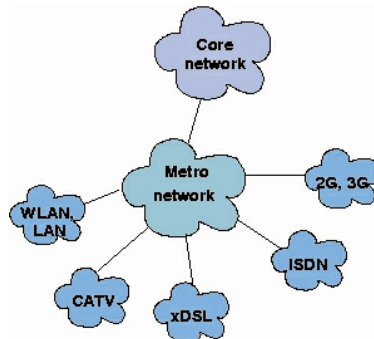

Figure 1: Metropolitan area network connects various access networks to the core network.

OAN Platform

The OAN network acts as a feeder network, connecting multiple subscribers to a core network. The test network prototype (Figures 2 and 3) exploits two counter-rotating 2.5 Gbit/s SDH rings in the transport network side, and two Gigabit Ethernet links together with one 2.5 Gbit/s SDH link in the access network side. The network functionality is implemented in Field Programmable Gate Arrays

Figure 2: Access node architecture.

Figure 3: An OAN prototype node is fitted into a standard CompactPCI-frame.

(FPGAs), which enable flexible network design and system testing. The physical equipment of the OAN node is fitted into a standard CompactPCI–frame, making the node easy to move and install. The CompactPCI-frame includes a Central Processing Unit (CPU) card, which is used by the management software.

The access network and transport network interfaces are implemented into separate interface and router cards respectively. Data between the interface and router cards is transported using a dedicated bus system and specific transport control. Flexible design of the system allows the interconnection of multiple router and interface cards. By developing new interface card versions, multiple technologies can be supported in the access network side.

To obtain both efficient optical protection and a cost-effective structure, the OAN network is physically a ring and logically a star. The OAN prototype network consists only of three to four
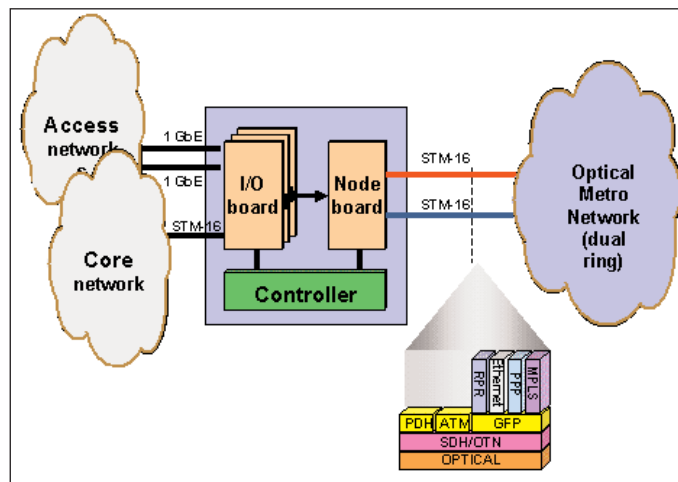
nodes, providing low network complexity. One node acts as a hub, connecting other nodes to the core network and to each other. The connections are established using WDM (Wavelength Division Multiplexing) technology.

Furthermore, a network-monitoring system for the OAN network was designed. In the design the usability of Simple Network Management Protocol (SNMP), standardised Management Information Bases (MIBs), and ready-to-use management software was considered.

Next-Generation SONET/SDH in Use
The next-generation SONET/SDH enables new types of services with more efficient network usage to be easily implemented by utilising existing infrastructure.

Corporations require diverse services (eg voice, VPN, data storage, and Internet connection services) from operators. Traditionally the different services are provided through technology-specific transport pipes. However, the next-generation SDH enables the simultaneous transport of heterogeneous services over one wavelength, thereby saving network-building and maintenance costs.

Usually a virtual private connection (VPN) is used to bridge operators' access points. In some applications however, it is desirable to transport the native network signal without extracting packets or frames. Normally the datacom protocols rely on 8B/10B coding, which causes a 25 percent increase in bandwidth. Using the next-generation SDH, which maps 8B/10B-coded data into 64B/65B-coded sequences, the required bandwidth is substantially decreased.

The ability to dynamically reallocate bandwidth allows Bandwidth on Demand (BoD) services. This will revolutionise the network service industry, since the users are able to specify their bandwidth requirements according to the time of day.

Link:
http://www.vtt.fi/tte/tte21/oan/

Please contact:
Kari Seppänen
VTT Information Technology, Finland
Tel: +358 9 456 5610
E-mail: Kari.Seppanen@vtt.fi

KOPI – An Online Plagiarism Search and Information Portal

by Máté Pataki

Due to the growing number of digitally available documents, the problem of plagiarism is becoming increasingly serious. The aim of the KOPI project, carried out at the Department of Distributed Systems (DSD) of SZTAKI, was to develop an online plagiarism search portal, which would help digital libraries to protect their documents, and teachers to identify copied texts or publications.

The work was performed as part of a joint project between SZTAKI and Monash University in Melbourne, with a series of investigations being made on document chunking and overlap detection, two techniques on which the detection of plagiarism is based. Continuing this work, development of the portal KOPI commenced in 2003, funded by the Hungarian Government. The portal will become available to users by the end of June 2004.

There are two different approaches to fighting plagiarism. The first is the protection of the document by preventing it from being copied or misused, and the second is the recognition of plagiarism. Protection is an important issue but it can cause difficulties for legal users. Moreover, all kinds of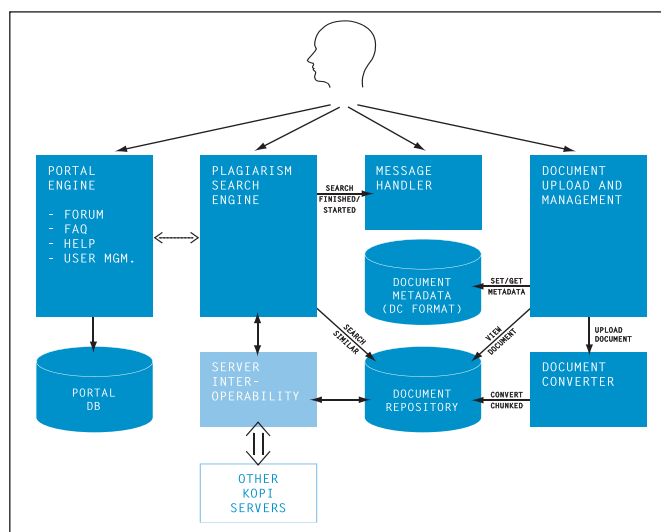 protection will be cracked in time. According to our view, the most effective technique in fighting plagiarism is the fast detection of document overlapping: in other words, there is no sense in copying a digital document if the copy can be detected within minutes. This method is used to protect documents that are part of the KOPI system from illegal use.

