institute imdea

madrid institute
for advanced studies

institute
imdea
software

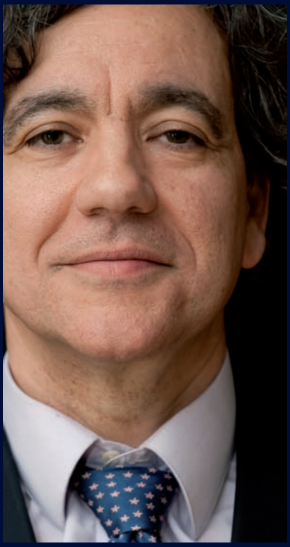
annual report
2011

madrid institute
for advanced studies

institute imdea software

annual report 2011

foreword



Manuel Hermenegildo
Director, IMDEA Software Institute

word

The IMDEA Software Institute was created by the Regional Government of Madrid under the strong belief that quality research in technology-related areas is the most successful and cost-effective way of generating knowledge, sustainable growth, and employment. This is more relevant in current times than ever, and software-related technology indeed has an immense potential for raising industrial competitiveness, opening whole new business areas, creating high added-value jobs, and, ultimately, improving quality of life. Today, gathering the material for this 2011 annual report, the Institute manifests itself as a vibrant and exciting reality, making significant progress towards its goals of excellence in research and technology transfer.

Without any doubt, the main acquired strength of the Institute is its people: its researchers and its support staff. The Institute has been very successful in attracting to Madrid top talent from all over the world and now includes 15 faculty (plus 2 more on leave or part time), 8 postdocs, 12 research assistants, and a number of interns, from 15 different nationalities. They joined after working at or obtaining their Ph.D. degrees from 32 different prestigious universities and research centers in 9 different countries, including Stanford University, Carnegie Mellon U., or Microsoft Research in the US, INRIA in France, U. of Cambridge in the UK, the Max Planck Software Institute in Germany, or ETH in Switzerland, to name just a few. In addition, 75 international researchers have visited and given talks at the Institute to date.

During 2011 IMDEA Software researchers have published 49 refereed articles (including some of the top-level venues in the field, such as POPL, ACM TOPLAS, CRYPTO, IEEE SSP, USENIX Security Symp., CSF, JCS, FM, TPLP, ICSOC, ICFP, ICLP, ICALP, etc.) and received *four best paper awards*, edited 6 proceedings, given 11 invited talks and 31 invited seminars, chaired 11 program committees, conferences or workshops, participated in 28 program committees, been members of 15 boards of journals and conferences, won a European best thesis award (for the second time in a row), and even won an international programming contest.

Also, during 2011 IMDEA Software Institute researchers have participated in 13 funded research projects and contracts, 4 funded by the European Union, and been beneficiaries of 16 fellowships. Through such projects and contracts the Institute has collaborated with a large number of companies including Atos, Siemens, Deimos, AbsInt, Microsoft, Fredhopper, Telefonica, and Thales (and with many others in other recent projects, such as France Telecom, SAP, Trusted Logic, Airbus, Alcatel, Daimler, and EADS). The Institute continues developing its strategic alliances with companies such as Atos, Telefonica, or BBVA, is in the process of completing two software registrations (with INRIA and Microsoft Research), and is also working on the commercialization of the ActionGUI technology developed by its Modeling Lab.

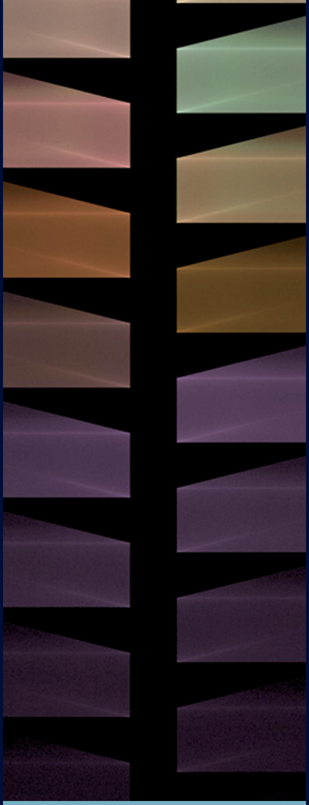
Major progress was also made in 2011 in the construction of the building for the Institute, in UPM's Montegancedo Science and Technology Park. At the time of writing construction is essentially completed and the building is going through the certification process before the move later in 2012.

Many thanks to all of those that have contributed to these achievements, and very specially to the Madrid Regional Government for their continuing vision and support.

table of c

table of contents

contents



1

general presentation

1.1. Profile

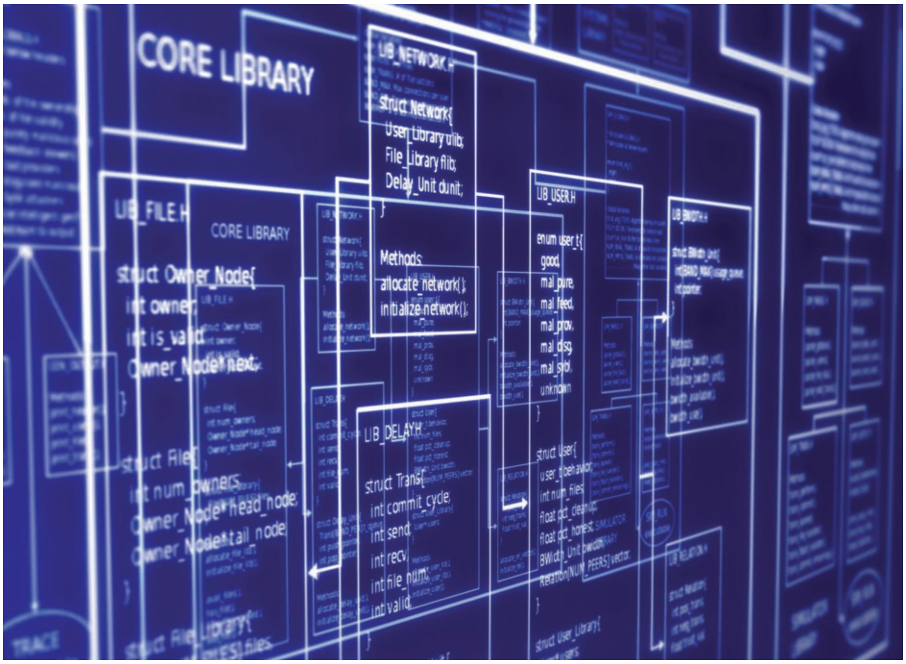
The IMDEA Software Institute (Madrid Institute for Advanced Studies in Software Development Technologies) is a non-profit, independent research institute promoted by the Madrid Regional Government (CM) to perform research of excellence in the methods, languages, and tools that will allow the cost-effective development of software products with sophisticated functionality and high quality, i.e., safe, reliable, and efficient.

The IMDEA Software Institute is part of the Madrid Institute for Advanced Studies (IMDEA) network, an institutional framework created to foster social and economic growth in the region of Madrid by promoting research of excellence and technology transfer in a number of strategic areas (water, food, social sciences, energy, materials, nanoscience, networks, and software) with high potential impact.

1.2. Motivation and Goals

It is difficult to overstate the importance of software both for our everyday lives and for the industrial processes which, running behind the scenes, are necessary to sustain the modern world. Software is the enabling technology in many devices and services which are now an essential part of our world and on which we, to different degrees, depend on: accounting, banking, cell phones, cars, flight control systems, behavior of the stock market, digital television, life support systems... not to mention tablets, computers, and the Internet itself. This pervasiveness explains the global figures around software and the IT services sector: the global software market has an estimated value of 225.000 M€ in 2008 and is estimated to grow to 360.000 M€ by 2013, being one of the few sectors which, despite the economic turmoil, continues to grow in terms of turnover, profit, and jobs. According to European Commission data, the ICT sector is directly responsible for 5% of EU GDP, with a market value of approximately 660.000 M€ and makes a proportionately much higher still contribution to the growth of productivity in general: 20% of such growth in productivity comes directly from the ICT sector itself and 30% from investments in ICT.

Given the economic relevance of software and its pervasiveness, errors and failures in software can have high social and economic cost: the results of malfunctioning software can range from being annoying (e.g., having to reboot), through posing serious social and legal problems (e.g., privacy and security leaks), to having high economic cost (e.g., software failures in financial markets and software-related product recalls) or even being a threat to human lives (e.g., a malfunctioning airplane or medical device). Unfortunately, developing software of an appropriate level of reliability, security, and performance, at a reasonable cost is still a challenge today. Some degree of correctness can be achieved by careful human or machine-assisted inspection at very high monetary costs, but the risk of errors produced by human mistakes will still be lurking in the dark. Reducing software errors in a cost-effective manner is therefore a task better left to automatic tools. These tools are, however, extremely hard to produce and pose scientific and technological challenges. Because of the ubiquity of software, solutions to these challenges can have a significant and pervasive impact on productivity and on the general competitiveness of the economy.



This is precisely the main mission of the IMDEA Software Institute: to perform research of excellence in methods, languages, and tools that will allow the cost-effective development of software products with sophisticated functionality and high quality, i.e., secure, reliable, and efficient. This research focus includes all phases of the development cycle (analysis, design, implementation, validation, verification, maintenance); its distinguishing feature is the concentration on approaches that are rigorous and at the same time allow building practical tools.

In order to achieve its mission, the IMDEA Software Institute is gathering a critical mass of world-wide, top class researchers, and is at the same time developing synergies between them and the already significant research base and industrial capabilities existing in the region. Indeed, most of the IT-related companies in Spain (and, specially, their research divisions) are located in the Madrid region, which facilitates collaboration and technology transfer. Thus, the IMDEA Software Institute brings about the opportunity of grouping a critical mass of researchers and industrial experts, which can allow for significant improvement in the impact of research.

1.3. Legal Status and Management Structure

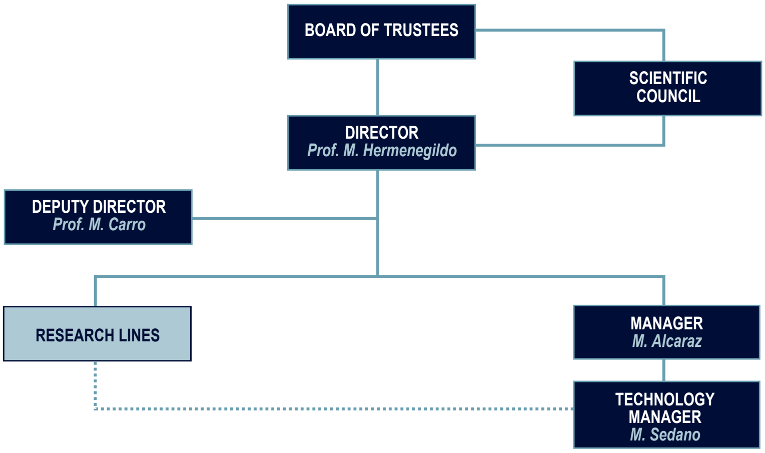


Figure 1.1: Management structure of the IMDEA Software Institute

The IMDEA Software Institute is a non-profit, independent organization, constituted as a Foundation. This structure brings together the advantages and guarantees offered by the foundation status with the flexible and dynamic management typical of a private body. This combination is deemed necessary to attain the goals of excellence in research, cooperation with industry, and attraction of talented researchers from all over the world. The Institute was created legally on November 23, 2006, following a design that was the result of a collaborative effort between industry and academia, at the initiative of the Madrid Regional Government, and started its activities during 2007.

The main governing body of the Institute is the Board of Trustees. The Board includes representatives of the Madrid Regional Government, universities and research centers of Madrid, scientists with an international reputation in software development technologies, and representatives of companies, together with independent experts.

The Board is in charge of guaranteeing the fulfillment of the foundational purpose and the correct administration of the funds and rights that make up the assets of the Institute, maintaining their appropriate returns and utility. The Board appoints the Director, who is the CEO of the Institute, among scientists with a well-established international reputation in software development technologies. The Director fosters and supervises the activities of the Institute, and establishes the distribution and application of the available funds among the goals of the Institute, within the patterns and limits decided by the Board of Trustees. The Director is assisted by the Deputy Director and the General Manager, who take care of the legal, administrative, and financial activities of the Institute. They are, in turn, helped by the Technology Manager, who is in charge of handling project preparation and development, industrial relations, and technology transfer. The structure is depicted in Figure 1.1.

The Board of Trustees and the Director are assisted in their functions by the Scientific Advisory Board, a scientific council currently made up of 9 scientists from 6 different countries with expertise in different areas of research covered by the Institute. The tasks of this scientific council include: to provide advice on and approve the selection of researchers; to provide advice and supervision in the preparation of yearly and longer-term (4-year) strategic plans; to evaluate the performance with respect to those plans; and to give general advice on matters of relevance to the Institute.

1.4. Location

During 2011 the IMDEA Software Institute has continued to be located temporarily in a newly renovated floor of the School of Computer Science of the Technical University of Madrid (UPM), in the Montegancedo Science and Technology Park. However major progress has been made in the construction of the new building which will be the permanent location for the Institute, also in the Montegancedo Park. Construction is expected to finish by mid 2012 and the move into this new facility is planned to be made after Summer 2012.

This location has excellent access to the UPM Computer Science Department as well as to other new research centers within the Montegancedo Park. These centers include

the Madrid Center for Supercomputing and Visualization (CESVIMA), the Montegancedo Campus UPM company "incubator," the Institute for Home Automation (CEDINT), the Institute for Biotechnology and Plant Genomics (CBGP), the Center for Biomedical Technology (CTB), the Microgravity Institute, and the Spanish User Support and Operations Centre for ISS payloads (USOC). In particular, the CESVIMA houses the second largest supercomputer in Spain and one of the largest in Europe, as well as a state-of-the-art visualization cave, and is equipped for massive information storage and processing, high performance computing, and advanced interactive visualization. A number of additional research centers are currently finishing construction in the campus.

The new site will also make use of all the convenient new infrastructures that have been completed recently around the campus, such as the recently opened "Montepríncipe" stop of the Madrid Underground. The campus has recently obtained the prestigious "International Campus of Excellence" label, and is the only campus in Spain to receive a "Campus of Excellence in Research and Technology Transfer" award in the Information and Communications Technologies area from the Spanish government.

1.5. Appointments to the Board of Trustees

During 2011, Jon Juaristi, Director General of Universities and Research, was appointed member of the Board of Trustees as representative of the Madrid Regional Government. Salvador Victoria Bolívar moved to the post of Counselor for Social Affairs, leaving the Board. José María Rotellar García moved to the post of Vicecounselor of the Treasury, remaining on the Board.

Narciso Martí Oliet replaced Carmen Fernández Chamizo as representative of Complutense University of Madrid. Eduardo Sicilia, former BBVA representative, was appointed as external expert. David Ríos, representative from Rey Juan Carlos University, left the Board at the end of his term.

Finally, at the beginning of 2012 Jorge Sáinz moved to a position in the Ministry of Education of the Spanish Government and Juan Ángel Botas, professor at Rey Juan Carlos University, replaced him in his position as Deputy Director of Research at the Comunidad de Madrid and as member of the Board of Trustees and Chair of the Standing Committee.

Section 1.6 presents the composition of the governing bodies on the date of the last Board of Trustees (November 22, 2011).

1.6. Members of the Governing Bodies

Board of Trustees

Chairman of the Foundation

Prof. David S. Warren
State University of New York at Stony Brook, USA.

Vice-chairman of the Foundation

Excma. Sra. Dña. Alicia Delibes Liniers
Vice-counselor for Education, Madrid Regional Government, Spain.

Madrid Regional Government

Excma. Sra. Dña. Alicia Delibes Liniers
Vice-counselor for Education, Madrid Regional Government, Spain.

Ilmo. Sr. D. José María Rotellar García
Vice-counselor of the Treasury, Madrid Regional Government, Spain.

Prof. Jon Juaristi
Deputy Director for Research, Madrid Regional Government, Spain.

Prof. Jorge Sáinz Gonzalez
Assistant Director of Research, Department of Education, Madrid Regional Government, Spain). Chairman of the Standing Committee.

Universities and Public Research Bodies

Prof. Narciso Martí Oliet
Professor, Universidad Complutense de Madrid, Spain.

Prof. Francisco Javier Segovia Pérez
Dean of the School of Computer Science, Universidad Politécnica de Madrid, Spain.

Prof. Carmen Peláez Martínez
Vice-president for Research, Consejo Superior de Investigaciones Científicas, Spain.

Scientific Trustees

Prof. David S. Warren
State University of New York at Stony Brook, USA. Chairman of the Board of Trustees.

Prof. Patrick Cousot
École Normale Supérieure de Paris (ENS), France and Courant Institute, New York University, USA.

Prof. Luis Moniz Pereira
Universidade Nova de Lisboa, Portugal.

Prof. José Meseguer
University of Illinois at Urbana Champaign, USA.

Prof. Roberto Di Cosmo
Université Paris 7, France.

Expert Trustees

Mr. José de la Sota Rius
Managing Director, Fundación para el Conocimiento (MadrI+D), Madrid, Spain.

Mr. Eduardo Sicilia Cavanillas
Escuela de Organización Industrial, Madrid, Spain.

Industrial Trustees

Board meetings have been attended, as invitees, by representatives of the following companies:

BBVA
Ms. María Carmen López Herranz. Innovation - Global Observatory & Portfolio Director at BBVA

Telefónica I+D
Mr. Francisco Jariego, Director for Technology Strategy at Telefónica R&D.

Deimos Space
Mr. Miguel Belló Mora, General Director and Mr. Carlos Fernández de la Peña.

Atos Origin
Mr. José María Cavanillas, Director Research & Innovation, and Ms. Clara Pezuela.

Secretary

Mr. Alejandro Blázquez Lidoy

Scientific Advisory Board

Prof. David S. Warren
State University of New York at Stony Brook, USA. Chairman of the Board.

Prof. María Alpuente
Universidad Politécnica de Valencia, Spain.

Prof. Roberto Di Cosmo
Université Paris 7, France.

Prof. Patrick Cousot
École Normale Supérieure de Paris (ENS), France and Courant Institute, New York University, USA.

Prof. Veronica Dahl
University Simon Fraser, Vancouver, Canada.

Prof. Herbert Kuchen
Universität Münster, Germany.

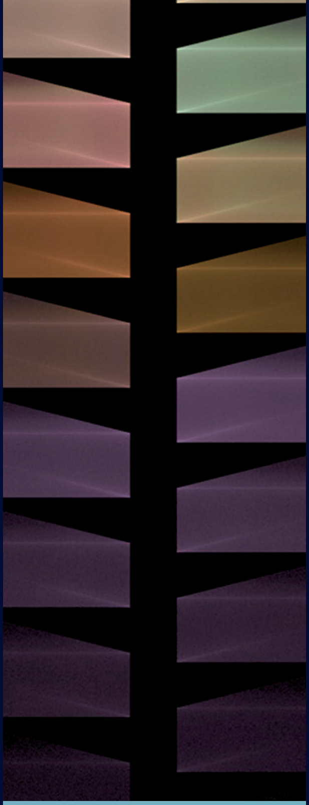
Prof. José Meseguer
University of Illinois at Urbana Champaign, USA.

Prof. Luis Moniz Pereira
Universidade Nova de Lisboa, Portugal.

Prof. Martin Wirsing
Ludwig-Maximilians-Universität München, Germany.

2

industrial and institutional partnerships



2.1. Industrial Partnerships

As mentioned before, one of the most successful and cost-effective ways of increasing the competitiveness of industry, and thus contributing to sustainable growth and employment, is by incorporating into processes and products new scientific results and technology. As a generator of such knowledge and technology, in an area with high potential economic impact, IMDEA Software collaborates with industry in a variety of ways in order to foster technology transfer.

The principal way in which this is carried out is through focused collaborations with companies in the framework of *competitive collaborative projects and direct contracts*. Figure 2.1 lists some of the companies with which the IMDEA Software Institute has collaborated to date in such projects and contracts (the currently active ones are described further in Chapter 5).

Other forms of industry collaboration include the *funding by industry of research assistantships* at the Institute (doctorate or masters work) on industry-relevant topics (for example, Microsoft funds a research assistant working on systems software verification and Deimos Space co-funds a research assistant working on rigorous development of satellite image processing), *transfer of research personnel trained at the Institute to companies* (IMDEA Software-trained personnel has already been transferred to companies such as Atos, Microsoft, or Google), funding by industry of research *stays of Institute researchers at company premises* (for example, Institute researchers have made industrially-funded extended stays at Deimos Space, Microsoft Redmond in the US, or Microsoft Cambridge in the UK), *access to the Institute's technology and scientific results* (for example, researchers of the Institute have met with personnel from BBVA, Telefónica I+D, Ericsson, GMV, INDRA, IBM, Canal de Isabel II, Interligare, or Lingway, among many others, to present their main research results), access to the Institute's researchers as consultants, participation of company staff in Institute activities, etc.

Another important form of technology transfer is the *commercialization of the technology* developed at the Institute. Given the controversy around software patents (and the impossibility of filing software patents in Europe) the Institute is combining the protection of its intellectual property (for example, two *software registrations* are currently being completed with INRIA and Microsoft Research) with other innovative business models, such as those based on open-source or free software licenses, together with the licensing of such technology and the *creation of technology-based companies*. In this line the Institute is actively working on the commercialization of the ActionGUI technology developed by its Modeling Lab.

The Institute has also established long-term *strategic alliances* with the main stakeholders in the sector which facilitate longer-term collaboration across projects. This includes specially the companies which are members of the Board of Trustees or invited to these meetings (currently BBVA, Atos, Telefónica, and DEIMOS Space). Also along the strategy line, the Institute participates jointly with industry in Spanish and EU *Technology Platforms*, such as the Technology Clusters of the Madrid Region, the INES Spanish Platform for Software and Services, the Internet of the Future Es.Internet Spanish platform, and the European Technology Platform for Software and Services. This allows aligning research agendas and facilitates joint participation in projects.

