SIXTH FRAMEWORK PROGRAMME
PRIORITY 2
"Information Society Technologies"

Project acronym: DESEREC
Project full title: Dependability and Security by Enhanced Reconfigurability
Proposal/Contract no.: IST-2004-026600-DESEREC

D<5.2>
<Report on the 1st DESEREC Dissemination Workshop>

Project Document Number: DESEREC/D<5.2>/<CO)>[1]/v<1.0> (official version)
DESEREC/D<5.2>/<CO >/<IEIIT>/v<1.0> (internal version / contributions)

Project Document Date: 20/07/2007
Workpackage Contributing to the Project Document: WP5
Deliverable Type and Security: <R>[2]-<CO>
Author(s): < Luca Durante> (<IEIIT>)

Abstract:
This document provides a report on the 1st DESEREC dissemination workshop.

Keywords: dissemination

[1] Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

[2] Type: P - Prototype, R - Report, D - Demonstrator, O - Other

History

Version	Date	Description, Author(s), Reviser(s)
1.0	20/07/2007	Document creation, Luca DURANTE

Executive Summary

This documents reports about the first DESEREC dissemination workshop, held in the 2nd ESFORS workshop “Trust, Security and Dependability in Service Oriented Infrastructures”, July 10th and 11th, 2007, Maribor, Slovenia.

This report mainly focuses on the contributions to the workshop organization and sessions given by DESEREC. A comprehensive description and a complete report on the workshop is available on www.esfors.org.

The report is organized as it follows: an introductory section briefly depicts the goals, the achievements and the results of the workshop, and then the full detailed program is introduced and commented. A more detailed section addresses the contributions to the workshop provided by DESEREC, and finally some results and conclusions are sketched.

NOTE: D5.2 was originally conceived as CD-ROM(s) and web pages, as shown in the DESEREC Annex 1 – Description of Work “CD-ROM(s) and web pages containing the research and technical results presented at the first workshop”. This was selected by assuming that the organization of the workshop would have been managed by the DESEREC project only.

On the other hand, the workshop has been organized jointly with other European Projects, and managed by the Coordination Action ESFORS, thus DESEREC hasn’t had the full management and control over the workshop organization, while the ESFORS consortium has managed the collection of the technical and scientific material presented at the workshop, making it available at www.esfors.org.

As a consequence of the change of the DESEREC role in the workshop, the DESEREC management and the WP5 leadership agreed on changing the structure and the title of the deliverable accordingly.

Contents

Page

1 Introduction 5

2 Workshop Organization 6

2.1 Workshop program 6

2.2 Parallel Sessions 8

3 DESEREC Role 12

3.1 Program Committee membership 12

3.2 Keynote speech and DESEREC project presentation 12

3.3 Day 1 / Session 1 - *Engineering dynamic & ad-hoc service coalitions: design and operational (run-time) TSD aspects* 12

3.3.1 Agenda 13

3.3.2 Talk highlights 13

3.3.3 Day 1 / Session 1 - Conclusions 17

3.4 Other DESEREC presentations 17

4 Conclusions 19

Appendix 20

Some thoughts for future RTD in secure software systems and services 21

An ICT for Trust and Security research project addressing the dependability of Information systems 26

Engineering Dynamic & ad-hoc Service Coalitions Design and Operational (run-time) TSD aspects 30

Serenity Framework Rules 33

Modeling services for trust and security assurance 38

Model-driven development of adaptive structures 43

Discovery, the Final Frontier 52

Security wrappers 59

Formal methods for the analysis of wide systems providing business services 61

Dependability and Security Metrics 70

Defining operational plans to provide dependability and security 81

Conclusions and plenary session 86

1 Introduction

The 2nd ESFORS Workshop on "Trust, Security and Dependability in Service Oriented Infrastructures" took place in Maribor (Slovenia) on July 10th and 11th, 2007, in the premises of the local Faculty of Electrical Engineering and Computer Science.

The workshop was organized by the Coordination Action ESFORS, with the cooperation of the European Commission DG INFSO unit F5, the European Technology Platform NESSI, the Slovenian Technology Platform NESSI, the Network of Excellence RESIST, and the Integrated Projects SERENITY and DESEREC.