As a portal site, KOPI includes common services such as a forum, context-sensitive help, FAQ, and static documents including information on plagiarism and university laws. In addition, two system-specific services are offered to users of the portal: a document upload and management service, and the plagiarism search engine. The first can be used to upload documents (html, rtf, doc, pdf, txt) or a batch of documents (zip), and to attach meta-information to them. The meta-data are stored in Dublin Core meta-data format to make possible future interoperability with other systems.

The uploaded documents can then be compared with each other, with previously uploaded documents, with all documents uploaded by users, or with collections of documents gathered from the Web or documents in digital libraries. The comparison is made offline, reducing waiting time and costs for the user. When the job is finished, the message handler unit sends the results to the user via e-mail.

The heart of the similarity search engine is the chunking method, which is used to chunk the given text into smaller pieces. This task and the conversion of the document to plain text are performed by the document converter subsystem. When comparing documents, only these

The structure of the KOPI system.

chunks or their so-called compressed fingerprints are examined to determine how many common parts the documents have. The KOPI system uses a combination of word chunking and overlapping word chunking to chunk the documents. This new algorithm provides a fast and accurate search, while keeping the size of the database small. (For more information on the chunking methods see the links below.)

In order to perform an efficient plagiarism search, the KOPI system needs to collect as many documents as possible. Four possible sources exist:
- documents on Internet
- digital library collections
- publications and theses from schools, universities, or conference organisers
- material uploaded by the users of the KOPI system.

Documents from the Internet are collected using a Web crawler. Digital libraries with an open interface to the Internet (eg OAI) can also be easily harvested. In the future, it is likely that university students will be requested to submit their theses in digital form, and so within a couple of years a large set of documents will have been collected.

KOPI is currently a stand-alone portal application. Future developments in the frame of a PhD project will target the creation of a distributed KOPI architecture. In such a system, institutes would use their own local copy of the KOPI engine, but could initiate a plagiarism search involving documents over the whole distributed KOPI system.

Links:
KOPI portal: http://kopi.sztaki.hu

Department of Distributed Systems of SZTAKI: http://dsd.sztaki.hu

Plagiarism Detection and Document Chunking Methods, The Twelfth International World Wide Web Conference: http://www2003.org/cdrom/papers/poster/p186/p186-Pataki.html

Match Detect Reveal Project at Monash University Melbourne: http://www.csse.monash.edu.au/projects/MDR/

The Dublin Core Metadata Initiative: http://dublincore.org/

Please contact:
László Kovács, SZTAKI, Hungary
Tel: +36 1 279 6212
E-mail: laszlo.kovacs@sztaki.hu

Assessing Quality of Service on the Internet

by Peder J. Emstad

Everyone who uses the Internet and mobile phones has an interest in their quality of service. We may notice delays when we browse the Web and transfer files, and we may wonder if our transactions are secure or if we can connect at all. New services, such as voice and video, are constantly being offered based on packet-switching technology, but provide varying quality for the user. A new centre at the Norwegian University of Science and Technology has been established to determine how to assess this quality of service.

Traditionally, telecommunication networks have been established to transport voice sig-nals between geographically disparate users, and for that purpose a worldwide network, albeit of widely varying technical quality, has emerged. However, the days of networks dominated by analogue voice signals are long gone. Today's network of networks, the Internet, has become the general vehicle for the transport of digital data representing all kinds of information and information services. These services are realised through the exchange of digital data between end-users, and between end-users and service providers. They cover most aspects of human activities, both private and professional: general information exchange, e-mail, voice and multimedia services, transfer and storage of medical information, economic transactions, location information – the list is already seemingly endless. In the industrialised world, everyday life as we know it would cease to exist if the Internet were to become permanently inoperable.

In order to address the problem of Quality of Service (QoS), the Centre for Quantifiable Quality of Service in Communication Systems has been established at the Norwegian University of Science and Technology. Research will be based around the continued evolution of packet-switching techniques for mobile networks and the Internet . Society's use of and dependence on such networks increases steadily, even though the QoS in a broad sense is unsatisfactory and at best variable. Services to be

considered include traditional teleservices, multimedia, messaging, Web and information services, and location- and content-aware services.

QoS as perceived by the users is a combination of factors, and relates to user expectation and satisfaction. It addresses quantifiable technical qualities and solutions, but other aspects such as billing and service management are not included. User (or source) and network

The Centre is financed by the Research Council of Norway, the Norwegian University of Science and Technology (NTNU), and UNINETT, the network operator and service provider for all educational and research organisations in Norway. Financing has been secured for five years, and may be extended for another five years. Telenor is also supporting the Centre, both financially and otherwise. The professors at the Centre are employed either in the Department of Telematics or the

heavily on these services, and it is of the utmost importance that they are perceived to be secure, dependable and of good quality. The Centre will assess technical solutions, including mechanisms, methods and specifications, and quantify the QoS using appropriate measurements.

The Centre works within the areas of multimedia signal processing, dependability, traffic and security as applied to multiparty communication. Theoretical studies will be undertaken based on analytical models, simulations and laboratory tests.

Theoretical results can then be verified using the unique laboratories available to the Centre for experimentation.

The idea of the Centre is to look at overall QoS in packet-based communication networks like the Internet. This is equally important for audio and video encoding, as is handling of these flows in the network, where they can suffer delays and losses. This can be due both to high traffic and to breakdown of equipment. Excessive traffic may be due to junk mail or hacker attacks, and for a user there may be a subtle difference between reliability and security. Indeed, QoS-relevant issues may be very entangled and should be studied as a whole.

ERCIM fellows Venkatesh Babu Radhakrishnan (center) and Yuming Jiang (right) hosted by NTNU at the Centre for Quantifiable Quality of Service in Communication Systems with Centre Director Peder J. Emstad.

behaviour and interaction in a broad sense are determining factors.