Project	Funding Agency	Industrial Partners
MOBIUS	FP6: IP	France Telecom, SAP AG, Trusted Labs
HATS	PF7: IP	Fredhopper
NESSoS	PF7: NoE	Siemens, ATOS
ES_PASS(*)	ITEA2, MITyC	Airbus France, Thales Avionics, CS Systèmes d'Information, Daimler AG, PSA Peugeot Citroen, Continental, Siemens VDO Automotive, EADS Astrium, GTD, Thales Transportation, and IFB Berlin
EzWeb	MITyC	Telefónica I+D, Alimerka, Integrasys, Codesyntax, TreeLogic, Intercom Factory, GESIMDE, Yaco, SoftTelecom
DESAFIOS-10	MICINN	BBVA-GlobalNet, LambdaStream, Deimos Space
PROMETIDOS	Madrid Regional Government	Deimos Space, Atos Origin, BBVA-GlobalNet, Thales, Telefónica I+D
MTECTEST	Madrid Regional Government	Deimos Space
AbsInt	AbsInt Gmbh	AbsInt
2 SEIF awards	Microsoft SEIF	Microsoft Research
ENTRA	FP7: STREP	XMOS
VARIES	FP7: ARTEMIS	HI iberia, IntegraSys, Tecnalia, ...

(*) Through associated group at Universidad Politécnica de Madrid.

Figure 2.1. Some companies with which the IMDEA Software Institute has collaborated through projects and contracts to date.

2.2. Cooperation with Research Institutions

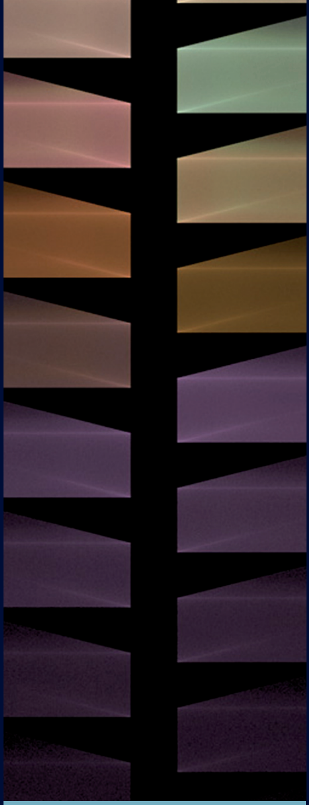
The Institute offers researchers access to and collaboration with universities and other research centers, in the Madrid region and beyond. The Institute is actively working with these institutions to create a critical mass of researchers capable of producing results which have significant potential impact on industry and society in general. At present the Institute has already signed agreements with the following universities and research centers:

- Universidad Politécnica de Madrid (from November 2007).
- Universidad Complutense de Madrid (from November 2007).
- Universidad Rey Juan Carlos (from January 2008).
- Roskilde University (from June 2008), Denmark.
- Consejo Superior de Investigaciones Científicas (from November 2008).

These agreements establish a framework for the development of collaborations and include the joint use of resources, equipment, and infrastructure, hiring of staff, joint participation in research projects, joint participation in graduate programs, or the association of researchers and research groups with the Institute. In particular, research assistants at the IMDEA Software Institute can follow graduate studies at any of the cooperating Institutions, while funded by IMDEA Software.

To illustrate the scope and importance for the Institute of these agreements, we offer here some highlights. The agreement with the Universidad Politécnica de Madrid includes provisions for the location of the Institute building in its Montegancedo Science and Technology Park as well as a joint graduate program, instrumented currently as a separate track on *Software Development through Rigorous Methods* in an existing Masters / PhD program at UPM ("MUSS / DSS"). Under the agreement with the Consejo Superior de Investigaciones Científicas, two of its researchers —Cesar Sánchez and Pedro López— are also part of the research staff of the Institute. Finally, under the agreement with Roskilde University, one of its full professors —John Gallagher— is also part-time senior researcher at the Institute.

The Institute also has already a strong presence in national and international bodies. In particular, it is a member of *Informatics Europe,* the organization of all deans, chairs, and directors of the leading Departments and Institutes of Computer Science in Europe, similar to the CRA in the US. In addition, the Institute is a member of ERCIM, the *European Research Consortium for Informatics and Mathematics* through SpaRCIM, the Spanish representative in ERCIM, where Manuel Hermenegildo, IMDEA Software Institute Director, is also the President of the Executive Board.



3 research lines

As briefly explained in Section 1.2, the cost-effective development of complex, safe, reliable, and efficient software is not a simple task, and it cannot be solved by simple “magic bullets” or more enlightened management. The problem affects all stages in the development lifecycle (analysis, design, implementation, verification, maintenance). The IMDEA Software Institute performs research on these aspects along a number of dimensions which include *Methodologies* (the development and industrial adoption of mathematically rigorous methodologies can improve the software process further), *Languages* (the basis for expressing software functionality, behavior, and properties), *Verification and Validation* (semantically well-founded, tool-supported methods to validate code or designs with respect to specifications), and *Adequacy/Optimization* (the optimal use of resources to achieve a desired goal). To this end:

- The research vision materializes in a number of *High-level Research Lines*. The current main lines are depicted as rows in Figure 3.1.
- The vision includes also a number of focusing *Areas of Application*: areas of engineering where the Institute aims and expects to make an impact and which have been identified as priorities in collaboration with industry. The main current areas of application are depicted as columns in Figure 3.1.

These areas of application and high-level research lines are explained further in the rest of the chapter. Finally, two fundamental, cross-cutting issues pervade the vision:

- *Tools*: well-founded and cost-effective (prototypes of) tools are fundamental study harnesses, demonstrators, and technology transfer vehicles for the techniques for automation of high quality software development.
- *Foundations*: methods and languages should be built on appropriate mathematical foundations, and at the same time be practical.

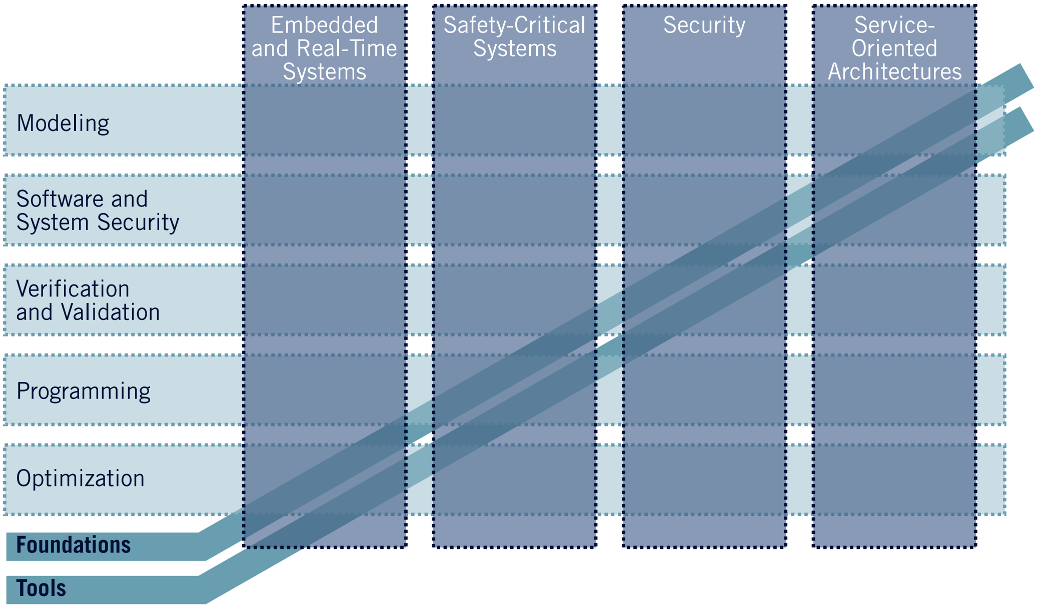
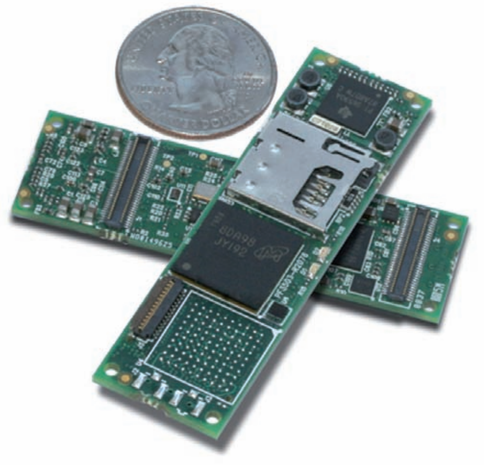

Embedded and Real-Time Systems
Safety-Critical Systems
Security
Service-Oriented Architectures
Modeling
Software and System Security
Verification and Validation
Programming
Optimization
Foundations
Tools

Figure 3.1: Main research lines, application areas, and cross-cutting issues.



Embedded systems are not only small: they need to be autonomous for large periods of time, which makes resilience and low energy consumption of utmost importance.

3.1. Areas of Application

The following are some areas of application: areas of engineering where the IMDEA Software Institute aims and expects to make an impact.

3.1.1. Embedded and Real-Time Systems

One of the application areas of software where correctness is most critical is embedded systems. An embedded system is a computational artifact that is subject to physical constrains, and whose correct functioning cannot depend on human guidance. In particular, embedded systems are involved in safety-critical applications (such as control systems of automobiles or aircrafts) or systems for highly remote operation (satellite, space, etc.). Embedded systems are also pervasive in areas of high economic impact, like mobile telephony or consumer electronics. Embedded systems must be resource-aware and are often also real-time systems. This means that the computation must be correctly performed within its time constraints, and also with an adequate use of resources. There is a common perception of the potential of rigorous techniques that can improve the quality of embedded software, or the time to market of new devices or families of devices.

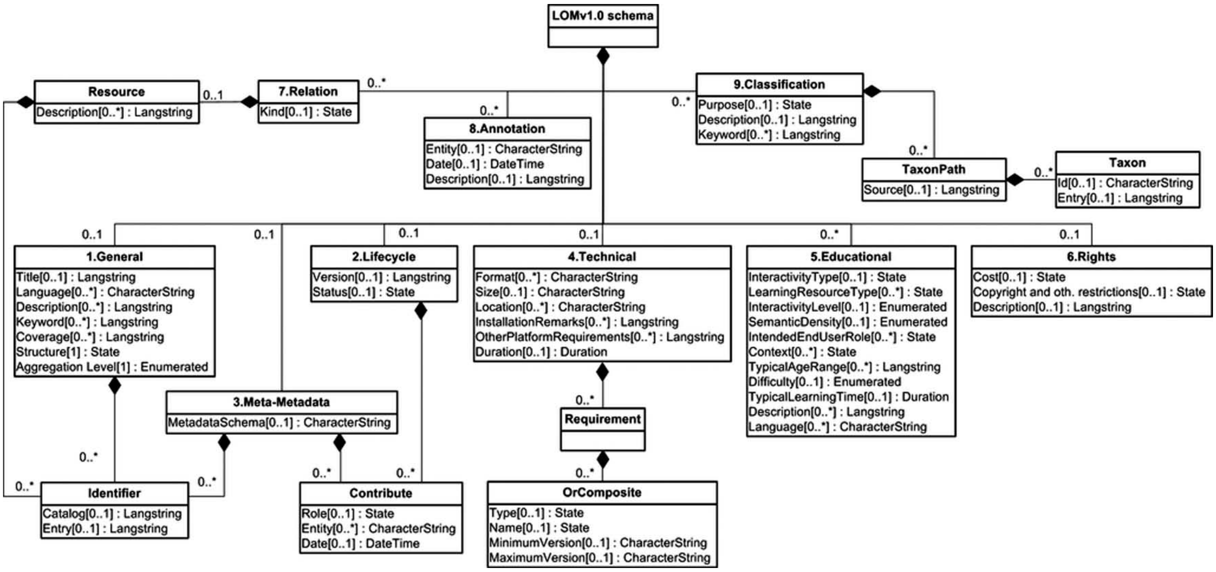
Most of the research activities required by embedded and real-time systems and planned at the IMDEA Software Institute are strongly related to the Strategic Research Agenda of the European Technology Platform on Embedded Computing Systems, ARTEMIS.

3.1.2. Safety-Critical Systems

Software is becoming pervasive in areas such as transportation (avionics, automotive), health (diagnosis, therapy), and control (of nuclear plants, of railway signaling systems, of conflict detection systems), where a failure or malfunction may be extremely damaging, even in terms of human lives. The constraints for such safety-critical systems are extremely stringent: the systems must be able to function during extremely long period of times, in presence of human mistakes or hardware or software failures, and provide an acceptable level of services at all times.

Thus, it is urgent to develop methods and tools that help support the development of such dependable software and its (quantitative) evaluation against the aforementioned constraints. To achieve this goal, it is important to build programming languages and software architectures that facilitate the development of fault-tolerant, resilient, and adaptable applications. One particular challenge is to scale existing methods so that they become effective in the context of distributed and networking systems.

Thousands of services are nowadays directly available on the Internet. This is a huge amount of power to tap from, but it needs to be done in a rational, controlled way in order to ensure correctness, dependability, and quality of service.


LOMv1.0 schema
Resource
Description[0..*] : Langstring
0..1
7.Relation
Kind[0..1] : State
0..*
8.Annotation
Entity[0..1] : CharacterString
Date[0..1] : DateTime
Description[0..1] : Langstring
0..*
9.Classification
Purpose[0..1] : State
Description[0..1] : Langstring
Keyword[0..*] : Langstring
0..*
TaxonPath
Source[0..1] : Langstring
0..*
Taxon
Id[0..1] : CharacterString
Entry[0..1] : Langstring
0..1
1.General
Title[0..1] : Langstring
Language[0..*] : CharacterString
Description[0..*] : Langstring
Keyword[0..*] : Langstring
Coverage[0..*] : Langstring
Structure[1] : State
Aggregation Level[1] : Enumerated
0..1
2.Lifecycle
Version[0..1] : Langstring
Status[0..1] : State
0..1
4.Technical
Format[0..*] : CharacterString
Size[0..1] : CharacterString
Location[0..*] : CharacterString
InstallationRemarks[0..*] : Langstring
OtherPlatformRequirements[0..*] : Langstring
Duration[0..1] : Duration
0..*
5.Educational
InteractivityType[0..1] : State
LearningResourceType[0..*] : State
InteractivityLevel[0..1] : Enumerated
SemanticDensity[0..1] : Enumerated
IntendedEndUserRole[0..*] : State
Context[0..*] : State
TypicalAgeRange[0..*] : Langstring
Difficulty[0..1] : Enumerated
TypicalLearningTime[0..1] : Duration
Description[0..*] : Langstring
Language[0..*] : CharacterString
0..1
6.Rights
Cost[0..1] : State
Copyright and oth. restrictions[0..1] : State
Description[0..1] : Langstring
0..1
3.Meta-Metadata
MetadataSchema[0..1] : CharacterString
0..*
Requirement
0..*
0..*
Identifier
Catalog[0..1] : Langstring
Entry[0..1] : Langstring
0..*
0..*
0..*
Contribute
Role[0..1] : State
Entity[0..*] : CharacterString
Date[0..1] : DateTime
0..*
OrComposite
Type[0..1] : State
Name[0..1] : State
MinimumVersion[0..1] : CharacterString
MaximumVersion[0..1] : CharacterString

3.1.3. Security

As our society increasingly relies on information technology, there is an urgent and unprecedented need to develop new security mechanisms for protecting infrastructures, data, and applications. Several concomitant factors aggravate the problems of information security.

In order to face this challenge, one must provide scalable and rigorous techniques that can be integrated in prevailing software development processes to enforce security of applications. Since many attacks arise at the application level, it is particularly important to achieve security at the level of programming languages, drawing from methods developed in programming language research (design, analysis, and verification), and developing security solutions at a level of abstraction that matches the programming language. At the same time it is important to target a wide range security- and privacy-related issues and scenarios, from whole systems to (big) data, and on the current plethora of platforms, ranging from apps to the cloud itself.

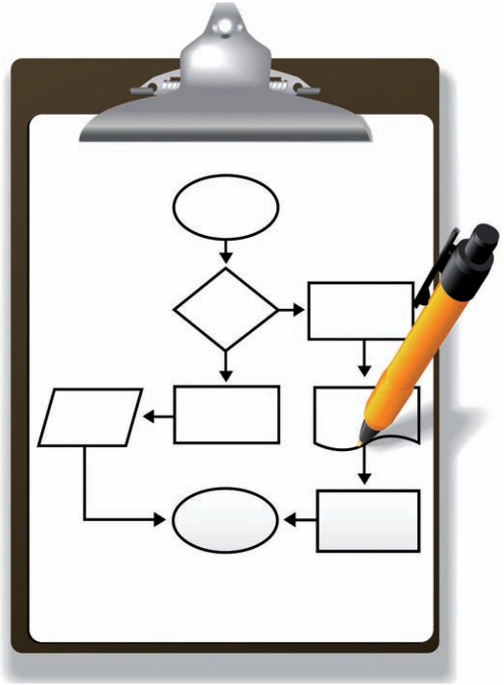
3.1.4. Service-Oriented Computing and Architectures

The evolution of computer infrastructures towards highly distributed networks makes it possible to provide users and software developers with a uniform and global access to software services. At the same time, selling services has become the biggest growth business in the IT industry. Service-Oriented Computing (SOC) is an attempt to provide at the level of software the necessary support for effectively programming, deploying, and maintaining services over highly-distributed networks. SOC draws from many areas of computer science, including software engineering, concurrent and distributed systems, and modular and component-based programming. While these areas are well developed in isolation, there remain significant challenges to combine the methodologies that stem from each area in order to deliver cost-effective approaches that support the construction and deployment of electronic services.

3.2. Research Lines

3.2.1. Modeling

A model is an abstraction of some aspect of a system (like a blueprint in engineering), which is created to serve particular purposes, for example, to present a human-understandable description of some aspects of the system or to present information in a form that can be mechanically analyzed. The term Model-Driven Engineering (MDE) is used to describe software development approaches in which abstract models of software systems are created and systematically transformed to obtain concrete implementations or skeletons. Model-driven development holds the promise of reducing system development time and improving the quality of the resulting products.



However, in mainstream MDE practice, models are usually informal, with no well-established semantics, and only used for documentation purposes. In fact, modeling has traditionally been a synonym for producing diagrams. Most models consist of a number of “bubbles and arrows” pictures and some accompanying text. The information conveyed by such models has a tendency to be incomplete, informal, imprecise, and sometimes even inconsistent.

In order to address the major challenges current MDE technologies are facing, we believe that the past and present work on formal methods is particularly relevant. Many of the flaws in modeling are caused by the limitations of the diagrams being used. A diagram simply cannot express some of the essential information of a thorough specification. To specify software systems, formal languages offer some clear benefits over the use of diagrams. Formal languages are unambiguous, and cannot be interpreted differently by different people, for example, an analyst and a programmer. Formal languages make a model more precise and detailed, and are subtle of manipulation by automated tools to ensure correctness and consistency with other elements of the model. On the other hand, a model completely written in a formal language is often not easily understood. In this sense, we believe that the interaction between the MDE and formal methods communities has a huge potential impact.

At the IMDEA Software Institute we are providing rigorous semantics for current MDE technologies (e.g., OCL, QVT) and we are developing tool-supported methodologies for applying these technologies for building *meaningful* models: i.e., models that have a clear and rich meaning, and that are therefore useful and valuable for developing quality software. At the same time, we are proposing new MDE technologies for specific areas of applications, including software and system security and graphical user interfaces.

3.2.2. Software and system security

The goal of this line is to develop methods and tools that provide an accurate security analysis of systems and software, together with some countermeasures to defeat malicious agents.

While software security traditionally focuses on low-level protection mechanisms such as access control, the popularization of massively distributed systems dramatically increases the number and severity of vulnerabilities at the application level. These vulnerabilities may be exploited by malicious software such as viruses, Trojan horses, *etc.*, but also (unintentionally) by buggy software, with disastrous effects.

Language-based security aims to achieve security at the level of the programming language, with the immediate benefit of countering application-level attacks at the same level at which such attacks arise. Language-based security is attractive to programmers because it allows them to express security policies and enforcement mechanisms within the programming language itself, using well-developed techniques that facilitate a rigorous specification and verification of security policies.

Language-based techniques can guarantee a wide range of policies including confidentiality, integrity, and availability, and their combination. However, their practical adoption has been hindered partly because known enforcement methods are confined to simple policies, such as non-interference for confidentiality. The most pressing challenges are defining

The aim of computer security is to create a virtual lock which can only be opened by the owner of the key and those the owner trusts. This makes it possible to distribute documents and data or to run systems and services while ensuring that only those entities (humans or programs) who are allowed can actually get access.

unified enforcement mechanisms that support flexible and customizable policies, and developing methods for providing a quantitative assessment of security.

The IMDEA Software Institute is developing rich policy languages that capture precisely common instances of information release. Moreover, these policy languages are directly applicable to powerful abstraction mechanisms that pervade modern programming languages. These policy languages are supported by automated verification procedures, that allow users to detect fraudulent software.

We are also developing accurate methods for a quantitative evaluation of program security. These methods account for covert channels, including timing behavior and resource consumption, and for resistance to common attacks, such as viruses. The ultimate goal is to develop comprehensive adversarial models and effective protection strategies against covert channels.

Language-based methods have been studied primarily for mobile code and very few methods are known to scale to distributed systems. One main challenge is to ensure security of distributed applications, using a combination of cryptographic and language-based methods. Programming language techniques provide an attractive approach to guarantee the security of distributed software, because they allow reasoning about programs and their cryptographic libraries in a unified framework. Moreover, programming language techniques are rigorous, and thus are useful to demonstrate beyond reasonable doubt that standard cryptographic systems, some of which have a long history of flawed security proofs and hidden but effective attacks, are secure.