Thanks to the valuable help of well known experts – such as Dr. Thomas Skordas (Project Officer, DG INFSO, EC), Prof. Antonio Lioy (Politecnico di Torino), Prof. Miroslaw Malek (Institute of Information, Humboldt-University of Berlin), Prof. Paulo Verissimo (University of Lisbon) and Dr. Gregory Chockler (IBM Haifa Research Laboratory) – researchers and managers, coming from academia and industry of several European countries and involved in European projects in the ICT field, found a good and valuable opportunity for exchanging and merging research experiences and drawing the lines for future activities.

The program of both days started with plenary sessions for introductory and keynote speeches, continued with the presentation of the European Projects officially represented and then went on with parallel sessions.

On the first day parallel sessions were dedicated to R&D gap analysis and future (long-term) research topics. These sessions were based on focused discussions to get common consensus around a selected set of topics, such as – but not limited to – security patterns, service composition and runtime issues, formal methods and specifications for design, development and testing of services, and presentations proposed by attendees and organizers.

On the second day parallel sessions were based on a "brainstorming" methodology, and instead of structured inputs, open contributions have driven the discussions, in particular focused on "Resilience in Services and Service Infrastructures".

Finally, a concluding plenary session has synthesized and merged the results coming from the previous sessions, designing future directions of research on Trust, Security and Dependability of service oriented infrastructures, from their design to run-time management.

DESEREC has provided several contributions to the workshop organization and sessions thanks to Prof. Antonio Lioy (Politecnico di Torino, Italy), keynote speaker on "Some thoughts for future RTD in secure software systems and services", and to Dr. Luca Durante (IEIIT/CNR, Italy), member of the program committee and chair of session 1 of day 1 "Engineering dynamic & ad-hoc service coalitions: Design and operational (run-time) TSD aspects", and to speakers from several other DESEREC partners.

More information is available at http://www.esfors.org.

2 Workshop Organization

The workshop was organized by Coordination Action ESFORS – European Security Forum for Web Services, Software and Systems, with the cooperation of:

- European Commission DG INFSO unit F5-Security
- European Technology Platform NESSI – Networked European Software & Services Initiative
- Slovenian Technology Platform NESSI - Slovenian Technology Platform for Software & Services
- Network of Excellence RESIST – Resilience for Survivability in IST
- Integrated Project SERENITY – System Engineering for Security & Dependability
- Integrated Project DESEREC – DEpendability & SEcurity by Enhanced REConfigurability

and people belonging to the program committee come from the above organizations and projects.

Program committee

Alberto Dainotti	Univ. Napoli, Italy
Aljosa Pasic	Atos Origin, Spain
Antonio Maña	Univ. Malaga, Spain
Bostjan Kezmah	Univ. Maribor, Slovenia
Domenico Presenza	Engineering, Italy
James Clarke	Waterford Institute of Technology, Ireland
Karama Kanoun	LAAS, France
Louis Marinos	ENISA Europe
Luca Durante	IEIIT - CNR, Italy
Pedro Carvalho	Univ. Lisbon, Portugal
Sandy Johnstone	HP, United Kingdom
Neeraj Suri	Univ. Darmstadt, Germany
Thomas Skordas	European Commission
Tomaz Domajnko	SRC, Slovenia
Volkmar Lotz	SAP, France

Thanks to the Program Committe efforts, almost 70 people attended to the workshop.

2.1 Workshop program

A plenary session has opened the workshop: welcome by Dr. Thomas Skordas (EC), two keynote speeches and the presentation of the main FP6 projects involved in the workshop organisation. Then three parallel sessions took place, addressing three different sides of "Services infrastructures". The second day has started with two keynote speeches, and then three parallel sessions addressing three sides of "Resilience in services and service infrastructures" followed. The workshop has been concluded by a plenary session, where the main results of the workshop have been introduced, and the future R&D trends in "Secure Software Systems and Services" have been summarized by each rapporteur.