To assess QoS it is necessary to thoroughly understand underlying network architecture, to make models and case studies, and to perform measurements. Research will therefore also assess methods and techniques. Furthermore, insights from the research will lead to improved technical solutions and strategies for network and service management. This will mean users can be guaranteed services without undesirable side effects.

The Centre was established on 1st January, 2003 and now includes six postdocs, fourteen PhD students, and seven professors, one of whom is a visiting academic. The Centre hosts two ERCIM fellows (see Figure).

Department of Electronics and Telecommunications at NTNU.

The Centre shares labs with the above-mentioned departments. In addition, UNINETT has made available laboratories, testbeds and advanced instrumentation for network measurements. The Centre is located on the premises of the Faculty of Information Technology, Mathematics and Electrical Engineering at NTNU in Trondheim. Trondheim is in the central part of Norway and offers excellent working and living conditions.

Mobile systems and the Internet are taking over services traditionally made available through dedicated networks, such as voice, audio and video. Furthermore, they are constantly offering new services, including banking, e-commerce, tax returns and so on. Society is relying more and more

The Centre is a member of EuroNGI, a Network of Excellence under the EU 6th Framework Programme working with problems related to the design and engineering of the Next Generation Internet. It will be represented at COST 276 - 'Information and Knowledge Management for Integrated Media Communication' and is a member of COST 290 – 'Traffic and QoS Management in Wireless Multimedia Networks'.

Link:
http://www.ntnu.no/Q2S/

Please contact:
Peder J. Emstad, NTNU, Norway
Tel: +47 73 59 4326
E-mail: peder@Q2S.ntnu.no

CWI and INRIA join Forces on Safety-Critical Systems

by Wan Fokkink, Hubert Garavel, Jaco van de Pol

The Embedded Systems Group (SEN2) at CWI and the Systems Validation Group (VASY) at INRIA have launched a collaboration in the area of the verification of safety-critical systems. Named SENVA ('System ENgineering and VAlidation'), this joint research team strengthens a long-standing collaboration.

Both teams have a lot of experience in the formal specification and computer-aided validation of safety-critical systems. Both teams share a common methodology based on the theory of process algebras established by prominent scientists like Hoare and Milner. The teams develop robust software tools, which are used in significant industrial case studies, in areas like railways, embedded systems, and hardware communication platforms.

The Netherlands Organisation for Scientific Research and INRIA have pledged financial support to the SENVA initiative, for the next two to four years.

This funding is used for mutual research visits and an annual SENVA workshop. The first SENVA workshop took place 21-24 June 2004, in Grenoble.

The SENVA team will pursue the efforts undertaken by SEN2 and VASY to interconnect their software tools, the muCRL and CADP verification tool sets respectively. It will also target a next generation of languages and tools for safety-critical systems. In 2004, the SENVA team focuses on the following themes:
- high-level specification languages
- tools for minimizing state spaces
- distributed algorithms for state space analysis
- compositional verification techniques.

In line with the current trend of integrating and strengthening the European Research Area, the SENVA research team will build upon the competences and workforce of both teams, in order to form a major player in formal methods for safety-critical systems.

Please contact:
Wan Fokkink, CWI, The Netherlands
E-mail: Wan.Fokkink@cwi.nl

Hubert Garavel, INRIA, France
E-mail: Hubert.Garavel@inria.fr

Discrete Tomography with an Eye towards Practice

by Joost Batenburg and Herman te Riele

Discrete Tomography, the reconstruction of binary (black-and-white) images from a small number of their projections, has received considerable attention over the past ten years. In a joint project, researchers at Leiden University and CWI perform research on new reconstruction algorithms and their mathematical background, with an eye towards possible practical applications.

The general discipline of tomography has been used over the past 40 years, particularly in medical imaging (CAT-scanners), for making noninvasive images of patients. When a large number of projections (X-rays) is available, accurate reconstructions can be made by a wide spectrum of available methods. Discrete tomography focuses on the case where only few projections are known and the images contain a small number of different colours (eg black-and-white). In this case, conventional techniques all fail. Yet, several practical

applications demand accurate reconstructions under these conditions.

Figure 1 shows a binary image and projections that are measured from two directions. The problem of reconstructing a binary image from only two projections is well understood and fast reconstruction algorithms are known. However, the solution is usually not unique: a huge number of other solutions may exist. When more than two projections are available, the number of solu-

tions becomes much smaller, but they also become more difficult to find.

At the start of the project, two years ago, reconstructing binary images of 50x50 pixels from a small number of projections was a major challenge. Since then, a new algorithm has been developed by the research team consisting of Robert Tijdeman (Leiden University), Herman Te Riele (CWI), and Joost Batenburg (CWI/Leiden University). This algorithm is capable of reconstructing
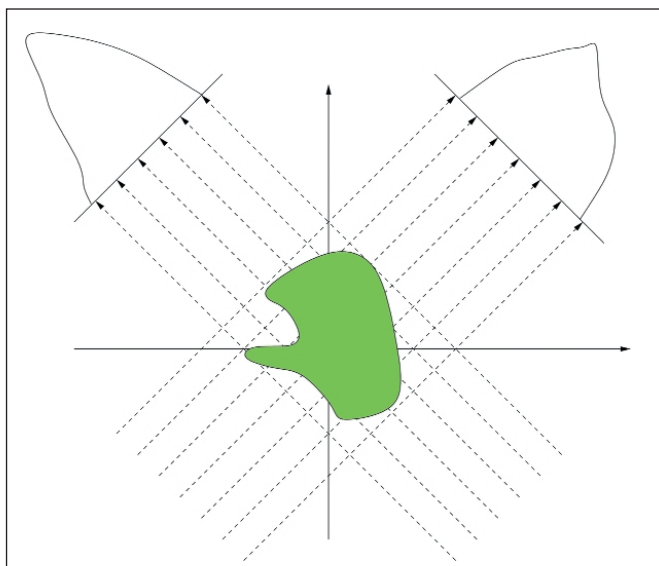
**Figure 1:
A binary image and two of its projections.**

images 100 times as large (500x500) with great accuracy.