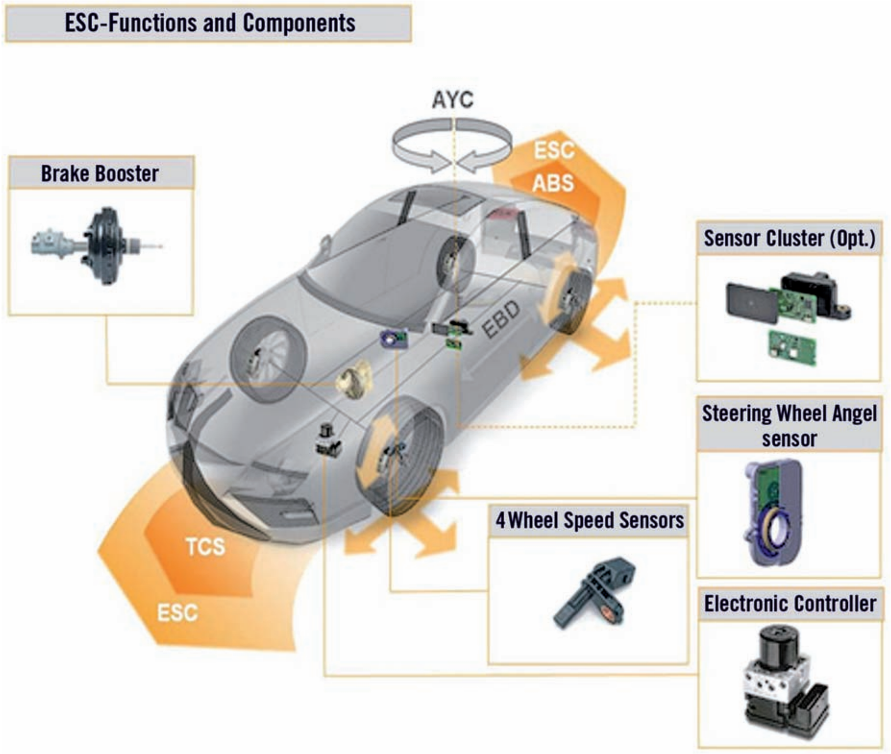
The IMDEA Software Institute is building tools that support the automated analysis of cryptographic systems and provide very strong guarantees of their correctness (cryptographic strength). The tools adopt the game-playing technique, that organizes the construction of cryptographic proofs as sequences of probabilistic games as a natural solution for taming the complexity of performing cryptographic proofs. The tools have been validated experimentally through the verification of widely deployed cryptographic standards.

3.2.3. Verification and Validation

Verification refers to the rigorous demonstration that software is correct; that is, it provides behavioral consistency according to a given specification of its intended behavior. By "intended behavior" we mean the properties that software is expected to satisfy when it is deployed. Software not possessing the properties might be defective: its execution might have unintended consequences. *Verified software* is software that is free of certain classes of defects because it has been rigorously proven that it satisfies its intended behavior. For these particular classes of defects, the verified software is termed zero-defect software. Such software does not require disclaimers that forgive developer error. Instead such software is guaranteed to be reliable — it behaves as intended.

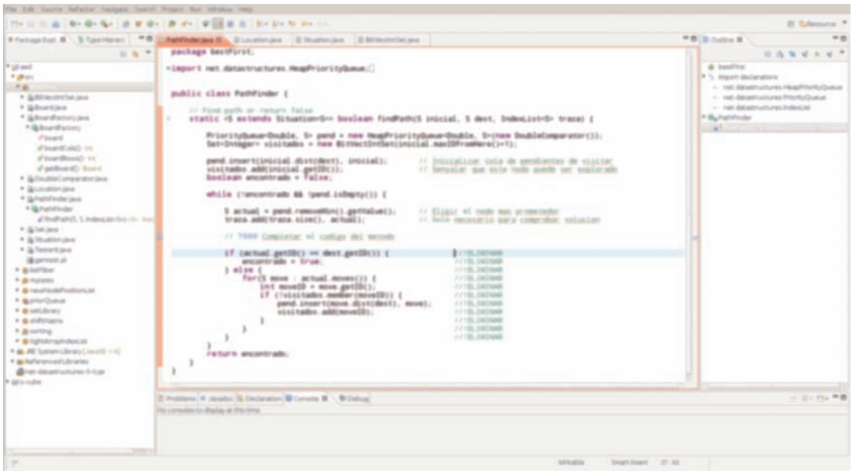
How do we "rigorously prove" that software is correct? The basic principle is to represent properties as logical formulas so that verification of the properties is akin to proving a theorem using proof techniques from mathematical logic. However, modern software is very complex and typically composed of several components, where each component can be written in a different programming language. For such complex software, proving properties manually is very difficult. The question that arises naturally is this: can the logic-based proof techniques be made to scale so that software can be automatically verified as much as possible so that the manual verification burden is minimized? Apart from managing the complexity of proofs, the benefits of automatic verification are as follows. First, the verification can be repeated whenever necessary and with the same results, thus attesting to the accuracy of verification. Second, proofs can be mechanically checked for correctness. Third, verification results can be reused: once a program has been verified, its specifications can be repeatedly used in verification of a larger piece of software without re-verification.

Researchers at the IMDEA Software Institute are involved in various aspects of automatic software verification. They study expressive languages and logics for specification of properties of software, particularly of software written in modern programming languages such as Java. Once a Java program is decorated with such specifications, off-the-shelf verifiers can be used to generate "verification conditions" which can then be discharged by theorem provers. Researchers are not only studying more efficient verification algorithms and decision procedures for improving theorem proving technology, but also are performing experiments on verifying realistic code such as Java libraries — whose programs are frequently used in building complex software — and design patterns, which provide generic solutions to common



Modern cars and trucks contain as many 100 million lines of computer code. This software runs on more than 30 on-board computers and controls vital functions, including the brakes, engine, cruise control, and stability systems. It is under increasing scrutiny in the wake of recent problems with major manufacturers and it is currently impossible to fully test.

Programming is notoriously difficult and error prone. Current tools (Eclipse editor, in the Figure) assist programmers in their task. However, much more sophisticated technology is still needed in order to reduce the time to market while actually increasing the degree of correctness of the delivered code beyond what is possible today.


```
package bestFirst;

import net.datastructures.HeapPriorityQueue;

public class PathFinder {

    // Find path or return false
    static <S extends Situation<S>> boolean findPath(S inicial, S dest, IndexList<S> traza) {

        PriorityQueue<Double, S> pend = new HeapPriorityQueue<Double, S>(new DoubleComparator());
        Set<Integer> visitados = new BitVectIntSet(inicial.maxIDFromHere()+1);

        pend.insert(inicial.dist(dest), inicial);      // Inicializar cola de pendientes de visitar
        visitados.add(inicial.getID());                // Senalar que este nodo puede ser explorado
        boolean encontrado = false;

        while (!encontrado && !pend.isEmpty()) {

            S actual = pend.removeMin().getValue();     // Elegir el nodo mas prometedor
            traza.add(traza.size(), actual);            // Solo necesario para comprobar solucion

            // TODO Completar el codigo del metodo

            if (actual.getID() == dest.getID()) {                  //!ELIMINAR
                encontrado = true;                                 //!ELIMINAR
            } else {                                               //!ELIMINAR
                for(S move : actual.moves()) {                     //!ELIMINAR
                    int moveID = move.getID();                     //!ELIMINAR
                    if (!visitados.member(moveID)) {               //!ELIMINAR
                        pend.insert(move.dist(dest), move);        //!ELIMINAR
                        visitados.add(moveID);                     //!ELIMINAR
                    }                                              //!ELIMINAR
                }                                                  //!ELIMINAR
            }                                                      //!ELIMINAR
        }
        return encontrado;
    }
}
```

No consoles to display at this time.

software problems. The automated proofs will be made publicly available in a repository linked to the Verified Software Repository of the international Verification Grand Challenge Project.

3.2.4. Advanced Programming and Optimization Tools

The goal of this line is to develop methods and tools that help programmers improve the quality and robustness of the programs they write, allow them to write better programs in a shorter time, and support efficient execution of code through highly optimizing compilers.

Regarding program correctness and robustness the aims are similar to those in verification, but the focus here is on tools that find errors and verify programs *during the process of writing such programs*, rather than a posteriori. This focus requires efficient and fully automatic program analysis methods.

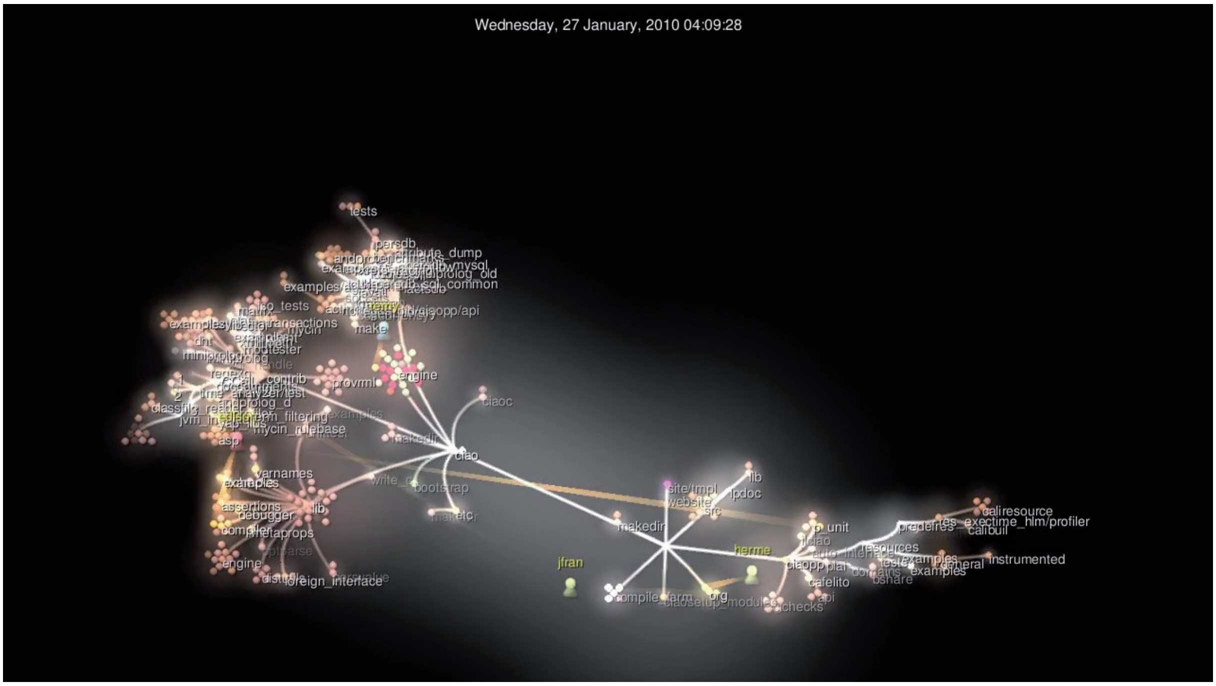
Abstraction-based techniques provide a unifying framework for this purpose. Their essence is abstract interpretation, a rigorous method which induces a dramatic reduction in the complexity of software analysis. It has been shown powerful enough to, for example, analyze automatically avionics software, a clear example of a large cyber-physical system, consisting of millions of lines of code, and subject to stringent conditions from the DO-178B standards. Researchers at the IMDEA Software Institute are developing tools that show that abstraction techniques can be embedded in development environments for routine use by programmers for on-line debugging, diagnosis, verification, and certificate generation, and that they combine naturally with (and reduce the need for) other techniques such as testing and run-time verification, which currently take more than 90% of overall development cost.

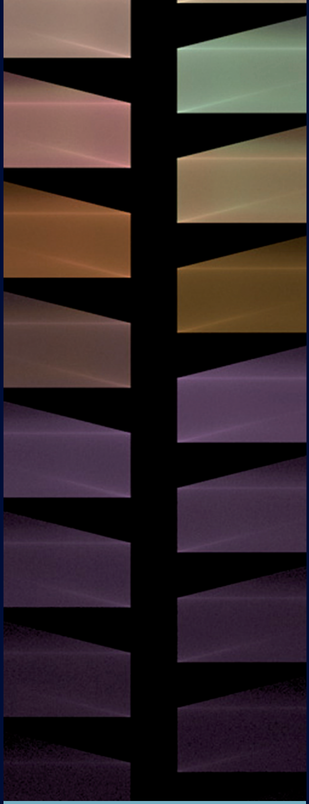
Abstraction-based techniques have also been shown particularly effective for high integrity and embedded software, where the properties of concerns are time and memory consumption, dynamic data sizes, energy consumption, termination, absence of errors or exceptions, etc. Researchers at the IMDEA Software Institute are developing advanced tools for debugging and verification of software with respect to these non-functional properties.

Another important way of improving the programming process, which allows programmers to write better programs in a shorter time, is by improving programming languages. Researchers at the IMDEA Software Institute are working on promising approaches such as extensible and multi-paradigm languages, support for domain-specific languages, support for multi-language applications, and service-oriented architectures.

Regarding the objective of supporting the efficient execution of code, abstraction-based techniques can also be used to ensure that programs are highly optimized before execution, i.e., that they run in the fastest and most resource-efficient way on the platforms and environmental conditions they are deployed on, while maintaining their observable behavior. Typical goals include saving on memory and processing time on sequential processors, adaptive task scheduling in parallel and distributed computers, self-reconfiguration, and automatic adaptation to environmental conditions.

A prominent form of such program optimization is automatic parallelization. As highly parallel processors are becoming an inexpensive and common facility in mainstream computing, there is an opportunity to build much faster, and eventually much better, software. Yet exploiting this enormous potential requires the development of new programming practices that reflect this profound change in the execution paradigm. Two common alternatives are to write parallel programs, using dedicated programming idioms and algorithms that help taming the complexity of parallel programs, or to automatically parallelize existing ones, using compilers for identifying parts of the application that are independent and can thus be run in parallel. Researchers at the IMDEA Software Institute are working on both approaches, developing languages and idioms more suited for parallelism and abstraction-based techniques and tools for allowing detection of common errors in parallel programs and for automatic parallelization of programs.





4 people

The IMDEA Software Institute strives towards excellence and being competitive with the highest-ranked institutions worldwide. Success in this goal can only be achieved by attracting highly-skilled personnel for the scientific teams and support staff. This is one of the main goals of the Institute, to the point of considering it a fundamental measure of its success.

Competition for talent in Computer Science is extremely high at the international level, since essentially all developed countries have identified the tremendous impact that IT has on the economy and the crucial competitive advantage that attracting truly first class researchers entails. To meet this challenge, the Institute is creating a world-class working environment that is competitive with similar institutions in Europe and in the US and combines the best aspects of a university department and a research laboratory.

Hiring at the Institute follows internationally-standard open dissemination procedures with public, international calls for applications launched periodically. These calls are advertised in the appropriate scientific journals and at conferences in the area, as well as in the Institute web page and mailing lists. Appointments at (or promotion to) the Assistant Professor, Associate Professor, and Full Professor levels (i.e., tenure-track hiring, tenure, and promotion decisions) are approved by the Scientific Advisory Board. In addition to staff positions, the Institute also has its own programs for visiting researchers, graduate scholarships, and internships. In all aspects related to human resources, the Institute follows the recommendations of the European Charter for Researchers and a Code of Conduct for their Recruitment (http://ec.europa.eu/), which it has duly signed.

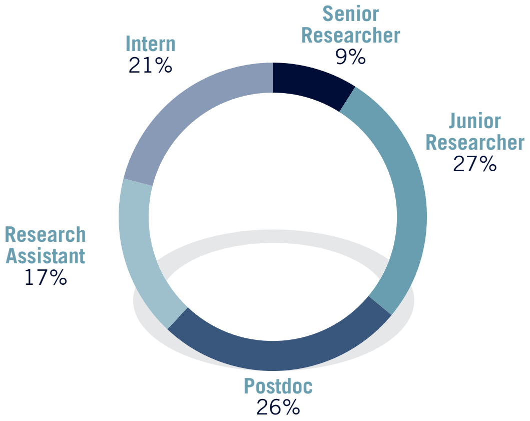


Figure 4.1. Type of position applied for.

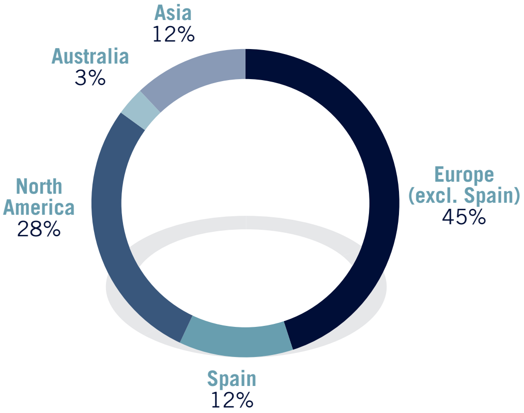


Figure 4.2. Location of previous institution for applicants at or above the postdoc level (by continent + Spain).

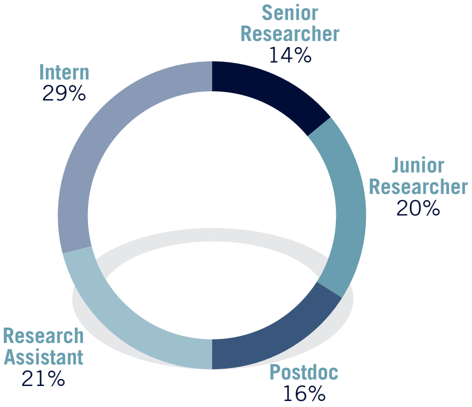


Figure 4.3. Location of previous institution for applicants at or above the postdoc level (by continent + Spain).

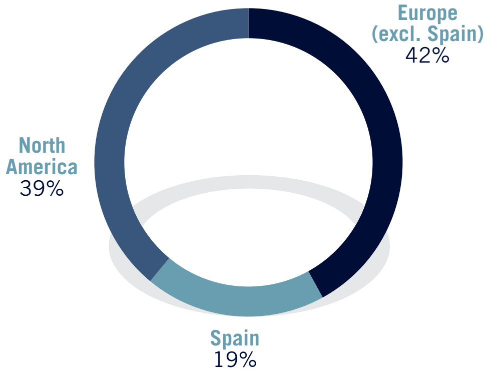


Figure 4.4. Type of position, all researchers

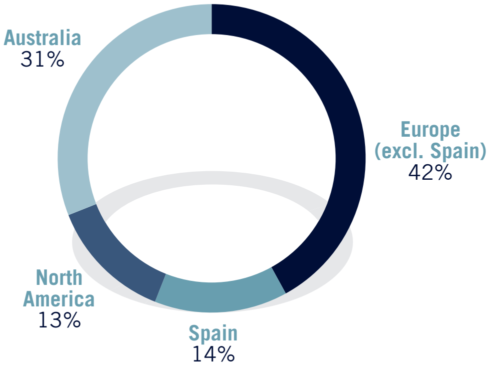


Figure 4.5. Where PhD was obtained (by continent + Spain).

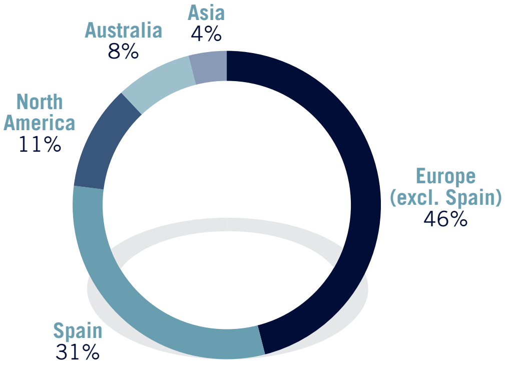


Figure 4.6. Location of previous institution, all (by continent + Spain).

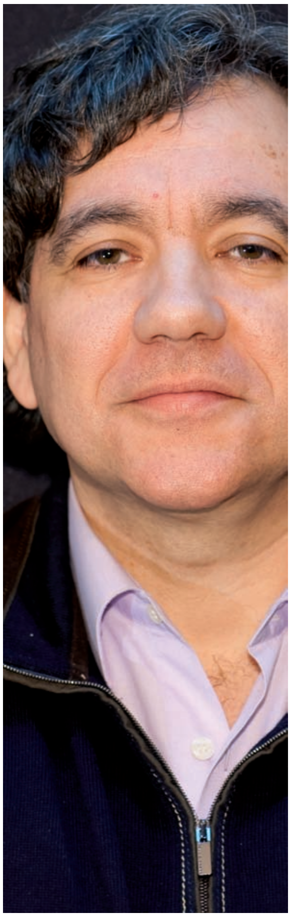
Applications

Figure 4.1 shows the proportions of applications received for each category during 2011: full and associate professors (senior researchers), assistant professors (junior researchers), postdoctoral researchers, research assistants, and interns. Figure 4.2 displays the location (by continents) of the institutions in which the applicants were at the time of application (for senior, junior, and postdoctoral positions). Spain is highlighted separately from the rest of Europe to provide a finer view of the data (level of internationalization).

Status

At the end of 2011, the scientific staff of the Institute was composed of six full or associate professors (plus one part-time), ten assistant professors (three non tenure-track and one on leave), eight postdoctoral researchers, and twelve research assistants (PhD candidates). Fifteen interns spent a variable length of time (from one month to half a year) at the Institute collaborating with the Faculty members. Additionally, a number of visitors have also been at the Institute during 2011. Figure 4.6 presents the nationalities of researchers at or above the postdoc level.

faculty



Manuel Hermenegildo

Professor and Scientific Director

Manuel Hermenegildo received his Ph.D. degree in Computer Science and Engineering from the University of Texas at Austin, USA, in 1986. Since January 1, 2007 he is Full Professor and Scientific Director of the IMDEA Software Institute. He is also a full Prof. of Computer Science at the Tech. U. of Madrid, UPM. Previously to joining the IMDEA Software Institute he held the P. of Asturias Endowed Chair in Information Science and Technology at the U. of New Mexico, USA. He has also been project leader at the MCC research center and Adjunct Assoc. Prof. at the CS Department of the U. of Texas, both in Austin, Texas, USA. He has received the Julio Rey Pastor Spanish National Prize in Mathematics and Information Science and Technology and the Aritmel National prize in Computer Science, and he is an elected member of the Academia Europaea. He is also one of the most cited Spanish authors in Computer Science. He has published more than 150 refereed scientific papers and monographs and has given numerous keynotes and invited talks in major conferences in these areas. He has also been coordinator and/or principal investigator of many national and international projects, area editor of several journals, and chair and PC member of a large number of conferences. He served as General Director for the research funding unit in Spain, as well as member of the European Union's high-level advisory group in information technology (ISTAG), and of the board of directors of the Spanish Scientific Research Council and the Center for Industrial and Technological Development, among other national and international duties.