Trust, Security and Dependability in Service Oriented Infrastructures
in the context of
Resilience in a Computer Supported Service Oriented Economy

10th July **Future R&D in Secure Software Systems and Services: Gap Analysis**		
09:30 10:00	Welcome by Organisers and EC	
	"Security, dependability and Trust in ICT-FP7: Coming Issues" **Dr. Thomas Skordas** (Deputy Head of Unit INFSO-F5 "Security", EC)	
10:00 11:00	**Keynote Speeches**	
	"Some thoughts for future RTD in secure software systems and services" **Prof. Antonio Lioy** (Politecnico di Torino, Italy)	
	"The Power of Prediction for Adaptive, Dependable Service-oriented Computing" **Prof. Miroslaw Malek** (Institute of Information, Humboldt-University of Berlin)	
11:00 11:30	Report on conclusions of previous Paris Workshop	
11:30 13:00	FP6 Project Presentations: **DESEREC, SERENITY, Resist**	
13:00 14:00	Lunch	
14:00 18:00	**Three parallel sessions** **Complementing previous workshop (prioritisation and gap analysis)**	
	1	Engineering dynamic & ad-hoc service coalitions: design and operational (run-time) TSD aspects Chair: **Dr. Luca Durante** (IEIIT - CNR, Italy) Rapporteur: **Dr. Jean Christophe Pazzaglia** (SAP Research Center, France)
	2	Scalable and adaptive ubiquitous service infrastructures Chair: **Dr. Antonio Maña** (University of Malaga, Spain) Rapporteur: **Aljosa Pasic** (Atos Origin, Spain)
	3	Alignment of security and trustworthy services: Interoperable security policies, business, socio-economic and legal aspects Chair: **Reijo Savola** (VTT Technical Research Centre of Finland) Rapporteur: **Prof. Bernhard M. Hämmerli** (ACRIS, Switzerland)

11th July **Resilience in Services and Service Infrastructures**		
08:30 09:30	**Keynote Speeches**	
	"Resilience Challenges in Service-Oriented Architectures" **Prof. Paulo Verissimo** (University of Lisbon, Portugal)	
	"Towards a Peer-to-Peer Middleware Platform for Highly Scalable and Robust Service-Oriented Computing" **Dr. Gregory Chockler** (IBM Haifa Research Laboratory)	
09:30 13:00	**Three parallel sessions** **Resilience in service oriented infrastructures**	
	1	Resilience in service oriented infrastructures Chair: **Dr. Edgar Weippl** (Secure Business Austria) Rapporteur: **DR. Pedro Carvalho** (University of Lisbon, Portugal)
	2	Resilience in Software Systems and Services Chair: **Prof. Peter Ryan** (Newcastle University, UK) Rapporteur: **Sandy Johnstone** (Hewlett-Packard, UK)
	3	Resilience in Business Processes Chair: **Luca Save** (DeepBlue, Italy) Rapporteur: **Domenico Presenza** (Engineering, Italy)
13:00 14:00	Lunch	
14:00 15:00	Conclusions and closing Plenary Session	

2.2 Parallel Sessions

Each parallel session has been organized in two parts: the first one for presentations proposed by voluntary attendees, selected during the weeks before the event and addressing one or more topics in a suitable set provided by the program committee. The goal of these presentations was to highlight the hottest topics in the field of security and dependability in service oriented infrastructures, and to provide the basis for the second part, where a public and open discussion about gap analysis, i.e. what has already been done and what it is expected or needed in mid / long term future research and development about "Trust, Security and Dependability in Service Oriented Infrastructures in the context of Resilience in a Computer Supported Service Oriented Economy".
Chairs and rapporteurs have collected the main emerged issues, and built a summary report presented in the closing plenary session where all summaries have been merged in order to provide a complete picture about the main lines of future R&D.

Day 1 / Session 1 - Engineering Dynamic & Ad-hoc Service Coalitions: Design and operational (run-time) TSD aspects

Rationale:
Dynamic and ad-hoc service composition processes and resulting coalitions have been introduced already in FP6 and a number of security RTD projects addressed trust, security and dependability (TSD) issues that these new trends will impose during the service engineering lifecycle, as well as during the actual deployment and operation of such coalitions. Some of the solutions that these FP6 projects are working on are typically making assumptions that might (or might be not) true. Furthermore, they are usually validated and tested in controlled closed environments where threats and vulnerabilities, as well as number of possible service combinations, are limited. Some of the research topics identified in the first ESFORS workshop, such as "Processes", "Security Engineering", "Formal

Methods, Semantics, Patterns, Design Tools, Validation Testing" , "Dynamic (run and real time) issues, context, dependence, complexity" etc belong to this group.

Objective:

The objective of this session is to perform gap analysis based on what has been done, under which assumptions (e.g. limited scale or not considering network infrastructure elements), and what else do we need. It should cover both the design and operational [run-time] aspects of SW systems and services, but also system testing and assurance related aspects. The architectural part needs to consider here the link with networks and network infrastructures, but also with embedded systems.