It repeatedly computes an approximate reconstruction, each time using only two of the projections. In general, there is a huge number of such two-projection solutions. The algorithm chooses one of these solutions based on a weight function. This weight function is constructed in such a way, that the new image satisfies the two prescribed projections completely, while still resembling the image that resulted from the previous iteration.

Assumptions on the image structure, such as a preference for smooth images, can also be incorporated in the weight function. Tests, using a large set of images from which projections were computed, have shown that when a sufficient number of projections is available, the algorithm is very likely to converge to the original image. Even very tiny

details, of only one pixel in size, are reconstructed with great accuracy.

The research uses techniques from a wide range of mathematical and computer science areas, including combinatorial optimization, number theory, numerical mathematics, and evolutionary computation. In particular, the subroutine that is computationally most expensive involves solving a minimum cost flow problem in a graph. Since this problem has been studied extensively in operations research, a wide range of efficient algorithms is available.

In the late 1980s, interest in discrete tomography flourished when new research results in material sciences called for the reconstrucion of crystal lattices from several projection images, obtained by electron microscopy (see Figure 2). Since then, mathematical research on discrete tomography and research on new electron microscopy

techniques has advanced both fields independently. In the tomography project at CWI/LU, efforts are made to bridge the gap between both fields, by cooperating with leading scientists in the microscopy field. If existing or new algorithms from discrete tomography can be applied successfully to the three-dimensional reconstruction of crystal lattices many open problems in material sciences, on the atomic structure of materials, can be solved.

Future research goals for the CWI/LU team include the development of a stronger theoretical foundation for the newly developed algorithms. Generalisations of the algorithm to other cases of interest, such as the reconstruction of images using more than two gray-levels, or the reconstruction of three-dimensional images are currently under consideration. From a more practical perspective: in addition to electron microscopy, several other potential application areas such as medical imaging and tomography of industrial objects are considered for future research.

Please contact:
Joost Batenburg, CWI, The Netherlands
Tel: +31 20 592 4176
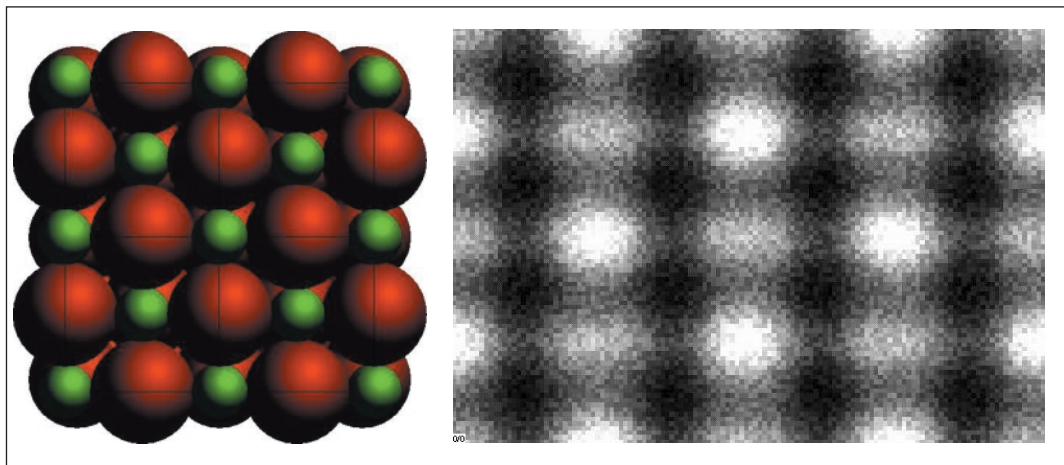E-mail: Joost.Batenburg@cwi.nl

**Figure 2:
A crystal lattice containing two atom types (left) and one of its projections (right).**

12th ERCIM Environmental Modelling Group Workshop: Current Trends and Developments

by Nikolaos A. Kampanis

The ERCIM Working Group on Environmental Modelling had its annual Workshop on 24 and 27 May 2004, at Hersonissos, Crete, Greece. The workshop was held during the ERCIM Meetings - Crete 2004 that took place in the week from 24 to 28 May at the Conference Center of Aldemar Knossos Royal Village Hotel at Hersonissos.

Members of the working group participated in the workshop along with researchers from FORTH-IACM and their cooperation partners, and presented their recent research work. Representatives of the regional authorities were also present and participated in the workshop. The presentations covered many fields of interest and promoted fruitful discussions among the participants.

Environmental research issues that have been addressed included: data mining, air pollution modelling, remote sensing, image analysis, simulation techniques and information systems. A round table gave the opportunity to the participants to further discuss on research topics related to environmental problems, discuss on future plans and propose further means of strengthening the activities of the working group.

Furthermore, workshop participants took part in a series of events organised throughout the week, including three short workshops about the 7th

Workshop participants.

Framework Programme co-organised with the European Commission and a familarization day at FORTH. On the occasion of the 15th Anniversary of ERCIM, a Celebration event was held.

Overall it was an enjoyable meeting with interesting scientific aspects as well as pleasant social activities. As part of the ERCIM Meetings – Crete 2004 the workshop especially benefited from the excellent organization of this whole event.

Links:
Workshop: http://www.iacm.forth.gr/numerical/Workshops/ERCIM.html

Working Group: http://wwwold.first.fhg.de/applications/em.html

Please contact:
Nikolaos Kampanis, IACM-FORTH, Greece
Tel: +30 2810 391780
E-mail: kampanis@iacm.forth.gr

Achim Sydow
Fraunhofer ICT Group, Germany
Tel: +49 30 6392 1813
E-mail: achim.sydow@first.fhg.de

Second OpenECG Workshop

by Catherine Chronaki

OpenECG is funded under the 5th framework to promote interoperability in digital electrocardiography through the consistent implementation of standards. OpenECG calls for open protocols and data formats and is driven by patient mobility, continuity of care, and comprehensive electronic health records (EHR).

The 2nd OpenECG workshop took place in Berlin, 1-3 April 2004, at DIN, the German institute for standardization. Gerd Herfurth, CEO of DIN-NAMED (Standardisation Committee for Medicine) delivered the welcome

address. About 60 experts from Europe, USA, and Japan, attended the workshop. The participants discussed barriers to interoperability and the driving forces towards the adoption of open standards

especially with regards to the integration of digital ECGs into the EHR.