Research Interests

His main areas of interest include programming language design and implementation; abstract interpretation-based program analysis, verification, debugging and optimization; logic and constraint programming; parallelizing compilers; parallel and distributed processing.

Manuel Carro

Associate Professor and Deputy Director.

Manuel Carro received his Bachelor degree in Computer Science from the Technical University of Madrid (UPM), and his PhD degree from the same University in 2003. He is currently Associate Research Professor and Deputy Director at the IMDEA Software Institute, and an Associate Professor at the Technical University of Madrid. He has previously been representative of UPM at the NESSI and INES technological platforms, and is now representative of UPM at SpaRCIM and deputy representative of IMDEA Software at ERCIM and Informatics Europe. He has published over 70 papers in international conferences and journals, some of which merited the "Best Conference Paper" award. He has been organizer and PC member of many international conferences and workshops and participated in research projects at the regional, national, and European level. He is UPM's principal investigator for the S-Cube European Network of Excellence. He has completed the supervision of two PhD thesis and is actively supervising two more.

Research Interests

His interests span several topics, including the design and implementation of high-level (logic- and constraint-based) programming languages for improving the quality of software production, the analysis of service-based systems, and the effective usage of formal specifications in the process of teaching programming. He has long been interested in parallel programming and parallel implementations of declarative languages, and visualization of program execution.

Gilles Barthe

Professor

Gilles Barthe received a Ph.D. in Mathematics from the University of Manchester, UK, in 1993, and an *Habilitation à diriger les recherches* in Computer Science from the University of Nice, France, in 2004. He joined the IMDEA Software Institute in April 2008. Previously, he was head of the Everest team on formal methods and security at INRIA Sophia-Antipolis Méditerranée, France, and a member of the Microsoft Research-INRIA Joint Centre. He also held positions at the University of Minho, Portugal; Chalmers University, Sweden; CWI, Netherlands; University of Nijmegen, Netherlands. He has published more than 100 refereed scientific papers. He has been coordinator/principal investigator of many national and European projects, and served as the scientific coordinator of the FP6 FET integrated project "MOBIUS: Mobility, Ubiquity and Security" for enabling proof-carrying code for Java on mobile devices (2005-2009). He has been a PC member of many conferences (CSF, ESORICS, FM, ICALP, ITP...), and served as PC (co-)chair of VMCAI'10, ESOP'11, FAST'11, and SEFM'11. He is a member of the editorial board of the Journal of Automated Reasoning.

Research Interests

Gilles' research interests include formal methods, programming languages and program verification, software security, and cryptography, and foundations of mathematics and computer science. His most recent research focuses on the automated certification of cryptographic schemes, and on correctness and security analysis of Java bytecode.

Anindya Banerjee

Professor

Anindya Banerjee received his PhD from Kansas State University, USA, in 1995. After his PhD, Anindya was a postdoctoral researcher, first in the Laboratoire d'Informatique (LIX) of École Polytechnique, Paris and subsequently at the University of Aarhus. He joined the IMDEA Software Institute Institute in February 2009 as Full Professor. Immediately prior to this position, Anindya was Full Professor of Computing and Information Sciences at Kansas State University, USA. He was an Academic Visitor in the Advanced Programming Tools group, IBM T. J. Watson Research Center in 2007 and a Visiting Researcher in the Programming Languages and Methodology group at Microsoft Research in 2007–2008. He was a recipient of the Career Award of the US National Science Foundation in 2001. He is an associate editor of the journal Higher-Order and Symbolic Computation.

Research Interests

Anindya's research interests lie in language-based computer security, program analysis and verification, program logics, concurrency, programming language semantics, abstract interpretation and type systems. His primary research activities over the past couple of years have centered around automatic and interactive verification of properties of pointer-based programs and in verification of security properties of such programs.

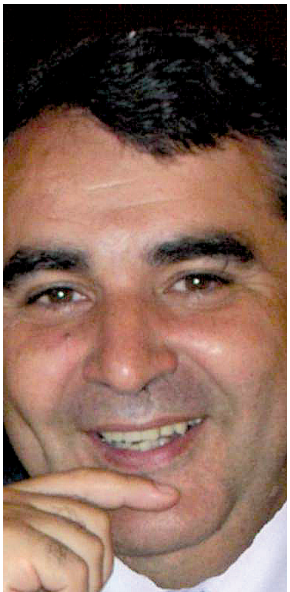
Juan José Moreno-Navarro

Professor (on leave)

Juan José Moreno-Navarro received his Ph.D. degree in Computer Science from the Technical U. of Madrid (UPM), Spain in 1989. He developed a large part of his Ph.D. at RWTH Aachen (Germany). He has been Full Professor at the UPM Computer Science Department since 1996 and joined the IMDEA Software Institute upon its foundation. He served as Deputy Director up to 2008. He has published more than 100 papers in international conferences, books, and journal publications. He has also participated in several EU-funded and other national and international research projects, founding, leading, and ensuring continuous funding for the BABEL research group for more than 17 years. He has organized, served in program committees, and given invited talks and tutorials in many conferences in the IST field. He is a member of the editorial board of the Electronic Journal of Functional and Logic Programming. He also coordinated the first Erasmus Mundus Master taught in Spain. He has been the founding director of SpaRCIM, the Spanish research consortium in Informatics and Mathematics. He has been responsible for the ICT research program, in charge on International Relations for IST, FP6 IST Committee member, and Spanish National Contact Point for the Spanish Ministry of Education and Science. He has also been the Spanish representative at the ICT COST Committee as well as COST-ICT liaison and observer from ERCIM at the European Science Foundation, vice-chair of the Spanish Society for Software Engineering, and vice-chair of the Spanish Technology Platform on Software and Services INES. Prof. Moreno-Navarro was during 2011 on leave as Director General for University Policies at the Spanish Ministry of Education.

Research Interests

His research interests include all aspects related to declarative and rigorous software development technologies. This includes software development techniques, specially specification languages, automatic generation of code (programming from specifications), and software services description. His interests also include declarative languages (functional and logic programming) and, specially the integration of functional and logic programming, including the expressiveness of such these languages for real world applications. He has led the design and implementation of the language BABEL and now takes part in the activities of the international committee involved in the design of the new language Curry.



John Gallagher

Professor (part-time)

Ph.D. (Computer Science) degrees from Trinity College Dublin in 1976 and 1983 respectively. He held a research assistantship in Trinity College (1983-4), and post-doc appointments at the Weizmann Institute of Science, Israel (1987-1989) and Katholieke Universiteit Leuven, Belgium (1989). From 1984-1987 he was employed in research and development in a software company in Hamburg, Germany. Between 1990 and 2002 he was a lecturer and later senior lecturer at the University of Bristol, UK. Since 2002 he has been a professor at the University of Roskilde, Denmark, where he is leader of the research group Programming, Logic and Intelligent Systems and the Experience Lab as well as (part-time) Professor and holds a dual appointment at the IMDEA Software Institute since February 2007. He is a member of the executive committee of the Association of Logic Programming (2008-2011) and of the steering committee of the ACM SIGPLAN workshop series on Partial Evaluation and Program Manipulation (PEPM). He is an area editor for the journal Theory and Practice of Logic Programming. He has published approximately 50 peer-reviewed papers which have over 1200 citations.

Research Interests

His research interests focus on program transformation and generation, constraint logic programming, rewrite systems, temporal logics, semantics-based emulation of languages and systems, analysis and verification of energy consumption of programs and other properties, and has participated in and led a number of national and European research projects on these topics.

Manuel Clavel

Associate Professor (Deputy Director until April 2011).

Manuel Clavel received his Bachelor's degree in Philosophy from the Universidad de Navarra in 1992, and his Ph.D. from the same university in 1998. Currently, he is Deputy Director and Associate Research Professor at the IMDEA Software Institute, as well as Associate Professor at the Universidad Complutense de Madrid. During his doctoral studies, he was an International Fellow at the Computer Science Laboratory of SRI International (1994-1997) and a Visiting Scholar at the Computer Science Department of Stanford University (1995-1997). His Ph.D. dissertation was published by the Center for the Study of Language and Information at Stanford University. Since then, he has published over 30 refereed scientific papers. He has also been involved in the supervision of 3 Ph.D. students (1 completed).

Research Interests

His research focuses on rigorous, tool-supported model-driven software development, including: modeling languages, model transformation, model quality assurance, and code-generation. Related interests include specification languages, automated deduction, and theorem proving.

César Sánchez

Assistant Professor

César Sánchez received his Ph.D. degree in Computer Science from Stanford University, USA, in 2007, studying formal methods for distributed algorithms. After a post-doc at the University of California at Santa Cruz, USA, César joined the IMDEA Software Institute in 2008, becoming a Scientific Researcher at the Spanish Council for Scientific Research (CSIC) in 2009. He holds a degree in Ingeniería de Telecomunicación (MSEE) from the Technical University of Madrid (UPM), Spain, in 1998. Funded by a Fellowship from La Caixa, he moved to Stanford University, USA, receiving a M.Sc. in Computer Science in 2001, specializing in Software Theory and Theoretical Computer Science. César is a recipient of the 2006 ACM Frank Anger Memorial Award. He keeps active collaborations with research groups in the USA and Europe.

Research Interests

César's research activities focus on formal methods for reactive systems with emphasis on the development and verification of concurrent, embedded and distributed systems. His foundational research includes the temporal verification of concurrent datatypes, runtime verification, and enhancements of linear temporal logics. In parallel, he is collaborating with industrial partners from the aerospace and embedded sectors to aid in the adoption of formal techniques for software development and validation. Current projects include the interactive formal generation of parallel software for satellite image processing, and the synthesis of advanced online debuggers for testing embedded software.

Pierre Ganty

Assistant Professor

Pierre joined the IMDEA Institute in September 2009 after completing a nearly two year postdoc at the University of California, Los Angeles (UCLA). He holds a joint PhD degree in Computer Science from the University of Brussels (ULB), Belgium and from the University of Genova (Unige), Italy that he obtained late 2007. Prior to his PhD, he completed a master and a DEA in computer science from the ULB that he obtained in 2002 and 2004, respectively. During his postdoc, Pierre has been nominated for a campus wide UCLA Chancellor's Award for Postdoctoral Research (15 nominees/1089 postdoctoral scholars).

Research Interests

Pierre's research studies automated analysis techniques for systems with infinitely many states. Many systems are, by nature, infinite and cannot be modeled precisely with finitely many states. Of particular interests are concurrent systems like multithreaded programs or communication protocols or event-based programs. In each of the above classes of systems, there is an unbounded dimension: the number of threads, the number of participants or the number of events; which is best modeled using an infinite state system. In theory, the analysis of such systems is infeasible unless some precision is lost. In his previous works, he defined over approximation analysis techniques which are useful to prove properties on such systems. His current research has a strong emphasis on complementary under approximation techniques which do not offer complete coverage but are relevant to catch bugs in those systems.

Aleks Nanevski

Assistant Professor

Aleks received his Ph.D. degree in Computer Science from Carnegie Mellon University, USA in 2004. After holding postdoctoral positions at Harvard University (USA), and Microsoft Research, Cambridge (UK), Aleks joined the IMDEA Software Institute in September 2009. Prior to the PhD, Aleks finished his undergraduate studies in Computer Science at the University of Skopje, Macedonia in 1995.

Research Interests

Aleks' research is in the design and implementation of programming languages that facilitate verification of various program properties, ranging from type and memory safety, lack of memory leaks or information leaks, all the way to full functional correctness. His languages and systems unify programming and specification with automated and interactive theorem proving, via a common foundational framework of type theory. He is particularly interested in verifying programs that combine modern higher-order linguistic features such as higher-order functions, polymorphism, abstract types, objects and modules, with imperative ingredients such as pointer arithmetic, pointer aliasing, unstructured control flow, and concurrency.

Boris Köpf

Assistant Professor

Boris joined the IMDEA Software Institute in September 2011 after completing a postdoc at the Max Planck Institute for Software Systems (MPI-SWS). He received a Ph.D. degree from ETH Zurich in 2007, investigating formal methods for countering sidechannel attacks. Before that, he studied mathematics at the Universidad de Chile, the Universidade Federal de Campinas, and the University of Konstanz, from which he received a M.Sc. degree. He is an alumnus of the German National Academic Foundation.

Research Interests

Boris' research focuses on the foundations of computer security. In particular, he is interested in quantitative notions of security, and in techniques for computing corresponding guarantees for real systems. He applies his research to the analysis of side-channel attacks (and countermeasures) and to privacy-preserving data publishing.

Alexey Gotsman

Assistant Professor

Alexey Gotsman received his Ph.D. degree in Computer Science from University of Cambridge, UK in 2009. During his Ph.D. studies, Alexey interned at Microsoft Research Cambridge, UK and Cadence Berkeley Labs, USA. He was a postdoctoral fellow at Cambridge before joining IMDEA in September 2011. Prior to his Ph.D., Alexey did his undergraduate and master studies in Applied Mathematics at Dnepropetrovsk National University, Ukraine, interning at the University of Trento, Italy in the process.

Research Interests

Alexey's research interests are in software verification, with particular focus on concurrent systems software. He is interested in developing both logics for reasoning about programs and automatic tools for verifying them. Alexey's research activities include development of such logics and tools for concurrent programs with data structures, liveness properties, and operating systems.

Juan Caballero

Assistant Professor

Juan Caballero joined the IMDEA Software Institute as an Assistant Research Professor in November 2011, after receiving his Ph.D degree in Electrical and Computer Engineering from Carnegie Mellon University, USA. Prior to joining the IMDEA Software Institute, Juan was a visiting graduate student researcher at University of California, Berkeley for two years. He was awarded the La Caixa fellowship for graduate studies in 2003. Juan also holds a M.Sc. in Electrical Engineering from the Royal Institute of Technology (KTH), Sweden, and a Telecommunications Engineer degree from the Technical University of Madrid (UPM), Spain.

Research Interests

Juan's research focuses on computer security, including security issues in systems, software, and networks. He enjoys designing program analysis techniques, specially techniques that work directly on program binaries. He applies those techniques for analyzing security properties of benign programs, as well as for malware analysis. In addition, he is interested in network security, the economic aspects of cybercrime, applying machine learning for security, and software engineering.

Pedro López-García

Researcher

Pedro López-García received a MS degree and a Ph.D. in Computer Science from the Technical University of Madrid (UPM), Spain in 1994 and 2000, respectively. In May 28, 2008 he got a Scientific Researcher position at the Spanish Council for Scientific Research (CSIC) and joined the IMDEA Software Institute. Immediately prior to this position, he held associate and assistant professor positions at UPM and was deputy director of the Artificial Intelligence unit at the Computer Science Department. He has published about 30 refereed scientific papers (50% of them at conferences and journals of high or very high impact.) He has also been coordinator of the international project ES_PASS and participated as a researcher in many other national and international projects.

Research Interests

His main areas of interest include automatic analysis and verification of non-functional program properties such as resource usage (user defined, energy, execution time, memory, etc.), non-failure and determinism; performance debugging; abstract interpretation; energy-aware software engineering; (automatic) granularity analysis/control for parallel and distributed computing; profiling; combined static/dynamic verification and unit-testing; type systems; constraint and logic programming.

Laurent Mauborgne

Researcher

Laurent Mauborgne received his Ph.D. in Computer Science from École Polytechnique, France, in 1999, and an Habilitation a diriger les recherches from University Paris-Dauphine (France) in 2007. He has been assistant professor at École normale supérieure, Paris, since 2000, and associate director of computer science studies there since 2006. He was also part-time professor at École Polytechnique. He was invited to spend a year at the IMDEA Software Institute in August 2009.

He published 16 refereed papers in international conferences and 3 papers in journals. He gave courses in research summer schools and participated in the European projects DAEDALUS and ES_PASS. He was program committee member of the Static Analysis Symposium for 4 years. He is one of the authors of the Astrée analyzer, a tool that proved the absence of run-time errors in critical avionic codes.

Research Interests

The research of Laurent Mauborgne is focused on static analysis of programs and abstract interpretation. The goal is to develop theoretical as well as practical tools to analyze the behaviors of programs. This includes proving safety or temporal properties, optimizing compilation and computing resource usage. Among the recent subjects, he studied the cooperative combination of analyzes in different frameworks.

Mark Marron

Researcher

He joined the IMDEA Software Institute as a postdoctoral researcher in June 2008. Following four months as a Visiting Researcher at Microsoft Research in Redmond he returned to IMDEA as a Researcher. Recent research highlights include the release of a robust and scalable heap analysis toolkit (Jackalope analysis tools) for public use and the award of a prestigious Microsoft Innovation Award for work on heap analysis and memory use.

Research Interests

His research interests are on developing practical techniques for modeling program behavior and using this information to support error detection and optimization applications. His work to date has focused on the development of static analysis for the program heap which infers region, sharing, footprint and heap based data dependence information. More recent work has focused on using the information extracted by the analysis to support program parallelization, memory management, error detection, and software engineering applications.

postdoctoral researchers

César Kunz

Postdoctoral Researcher

César Kunz received a Computer Science degree from the National University of Córdoba (UNC), Argentina in 2004. He continued his studies at INRIA, France, funded by the FP6 FET integrated project «MOBIUS: Mobility, Ubiquity and Security», and received a Ph.D. from the École des Mines de Paris (ENSMP), France in February, 2009. He joined the IMDEA Software Institute as a postdoctoral researcher in February 2009.

Research Interests

His research interests lie around formal program analysis and verification, abstract interpretation, and program transformation. His primary research activities are centered on the certification of program correctness, the verification of compiler optimizations, and the transformation of verification results in the presence of program transformations.

Alexander Malkis

Postdoctoral researcher

Alexander has obtained his Diploma degree from the University of Saarland, Germany, in 2004-2005, for a work on polyforms (in other terminology, bond animals) under the guidance of Prof. Dr. Raimund Seidel; during his studies Alexander was financed by the prominent foundation "Studienstiftung des deutschen Volkes". He continued his studies in Saarbruecken and Freiburg, funded by the Max-Planck society and the DFG (German science foundation), obtaining his PhD thesis in 2011 at the University of Freiburg for a work on verification of multithreaded programs under guidance of Prof. Dr. Andreas Podelski. In April 2011, he joined the IMDEA Software Institute.

Research Interests

There is a range of topics in which Alexander is interested in, among them: polynomial verification of large program classes; emptiness of language intersection (complexity and algorithms); thread simulations, liveness, procedure abstractions under concurrency; a working verifier for multithreaded C; verifying multithreaded programs with rich structure and semantics, e.g. with heap, probabilism, recursion, for multicore systems; modeling biological and social systems; and synthesis of multithreaded embedded software.

José Francisco Morales

Postdoctoral researcher

Jose F. Morales joined the IMDEA Software Institute as a postdoctoral researcher in November 2011, after receiving his Ph.D degree in Computer Science from the Technical University of Madrid (UPM), Spain. Previously, he held a teaching assistant position at the Universidad Complutense de Madrid, starting in 2005. Jose's work to date has focused on mechanisms for the efficient execution of logic programs: inference of program properties by abstract interpretation, highly-optimizing translation to low-level code using those properties, and the development of abstractions for the specification and automated construction of abstract machines.

Research Interests

His current research interests include the design of multiparadigm languages (combining imperative, logic, functional, and object-oriented programming), assertion languages and type systems, abstract interpretation, abstract machines, compiler optimizations, and native code generation.

Ruy Ley Wild

Postdoctoral researcher

Ruy Ley-Wild joined the IMDEA Software Institute as a postdoctoral researcher in December 2011. He received his Ph.D. degree in Computer Science from Carnegie Mellon University under the supervision of Guy Blelloch. During his Ph.D. studies, he was funded by a Bell Labs Graduate Research Fellowship and interned at Bell Labs, Toyota Technological Institute at Chicago, and Microsoft Research Cambridge.

Research Interests

Ruy is broadly interested in the design and implementation of programming languages that express computation at a suitable level of abstraction and logics that enable high-level reasoning about the correctness and complexity of such programs. In particular, he has worked on compilation, cost semantics, and high-level dependence-tracking for self-adjusting computation. He is currently working with Aleks Nanevski on a type-theoretic approach to semantics and logics for a higher-order, stateful, concurrent language.

Noam Zeilberger

Postdoctoral Researcher

Noam Zeilberger joined IMDEA Software in October 2011 as a postdoctoral researcher. Previously, he held a two-year postdoctoral fellowship of the *Fondation Sciences Mathématiques de Paris*, working at Université Paris 7. He obtained his Ph.D. in May 2009 from the Computer Science Department of Carnegie Mellon University, under the supervision of Peter Lee and Frank Pfenning.

Research Interests

Noam is interested broadly in the connections between logic and language and computation, and is excited by the potential of type theory (and its twin sister category theory) as a common foundation for (and a means of facilitating communication between) different areas of computer science. His work has focused on the Curry-Howard correspondence in general, and more specifically on: the problem of side-effects; continuations and computational duality; linear logic and focalisation; refinement types and dependent types. Since joining IMDEA and working with Gilles Barthe, he has also become interested in the notion of zero-knowledge from cryptography/complexity theory, and how it relates to notions of knowledge from proof theory.

Zoé Drey

Postdoctoral Researcher

Zoé Drey joined the IMDEA Software Institute as a postdoctoral researcher in October 2011. She received a Ph.D. degree in Computer Science from the University of Bordeaux 1, France, under the supervision of Charles Consel. Before joining IMDEA, she held a full-time teaching assistant position at the ENSEEIHT engineering school in Toulouse, France. Her Ph.D. studies were focused on making accessible the programming task in the field of networked entity orchestration, by providing adapted tools and methodologies to ease the development of reliable applications.