Day 1 / Session 2 - Scalable and Adaptive Ubiquitous Service Infrastructures

Rationale:

Introduction of service oriented software and systems is bringing various new challenges, besides those related to dynamic and ad-hoc nature of service coalitions, and these have to be also taken into account. The main problem is, that at this stage, we are not even fully aware of possible problems that service orientation will bring, or, in some cases, we are not aware of the scale of potential problems. In other words, this session will deal with UNKNOWN or underestimated problems of future internet and service oriented software and systems. This includes prediction and behavioral analysis, scalability of services and infrastructure components, service mutations and adaptability, co-existence and compatibility, flexible TSD etc. We also need to address the current solutions that, while solving one particular problem, potentially introduce a new vulnerability or threat in the future service ecosystems. Some of the research topics identified in the first ESFORS workshop include virtualization, trustworthy computing, trust and reputation engines, security as service, distributed authentication and authorization services, but also issues not mentioned in the first workshop such as on demand security level (and TSD trade off mechanisms), hosting and outsourcing TSD, security process networks, event driven security etc.

Objective:

This would be forward looking type of session with visionary and roadmapping objectives. In essence, the basic objective of this session is obtaining a successful prediction and for this purpose “as if” methodology could be used. We will stimulate innovative thinking by introducing new ICT scenarios and elements with stronger limitations when attempting the simulation or replication of the outcomes from previous research.

Day 1 / Session 3 - Alignment of Security and Trustworthy Services: Interoperable security policies, business, socio-economic and legal aspects

Rationale:

In the real world, decision making and investment in security is often a result of previous process, such as risk analysis, non-functional requirements, regulatory compliance etc., Building secure, yet dynamic business processes and business coalitions, should be also aligned with other issues such as business resilience or socio-economic aspects. Some of the research topics identified in the first ESFORS workshop include alignment with Business Objectives, change management, standardization and certification, interoperability, openness, the management of the security lifecycle, dynamic risk assessment and risk management, metrics, multidisciplinary approach, legal, socio-economic and socio-technical dimensions.

Objective:

In this session we look also on TSD environment and what we did wrong, what could have been done better or what changes have to be done in this environment. TSD is not standalone element nor does it have an objective in itself. This session should produce a list of recommendations of what further actions should complement technical work on secure services and systems.

Day 2 / Session 1 – Resilience in Service Oriented Infrastructures (SoI)

Rationale:
The "services" notion provides for a wide range of on-demand, scalable and adaptive functionality extending beyond the constraints of classical "systems". It also makes the user transparent to the details of the systems/infrastructures providing the services. However, the SoI level transparency/adaptability comes at the cost of ever increasing complexity of the underlying service interactions, and the infrastructures provisioning the services. Obviously, the utility of SoI's exists if they can provide stable services, resilient to either system, network or user level disruptions. SoI-resilience is not only a fundamental challenge but also an unavoidable one to make SoI's meaningful.

Objective:
The objectives of this session will cover (a) specification of the SoI operational space (the services, their interactions, the infrastructural aspects), and the related resilience considerations, (b) the mechanisms to provide for resilience in SoI (design-stage, operational-stage) and (c) measures of resilience and their validation.

Day 2 / Session 2 – Resilience in Software Systems and Services

Rationale:
Software and middleware increasingly determines the core functionality and the services offered by systems and Service oriented Infrastructures (SoI's). As the diversity and the underlying systems/SoI grow, it is the underlying SW/MW primitives that facilitate this growth. Unfortunately, to adapt to the dynamic growth of systems/SoI's, the SW/MW approaches are often reactive in nature, and with resilience as an add-on property. Our current SW engineering approaches to resilient SW/MW work well for discrete systems and applications. As large scale, adaptive functionality systems/SoI's are the future, and also operating in unpredictable environments, a paradigm shift in designing SW/MW is warranted! Especially the need to explicitly incorporate resilience into scalable, adaptive (for functionality, environment and threats) and distributed SW/MW design is needed. The need exists to develop theoretically well founded principles and practical techniques, methods and tools for engineering such future resilient software systems, middleware and services to help software designers and developers cope with the complexity of upcoming systems and SoI's.