Prizes for the OpenECG programming contest were awarded. E. Cervesato & G. De Odorico received the first prize (6000

Euro) for an SCP-ECG viewer written in C. The 2nd prize (3000 Euro) was awarded to D. Clunie, for an ECG viewer in Java able to display ECG records in the DICOM and SCP-ECG formats. The recipient of the 3rd prize (1000 Euro), A. Schloegl created in Octave a converter/parser for SCP-ECG records. All submissions are available in open source.

Since July 2002, OpenECG has established an advisory board and the www.openecg.net portal to consolidate best practice in interoperability of ECGs. Conformance testing, a helpdesk, converters, specifications, and tutorials for ECG standards attract people to the portal. A members' mailing list provides a forum for discussion. OpenECG has established a community of more than 200 members and vendors see it as a quality label. The first online service for

frequent non-invasive cardiology examination, its utilisation is limited by closed data formats and non-disclosure of quality characteristics that affect the ability to compare ECGs of the same patient over time (a technique called serial comparison).

According to Mr. Rossing, former head of the European Health Telematics program, the eHealth sector in Europe demonstrates tardiness when compared to other sectors. Lack of documented benefits, slow adoption of EHR and limited interoperability are contributing factors. The low cost of ECG recorders has allowed ECG vendors to get away with proprietary protocols and data formats. However, interoperability MANDATES adoption of standards!

Dr. Badillini presented developments in USA, where in 2001 FDA demanded full

Data format and protocol specifications should be freely available to reduce the cost of integrating medical devices. Integrating legacy ECG devices to EHRs is now costly and requires a non-disclosure agreement. Furthermore, there are no widely adopted data formats for examinations like Holter, stress ECG, and vital signs monitoring. The FEF prestandard as well as successful extensions of SCP-ECG for such examinations were presented by A. Varri, F. Chiarugi, and Chr. Zywietz. The participants recognised SCP-ECG as a data format that addresses resting ECG quality and accuracy, and supports compression that makes it suitable for storage in the European health card (~8kb).
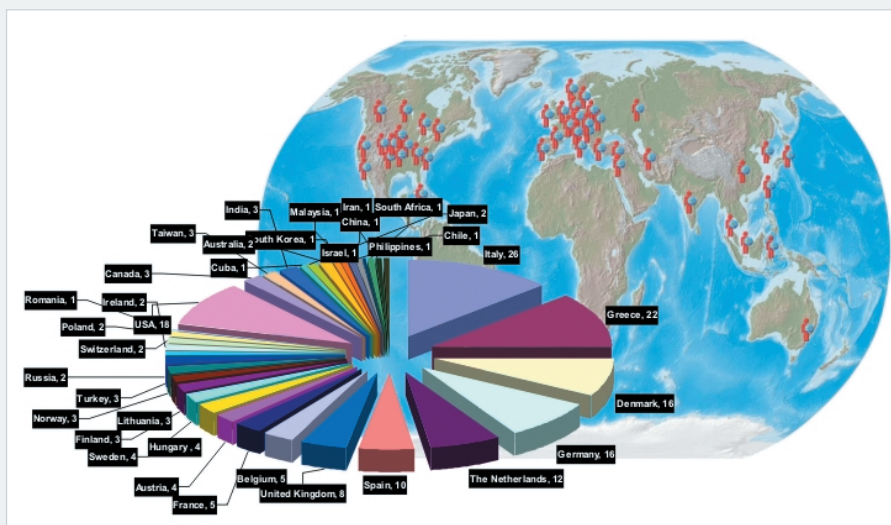
Dr. Lymberis commended that despite advances in technology in the eve of the 'Knowledge society,' most ECGs are reported and archived on paper; a limiting factor for broad access to high quality care. Work on standard ECG layouts that look as good on the screen as on paper and consistent terminology are steps towards the acceptance of digital electrocardiography in Europe.

Dr. Siebold of Siemens presented the IHE initiative and suggested that OpenECG should collaborate with IHE in cardiology to create added value for the community.

OpenECG community in March 2004.

The high market penetration of low-cost wearable and portable devices used in personal health management is expected to play a key role in the evolution of eHealth. This tendency pronounces the need for testing of quality and safety issues, an aspect of which is interoperability. OpenECG's envisioned role is to promote interoperability testing, open standards and harmonization of data formats, to reduce the risks involved in the use eHealth. In the long term, establishing consumer confidence will contribute to an increase in market acceptance of eHealth.

conformance testing of ECG records to SCP-ECG, the European standard for ECG interoperability, was used more than 280 times by 25 members only in the first six months. Four manufacturers and several integrators have received assistance in implementing the SCP-ECG standard by the OpenECG helpdesk.

Dr. Becks from VDE, Germany, identified cardiovascular diseases as the leading cost of death and the most expensive disease in the western world. However, although the ECG is the most

disclosure of waveform data recorded in clinical trials. Specifications, samples, and a viewer are now available for the HL7/FDA Annotated ECG format. Ken Kawamoto and Masaaki Hirai presented a freely available viewer for MFER, a Japanese standard for the storage of waveform data. In Europe, the German IMEX project of VDE is developing MSD, a new micro-system data format (based on SCP-ECG section 1), to address the needs for wearable devices, as reported by Dr. Jettkant.

Link:
http://www.openecg.net

Please contact:
Catherine Chronaki, ICS-FORTH, Greece
Tel: +30 2810 391691
E-mail: chronaki@ics.forth.gr

InfoSam2020 –
Planning for the Future Information Society

by Arne Solvberg and Tore R. Jørgensen

A two-day seminar for exploring relevant features of the future (year 2020) information society initiated the InfoSam activity at the Faculty of Information Technology, Mathematics and Electrical Engineering (IME) at The Norwegian University of Science and Technology (NTNU) at the end of April.

Different working groups presented position papers on a number of sub themes within this wide framework. Papers and input from invited key-note speakers:

- Keith Jeffery, Director of Information Technology and Head of Business and Information Technology Department, CLRC
- Michael L. Brodie, Chief Scientist at Verizon Information Technology
- Johan de Kleer, Manager of the Systems and Practices Laboratory, Palo Alto Research Center (PARC).