Research Interests

Her interests revolve around the design and implementation of domain-specific languages which reconcile language usability and reliability of the developed programs. In particular, she is interested in combining programming language semantics, logic and functional programming techniques, as well as software engineering methodologies to address this challenge. She is currently exploring the ways to make both the development process and the verification more usable to non-expert programmers, by adapting existing static analysis techniques to ease the instrumentation of domain-specific languages with debugging and verification tools (e.g., by automatically specializing existing debuggers/analyzers for user-friendly error reporting, and/or by providing high-level interfaces to existing specification languages)

Marina Egea

Postdoctoral Researcher

Marina Egea is holding a postdoctoral position at IMDEA Software Institute. Previously she held a postdoctoral position in the Information Security Group at ETH Zurich. She received her doctoral degree in Computer Science at the University Complutense of Madrid in 2008. Her thesis proposes an executable formal semantics for a significant subset of OCL, which is based on a novel mapping from UML models with OCL expressions to equational theories which are proved to be Church-Rosser and terminating and are shown to allow rigorous analysis and validation of the corresponding model. She received her bachelor degree in Mathematics from the University of Granada in 2001, and her Master Thesis from the University Complutense of Madrid in 2005 by the Department of Computer Science.

Research Interests

Her research focuses on the use of formal methods for improving the quality of software engineering products. She is actively involved in the development of a research line on rigorous, tool-supported modeling and validation of software systems. Some of her recent and current research focuses on integrating security policies in system design models and automatically analyzing the resulting model, automatically transforming system design models (including security properties) in a provably correct way, and on bridging the gap between the software design and deployment by helping the fully automation of the code generation.

Santiago Zanella Béguelin

Postdoctoral researcher

Santiago Zanella Béguelin obtained his degree in Computer Science from Universidad Nacional de Rosario (UNR), Argentina in 2006. He received his Ph.D. degree from École Nationale Supérieure des Mines de Paris in 2011 under the supervision of Gilles Barthe. From 2006 to 2011 he was a member of the *Secure Distributed Computations and their Proofs* team at the Microsoft Research-INRIA Joint Centre, Paris. He joined IMDEA in November 2009.

Research Interests

His main areas of interest include program specification and verification, quantitative analysis of programs, security proofs of cryptographic systems, language-based security, and proof assistants. Santiago has devised novel program logics and programming language techniques that can be used to establish the security of cryptographic systems with an unprecedented level of assurance, making a jump from qualitative to quantitative guarantees, and from informal arguments to fully formalized, independently verifiable proofs. These ideas have been realized in the CertiCrypt framework, and applied to obtain certified security proofs of prominent and practically-relevant cryptographic systems, such as the Optimal Asymmetric Encryption Padding (OAEP) scheme.

He is currently working on developing automated tools to bring verification of security of cryptographic systems to practice, using off-the-shelf SMT solvers and automated theorem provers.

visiting faculty

David Naumann

Visiting Professor

Stevens Institute of Technology
Visiting during Mar. 2011–May 2011

Martin Wirsing

Visiting Professor

Ludwig-Maximilians Universität München
Visiting during Sep. 2011–Dec. 2011

research assistants

ph.d. students

Alvaro García

Research Assistant

Degree: Technical University of Madrid (UPM), Spain
Research: Functional Programming, Lambda Calculus, Type Theory, Program Transformation and Abstract Machines.

Miguel Angel García de Dios

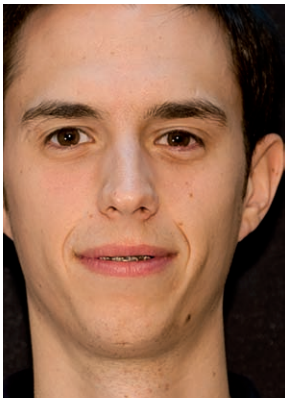
Research Assistant

Degree: Universidad Complutense de Madrid (UCM), Spain
Research: Formal specification and verification, and rigorous tool supported modeling and validation of software systems.

Julian Samborski-Forlese

Research Assistant

Degree: Universidad Nacional de Rosario (UNR), Argentina
Research: Applications of formal methods and abstract interpretation to program verification; quantum computing; functional programming languages; semantics.



Juan Manuel Crespo

Research Assistant

Degree: Universidad Nacional de Rosario (UNR), Argentina

Research: Programming language semantics, type theory, functional programming, category theory, logic and software verification.

Federico Olmedo

Research Assistant

Degree: Universidad Nacional de Rosario (UNR), Argentina

Research: Verification of cryptographic systems and semantics of programming languages

Alejandro Sánchez

Research Assistant

Degree: Universidad Nacional de Córdoba (UNC), Argentina

Research: Formal methods, program verification, dynamic memory analysis, concurrent systems, type theory, functional programming.

Carolina Inés Dania

Research Assistant

Degree: Universidad Nacional de Córdoba (UNC), Argentina

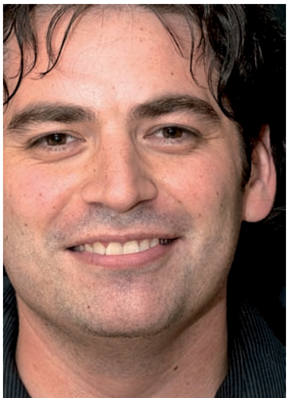
Research: Tool-supported model-driven software development. Oriented on formal specification languages, security models, transformation and code generation.

Javier Valdazo Parnisari

Research Assistant

Degree: Universidad Nacional de Córdoba (UNC), Argentina

Research: Formal specification and verification. Rigorous tool supported modeling and validation of software systems. Model driven software engineering. Model transformations. Security models, transformation and enforcement.

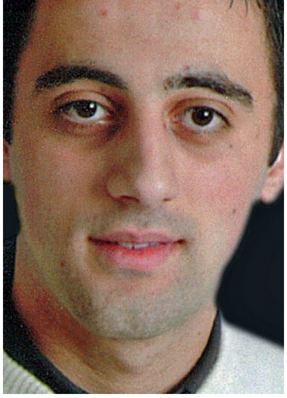


Germán Andrés Delbianco

Research Assistant

Degree: Universidad Nacional de Rosario (UNR), Argentina

Research: Theorem proving and software verification. Programming languages semantics, type theory, functional programming and the application of concepts from category theory to Computer Science.



Umer Liqat

Research Assistant

Degree: Dresden University of Technology (TUD), Germany
Research: Program analysis and verification, Automatic analysis and verification of (user definable) resource usage. In particular energy consumption analysis and optimization. Constraint and Logic programming.

Teresa Trigo

Research Assistant

Degree: Technical University of Madrid (UPM), Spain
Research: Software verification techniques based on static analysis and its application to embedded systems. Resource usage analysis and automatic parallelization.

Antonio Artés

Research Assistant

Degree: Technical University of Madrid (UPM), Spain
Research: Power-aware, temperature-aware and reliability-aware design of low power semiconductor devices.

interns

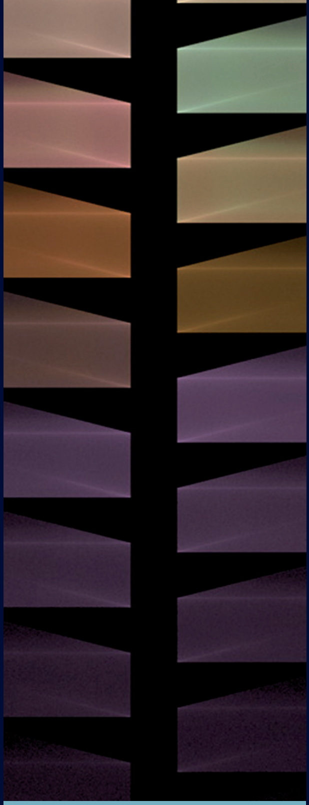
Intern	Period	Nationality
Lufthi Darmawan	Dec. 2010 - Dec. 2011	Indonesia
Tomas Poch	Dec. 2010 - Jan. 2011	Czech Republic
Germán Delbianco	Jan. 2011 - Jul. 2011	Argentina
Exequiel Rivas	Jan. 2011 - Jul. 2011	Argentina
Goran Doychev	Mar. 2011 - May. 2012	Germany
Clementina Latanzi	Mar. 2011 - Jul. 2011	Argentina
Florence Clerc	Apr. 2011 - Aug. 2011	France
Remi Geraud	Apr. 2011 - Aug. 2011	France
Santiago Gonzalez	May. 2011 - Sep. 2011	Argentina
Gustavo Grieco	May. 2011 - Sep. 2011	Argentina
James Gordon Stewart	Jun. 2011 - Aug. 2011	USA
Adrián Silveira	Aug. 2011 - Nov. 2011	Uruguay
Stephan Max	Sep. 2011 - Oct. 2011	Germany
Bishesh Adhikari	Oct. 2011 - Dec. 2011	Belgium
Guido Genzone	Nov. 2011 - Apr. 2012	Argentina

administration

& IT support

Researchers at the IMDEA Software Institute are provided with adequate administrative and technical support such that they can concentrate their efforts on scientific activities. Our administrative and technical support staff is currently co-funded by different projects.

María Alcaraz	General Manager	full time	MBA, MSc. Economics
Marta Sedano	Technology Manager	full time	BA (Hons) Business
Paola Huerta	Assistant	full time	MSc. History
Tania Rodríguez	Assistant	part time	MSc. Economics
Juan Céspedes	System Administrator	part time	MSc. Electrical Engineering
Inés Huertas	System Administrator	part time	Bach. Telematics



5 research projects and contracts

Projects funded by national or international funding agencies or directly through contracts with industry are an important framework within which the IMDEA Software Institute funds and carries out research activities and technology transfer to industry. The Institute is currently participating in a number of such projects, contracts, grants, and fellowships which are briefly summarized in this chapter.

5.1. Ongoing Projects

HATS

Highly Adaptable and Trustworthy Software using Formal Models

Funding: European Union, FET Proactive Call "Forever Yours" – 7th Framework Program
Duration: 2009-2013
Principal Investigator: Prof. Gilles Barthe

HATS is an Integrated Project funded by the European Union within the 7th Framework Program. The main outcome envisaged by this project is an integrated architectural framework and a methodology for rigorous development of highly adaptable and trustworthy software. The IMDEA Software Institute is one of the research centers in a consortium of 8 academic partners, 2 industrial research centers, and 1 SML, from 7 countries. The budget for the project is approximately 6 M Euros.

Specifically, HATS will turn software product family (SWPF) development into a rigorous approach. The technical core of the project is an Abstract Behavioral Specification language which will allow precise description of SWPF features and components and their instances. The main project outcome is a methodological and tool framework achieving not merely far-reaching automation in maintaining dynamically evolving software, but an unprecedented level of trust while informal processes are replaced with rigorous analyses based on formal semantics.

The IMDEA Software Institute is responsible for the development of a highly adaptable architecture that allows cost-effective verification of the executable programs that will be automatically generated from Abstract Behavioral Specifications. The security architecture will be specifically directed towards security policies expressed using information flow and functional correctness policies.

SEVENTH FRAMEWORK PROGRAMME

SIEMENS

AtoS

NESSoS

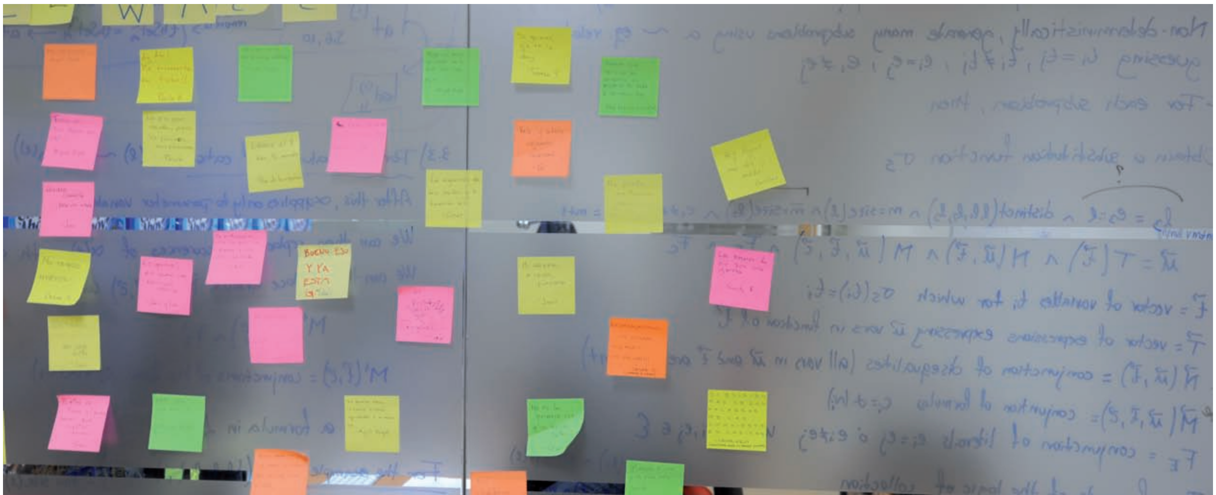
Network of Excellence on Engineering Secure Future Internet Software Services and Systems

Funding: European Union, Cooperation Program (NoE) – 7th Framework Program
Duration: 2011-2013
Principal Investigator: Prof. Manuel Clavel

The Network of Excellence on Engineering Secure Future Internet Software Services and Systems (NESSoS) aims at constituting and integrating a long lasting research community on engineering secure software-based services and systems. The NESSoS consortium involves 12 partners, including 2 companies (namely, Siemens and ATOS), from 7 countries. The budget for the project is approximately 3.5 M Euros.

The domain of Engineering Secure Software Services covers a collection of engineering activities that aim at the creation of software services —i.e. ICT services delivered through the deployment of software systems— that are both behaviorally correct (typically guided by software engineering principles) as well as secure (typically guided by security engineering principles). The approach of engineering secure software services is based on the principle of addressing security issues from the very beginning in system design and analysis, thus contributing to reducing system and service vulnerabilities, improving the necessary assurance level, thereby considering risk and cost issues during development in order to prioritize investments.

IMDEA Software plays a prominent role in three research workpackages: secure service architectures and design; programming environments for secure and composable services; and security assurance for services. Also, IMDEA Software leads the researcher mobility program within the consortium. This program is a mechanism that supports the integration of activities across the various sites: it brings together researchers working on related topics; it drives knowledge exchange and knowledge generation through union and diversity; and, finally, it increases the capability of joint cooperation among researchers.



DESAFIOS-10

High-Quality, Reliable, Distributed, and Secure Software Development

Funding: Spanish Ministry of Science and Innovation
Duration: 2011-2013
Principal Investigator: Prof. Gilles Barthe

The overall goal of the DESAFIOS-10 project is to contribute both foundations and technologies helpful in the development of software systems with certified quality and reliability, typically based on formal methods and declarative programming. The consortium involves groups from three different Institutions (Universidad Complutense de Madrid, Universidad Politécnica de Madrid, and IMDEA Software) and a number of industrial users.

This project arises as a natural evolution of previous coordinated project DESAFIOS, which involved only the research groups from the Universidad Complutense de Madrid and the Universidad Politécnica de Madrid. However, DESAFIOS-10 emphasizes the security and reliability aspects of this research, which is precisely the workpackage led by IMDEA Software.

PROMETIDOS

Methods for Rigorous Software Development

Funding: Regional Government of Madrid
Duration: 2011-2013
Principal Investigator: Prof. Gilles Barthe

The PROMETIDOS-CM research program is focused on four main areas: specification and validation, to provide a solid foundation for the description and analysis of services; reliability and security, to guarantee robust solutions from start to end; declarative programming, to develop the next generation of languages for services; and efficiency, to optimize quality of service with respect to performance. A common goal for all these research lines is the development of tools that will rigorously support their scientific results and that could be eventually transferred to industry.

PROMETIDOS-CM is a consortium involving groups from Universidad Complutense de Madrid, Universidad Politécnica de Madrid, and the IMDEA Software Institute, which is the project coordinator.

GOBIERNO DE ESPAÑA
MINISTERIO DE ECONOMÍA Y COMPETITIVIDAD

PARAN-10

Parametrized Verification of Computing Systems

Funding: Spanish Ministry of Science and Innovation Duration: 2011-2012
Principal Investigator: Pierre Ganty

This project aims at developing novel techniques for production, verification and certification of computing systems where *parameters* play an essential role. Parameters either at the level of the system specification or at the level of the verification technique make it possible to address scalability and undecidability issues. However, specification and verification in the presence of parameters are highly non-trivial, and pose problems for automated verification methods (such as model checking) as well as interactive approaches to computing systems verification (such as theorem proving), both of which are relevant in practice.

The project is organized along three research lines: model-checking of parametrized systems, parametric model-checking, and programming languages and logics for parametrization. In these three lines the project aims at making fundamental contributions to advance the state of the art as well as develop prototype implementations in order to explore and demonstrate the practical relevance of the proposed approaches.

cost
EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

EUROPEAN UNION

RMT

Rich-Model Toolkit – An Infrastructure for Reliable Computer Systems (COST Action IC0901)

Funding: European Union, Cost action
Duration: 2011
Principal Investigator: César Sánchez

This initiative explores directions and techniques for making automated reasoning (including analysis and synthesis) applicable to a wider range of problems, as well as making them easier to use by researchers, software developers, hardware designers, and information system users and developers. It includes participants from over 20 countries. A selection of the topics of interest is:

Standardization of expressive languages: Definitions of formats to represent systems, formulas, proofs, counterexamples. A framework to specify translations between specification languages, as well as benchmarks and competitions for automated reasoning, verification, analysis, and synthesis.

Decision procedures: Creation of decision procedures for new classes of constraints, including implementation of SAT and SMT and their certification. This will need the encoding of synthesis and analysis problems into SMT. We will also tackle the encoding of description logics (widely used in the Semantic Web) and the problem of scalable reasoning about knowledge bases.

Transition system analysis: One key of study is the abstraction-based approaches and refinement for verification of infinite-state systems. The application of constraint-based program analysis will also be analyzed, as well as data-flow analysis for complex domains. The application of TSA to programming languages and bytecodes will be explored by extracting transition systems from them.

High-level synthesis: The project will devise new algorithms for synthesis from high-level specifications, and decision procedures will be extended to perform synthesis tasks. A relevant point to explore will the connection between invariant generation and code synthesis.

MTECTEST

New testing techniques for on-board software

Funding: Regional Government of Madrid
Duration: 2011
Principal Investigator: César Sánchez

This project explores several techniques of software testing geared towards embedded systems. These techniques are being selected and tried in collaboration with DEIMOS Space, which participates in ESA projects focusing on software validation and verification. These techniques try to overcome practical limitations of existing approaches to verification, which may be too formal in some cases. They will work directly on executing programs and try to reach a level of accuracy in the tests, control of scenarios, and automation of the verification of the results higher than usual. An additional goal of this project is to study the effectiveness of these techniques when taking into account actual constraints of actual projects, such as development environments and testing frameworks.

AbsInt Gmbh

Funding: AbsInt Gmbh
Duration: 2011-2012
Principal Investigator: Laurent Mauborgne

This project is a contract with AbsInt Angewandte Informatik Gmbh to collaborate in the development of static analyzers by abstract interpretation, coordinated by Laurent Mauborgne. The goal of this contract is to develop advanced abstract interpretation techniques allowing the fine tuning and increasing the precision and efficiency of the Astrée static analyzer sold and maintained by AbsInt. IMDEA Software brings its expertise and advices on sound abstractions of the memory model of the C language and on adaptive relational abstract domain tuning.

AMAROUT Europe

Funding: European Union, Marie Curie Action (PEOPLE-COFUND) - 7th Framework Program
Duration: 2009-2013
General Coordinator: Prof. Manuel Hermenegildo

AMAROUT Europe is a Marie Curie Action (PEOPLE-COFUND) to foster and consolidate the European Research Area by attracting to Europe and, in particular, to the region of Madrid (Spain) top research talent. AMAROUT contributes with IMDEA to the goal of turning Madrid into one of the top knowledge generation regions in Europe. To accomplish this, the AMAROUT program finances up to 132 researchers to join the IMDEA network of research institutes for one year (renewable up to twice). The total budget for the program is around 11 M Euros of which the European Union cofinances 40%.

Both “experienced” and “very experienced” researchers from any country (worldwide) can apply for AMAROUT fellowships at any of the eight IMDEA Institutes participating in the program (Software, Energy, Food, Materials, Nanoscience, Networks, Water, and Social Sciences). The AMAROUT Selection Committee consists of eight Evaluation Panels, one for each of the participating IMDEA Institutes. Each Evaluation Panel is formed by the Director of the Institute, three members of its Scientific Advisory Board, and two external, independent peer reviewers. The main AMAROUT selection criteria is the candidate’s demonstrated ability and commitment to research, as well as the match of experience and interests with the research theme and lines of the IMDEA Institute chosen by the candidate.

The AMAROUT Program is a joint initiative from the eight IMDEA research institutes. The IMDEA Software Institute operates as the proposer and beneficiary. As such, IMDEA Software is also in charge of the project management and its structure: Scientific Committee (SC); Fellowships Management Unit (FMU); Secretary; and Local Board of Prospective (BP). The FMU is responsible for the overall program management. IMDEA Software chairs the project team meetings (quarterly). The FMU is supported in its activities by the Secretary (administration, financial, H&M, welcoming) to fulfill the personnel-related, administrative and financial requirements of the Program and the EC. The secretary is commanded by IMDEA Software. The SC is responsible for the definition of the scientific lines and for the appraisal of the correct implementation of the scientific Program. The IMDEA Software director is the chair of the SC.

NUSA

Numeric and Symbolic Abstractions for Software Model Checking

Funding: The Danish Council for Independent Research - Natural Sciences
Duration: 2011-2013
Principal Investigator: John Gallagher

Danish Agency for Science Technology and Innovation
Ministry of Science, Innovation and Higher Education

UNIVERSITAS ROSKILDENSIS · IN TRANQUILLO MORS · IN FLUCTU VITA

Abstract interpretation and model checking are two approaches to verifying or deriving properties of software and hardware systems. While model checking is applied to finite-state systems (typically hardware), abstract interpretation is usually aimed at infinite-state software systems. Indeed, the very notion of verification by abstraction starts from the assumption that the system under consideration is infinite or very large. Both abstract interpretation and model checking are the subject of major research efforts, both in academic and industrial laboratories, since they hold out the promise of an automatic, push-button approach to obtaining guarantees of system behavior. This proposal lies in the intersection of abstract interpretation and model checking. The main question for investigation in this project is how the framework and accumulated experience of abstract interpretation can be applied to model checking infinite state systems - in short, to define abstract model checking methods that exploit the generality and power of the framework of abstract interpretation.