Objective:
The session aims to promote new software and middleware paradigms. The intended coverage is for design and development methodologies, practices, techniques and tools which will promote a resilience-driven approach to the design, performance, scalability, extensibility and maintainability of the resulting systems, together with means of assessing these properties at run-time. The session will: (a) review the state of the art in foresight-based architecting and design processes for engineering resilience into software systems, middleware and services; This includes examining complexity, dynamicity and context from different angles; in order to define characteristics of all resilience dimensions; and, considering the extent to which existing technology (e.g., trusted computing, virtualization) in system designs provides options to aid the attainment of these; (b) perform a gap analysis on the challenges facing practical deployment of new methods.

Day 2 / Session 3 – Resilience in Business Processes

Rationale:
This session is about governance and engineering at organizational level of resilient business-critical services, related business processes, and associated assets. In line with the other sessions, Resilience is interpreted as the ability of an organization to perceive and cope with changes in the shape of risk induced by both internal and external events. The socio-technical nature of business organization has a profound impact on how resilience can be achieved in business-critical services and processes. The specificity of human "components" (or "liveware". as some authors refer to them) prevent from adopting at organizational level the traditional tools conceived for pure technical infrastructure. When considering complex socio-technical system like large business organizations, performance variability of people, along with variability of business, customer risk and technological environment, prevent from obtaining an adequate predictability of possible future events required by traditional risk models.

In such a kind of contexts threats could also emerge from combination of normal behaviors. Resilience in business processes of organizations/enterprises, hence, requires both new conceptual tools and techniques and a shift towards a new resilient way of thinking.

Objective:
The objectives of the session are to investigate: (a) which are the currently available solutions/practices to govern shift of processes, strategies, and responsibilities required to improve resilience and (b) directions and priorities for future scientific investigations. and technological developments.

3 DESEREC Role

This secsion is mainly focused on the role of the DESEREC project in the workshop, so it gives a partial view of the workshop. The complete view of the workshop is available at http://www.esfors.org. DESEREC has contributed to the workshop by means of:

1. a program committee member (L. Durante – IEIIT)
2. a keynote speech – "*Some thoughts for future RTD in secure software systems and services*" (A. Lioy – POLITO)
3. DESEREC presentation (A. Lioy – POLITO)
4. a session chair – Session 1 / Day 1 "*Engineering dynamic & ad-hoc service coalitions: design and operational (run-time) TSD aspects*" (L. Durante – IEIIT)
    a. some speakers in the session chaired by L. Durante:
        i. M. Aime (POLITO)
        ii. G. Csertán (BUTE)
        iii. L. Durante (IEIIT)
5. some speakers in other sessions:
    a. Session 2 / Day 1 "*Scalable and adaptive ubiquitous service infrastructures*"
        i. G. Csertán (BUTE)
    b. Session 2 / Day 2 "*Resilience in Software Systems and Services*"
        i. G. López (UMU)

This session addresses each contribution, and gives a detailed view, including presentations done by speakers not belonging to DESEREC, of the session chaired by L. Durante.

3.1 Program Committee membership

The L. Durante's activity in the workshop program committee has been mainly focused on:

1. the selection of the workshop topics
2. the definition of the workshop program
3. the selection of the contributions and speakers both of plenary (keynote speechs and speakers) and parallel sections

3.2 Keynote speech and DESEREC project presentation

The keynote speech "*Some thoughts for future RTD in secure software systems and services*" (A. Lioy – POLITO) was focused on two main subjects:

1. conflicts between business requirements and protection of the ICT infrastructure (and ways to solve these conflicts)
2. compliance checking both for internal (e.g. audit) and external (e.g. regulatory) purposes

Slides of this presentation can be found in Appendix and on the ESFORS website.

The talk about DESEREC "An *ICT for Trust and Security research project addressing the dependability of Information systems*" (A.Lioy – POLITO) described the scientific approach (based on formal specification and analysis tools) and the foreseen 3-level architecture (self-healing, fast reaction and optimal planning).
Slides of this presentation can be found in Appendix and on the ESFORS website.

3.3 Day 1 / Session 1 - *Engineering dynamic & ad-hoc service coalitions: design and operational (run-time) TSD aspects*

This section summarizes the output of the first session of the 2nd ESFORS Workshop entitled *"Engineering Dynamic & ad-hoc Service Coalitions Design and Operational (run-time) TSD aspects"* chaired by Dr Luca Durante IEIIT/CNR.