No final conclusions were drawn. However, a wide hearing has been started and the process will end late 2004. In general, when the basic technologies get continuously cheaper, faster and smaller, the broad outline seems to be:

- further growth of ICT systems into very huge and complex networks in most application areas; interoperability, security, privacy and reliability are crucial factors
- simulations and visualization will be of increased importance to reproduce reality (like petroleum reservoirs, ultrasound scans of the human body, etc), construct reality from models (like automobiles), understand reality (genomics, etc.) and as tools to control reality (like process control)
- how to manage and exploit the enormous amounts of information stored
- self-configuring systems (connected to networks or stand-alone) from 'smart dust' to complex machines operating with mission statements.

Furthermore, the interaction between ICT and each application area specialists is assumed to be crucial for producing the competitive solutions in an international business. The number of highly qualified ICT experts will limit the speed of development. Without sufficient quality, the various systems will brake down or require more resources to build and operate than they are supposed to release.

IME has an annual educational output of approximately 500 master students and 30 PhDs, a significant part (60-65%) of the capacity in Norway, and thus the decisions by the faculty will be of national importance. The rapid pace of technological change impacts strongly on curricula and research. Technological change is also a key driver for changing the way we live, due to the deep penetration of computers in all realms of society. This has an impact on student recruitment, as well as on the relevance of research themes. The quality of our teaching depends on the quality and relevance of our research. In order to stay in the forefront the faculty has to relate to future needs.

The overall objective of InfoSam2020 is to determine which knowledge profile the students should have in 2020, and the associated teaching and research. The course for the faculty for the next 15-20 years is to be charted. This may seem a long time, but it is not when compared to the 10 years it takes to set up high quality research in a new field.

The first step in the process is to explore the space of future possibilities, to describe possible technological changes and subsequent changes in our society. The next step comprises the charting of the course.

InfoSam2020 is arranged in cooperation with The Norwegian Board of Technology, an independent body for technology assessment and foresight established by the Norwegian Government in 1999. The Board aims to assess impacts and options of technology in all areas of society; to stimulate public debate on technology; and to support the political decision-making process and shaping of technological change. Insights from InfoSam2020 will thus be presented by the Board to relevant political authorities.

To serve as a framework, two major themes have been selected. The first theme is devoted to the use of 'intelligent' artefacts in various realms of human endeavour. The second theme is devoted to enabling technologies, in particular within the fields of mathematics, ICT and electrical engineering.

Link:
http://www.ime.ntnu.no/infosam2020/

Please contact:
Arne Sølvberg, Tore R. Jørgensen
NTNU, Norway
E-mail: Arne.Soelvberg@idi.ntnu.no,
Tore.R.Jorgensen@ime.ntnu.no

Third Workshop on 'System Testing and Validation '

Paris, France, 30 November-2 December 2004

Software and system assurance is a growing concern. A look at some of the domains involved, including at least medical devices, automotive, railway, aeronautical, space and telecommunications provides a feeling on the importance of the topic. For many of these domains, and partly because of the increasing complexity of software, validation is an open and not straightforward at all, both from a technical and from a managerial point of view. Another topic to consider is that validation is, most usually, performed at the end the development lifecycle, what results in obvious calendar problems. One of the consequences is that the relevance of lifecycle processes such as system testing and validation must be broadened. Therefore inputs from research contributions and experience from industry is required in order to enable innovative, and often more rigorous, approaches.

Topics

Different approaches and methodologies to system validation. Issues to be considered are:
- innovative approaches to validate different types of requirements: functional, and non-functional (reliability, safety, maintainability, etc.)
- automation and tool support; validation for systems of formerly-validated components;
- validation in the context of different lifecycle process models: eg conventional and agile
- validation and certification.

Both methodological issues and managerial (cost, organization required) are welcome.

Contributions that put special emphasis to the usage and suitability of advanced test languages such as TTCN-3 or the UML profile for testing are also encouraged. Practical aspects of validation are welcome, though their foundation will be equally considered.

Important Dates
- Submission deadline: Friday 10th September 2004
- Notification of acceptance for participation/presentation: Monday 11th October 2004

More information:
http://syst.eui.upm.es/conference/sv04/

FMICS 04 — Ninth International Workshop on Formal Methods for Industrial Critical Systems

Linz, Austria, 20-21 September 2004, co-located with ASE 2004

The aim of the FMICS workshops is to provide a forum for researchers who are interested in the development and application of formal methods in industry. In particular, these workshops are intended to bring together scientists who are active in the area of formal methods and interested in exchanging their experiences in the industrial usage of these methods. These workshops also strive to promote research and development for the improvement of formal methods and tools for industrial applications.

Topics

Topics include, but are not restricted to:
- tools for the design and development of formal descriptions
- verification and validation of complex, distributed, real-time systems and embedded systems
- verification and validation methods that aim at circumventing shortcomings of existing methods in respect to their industrial applicability
- case studies and project reports on formal methods related projects with industrial participation (eg safety critical systems, mobile systems, object-based distributed systems)
- application of formal methods in standardization and industrial forums

More information:
http://www.fmics04.cclrc.ac.uk/

TED Conference on e-Government
Bolzano, 2-4 March 2005

The European Science Foundation (ESF) has established the program 'Towards Electronic Democracy' (TED) (http://bayes.escet.urjc.es) with the overall objectives:
- to discuss and evaluate how advances of interactive decision-analytic tools might help develop e-democracy.
- to develop e-government systems which involve their citizens more fully in the public decision making process.

The conference is held as part of the TED program and aims at addressing a large spectrum of issues that are relevant and have to be investigated for a successful transition from the traditional form of government to a new form known as e-government. The conference has a strong interdisciplinary character and aims to bring together people from different sectors, including researchers, technology providers, administrative staff, and politicians.

Special Tracks
- e-Democracy: Improving citizen participation and policy making
- e-Government in small and medium sized government organizations (SMGO)
- e-Decision: Decision support systems on the web
- automation of cross-border and inter-organizational services
- data and application integration in the public administration.