5.2. Projects with Associated Groups

Part of the research of the Institute is performed in collaboration with research groups at associated institutions. This is exemplified by the existence of research projects led by these institutions but in which IMDEA personnel take part (and the resulting joint publications and results). We provide a summary list of the most relevant such projects which were active during year 2011.

Project	Duration	Description	Funding Agency
S-CUBE	2008-2012	The European network of excellence in software and services	European Union – Network of Excellence
DOVES	2009-2013	Development of verifiable and efficient software	MINECO
SpaRCIM	2003-. . .	Spanish Research Consortium for Informatics and Mathematics	European Union / MINECO

5.3. Recently Granted Projects (not started in 2011)

ENTRA

Whole-systems energy transparency

Funding: European Union - 7th Framework Program
Duration: 2012-2015
Principal Investigator: John Gallagher

ENTRA is an FP7 “Future and Emerging Technologies” project under the proactive “MINECC” objective - “Minimizing Energy Consumption of Computing to the Limit”. The ENTRA project proposes radical advances in energy-aware software design and management that will provide the key to the pervasive realization of energy-aware computing. Though huge advances have been made in power-efficient hardware, most of the potential energy savings are wasted by software that does not exploit energy-saving features of hardware, and by poor dynamic management of tasks and resources.

The project is built around the central concept of *energy transparency* at every stage of the software lifecycle. The project work packages will develop novel *program analysis* and *energy modeling* techniques, making energy usage transparent through the system layers. This will enable *energy optimizations* both during code development and at run-time, and promote energy efficiency to a first-class software design objective.

VARIES

Variability in safety critical embedded systems

Funding: ARTEMIS- European Union - 7th Framework Program
Duration: 2012-2015
Principal Investigator: Laurent Mauborgne

VARIES is an ARTEMIS Joint Undertaking project granted under the FP7 ARTEMIS-2011-1 Call. The 26 partners-strong international consortium includes the participation of national partners Hi-Iberia, Integrasys, and Tecnalia. The main goal of the VARIES project is to help Embedded Systems (ES) developers to maximize the full potential of variability in safety-critical ES. The objectives of this project will be therefore (i) to enable companies to make informed decisions on variability use in safety-critical ES; (ii) to provide effective variability architectures and approaches for safety-critical ES; and (iii) to offer consistent, integrated and, continuous variability management over the entire product life cycle.

The VARIES project will deliver the VARIES Platform: a complete, cross-domain, multi-concern, state-of-the-art reference platform for managing variability in safetycritical ES. Special attention will be given to aspects specific to safety-critical ES, in particular the impact of reuse and composition on certification.

In addition to this ambitious goal, the VARIES project will create a Center of Innovation Excellence (CoIE) for managing variability in ES. The VARIES CoIE will support the European ES industry on the 3 aforementioned objectives.

Microsoft Research Software Engineering Innovation Foundation Awards

The *SEIF (Software Engineering Innovation Foundation)* awards are given by Microsoft to support research in software engineering technologies, tools, practices, and teaching methods. The awards are given to project proposals which can be related to any of the core areas of interest in software engineering and have been given in 2012 for the third time. More than 100 proposals were received this year, among which 10 projects were selected to receive the prize and the associated grant. Out of these 10 selected projects, two have been granted to the IMDEA Software Institute for the following projects:

Mark Marron with the project *MemAlyzer: Finding and Fixing Memory Usage Problems*.
Alexey Gotsman with the project *Specifying and Validating Components on Memory Models of Mobile Platforms*.

The rest of the selected applications were from centers in the US (6), Canada (1 – U. of Calgary) and Switzerland (1 – ETH Zurich).

SEVENTH FRAMEWORK PROGRAMME

COFUNDED BY MARIE CURIE ACTIONS

AMAROUT II Europe

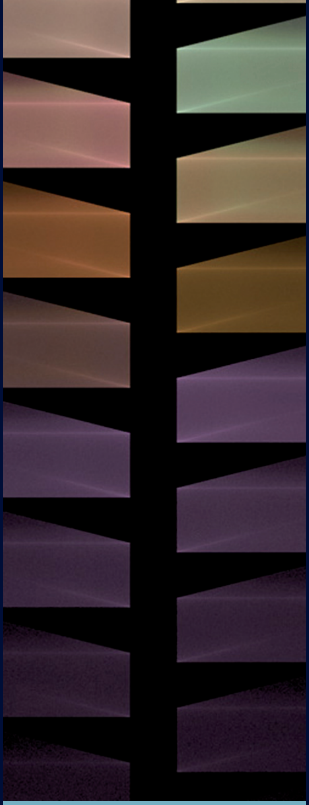
Funding: European Union, Marie Curie Action (PEOPLE-COFUND) - 7th Framework Program
Duration: 20012-2016
General Coordinator: Prof. Manuel Hermenegildo

AMAROUT-II Europe is a Marie Curie Action (PEOPLE-COFUND), a continuation of the AMAROUT project which shares the aim of fostering and consolidating the European Research Area by attracting to Europe and, in particular, to the region of Madrid (Spain) top research talent. As in the previous AMAROUT program "experienced" and "very experienced" researchers from any country (worldwide) can apply for AMAROUT II fellowships at any of the IMDEA Institutes participating in the program. The programme will seek to attract, over 4 years, 152 experienced researchers to carry out their individual research projects within the IMDEA network. The programme will keep a call open permanently between months 1 and 36. Applications will be evaluated by batches, according to quarterly cut-off dates. To promote the programme and its calls, both nationally and abroad, best practices developed during the previous AMAROUT program will be integrated. IMDEA Software will be the mono-beneficiary of the AMAROUT-II programme, the same role that it is currently performing for the previous AMAROUT programme.

5.4. Fellowships

1. *Microsoft Research PhD Scholarship funds*, awarded in 2011, active in 2012-2015. **Alexey Gotsman**.
2. *Juan de la Cierva Postdoc grant*, Spanish Ministry of Science and Innovation, awarded in 2010 and ending in 2014, **César Kunz** (through UPM).
3. *Juan de la Cierva Postdoc grant*, Spanish Ministry of Science and Innovation, awarded in 2011 and ending in 2015, **Juan Caballero**.
4. *Ramón y Cajal grant*, Spanish Ministry of Science and Innovation, awarded in 2010 and ending in 2015, **Aleksandar Nanevski**.
5. *Marie Curie AMAROUT Incoming Fellowships*, European Union – 7th Framework Program, awarded in 2009 and active in 2011. **Aleks Nanevski**, **Pierre Ganty**, and **Laurent Mauborgne**.
6. *Marie Curie AMAROUT Reintegration Fellowships* awarded in 2010 and active in 2011. **Marina Egea** and **Juan Caballero**.
7. *Marie Curie AMAROUT Incoming Fellowships* awarded in 2010 and active in 2011. **Ruy Ley Wild**, **Boris Köpf** and **Alexey Gotsman**.
8. *ERCIM Grants 2011*, European Union, 7th Framework Program. **Noam Zeilberger** (December 2011 – November 2012) and **Zoé Drey** (September 2011 – August 2012).
9. *Predoctoral Grants*, Madrid Regional Government, awarded in 2009 and continuing in 2011. **Álvaro García.**
10. *FPI Doctoral Grant*, Spanish Ministry of Science and Innovation, awarded in 2010 and continuing until 2014. **Juan Manuel Crespo**.

6 dissemination of results



6.1 Publications

6.1.1 Refereed Publications

1. Isabella Mastroeni, *Anindya Banerjee*. *Modelling Declassification Policies using Abstract Domain Completeness.* Mathematical Structures in Computer Science, Vol. 21, Num. 6, pages 1253–1299, December 2011.

2. *Aleksandar Nanevski*, *Anindya Banerjee*, Deepak Garg. *Verification of Information Flow and Access Control Policies with Dependent Types.* IEEE Symposium on Security and Privacy, pages 165–179, 2011.

3. Benjamin Livshits, Aditya V. Nori, Sriram K. Rajamani, *Anindya Banerjee*. *Merlin: Specification Inference for Explicit Information Flow Problems.* Mining Software Specifications, pages 377–410, Chapman & Hall/CRC, May 2011.

4. *Gilles Barthe*, Pedro D'Argenio, Tamara Rezk. *Secure Information Flow by Self-Composition.* Mathematical Structures in Computer Science, Vol. 21, Num. 6, pages 1207–1252, Cambridge University Press, October 2011.

5. *Gilles Barthe*, *César Kunz*. An Abstract Model of Certificate Translation. ACM Trans. Program. Lang. Syst., Vol. 33, Num. 4, pages 1–13, ACM, July 2011.

6. *Gilles Barthe*, *Federico Olmedo*, *Santiago Zanella Béguelin*. *Verifiable Security of Boneh-Franklin Identity-Based Encryption.* 5th International Conference on Provable Security – ProvSec 2011, Lecture Notes in Computer Science, Vol. 6980, pages 68–83, Springer, October 2011.

7. *Gilles Barthe*, Benjamin Grégoire, Sylvain Heraud, *Santiago Zanella Béguelin*. *Computer-Aided Security Proofs for the Working Cryptographer.* Advances in Cryptology – CRYPTO 2011, Lecture Notes in Computer Science, Vol. 6841, pages 71–90, Springer, August 2011. **Best paper award**.

8. *Gilles Barthe*, *Juan Manuel Crespo*, *César Kunz*. *Relational Verification Using Product Programs.* FM 2011: 17th International Symposium on Formal Methods, LNCS, Vol. 6664, pages 200–214, Springer, June 2011.

9. *Gilles Barthe*, Gustavo Betarte, Juan Diego Campo, Carlos Luna. *Formally Verifying Isolation and Availability in an Idealized Model of Virtualization.* FM 2011: 17th International Symposium on Formal Methods, LNCS, Vol. 6664, pages 231-245, Springer, June 2011.

10. *Gilles Barthe*, Benjamin Grégoire, Yassine Lakhnech, *Santiago Zanella Béguelin*. *Beyond Provable Security Verifiable IND-CCA Security of OAEP.* Topics in Cryptology — CT-RSA 2011, Lecture Notes in Computer Science, Vol. 6558, pages 180-196, Springer, February 2011.

11. *Gilles Barthe*, *Exequiel Rivas*. *Static Enforcement of Information Flow Policies for a Concurrent Object-Oriented Language.* Sixth Trusted Global Computing Conference 2011 (TGC'11), LNCS, Springer, 2011.

12. *Gilles Barthe*, *Boris Köpf*. *Information-theoretic Bounds for Differentially Private Mechanisms.* Proc. 24rd IEEE Computer Security Foundations Symposium (CSF'11), pages 191–204, IEEE, 2011.

13. *Juan Caballero*, Chris Grier, Christian Kreibich, Vern Paxson. *Measuring Pay-perInstall: The Commoditization of Malware Distribution.* Proceedings of the 20th USENIX Security Symposium, August 2011. **Best paper award**.

14. Noah M. Johnson, *Juan Caballero*, Kevin Chen, Stephen McCamant, Pongsin Poosankam, Daniel Reynaud, Dawn Song. *Differential Slicing: Identifying Causal Execution Differences for Secu-*

rity Applications. Proceedings of the IEEE Symposium on Security and Privacy, May 2011.

15. Pablo Chico de Guzmán, Amadeo Casas, *Manuel Carro*, *Manuel Hermenegildo*. Parallel Backtracking with Answer Memoing for Independent And-Parallelism*.* Theory and Practice of Logic Programming, 27th Int'l. Conference on Logic Programming (ICLP'11) Special Issue, Vol. 11, Num. 4–5, pages 555–574, Cambridge U. Press, July 2011.

16. Dragan Ivanović, *Manuel Carro*, *Manuel Hermenegildo*. *Constraint-Based Runtime Prediction of SLA Violations in Service Orchestrations.* Service-Oriented Computing – ICSOC 2011, LNCS, Num. 7084, pages 62–76, Springer Verlag, December 2011. **Best paper award**.

17. Dragan Ivanović, *Manuel Carro*, *Manuel V. Hermenegildo*. *Automated Attribute Inference in Complex Service Workflows Based on Sharing Analysis.* Proceedings of the 8th IEEE Conference on Services Computing SCC 2011, pages 120-127, IEEE Press, July 2011.

18. *Manuel Carro*, Dimka Karastoyanova, Grace A. Lewis, Anna Liu. *Third International Workshop on Principles of Engineering Service-Oriented Systems (PESOS 2011).* ICSE, pages 1218–1219, 2011.

19. *Manuel Carro*, *Manuel V. Hermenegildo*. *Logic Languages.* Encyclopedia of Parallel Computing, pages 1057–1068, Springer, 2011.

20. David A. Basin, *Manuel Clavel*, *Marina Egea*. *A Decade of Model-Driven Security.* SACMAT 2011, 16th ACM Symposium on Access Control Models and Technologies, pages 1–10, ACM, June 2011.

21. David A. Basin, *Manuel Clavel*, *Marina Egea*, *Miguel Angel García de Dios*, *Carolina Dania*, *Gonzalo Ortiz*, *Javier Valdazo*. *Model-Driven Development of Security-Aware GUIs for Data-Centric Applications.* Foundations of Security Analysis and Design VI (FOSAD 2010), LNCS, Vol. 6858, pages 101–124, Springer, 2011.

22. *Manuel Clavel*, Narciso Martí-Oliet, Miguel Palomino. *Parameterized Metareasoning in Membership Equational Logic.* Formal Modeling: Actors, Open Systems, Biological Systems - Essays Dedicated to Carolyn Talcott on the Occasion of Her 70th Birthday, LNCS, Vol. 7000, pages 277–298, Springer, 2011.

23. *Juan Manuel Crespo*, *César Kunz*. *A Machine-Checked Framework for Relational Separation Logic.* Software Engineering and Formal Methods - 9th International Conference, SEFM 2011, LNCS, Vol. 7041, pages 122–137, Springer, November 2011.

24. *Germán Andrés Delbianco*, Mauro Jaskelioff, Alberto Pardo. *Applicative Shortcut Fusion.* 12th International Symposium on Trends in Functional Programming, TFP'11, May 2011.

25. Javier Esparza, *Pierre Ganty*, Stefan Kiefer, Michael Luttenberger. Parikh's Theorem: A sim-

ple and direct automaton construction. Information Processing Letters, Vol. 111, pages 614–619, 2011.

26. Mohamed Faouzi Atig, *Pierre Ganty*. *Approximating Petri Net Reachability Along Context-free Traces.* FSTTCS '11: Proc. 31st Int. Conf. on Fondation of Software Technology and Theoretical Computer Science, Leibniz International Proceedings in Informatics (LIPIcs), Vol. 13, Leibniz-Zentrum fuer Informatik, 2011.

27. Laura Bozzelli, *Pierre Ganty*. *Complexity Analysis of the Backward Coverability Algorithm for VASS.* RP '11: Proc. 5th Workshop on Reachability Problems, LNCS, Vol. 6945, Springer, 2011.

28. Javier Esparza, *Pierre Ganty*. *Complexity of Pattern-based Verification for Multithreaded Programs.* POPL '11: Proc. 38th ACM SIGACT-SIGPLAN Symp. on Principles of Programming Languages, pages 499–510, ACM Press, 2011.

29. *Alexey Gotsman*, Hongseok Yang. *Modular Verification of Preemptive OS Kernels.* Proceedings of the 16th ACM International Conference on Functional Programming (ICFP'11), Tokyo, Japan, pages 404–417, ACM Press, 2011.

30. *Alexey Gotsman*, Hongseok Yang. *Liveness-Preserving Atomicity Abstraction.* Proceedings of the 38th International Colloquium on Automata, Languages and Programming, (ICALP'11), Zurich, Switzerland, LNCS, Vol. 6756, pages 453-465, Springer, 2011.

31. *Alexey Gotsman*, Josh Berdine, Byron Cook. *Precision and the Conjunction Rule in Concurrent Separation Logic.* Proceedings of the 27th Conference on the Mathematical Foundations of Programming Semantics (MFPS'11), Pittsburgh, PA, USA, Elsevier, 2011.

32. *Manuel V. Hermenegildo*, Francisco Bueno, *Manuel Carro*, *Pedro López-García*, Remy Haemmerlé, Edison Mera, *José F. Morales*, G. Puebla. *An Overview of the Ciao System.* Proc. of Symposium on Rule-Based Computing (RuleML-Europe 2011), LNCS, Num. 6826, pages 2–3, Springer-Verlag, July 2011.

33. *Manuel V. Hermenegildo*, Francisco Bueno, *Manuel Carro*, *P. López*, Edison Mera, *José F. Morales*, *Germán Puebla*. *The Ciao Approach to the Dynamic vs. Static Language Dilemma.* Proceedings for the International Workshop on Scripts to Programs, STOP'11, 4 pages, ACM, 2011.

34. Germán Puebla, Elvira Albert, *Manuel Hermenegildo*. *Efficient Local Unfolding with Ancestor Stacks.* Theory and Practice of Logic Programming, Vol. 11, Num. 1, pages 1–32, Cambridge U. Press, January 2011.

35. Remy Haemmerlé, *Pedro López*, *Manuel Hermenegildo*. *CLP Projection for Constraint Handling Rules.* Proceedings of the 13th International ACM SIGPLAN Conference on Principles and Practice of Declarative Programming, pages 137-148, ACM Press, July 2011.

36. *José F. Morales*, *Manuel V. Hermenegildo*, Remy Haemmerlé. *Modular Extensions for Modular (Logic) Languages.* 21th International Symposium on Logic-Based Program Synthesis and Transformation (LOPSTR'11), July 2011.

37. Francisco Bueno, María García de la Banda, *Manuel V. Hermenegildo*, *Pedro López-García*, Edison Mera, Peter J. Stuckey. Towards Resource Usage Analysis of MiniZinc Models. MiniZinc Workshop (MZN'11), 15 pages, September 2011.

38. *Pedro López-García*, *Luthfi Darmawan*, Francisco Bueno, *Manuel Hermenegildo*. *Interval-based Resource Usage Verification: Formalization and Prototype.* 2nd International Workshop on Foundational and Practical Aspects of Resource Analysis (FOPARA'2011), May 2011.

39. *Boris Köpf*, David Basin. *Automatically Deriving Information-theoretic Bounds for Adaptive Side-channel Attacks.* Journal of Computer Security, Vol. 1, pages 1-31, 2011.

40. Michael Backes, Matthias Berg, *Boris Köpf. Non-Uniform Distributions in Quantitative Information-Flow.* Proc. 6th ACM Conference on Information, Computer and Communications Security (ASIACCS '11), pages 367–375, ACM, 2011.

41. Patrick Cousot, Radhia Cousot, *Laurent Mauborgne. The Reduced Product of Abstract Domains and the Combination of Decision Procedures.* 14th International Conference on Fondations of Software Science and Computation Structures (FoS- SaCS 2011), Lecture Notes in Computer Science, Vol. 6604, pages 456–472, Springer-Verlag, 2011.

42. Julien Bertrane, Patrick Cousot, Radhia Cousot, Jérôme Feret, *Laurent Mauborgne*, Antoine Miné, Xavier Rival. *Static Analysis by Abstract Interpretation of Embedded Critical Software.* ACM SIGSOFT Software Engineering Notes, Vol. 36, Number 1, pages 1–8, ACM, 2011.

43. *Alexander Malkis*, *Laurent Mauborgne*. *On the Strength of Owicki-Gries for Resources.* 9th Asian Symposium on Programming Languages and Systems (APLAS 2011), Lecture Notes in Computer Science, Springer-Verlag, 2011.

44. Patrick Cousot, Radhia Cousot, *Laurent Mauborgne. Logical Abstract Domains and Interpretations.* The Future of Software Engineering, pages 48–71, SpringerVerlag, 2011.

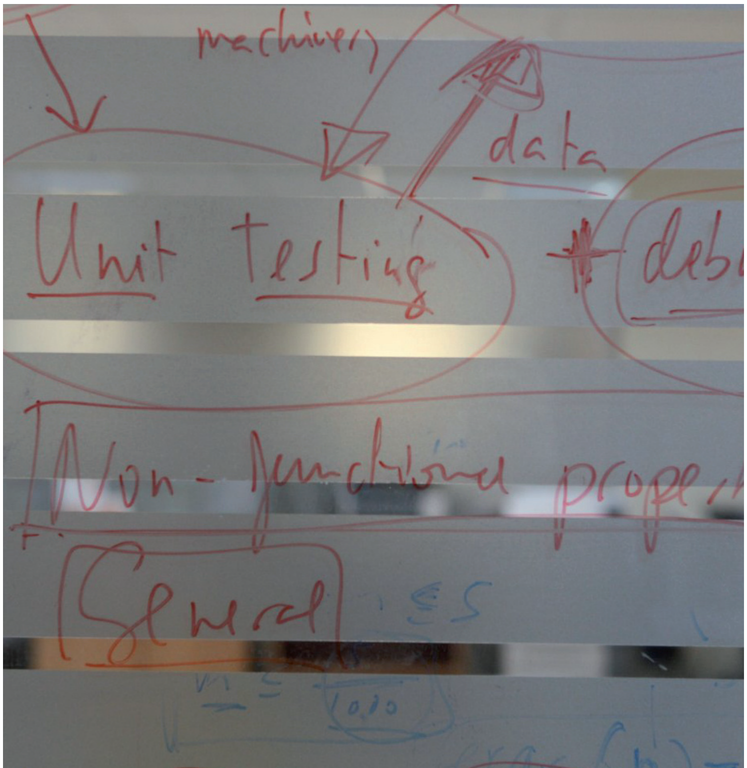
45. Georges Gonthier, Beta Ziliani, *Aleksandar Nanevski*, Derek Dreyer. *How to Make Ad-Hoc Proof Automation Less Ad-Hoc.* 16th ACM SIGPLAN international conference on Functional Programming, ICFP 2011, pages 163–175, ACM, September 2011.