Slides of the chair presentation can be found in Appendix and on the ESFORS website

Each speaker also provided some "*State of the Art and Current Limitation*" background about the subject of her/his talk, and sketched some "*Gap Analysis*", i.e. which theoretical and/or technical limitation affects what she/he talked about. The concluding open discussion allowed to deepen the discussion, here summarized in the corresponding sections of each talk. Some further synthesis has been done, taking into account all the presentations, and summarized here in section 3.3.3 - *Day 1 / Session 1 – Conclusions.*

3.3.1 Agenda

The session was organised around three main topics related to the lifecycle of software components aiming to be deployed in the context of SOA. The speakers presented the vision developed within several EU funded projects namely SERENITY, DESEREC, DECOS, POSITIF and MOSQUITO but also by internal projects.

- **Design and Methodology**
  - "*Serenity Framework Rules*" (P. Soria - ATOS Research)
  - "*Modeling services for trust and security assurance*" (M. Aime - Politecnico di Torino)
  - "*Model-driven development of adaptive structures*" (G. Csertán - BUTE)
- **Dynamic and Operational Aspects of SW Systems & Services**
  - "*Discovery, the Final Frontier*" (J. C. Pazzaglia – SAP Research)
  - "*Security wrappers*" (A. Waller - THALES)
- **Formal Methods**
  - "Formal *methods for the analysis of wide systems providing business services*" (L. Durante - IEIIT/CNR)

Chair

Dr. Luca Durante (IEIIT - CNR, Italy)

Rapporteur

Dr. Jean Christophe Pazzaglia (SAP Research Center Sophia Antipolis, France)

Slides of the following presentations can be found in Appendix and on the ESFORS website.

3.3.2 Talk highlights

3.3.2.1 "Serenity Framework Rules"

P. Soria - ATOS Research – Serenity Project

Abstract

In the future, it is foreseen that large numbers of devices will have the capability and the need to communicate among themselves, as part of the provision of new services. Managing the security of such a large number of interconnections and devices will become even more complex. This talk presents a model for security and dependability solutions, developed by the SERENITY project, that will allow for easy, centralized management of security properties, while also enhancing the security of devices and applications.

State of the Art and Current Limitation

Design patterns in Software Engineering gained popularity in the mid-90's to capitalize the knowledge and to capture good practices in development. Initially dedicated to basic and well-known programming techniques (singleton, MVC, etc), they quickly broad their scope to address different issues including security.

Despite these efforts, patterns dedicated to TSD aspects are still problematic to capture and to reuse. These limitations are mainly due to the different models/abstractions used to describe them but also to the lack of tools enabling to use and to compose them automatically during the development phase.

Gap Analysis

The development of a framework enabling to capture, manipulate and inject patterns in the different software phases (requirement, development and monitoring) will greatly benefit to the community. This framework should be able to cope with automated combination of orthogonal (Security, Dependability, …) and vertical (Organization, Processes, Infrastructure) aspects. Once an adequate framework developed, the expert community will be able to create a large library of reusable TSD patterns that can be used in everyday work by business analyst, software developer and network administrator.

3.3.2.2 "Modelling services for trust and security assurance"

M. Aime - Politecnico di Torino – DESEREC Project

Abstract

The DESEREC project has defined a modelling framework for describing services and their configuration on top of the system infrastructure. Based on these models we compute security and dependability metrics for planning and deploying management procedures in the live system.

In this talk we discuss how to extend this approach to ad-hoc service compositions. Moreover, we discuss how assess security and dependability assurance of resulting coalitions.

State of the Art and Current Limitation

In the past decade, specific description languages have been tailored to model different aspects of systems. Concurrently reasoning techniques have been enhanced to analyse the security and dependability properties of different subcomponents of the solution (for example network architecture). The modelling cost and the absence of overall framework suitable to analyze the global properties (for example security level) and to capture the dynamic of SOA solutions is a major limitation of the existing work.

Gap Analysis

The introduction of an overall modelling framework enabling to capture TSD & compliance for SOA will enable to analyse their expected behaviour, reliability and threat resistance. In order to provide both qualitative and quantitative analysis, this framework would benefit to the definition of metrics suitable to measure TSD properties, these metrics could then be used at run-time for monitoring purpose. Finally, in order to circumvent the cost associated with a formal definition of the system, we can envisage to upgrade automatically the model based on the observation of the running system.