Additional Topics
- e-negotiations
- e-procurement
- digital divide and social and cultural issues
- political and societal implications of e-government
- security issues and electronic identity card
- e-government and multilingualism
- open source in the public sector
- semantic web and web services for e-government
- best practice experiences and case studies

Important Dates
- paper submission: 10 September 2004
- notification of acceptance: 20 October 2004
- final paper submission: 26 November 2004

More information:
http://www.inf.unibz.it/tcgov2005

EURO-LEGAL

News about legal information relating to Information Technology from European directives, and pan-European legal requirements and regulations.

New Copyright Legislation in the UK

The Copyright and Related Rights Regulations 2003 and the Copyright (Visually Impaired Persons) Act 2002 came into force on 31 October 2003. This followed a long consultation process which generated a huge response from the copyright industry and users alike.

The new regulations introduce a monopoly right for copyright owners of 'communicating a work to the public', which includes broadcasting and any transmission of a copyright work by electronic means. Posting copyright material on a website is now clearly an infringement unless the owner of the copyright has given permission for the posting.

The use of a work will only be fair dealing providing it is for a non-commercial purpose, eg research or private study, and providing that it is not directly or indirectly for commercial purposes. In fact librarians and archivists, through whom copies of works are requested, must be satisfied that they are for non-commercial research or study.

The visually impaired are able to have full access rights to make an accessible copy of a work providing it is for their own personal use.

The new rules make it clear that making a temporary copy which is transient or incidental, eg within a digital network, that has no independent economic significance, will not be an act of infringement. However, important new rights have been introduced to prevent the use of products and devices which circumvent technical protective measures (TPMs), such as encryption. New criminal offences for dealing in such devices or products have been introduced which permit the police to obtain search warrants and forfeiture orders in the course of investigating these crimes. The Regulations also provide a further right against persons knowingly removing or altering rights of management information (RMI) when they are communicated to the public in digital form.

Whilst it has always been possible under common law to seek an injunction against service providers who knowingly infringe copyright, the new regulations now give the courts statutory powers to grant injunctions in these circumstances.

Working Hours in the UK

A recent labour force survey found that around 4 million white-collar workers in the UK work in excess of 48 hours per week, and that 16% of all those surveyed worked more than 60 hours per week. This is a rise of 4% since the year 2000. Employers argue that working more than the statutory limit of 48 hours a week should be a matter of individual choice on the part of workers, but official figures from the Trades Union Congress suggest that two thirds of those working more than 48 hours per week would like to work fewer hours.

The UK is the only European Member State that allows its workers to opt out of the 48 hour limit on the average working week. The EU will begin a review of the UK opt-out shortly.

Internet Crime

The National High-Tech Crime Unit in the UK have successfully brought a case of fraudulent theft against six men for the largest identity theft racket to be uncovered in the UK. The men are alleged to have used internet property auction sites and credit reference agency databases to steal the identities of people with good credit ratings who have recently died, to secure loans, cheque books, credit cards, passports and driving licences. With their ill-gotten gains they bought houses, high performance cars and expensive jewellery. Detectives were able to trace the gang through an IP address.

by Heather Weaver, CCLRC
Heather Weaver regrets that she is unable to reply personally to emails or telephone calls seeking legal advice.

🌸 SARIT — **Daniel Thalmann** is the new president of the Swiss Association for Research in Information Technology; he is also the SARIT representative to the ERCIM Board of Directors. Professor Thalmann is Director of the Virtual Reality Laboratory at the Swiss Federal Institute in Lausanne. **Patrick Furrer** from Euresearch is SARIT's new representative to the ERCIM executive committee.

🌸 CWI — **Ute Ebert** has won the Minerva Prize 2004. This was announced by the Dutch Foundation for Fundamental Research on Matter, FOM. Ebert's publication was chosen unanimously out of 28 submissions. The Minerva Prize is awarded every two year for the best scientific publication by a Dutch female writer on a physics topic. Ebert, leader of the Nonlinear Dynamics and Complex Systems theme MAS3, receives the award of 5000€ for the paper Spontaneous Branching of Anode-Directed Streamers Between Planar Electrodes, published in 2002 in the journal Physical Review Letters. Using accurate computer simulations and careful analysis of the observed phenomena, Ebert and her co-authors Manual Arrayás and Willem Hundsdorfer showed that the branching of sparks can be explained with a simple physical gas model. Ebert received the Minerva Prize on 26 May. The Ministery of Education, Culture and Science doubled the prize.

Photo: Liesbeth Sluiter

🌸 INRIA — **Gilles Kahn** was appointed Chairman of INRIA's Board of Directors by decree of the President of the French Republic dated May 26, 2004. According to INRIA's statutes, the Chairman of the Board of Directors also assumes the function of Chief and Executive Officer of the Institute.

Gilles Kahn, 58, is a member of the Academy of Sciences and is also INRIA's Scientific Director. Gilles Kahn is an expert in programming environments and computer aided proof environments. A former student of the École Polytechnique (1964), he spent several years conducting research abroad — Stanford University, U.S.A. (1968-71), University of Edinburgh, UK, (1975-76), the Isaac Newton Institute, Cambridge, UK (1995)-in collaboration with scientists of renown.

© INRIA / Photo: F. Jannin

Gilles Kahn was hired by the CEA and then by the CISI, while carrying out his research at Rocquencourt. He joined INRIA (then called IRIA) in 1977 as the head of a project dedicated to the development of programming environments. In 1983, Gilles Kahn participated in the foundation of the INRIA Sophia Antipolis Research Unit where he continued to head a research project while simultaneously assuming the responsibility of steering research as deputy to the Unit's director. In 1993, he joined the Institute's General Management and became INRIA's Scientific Director.

Gilles Kahn is a member of many scientific boards of companies and research institutions in France and abroad (Onera, BRGM, CNES, EDF, Ilog, among others). He belongs to the Board of Directors of CNRS and the General Council for Information Technology. Gilles Kahn also participates in numerous international evaluations of scientific organizations all around the world and in international scientific award juries.

Gilles Kahn was also called upon for matters of national interest. In particular, he was a member of the investigating committee concerning the failure of the 501 Ariane flight in 1996. He co-authored a report on research in telecommunications with Didier Lombard, General Director for Industrial Strategy and the Ministry of industry in 1997. He also co-authored a report on access to knowledge for all addressed to the President of the Republic.