46. Kasper Svendsen, Lars Birkedal, *Aleksandar Nanevski*. *Partiality, State and Dependent Types.* Typed Lambda Calculi and Applications - 10th International Conference, TLCA 2011, LNCS, Vol. 6690, pages 198–212, Springer, June 2011.

47. *Alejandro Sánchez*, *César Sánchez*. A Theory of Skiplists with Applications to the Verification of Concurrent Datatypes. Proc. of the 3rd NASA Formal Methods Symposium (NFM'11), LNCS, Vol. 6447, pages 343–358, Springer, 2011.

48. Edison Mera, *Teresa Trigo*, *Pedro López-García*, *Manuel Hermenegildo*. *Profiling for Run-Time Checking of Computational Properties and Performance Debugging.* Practical Aspects of Declarative Languages (PADL'11), LNCS, Vol. 6539, pages 38–53, Springer-Verlag, January 2011.

49. *Teresa Trigo*, *Pedro López-García*, Susana Muñoz-Hernandez. *A Fuzzy Approach to Resource Aware Automatic Parallelization.* 2nd International Joint Conference on Computational Intelligence, Selected Papers, Studies in Computational Intelligence (SCI), 19 pages, Springer-Verlag, 2011.



6.1.2. Edited Volumes

1. *Gilles Barthe*, Alberto Pardo, Gerardo Schneider. *Software Engineering and Formal Methods - 9th International Conference, SEFM 2011.* Lecture Notes in Computer Science, Vol. 7041, Springer, November 2011.

2. *Gilles Barthe*. *Programming Languages and Systems - 20th European Symposium on Programming, ESOP 2011, held as part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2011.* Lecture Notes in Computer Science, Vol. 6602, Springer, March 2011.

3. Jorge Cuéllar, Javier Lopez, *Gilles Barthe*, Alexander Pretschner. *Security and Trust Management - 6th International Workshop, STM 2010, Athens, Greece, September 23-24, 2010, Revised Selected Papers.* Lecture Notes in Computer Science, Vol. 6710, Springer, 2011.

4. *M. Carro*, J.H. Reppy. *ACM SIGPLAN Proceedings of the Workshop on Declarative Aspects of Multicore Programming.* ACM, January 2011.

5. *John P. Gallagher*, Michael Gelfond. *Theory and Practice of Logic Programming. 27th Int'l. Conference on Logic Programming (ICLP'11) Special Issue.* Vol. 11 (4–5), pages 429–839, Cambridge University Press, July 2011.

6. *John P. Gallagher*, Michael Gelfond. *Technical Communications of the 27th International Conference on Logic Programming (ICLP'11).* Leibniz International Proceedings in Informatics (LIPIcs), Vol. 11, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, July 2011.

6.1.3. Master's Theses

1. Julian Samborski-Forlese. Two Algorithms for Model Checking of Regular Linear Temporal Logic. Master's Thesis, Universidad Complutense de Madrid, September, 2011. Adviser: César Sánchez (IMDEA Software Institute) and Miguel Palomino (UCM).

2. Alejandro Sánchez. Decision Procedures for the Temporal Verification of Concurrent Data Structures. Master's Thesis, Universidad Complutense de Madrid, July, 2011. Adviser: César Sánchez (IMDEA Software Institute) and Miguel Palomino (UCM).

3. Carolina Inés Dania. MySQL4OCL: A Compiler from OCL to MySQL. Master's Thesis, Universidad Complutense de Madrid, September, 2011. Adviser: Manuel Clavel and Marina Egea (IMDEA Software Institute).

invited talks

6.2. Invited Talks

6.2.1. Invited and Plenary Talks by IMDEA Scientists

1. *Anindya Banerjee*. Modular Reasoning about Object-based Programs. *Second International Conference on Formal Verification of Object-Oriented Software*, Turin, Italy, October 2011.

2. *Anindya Banerjee*. Verification of Access Control and Information Flow Policies using Dependent Types. *Workshop on Formal Methods and Security*, Chalmers University, August 2011.

3. *Gilles Barthe*. Probabilistic Reasoning About Differential Privacy. Invited talk at the *Workshop on Games, Logic and Security* (GIPSy'11), October 2011, Rennes, France.

4. *Manuel Clavel*. A Decade of Model-Driven Security (with David Basin). *ACM Symposium on Access Control Models and Technologies* (SACMAT 2011).

5. *Manuel Clavel*. Model-Driven Security: Foundations, Tools, and Practice (with David Basin). *11th International School on Foundations of Security Analysis and Design* (FOSAD 2011).

6. *John P. Gallagher*. Program Analysis With Regular Tree Languages. *Logic-Based Program Synthesis and Transformation Symposium*, Odense, Denmark, July 2011.

7. *Manuel V. Hermenegildo*. An Overview of the Ciao System. *Symposium on Rule-Based Computing* (RuleML-Europe 2011).

8. *Manuel V. Hermenegildo*. The Ciao Approach to the Dynamic vs. Static Language Dilemma. *Workshop on Logic-based Methods in Programming Environment* (WLPE 2011).

9. *Boris Köpf*. Quantitative Information-Flow Analysis. Invited tutorial at the 8th International Conference on Quantitative Evaluation of Systems (QEST 2011), Aachen, Germany, September 2011.

10. *Aleksandar Nanevski*. How to Make Ad-Hoc Proof Automation Less Ad-Hoc. *6th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice* (LFMTP 2011), Nijmegen, Holland.

11. *Aleksandar Nanevski*. Scrap Your Tactics: Automating Coq Proofs with Generalized Type Classes. In *Functional Programming Workshop* (TFP'11).

6.2.2. Invited Seminars and Lectures by IMDEA Scientists

Anindya Banerjee

1. Modular Reasoning about Object-based Programs. ETH Zürich, December 2011.

Gilles Barthe

2. IFIP Working Group 2.1. Computer-aided Cryptographic Proofs. Paris, France. September 2011.

Manuel Carro

3. Tabled Logic Programming and Applications. Invited talk at Prometidos-CM Summer School, Madrid, September 2011.

Pierre Ganty

4. Approximating Petri Net Reachability Along Context-free Traces. Seminar at the Computer Science Department, Université Libre de Bruxelles, Belgium, June 2011.

Pierre Ganty

Pattern-based Verification for Multithreaded Programs, given at:

5. "Séminaire du LSV", LSV, École Normale Supérieure de Cachan, France, April 2011.

6. Seminar of the "Centre Fédéré en Vérification", Belgium, June 2011.

7. Upmarc Seminar, Uppsala Univeristy, Sweden, June 2011.

8. Verimag, France, July 2011.

Pierre Ganty

Petri Nets and Finite Index Context-Free Languages:

9. LIAFA, Université Paris 7, France, April 2011.

Pierre Ganty

Verification of Systems with Infinitely Many States: Underapproximations and Overapproximations.

10. Departamento de Sistemas Informáticos y Computación, Facultad de Informática, Universidad Complutense de Madrid, Spain, January 2011.

Alexey Gotsman

The Importance of Being Linearizable. Seminar given at:

11. Microsoft Research Redmond, USA

12. University of Cambridge, UK.

13. ETH Zurich, Switzerland.

14. Imperial College London, UK.

15. University of Oxford, UK.

16. University of Washington, USA.

17. Portland State University, USA.

18. University of Cambridge, UK.

19. Galois Inc., Portland, USA.

Alexey Gotsman

Modular Verification of Preemptive OS Kernels. Talk given at:

20. Dagstuhl Seminar on Multi-Core Memory Models and Concurrency Theory.

21. Microsoft Research Redmond, USA.

22. IBM Research TJ Watson Center, USA.

23. Yale University, USA

Alexey Gotsman

24. Separation Logics and Applications. Invited lecturer at the *ENS Lyon Winter School*, January 31 - February 4, 2011.

Boris Köpf

25. Quantitative Information-Flow Analysis Tutorial. Talk given at the Prometidos Summer School, Madrid, Spain, September 2011.

César Kunz

26. Relational Verification Using Product Programs. Prometidos Summer School, Madrid, Spain, September 2011.

Pedro López

27. Resource Usage Analysis and Verification in the CiaoPP System. University of New Mexico, August 25, 2011.

Alexander Malkis

Modular Verification of Multithreaded Programs. Talk given at:

28. The *Sixth International Conference on Information Systems Security* (ICISS).

29. The Prometidos Summer School, Madrid, Spain, September 2011.

José F. Morales Caballero

30. Optimizing Compilation Techniques for Logic Programming. Prometidos Summer School, Madrid, Spain, September 2011.

Mark Marron

31. Understanding Data Structures: From Design to Debugging. University of California at Davis, August 2011.

6.2.3. Invited Speaker Series

During 2011, a total of 22 external researchers gave invited talks at the IMDEA Software Institute. The list of researchers and their talks follow:

1. Natasha Sharygina, Universita della Svizzera Italiana (University of Lugano), Italy. Local Proof Transformations for Flexible Interpolation and Proof Reduction.

2. Pavithra Prabhakar, University of Illinois at Urbana Champaign, USA. Approximations for Verification of Cyber Physical Systems.

3. Doron Peled, Bar Ilan University, Israel. Knowledge-based Synthesis of Control for Distributed Systems.

4. Deepak Kapur, University of New Mexico, USA. Induction, Invariants, and Abstraction.

5. Isil Dillig, Stanford University, USA. Precise and Fully-Automatic Verification of Container-Manipulating Programs.

6. Thomas Dillig, Stanford University, USA. Program Paths Simplified: Scalable Path-Sensitive Analysis without Heuristics.

7. Vijay Ganesh, MIT Cambridge, MA, USA. Solvers for Software Reliability and Security.

8. Raul Santelices, Georgia Institute of Technology, USA. Change-effects Analysis for Effective Testing and Validation of Evolving Software.

9. Graham Steel, INRIA Rocqcuencourt, France. Attacking and Fixing PKCS#11 Security Tokens.

10. Vasu Singh, Institute of Science and Technology (IST), Austria. Exporting the Art of Formal Verification.

11. Filip Pizlo, Lafayette Indiana, USA. Fragmentation Tolerant Real Time Garbage Collection.

12. David Basin, ETH Zurich, Switzerland. Policy Monitoring in First-order Temporal Logic.

13. Reinhard Wilhelm, University of Saarland, Germany. Ongoing Work and Open Questions in Timing Analysis.

14. Giorgio Delzanno, Università di Genova, Italia. Monotonic Approximations in Parameterized Verification.

15. Nazareno Aguirre, Universidad Nacional de Río IV, Argentina. Incorporating Coverage Criteria in Bounded Exhaustive Black Box Test Generation of Structural Inputs.

16. Martin Wirsing, Ludwig-Maximilians University of München, Germany. Adaptation and Awareness in Ensembles.

17. Rodrigo Rodrigues, École Polytechnique Federale de Lausanne, Switzerland. Gaining Customer Trust in Cloud Services with Excalibur.

18. Dejan Kostic, École Polytechnique Federale de Lausanne, Switzerland. Online Testing of Deployed Federated and Heterogeneous Distributed Systems.

19. Matthieu Sozeau, INRIA Rocqencourt, France. First-Class Type Classes for Programs and Proofs.

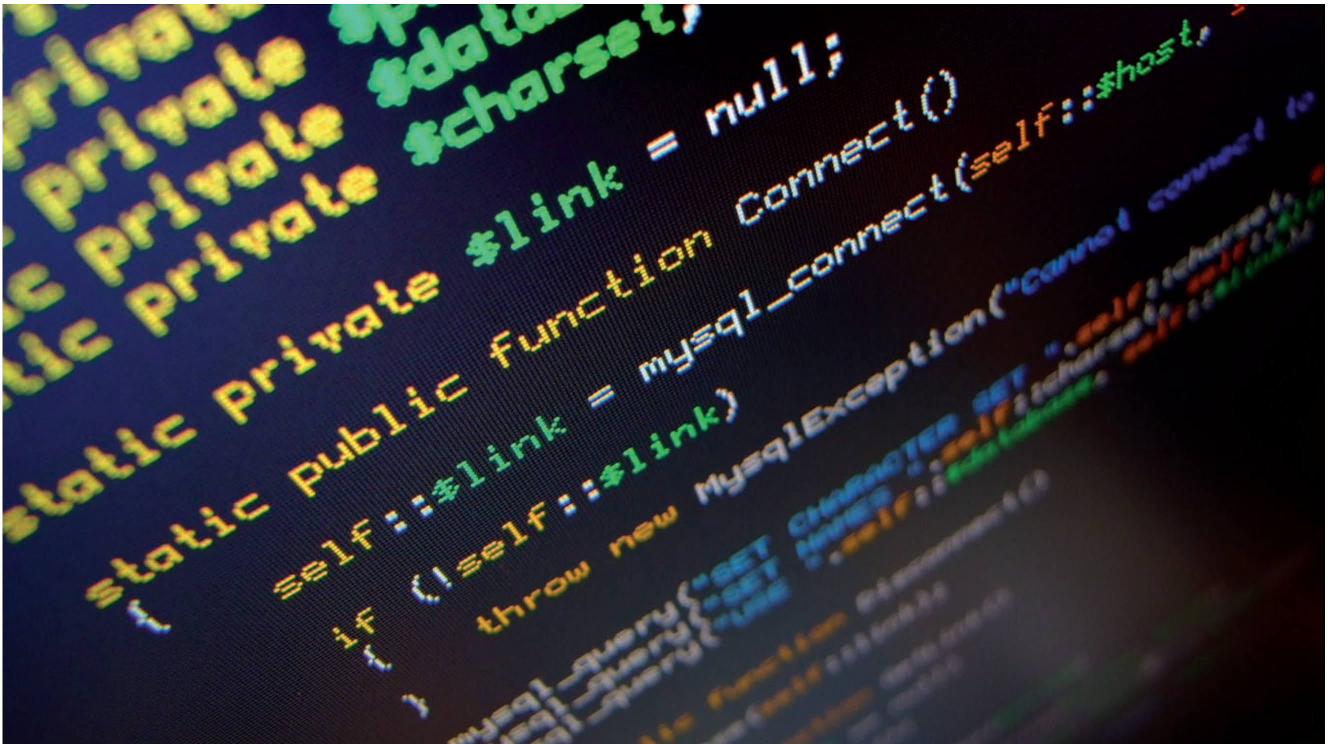
20. Ivan Beschastnikh, University of Washington, USA. Leveraging Existing Instrumentation to Automatically Infer Invariant-Constrained Models.

21. Antoine Miné, École Normale Supérieure, France. Astrée: A Static Analyzer for Embedded Multi-Threaded C Programs.

22. Misha Aizatulin, Open University, UK. Extracting and Verifying Cryptographic Models from C Protocol Code by Symbolic Execution.

6.2.4. Theory Lunch Series

The Institute also holds an internal seminar series to foster communication and collaboration. A total of **26** seminars were given in 2011.



```
private $charset,
static private $link = null;
static public function Connect()
{
self::$link = mysql_connect(self::$host,
if (!self::$link)
throw new MysqlException("Cannot connect
```

6.3. Scientific Service & Other Activities

6.3.1. Participation in Program Committees

Anyndia Banerjee

1. 17th International Symposium on Formal Methods (FM 2011).

2. 18th International Static Analysis Symposium (SAS 2011).

Gilles Barthe

3. 24th IEEE Computer Security Foundations Symposium (CSF 2011).

4. First International Conference on Certified Programs and Proofs (CPP 2011).

5. 38th International Colloquium on Automata, Languages and Programming (ICALP 2011).

Juan Caballero

6. 14th International Symposium on Recent Advances in Intrusion Detection (RAID 2011).

7. 8th Conference on Detection of Intrusions and Malware & Vulnerability Assessment (DIMVA 2011).

Manuel Carro

8. Business Process and Services Computing (BPSC 2011).

9. 11th Colloquium on Implementation of Constraint and LOgic Programming Systems (CICLOPS 2011).

10. Sixth International Workshop on Declarative Aspects of Multicore Programming (DAMP 2011).

11. 27th International Conference on Logic Programming (ICLP 2011).

12. International Conference on Service Oriented Computing (ICSOC 2011).

13. 21st International Symposium on Logic-Based Program Synthesis and Transformation (LOPSTR 2011).

14. XI Jornadas sobre Programación y Lenguajes (PROLE 2011).

15. First International Workshop on Quality Assurance for Service-based Applications (QASBA 2011).

16. ServiceWave 2011.

Manuel Clavel:

17. 8th International Workshop on Rewriting Logic and its Application (WRLA 2011).

18. Fifth International Conference on Secure Software Integration and Reliability Improvement (SSIRI 2011).

John Gallagher

19. 11th Scandinavian Conference on Artificial Intelligence (SCAI 2011).

20. 21st Workshop on Logic-based Methods in Programming Environments (WLPE 2011).

Alexey Gotsman

21. 6th International Workshop on Systems Software Verification (SSV 2011).

Manuel Hermenegildo

22. 32nd ACM SIGPLAN conference on Programming Language Design and Implementation (PLDI 2011).

23. Sixth International Workshop on Declarative Aspects of Multicore Programming (DAMP 2011).

Boris Köpf

24. 24th IEEE Computer Security Foundations Symposium (CSF 2011).

25. 8th International Conference on Quantitative Evaluation of Systems (QEST 2011).

Mark Marron

26. Workshop on Developing Tools as Plug-ins (TOPI 2012, co-located with ICSE 2012).

César Sánchez

27. 18th International Symposium on Temporal Representation and Reasoning (TIME 2011).

28. XI Jornadas sobre Programación y Lenguajes (PROLE 2011).

6.3.2. Conference and Program Committee Chairmanships

Gilles Barthe

1. PC co-chair of the 2011 International Symposium on Engineering Secure Software and Systems (ESSoS 2011).

2. PC chair of the 2011 European Symposium on Programming (ESOP 2011).

3. PC co-chair of the 8th International Workshop on Formal Aspects of Security & Trust (FAST'11).

4. PC Co-chair of Software Engineering and Formal Methods (SEFM'11).

Manuel Clavel

5. General Chair of the 2011 International Symposium on Engineering Secure Software and Systems (ESSoS 2011).

Manuel Carro

6. Co-chair of the Third International Workshop on Principles of Engineering Service-Oriented Systems (PESOS'11 — satellite workshop of ICSE 2011).

7. Co-chair of the Sixth International Workshop on Declarative Aspects of Multicore Programming (DAMP 2011 – satellite workshop of POPL 2011).

John Gallagher

8. PC co-chair of the 27th International Conference on Logic Programming (ICLP 2011).

Pierre Ganty

9. Co-Chair of the 2011 Bytecode Semantics, Verification, Analysis and Transformation workshop (BYTECODE 2011 — satellite workshop of ETAPS 2011).

Mark Marron

10. Co-Chair of the 2011 Bytecode Semantics, Verification, Analysis and Transformation workshop (BYTECODE 2011 — satellite workshop of ETAPS 2011).

Laurent Mauborgne

11. Chair of the Third Numerical and Symbolic Abstract Domain Workshop (NASA 2011 – collocated with SAS 2011).

6.3.3. Editorial Boards and Conference Steering Committees

Anindya Banerjee

1. Associate Editor of Journal of Higher Order and Symbolic Computation.

Gilles Barthe:

2. Editor of the Journal of Automated Reasoning.

Manuel Carro:

3. Steering Committee member of the International Workshop on Declarative Aspects of Multicore Programming (DAMP).

John Gallagher:

4. TPLP Area Editor (Technical Notes and Rapid Publications).

5. Steering Committee member of the conference Partial Evaluation and Program Manipulation (PEPM).

Manuel Hermenegildo

6. Editorial Adviser of Theory and Practice of Logic Programming.

7. Area Editor of the Journal of Applied Logic.

8. Associate Editor of the Journal of New Generation Computing.

9. Member of the Journal of Algorithms in Cognition, Informatics, and Logic.

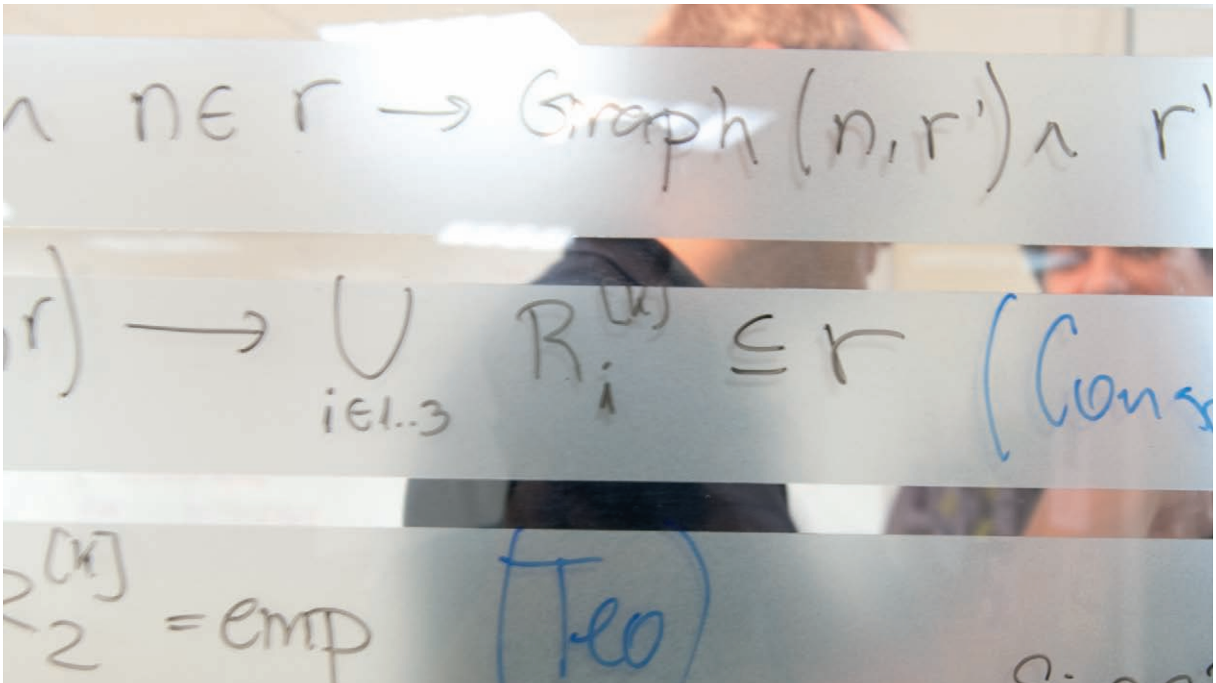
10. Steering Committee member of the ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL).