3.3.2.3 "Model-driven development of adaptive structures"

G. Csertán – BUTE – DECOS and DESEREC Projects

Abstract

The DECOS project addresses the development of safety-critical and dependable systems in the automobile, aerospace and industrial control domain. One of its main goals is to provide model based methods and tools for the different development phases including specification, design, implementation, validation and verification. Tools are integrated into a coherent tool chain in order to fully support the MDD (model driven development) approach on an integrated platform.

State of the Art and Current Limitation

In order to cope with the critical constraints involved in the development addressed by DECOS, the chosen approach was heavily model driven. The broad scope of the solution raises the issue of the lack

of integration between the different tools and the difficulty to introduce new tools in the development chain. Using a formal approach, required by the model-driven nature of development often requires *mathematical* tools and skills that are not found in mainstream developers and analysts. Formerly models have been transformed in an ad hoc manner between tools (point-to-point).

Gap Analysis

The main challenge of such approach is to seamlessly integrate formal analysis in the software development chain. In order to achieve this goal, tools should be open and enable multi-model manipulation in a coherent fashion. Models and analysis techniques should also evolve to deal with more dynamic aspects that are present in modern hardware and software platforms.

3.3.2.4 "Discovery, the Final Frontier"

J. C. Pazzaglia – SAP Research – MOSQUITO Project

Abstract

Highly dynamic systems such as Internet-wide SOA model face a set of challenges with respect to Services Discovery. In such applications, the discovery strategy should cope with the heterogeneity of services and platforms, with the complex semantics of service descriptions, with the scalability of the solution, and with mandatory requirements from a security and dependability perspective. In this talk we will highlight the research challenges related to Services Discovery, a field which is largely ignored.

State of the Art and Current Limitation

Service Discovery is a critical element to achieve a vision of highly dynamic and self configurable delivery of services in SOA environment. Despite this importance, current service discovery are limiting themselves to core technical functions (WS-Discovery) or mimics registries (white, yellow and green pages in UDDI). Although extensions have been proposed to use metadata or ontology based description of services, discovery protocols assume basic matching capabilities on the registry side and never take into account how to build a coherent chain of services complying with non-functional properties (SLA, Security, etc). Similarly, specific threats to the discovery phase are not addressed in standards and require enhanced cryptographic mechanisms to cope with large population of services.

Gap Analysis

New Service Discovery protocols should go behind state of the art security mechanisms (SSL) to take into account security and privacy concerns specific to the lookup or advertisement phases. The introduction of a common, standard, language dedicated to TSD properties would also enable to take into account non-functional aspects by the registries or in peer to peer setup. The description of these features should be used during the matching stage; moreover they may require using advanced discovery protocol enabling negotiation. Ultimately, discovery services should also be able to propose coalition of services able to insure end-to-end guarantee in term of security, QoS or SLAs.

3.3.2.5 "Security wrappers"

A. Waller – THALES

Abstract

When dynamically constructing systems from component services, there is a problem of matching the security requirements of the system as a whole to the security provided by these services. Component services that meet all of the security requirements may not be available, and even if they are, they may not be fully trusted. In this talk, we put forward the idea of "security wrappers" as a way to deal with this problem.

State of the Art and Current Limitation

When creating an ad hoc, dynamic system-of-systems from component services one should perform a security requirements analysis for the whole system, and should select individual services to meet the system requirements (including security requirements). However, it is difficult to discover the security

properties of component services and services meeting these security requirements may not be available. Finally, even if they are available, they (and their operators) may not be fully trusted. Wrapper technologies have been developed to address interoperability or safety features, we argue that similar technologies can be used to insure TSD properties.

Gap Analysis

The development of "security wrapper" around each individual component service could be part of the solution. They may enable monitoring the compliance of partially trusted component services with the security policy/requirements. They can also be used to enforce or otherwise compensate for security requirements that are not met by component services. These security wrappers would need to be tailored to individual services, and, ideally, could be created at runtime. Security patterns may be used to select appropriate monitoring and enforcement technologies from a "toolbox" for wrapper creation.

Only, a small amount of research has been done in this area and in related technologies, but it appears to be a potential long-term research challenge candidate.