Gilles Kahn succeeds Michel Cosnard, who had been appointed in December 2003 and assumed the functions of President of INRIA until May 26, 2004. Michel Cosnard had asked to be relieved from his functions following the sudden demise of his son.

One of the first priorities of the new CEO will be to implement the Institute's 2004-2007 Strategic Plan that was approved by the Board of Directors in July 2003. He himself greatly contributed to this Strategic Plan.

Gill Kahn represents INRIA on ERCIM's board of directors.
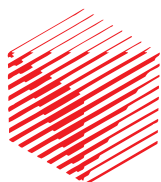
ERCIM – The European Research Consortium for Informatics and Mathematics is an organisation dedicated to the advancement of European research and development, in information technology and applied mathematics. Its national member institutions aim to foster collaborative work within the European research community and to increase co-operation with European industry.

W3C **ERCIM is the European Host of the World Wide Web Consortium.**

AARIT
Austrian Association for Research in IT
c/o Österreichische Computer Gesellschaft
Wollzeile 1-3, A-1010 Wien, Austria
Tel: +43 1 512 02 35 0, Fax: +43 1 512 02 35 9
http://www.aarit.at/

CCLRC
Council for the Central Laboratory of the Research
Councils, Rutherford Appleton Laboratory
Chilton, Didcot, Oxfordshire OX11 0QX, United Kingdom
Tel: +44 1235 82 1900, Fax: +44 1235 44 5385
http://www.cclrc.ac.uk/

CNR
Consiglio Nazionale delle Ricerche, ISTI-CNR
Area della Ricerca CNR di Pisa,
Via G. Moruzzi 1, 56124 Pisa, Italy
Tel: +39 050 315 2878, Fax: +39 050 315 2810
http://www.isti.cnr.it/

Crcim
Czech Research Consortium
for Informatics and Mathematics
FI MU, Botanicka 68a, CZ-602 00 Brno, Czech Republic
Tel: +420 2 688 4669, Fax: +420 2 688 4903
http://www.utia.cas.cz/CRCIM/home.html

CWI
Centrum voor Wiskunde en Informatica
Kruislaan 413, NL-1098 SJ Amsterdam,
The Netherlands
Tel: +31 20 592 9333, Fax: +31 20 592 4199
http://www.cwi.nl/

fonds national de la recherche
Fonds National de la Recherche
6, rue Antoine de Saint-Exupéry, B.P. 1777
L-1017 Luxembourg-Kirchberg
Tel. +352 26 19 25-1, Fax +352 26 1925 35
http:// www.fnr.lu

FWO FNRS
FWO
Egmontstraat 5
B-1000 Brussels, Belgium
Tel: +32 2 512.9110
http://www.fwo.be/

FNRS
rue d'Egmont 5
B-1000 Brussels, Belgium
Tel: +32 2 504 92 11
http://www.fnrs.be/

FORTH
Foundation for Research and Technology – Hellas
Institute of Computer Science
P.O. Box 1385, GR-71110 Heraklion, Crete, Greece
Tel: +30 2810 39 16 00, Fax: +30 2810 39 16 01
http://www.ics.forth.gr/

Fraunhofer IuK Gruppe Informations- und Kommunikationstechnik
Fraunhofer ICT Group
Friedrichstr. 60
10117 Berlin, Germany
Tel: +49 30 726 15 66 0, Fax: ++49 30 726 15 66 19
http://www.iuk.fraunhofer.de

INRIA
Institut National de Recherche en Informatique
et en Automatique
B.P. 105, F-78153 Le Chesnay, France
Tel: +33 1 3963 5511, Fax: +33 1 3963 5330
http://www.inria.fr/

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology, Mathematics and
Electrical Engineering, N 7491 Trondheim, Norway
Tel: +47 73 59 80 35, Fax: +47 73 59 36 28
http://www.ntnu.no/

SpaRCIM
Spanish Research Consortium for Informatics
and Mathematics
Esperanza Marcos, Rey Juan Carlos University,
C/ Tulipan s/n, 28933-Móstoles, Madrid, Spain,
Tel: +34 91 664 74 91, Fax: 34 91 664 74 90

SICS
Swedish Institute of Computer Science
Box 1263,
SE-164 29 Kista, Sweden
Tel: +46 8 633 1500, Fax: +46 8 751 72 30
http://www.sics.se/

SARIT
Swiss Association for Research in Information Technology
c/o Prof. Dr Alfred Strohmeier, EPFL-IC-LGL,
CH-1015 Lausanne, Switzerland
Tel +41 21 693 4231, Fax +41 21 693 5079
http://www.sarit.ch/

SRCIM
Slovak Research Consortium for Informatics and
Mathematics, Comenius University, Dept.of Computer
Science, Mlynska Dolina M, SK-84248 Bratislava, Slovakia
Tel: +421 2 654 266 35, Fax: 421 2 654 270 41
http://www.srcim.sk

MTA SZTAKI
Magyar Tudományos Akadémia
Számítástechnikai és Automatizálási Kutató Intézet
P.O. Box 63, H-1518 Budapest, Hungary
Tel: +36 1 279 6000, Fax: + 36 1 466 7503
http://www.sztaki.hu/

Trinity College
Trinity College
Department of Computer Science,
Dublin 2, Ireland
Tel: +353 1 608 1765, Fax: 353 1 677 2204
http://www.cs.tcd.ie/ERCIM

VTT
Technical Research Centre of Finland
P.O. Box 1200
FIN-02044 VTT, Finland
Tel:+358 9 456 6041, Fax :+358 9 456 6027
http://www.vtt.fi/tte

Order Form

If you wish to subscribe to ERCIM News
free of charge
or if you know of a colleague who would like to
receive regular copies of
ERCIM News, please fill in this form and we
will add you/them to the mailing list.

Send, fax or email this form to:
ERCIM NEWS
2004 route des Lucioles
BP 93
F-06902 Sophia Antipolis Cedex
Fax: +33 4 9238 5011
E-mail: office@ercim.org

I wish to subscribe to the

❏ *printed edtion*   ❏ *online edition (email required)*

Name:

Organisation/Company:

Address:

Post Code:

City:

Country

E-mail:

You can also subscribe to ERCIM News and order back copies by filling out the form at the ERCIM website at http://www.ercim.org/publication/Ercim_News/