11. Steering Committee member of the International Static Analysis Symposium (SAS).

12. Steering Committee member of the International Symposium on Functional and Logic Programming (FLOPS).

13. Steering Committee member of the International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI).

14. Steering Committee member of the Federated Logic Conference (FLoC).



6.3.4. Association and Organization Committees

Gilles Barthe

1. Co-Chair of the Working Group on Concurrency and Distribution for COST Action 0701 *Formal Verification of Object Oriented Software*.

Manuel Carro

2. Deputy representative at Informatics Europe.

John Gallagher

3. Executive Committee member for the Association for Logic Programming.

Alexey Gotsman

4. Steering Committee member of the EPSRC Programme Grant on Resource Reasoning.

Manuel Hermenegildo

5. Member of the Academia Europaea.

6. Member of the IFCoLog advisory board.

7. Elected President of SpaRCIM.

8. Member of Informatics Europe department evaluation advisory board.

9. Member of IRILL (French Free Software Institute) scientific board.

10. Member of CSIC (National Research Council) scientific board.

11. External department evaluator for the Katholieke Universiteit Leuven.

12. Member of the Comunidad de Madrid high school honors program faculty selection committee board.

6.3.5. Awards and Competitions

Best conference paper awards

1. *Barthe, Gilles*, Grégoire, Benjamin, Heraud, Sylvain, *Zanella Béguelin, Santiago*. *Computer-Aided Security Proofs for the Working Cryptographer.* Advances in Cryptology – CRYPTO 2011, Lecture Notes in Computer Science, Vol. 6841, pages 71–90, Springer, August 2011.

2. *Juan Caballero*, Chris Grier, Christian Kreibich, Vern Paxson. *Measuring Pay-perInstall: The Commoditization of Malware Distribution.* Proceedings of the 20th USENIX Security Symposium, August 2011.

3. D. Ivanović, *M. Carro*, *M. Hermenegildo*. *Constraint-Based Runtime Prediction of SLA Violations in Service Orchestrations.* Service-Oriented Computing – ICSOC 2011, LNCS, Num. 7084, pages 62–76, Springer Verlag, December 2011.

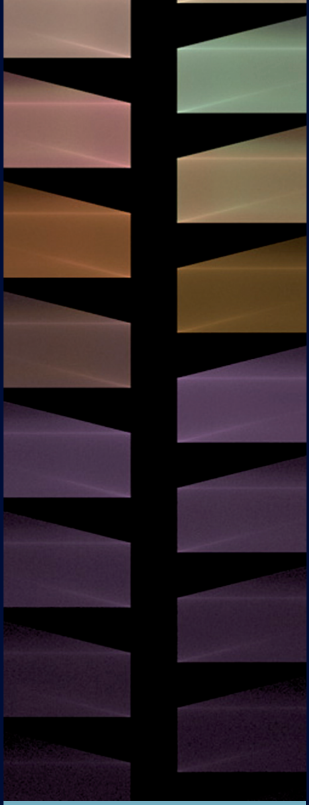
4. Julien Bertrane, Patrick Cousot, Radhia Cousot, Jérôme Feret, *Laurent Mauborgne*, Antoine Miné, Xavier Rival. *Static Analysis and Verification of Aerospace Software by Abstract Interpretation*. Granted by the American Institute of Aeronautics and Astronautics, Intelligent Systems Technical Committee.

Thesis awards

1. Santiago Zanella Béguelin received the 2011 European Association for Programming Languages and Systems Best PhD Dissertation Award, completed under the supervision of Prof. Gilles Barthe.

Competitions

1. Manuel Hermenegildo, member of 2011 Prolog programming contest winning team. International Conference on Logic Programming, Lexington, Kentucky, USA July 2011.



7 scientific highlights

towards “gree

Towards “Greener” Software: Verifying & Controlling Computing Resource Consumption

verifying & controlling computing resource consumption

Energy consumption and the environmental impact of technology have been major worldwide concerns for years. However, these concerns have not included computing technologies until very recently. This new focus on the energy-related costs of computing is motivated in part by recent sustainability studies and also by the increasing demand for complex computing systems which have to operate with limited batteries, such as portable medical devices and mobile computing (e.g., phones or tablets), and the ensuing need to optimize energy consumption in such services. Energy consumption has also become a major concern in high-performance computing, distributed applications, and data centers. In office environments, computers and monitors account for the highest energy consumption after lighting. It has been estimated that cloud computing-related energy consumption has been increasing by 14% per year, while Internet traffic has increased by 50% per year. Such growth is not sustainable with the energy efficiency levels of current computing technology. Thus, we face the challenge of reducing the energy usage of computing significantly.

In spite of the huge recent advances in power-efficient hardware, most of the potential energy savings are wasted by software that does not exploit these hardware energy-saving features, and by poor dynamic management of tasks and resources. For instance, a recent estimate by Intel states that energy savings by a factor of 3 to 5 could be achieved using software optimizations alone.

The IMDEA Software Institute aims at promoting energy efficiency to a first-class software design goal, and, in a more general context, to developing tools that facilitate the production of “greener” devices, i.e., devices that make a certifiably more efficient use of their available *resources*. Our very general concept of resource includes classical concerns like execution time, memory, or disk space as well as other user-defined or platform-dependent magnitudes like energy, network accesses, or opened files, to name a few.

IMDEA Software researchers are working towards achieving radical advances in energy-aware software design and management that aims to provide the key to the pervasive realisation of energy-aware computing. These advances include the explicit exploitation of power-efficient features offered by hardware supporting conventional computation models, as well as by emerging approaches such as massively parallel systems and biologically

ner” software

inspired computation models. A novel, holistic, energy-aware system development approach is being developed that covers hardware, software, and the run-time environment, making information on energy usage available throughout the system layers and promoting optimizations both during code development and at runtime. This approach enables a flexible trade-off between energy and behavioral aspects of the software, including precision and performance, and requires an important effort on developing novel analysis, combined hardware-software energy modeling, verification and optimization techniques.

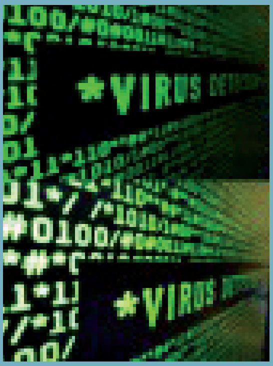
Most of this research will be performed within the scope of the recently granted project ENTRA, an FP7 “Future and Emerging Technologies” project under the proactive “MINECC” objective, which focuses on “Minimising Energy Consumption of Computing to the Limit,” in collaboration with XMOS (UK), The University of Bristol (UK) and Roskilde University (Denmark).

IMDEA Software researchers have extended conventional debugging and verification techniques to deal with resource usage properties, allowing automated performance debugging and certification of programs. A novel aspect of resource verification is that static checking generates answers that go beyond the classical outcomes (true/false/unknown). These answers include conditions under which these classical outcomes are obtained, including input data size or value ranges. For example, it is possible to infer that the outcome is true if the input data size is in a given range.

IMDEA Software researchers have also developed and keep working on automatic optimization techniques that reduce significantly the resource usage of programs, in particular the total execution time and power use. For instance, automated refactoring of reference types into value types produce programs with lower memory consumption, garbage collection times, and better memory locality. All of which reduce the resource usage of the programs. Automated synthesis techniques (e.g., autovectorization) of programs for SIMD architectures also reduces the total execution time and power use considerably.

All the developed techniques are implemented and integrated into IMDEA’s state-of-the-art tools, which are demonstrated to industrial collaborators and tested on concrete examples extracted from their application codes. For example, the pioneering CiaoPP system provides a general framework for estimating with high precision the resources consumed by a given piece of software and for debugging/certifying such consumption with respect to specifications. The tool is highly adaptable to different languages, hardware, and resources because it is built around a customizable static analyzer with a versatile and clear assertion language.

Understanding the Role of Malware in Cybercrime



Cybercrime, criminal activity conducted via computers connected to the Internet, is a growing threat for developed regions like Europe where nearly three quarters of the households and a large number of the infrastructures are connected to the Internet, and an increasingly number of services and transactions happen on line.

At the core of most cybercrime operations is the attacker's ability to install malicious programs (i.e., malware) on Internet-connected computers without the owner's informed consent. Malware includes bots, viruses, trojans, rootkits, fake software, and spyware. Malware enables attackers to establish a permanent presence in the compromised computer and to leverage it for their cybercrime operations. The target of those operations may be the compromised computers themselves (e.g., stealing from the computer an organization's intellectual property or a user's banking credentials), but also third parties. In the latter case, the compromised computers are simply assets, which the attacker employs to launch malicious activities such as sending spam, launching denial-of-service (DoS) attacks, faking user clicks on online advertisements (i.e., click-fraud), or simply as a stepping stone to hide its location.

The goal of the MALICIA project at the IMDEA Software Institute is to study the crucial role of malware in cybercrime and the rise in recent years of a far-reaching "underground economy" associated with malware and the subversion of Internet-connected computers. Long gone are the days where attackers compromised computers and built malware to show off their skills to peers. Nowadays, the malware ecosystem revolves around cybercrime and the monetization of compromised computers.

As the malware ecosystem has grown larger and more profitable, specialization has come to the marketplace. Attackers have understood that tackling the entire value-chain from malware creation to monetization poses a daunting task requiring highly developed skills and resources. As a result, specialized services have been created at all stages in the malware-monetization chain, such as toolkits to automate the construction of malware, program encryption tools to evade antivirus (AV) software, "bullet-proof" hosting, and forums for buying and selling ill-gotten gains. Specialized services lower the barrier to enter the malware ecosystem. However, defenders can also take advantage of specialization, as disrupting the specialized services disrupts the different malware operations using them.

As a first step in the MALICIA project, we have collaborated with researchers at the University of California, Berkeley and the International Computer Science Institute to investigate the commoditization of malware distribution in the form of *pay-per-install* (PPI) services. PPI services offer criminals a simple way to outsource the distribution of their malware. The clients provide their malware to the PPI service and select the

f malware in cybercrime

number of desired installations (called *installs*) in each geographical area. The PPI service takes care of installing the malware on compromised computers in exchange for a small fee that ranges from $180 for a thousand computers in some European countries and the US, down to $7 for a thousand computers in Asia.

To satisfy the clients' demand for installs, the PPI provider typically outsources malware distribution to third parties called *affiliates*. PPI providers pay affiliates for each compromised computer, acting as a middle man that sells installs to the clients while buying installs from affiliates. Each affiliate may specialize in some specific malware distribution method (e.g., bundling malware with a benign program and distributing the bundle via file-sharing networks; exploiting web browsers through drive-by-downloads; or social engineering). The PPI service gives each affiliate a downloader program customized with a unique affiliate identifier. When the affiliate installs the downloader in a compromised computer, the downloader connects back to the PPI service to download the client programs. After installing the client programs on the compromised host, the downloader reports the affiliate identifier and the affiliate is credited with an install.

To understand the PPI market we *infiltrated* four PPI services. For this, we developed infrastructure enabling us to (1) interact with PPI services by mimicking the protocol interactions they expect to receive from affiliates, and (2) collect and classify the malware being distributed by the PPI services. Using this infrastructure we harvested over a million malware programs and classified them by malware family as well as monetization methods. Our analysis revealed that of the world's top 20 malware families, 12 employed PPI services for their distribution. It also revealed that some malware families exclusively target the US and a variety of European countries. The monetization methods in use are wide including: spam, installing fake antivirus software, information-stealing, denial-of-service, click-fraud, and adware.

Much remains to be learnt about the malware ecosystem and the specialized economy supporting cybercrime. Our current work strives on deepening our understanding of other parts of the ecosystem. One overarching goal is evolving malware analysis from understanding what a malware program does, to also cover *why* it does it, i.e., what role the malware program plays in the cybercrime operation where it is used.

This work, in addition to appearing in a number of scientific dissemination fora, has been covered by different media, including MIT's Technology Review.

When Security Matters: Computer-Aided Cryptographic Proofs

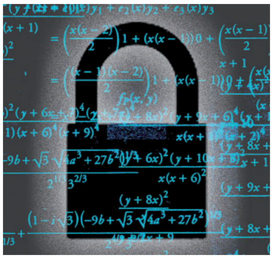
Modern cryptography is the science of developing methods for protecting information and communication against misbehaving parties. Initiated with the pioneering work of Shannon on secrecy in encryption systems in 1949, modern cryptography has become an active field of research with the discovery of public-key cryptography by Diffie and Hellman in 1976, the invention of the RSA algorithm by Rivest, Shamir and Adleman in 1978, and the conception of provable security by Goldwasser and Micali in 1984. Initially focused on encrypted communications over insecure channels, cryptography has expanded considerably to achieve a broad range of security goals, from basic ones such as confidentiality and integrity, to elaborate goals such as proofs of knowledge. In parallel, applications of cryptography have outgrown the domain of military and diplomatic communications, to play a central role in the Internet, the Cloud, and more generally in any massively distributed infrastructure that can store and process huge quantities of data and computations. In effect, billions of individuals, companies, and institutions use cryptography routinely for interacting with each other. Online banking systems, electronic health records, cash machines, cell phones, or digital identity management systems are only a few examples of the numerous applications that rely on cryptography.

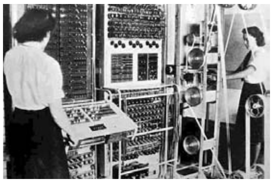
16th century French cypher machine in the shape of a book with arms of Henri II

German Enigma machine

matters:

Smart Card





Modern cryptography achieves its goals by extracting from specialized branches of pure mathematics, such as number theory, very efficient algorithms that can be used in practical systems for the purpose of granting them an adequate level of security. The downside to such an amazing feat is that cryptographic algorithms are very difficult to analyze. Thus, cryptographers devote a significant amount of time to building rigorous mathematical proofs of the security of a cryptographic scheme. The goal of these proofs is to show that the cost of attacking a cryptographic system is prohibitive, and to determine the parameters that must be used—for example, the size of keys—in any practical realization. Building such cryptographic proofs is far from being an academic problem: in fact, there are critical gaps in the security proofs of the most widely used cryptographic schemes and protocols. For instance, researchers from IMDEA Software Institute, INRIA, and Université of Grenoble have recently unveiled an inaccuracy in the proof of the RSA-OAEP, a widely deployed encryption scheme that is recommended by several standards, including IEEE P1363, PKCS, ISO 18033-2, ANSI X9, CRYPTREC and SET.

In order to ensure that cryptographic systems achieve their purported security requirements, researchers from IMDEA Software Institute, INRIA and Microsoft are developing EasyCrypt, an automated tool that supports a radically new approach to cryptographic proofs. While adhering to the principles and the methods of provable security, EasyCrypt takes the view that cryptographic proofs should be treated in a manner similar to high-integrity software, so that confidence in the design of a cryptographic system is no lower than confidence in the software systems that use it. In order to realize its ambition, and to provide working cryptographers with practical tools for building trustworthy and verifiable proofs, EasyCrypt builds upon state-of-the-art verification tools, including SMT solvers, automated theorem provers, and proof assistants.

Although its development is in its initial stage, EasyCrypt has attracted considerable interest from the academic and industrial communities. The article “Computer-Aided Security Proofs for the Working Cryptographer”, by Gilles Barthe, Benjamin Grégoire, Sylvain Heraud, and Santiago Zanella Béguelin, received the Best Paper Award at CRYPTO’11, the premier conference in cryptography; besides, Santiago Zanella Béguelin received the 2011 EAPLS Best PhD Dissertation Award. Researchers at IMDEA Software and INRIA have also initiated collaborations around EasyCrypt with many leading researchers in several European and US academic and research institutions.

high integri

when software must not fail

High Integrity Software: When Software Must Not Fail

Some software cannot fail, in some real world applications. This software is called high integrity software and must be trusted to work dependably in some critical function. Failure in these programs may have catastrophic results in terms of lives or have high economic cost. For example, failure in a program used by air traffic controllers could lead to fatal accidents; a failure in a medical system or a medical device could lead to irreversible damage; failures in parts of automobile systems such as brake controllers, apart from being potentially dangerous, could lead to massive and costly recalls. In fact, all of these scenarios have occurred already.

Examples of high integrity software include safety systems of nuclear power plants, medical devices, air traffic control, automated manufacturing and satellite control. In software that runs power grids, financial systems, water treatment plants and other critical elements of a country's national infrastructure another dimension of high integrity must be guaranteed: such software must further be secure against cyberattacks and thus must guarantee that they meet security requirements.

Current software engineering practices balance between the cost (and time) to complete a project on the one hand, and software quality on the other. The rapidly growing demand for software that provide ever-more complex functionality has increased industry demand for software developers. In turn, this need has motivated a trend towards reducing the

Satellites have to be autonomous up to a certain degree. The programs running in their computers continuously monitor for deviations from their scheduled trajectories and take the appropriate decisions to correct them.

ty software

training necessary for software engineers and developers to enter the job market. However, at the same time, the quality of the software produced has become more and more difficult to guarantee. The issue with software quality is witnessed by the fact that the dominant factor of the overall cost of current industrial software projects is testing, and not building the product itself. Even in non-critical projects testing dominates more than 90% of the total cost.

High integrity software requires the replacement of disclaimers (as is current practice) by guarantees such as functional correctness and security. The US government is currently discussing legislation to ensure such guarantees from software vendors.

The quality and reliability requirements of high integrity software justifies the investment in scientific undertakings to create a body of knowledge about how to build more reliable software. These new techniques intend to provide better guarantees of quality, at the price of using more sophisticated methods and tools by properly trained software engineers. These methods encompass foundational theories, tools and convincing experiments and their importance cannot be overstated. While in the "here and now" they lead to more productive software processes for industry, their long term goal is to develop processes that can be applied to the widest possible range of application software, far beyond current demands of industry or popularity in the market place.

Researchers from the IMDEA Software Institute have developed – and continue to develop – cutting-edge technologies for high-integrity software following two approaches.

The first approach is foundational: it is aimed towards creating the basic science, based on rigorous mathematics, that can be used to craft the high-integrity software of the future. These techniques are designed to provide the best guarantee of adherence to intended behavior and are at the heart of the robust, scalable tools, that serve as testbeds for our foundational approach. Completed and ongoing projects include the use of high-order theorem proving to verify programs and libraries, static analysis for functional and non-functional properties of real-time, embedded and reactive systems.

The second approach concerns the development of novel lightweight and applicable techniques that can be directly incorporated to improve existing software practices: advanced visualization of heap-manipulating programs, debugging of production system programs, and online monitoring of embedded reactive programs based on runtime verification.

Institute researchers have also launched a significant effort towards verifying concurrent software, particularly concurrent data type libraries and operating systems software. This is a particularly challenging area and requires a foundational re-think from current practices which, in industry, are based predominantly on testing. For instance, it is well known that concurrency bugs owing to race conditions and deadlocks are very hard to detect. Even the seemingly easy task of reproducing a bug becomes a challenge due to influence of factors such as the current CPU workload.

The IMDEA Software Institute is collaborating with the leading aerospace company Deimos, located in the area of Madrid, on the technology transfer of these techniques. This continuing effort started with the rigorous and systematic development of software for satellite image processing. The aim of this project is to develop the tools to interactively synthesize provably correct software, based on a formal approach to software families, applied to image processing. Using these tools software engineers can develop very efficient parallel software that can be verified with independent tools. Moreover, different projects can experience dramatic cost gains by the increase in the level of reuse by the use of software families.

As mentioned before, an important dimension of high integrity is that software meets security requirements. Current computing environments and infrastructures are increasingly heterogeneous and dynamically changing. Executable programs are everywhere: web pages, email, plug-and-play extensions, JavaScript, on-line games, Word and PowerPoint documents and attachments, electronic banking, etc. Software is constantly being updated and downloaded over the Internet, sometimes without our knowledge or consent.

Yet, today's security architectures provide poor protection from faulty software, and even less from malicious software. These security architectures were developed at a time when software was managed and updated infrequently by an experienced administrator, when we trusted the (few) programs we ran, when physical access to the systems was required to cause any damage to the data, and crashes and outages did not cost billions.

As none of these conditions is valid anymore, our information systems have become increasingly susceptible to attacks with potentially devastating consequences.

To accommodate for the new trends in software use and deployment, researchers at the IMDEA Software Institute are working on theories and tools for building trustworthy software that are are well suited for networked computing systems built from diverse and extensible components. By leveraging techniques from programming languages and program logics researchers are addressing the following fundamental issues: (a) what is the precise meaning of security policies, (b) how to correctly specify security policies, (c) how to prove that programs respect the policies and (d) how to provide verifiable evidence, checked by a machine, of proofs of conformance of programs to security policies.

Institute researchers have also shown that such foundational security infrastructure can be put to use in practice. For example, in the Mobius project, jointly with France Telecom and INRIA, they have shown the feasibility of on-device checking of mathematical proofs, using dedicated checkers developed and extracted from rigorous mathematical formalizations in the proof assistant Coq.

editor

imdea software institute

graphic design

base 12 diseño y comunicación

www.software.imdea.org

La Suma de Todos
Comunidad de Madrid
www.madrid.org

EUROPEAN UNION
STRUCTURAL FUNDS

madrid institute
for advanced studies

institute
imdea
software

www.software.imdea.org

Contact
software@imdea.org
tel. +34 91 101 22 02
fax +34 91 101 13 58

Instituto IMDEA Software
Campus de Montegancedo
28223 Pozuelo de Alarcón
Madrid, Spain