3.3.2.6 "Formal methods for the analysis of wide systems providing business services"

L. Durante - IEIIT/CNR – DESEREC Project

Abstract

On large and complex distributed systems hardware and software faults, as well as vulnerabilities, exhibit significant dependencies and interrelationships. Being able to assess their actual impact on the overall system dependability by means of exhaustive formal analysis is especially important. Here we propose a unifying way of describing a complex hardware and software system, in order to assess the impact of both vulnerabilities and faults by means of the same underlying reasoning mechanism, built on a standard Prolog inference engine. Some preliminary experimental results show that a prototype tool based on these techniques is both feasible and able to achieve encouraging performance levels on several synthetic test cases. Moreover, the underlying simplifying assumptions are deeply investigated and justified, and the main open issues are highlighted.

State of the Art and Current Limitation

Formal methods allow performing exhaustive analysis with mathematical rigor on systems according that the system has been formally described.

Security analysis deals with the security properties of a system. It checks what can happen if a malicious agent can gain access to hosts in the network. Typically a host can be attacked if it suffers from some vulnerabilities. Indeed, our analysis in this sense in focused on the study of potentially exploitable vulnerabilities in the system. This kind of analysis bring as results, information on which hosts in the network can be compromised by an attacker.

Dependability analysis, instead, takes the form of determining the dependencies within the nodes and/or services of the network. Then it shows the effects / consequences of faults in an element (node, service) of the system.

These analysis have much in commons since they both need a precise modelling of the elements of the system analysed and they work on dependency models. While these techniques are able to provide meaningful results, they have to be scalable in order to handle real large network. Moreover to insure a large coverage they should cope with a large set of languages used to describe different elements, and they will greatly benefit from standard scanners able to gather information from heterogeneous systems both at deployment and run-time.

Gap Analysis

Today standard languages able to describe the whole system in an integrated way do not exist, while different languages can be used, the semantic attached to shared concepts may be slightly different and may cause some inconsistencies to compute global analysis. Moreover, in order to perform formal analysis these languages are processed or, even worst, translated *by hand* to an abstract model and the

analysis results are expressed in term of the abstract system models. However, to help the designer, the results of the formal analysis (carried out on abstract system models) have to be translated back to the abstraction level daily used in the design stage. The current models and analysis techniques have difficulties to handle large systems: this is mainly due to the complexity of the involved computation but sometime also to the complexity to handle result of the analysis. Moreover vulnerability and fault propagation analysis is feasible on networks providing high level business services if simple vulnerability, fault and dependency models are used. Dynamic behaviours of components such as modern firewall and routers are difficult to model and naïve approaches are suffering from an explosion of states during formal analysis.

The current analysis frameworks also lack of integration among the different useful tools that can be used in the design and run-time phases and that may involve simulating, emulating, testing and monitoring techniques. Finally the lack of suitable languages and standards for representing all the involved information is a limitation both for designers and for the development of tools able to automatically collect information.

3.3.3 Day 1 / Session 1 - Conclusions

A consensus quickly emerged on the lack of, and the difficulty, in providing a global, integrated and uniform vision of the different components of IT services. The problem is complex and may involve the development of languages able to provide a horizontal (functional and non functional properties of systems) and vertical (different layers form organization structure to the operating system and devices) coverage of systems. Since we are conscious of the difficulty to populate such model when dealing with already existing systems, we do believe that this development should take into the very beginning the possibility to implement a feedback loop between runtime and the model (monitoring, self configuration, …) by means of scanners, for example, together with the ability to populate automatically the runtime aspects of the model during deployment.

To be effective such model should also be able to adapt itself to the different usage, needs and users (from business analyst to system administrator) and be natively integrated in standard software development tools. Model and associated analysis techniques should evolve to enable modular design and to offer better support for dynamic aspects of resource access and loosely coupled systems supported by a set of flexible and open tool chain. This will ultimately benefit to the simplification of the design, development and deployment phases.

3.4 Other DESEREC presentations

A couple of presentations about DESEREC, done by DESEREC partners, have been done in other sessions. Here some details.

A copy of the presentations can be found in Appendix and on the ESFORS website.

Day 1 / Session 2 – "*Scalable and adaptive ubiquitous service infrastructures*"

Dependability and Security Metrics

G. Csertán - BUTE – DESEREC and DECOS Projects

Abstract

A metric is a precisely defined method to associate a number with an attribute. In IT systems the metrics are used to assess the system against non-functional requirements such as performance, availability, throughput, security. The main problem in defining metrics is that their semantics should be clear, but diverse interpretations in various contexts should be allowed. In DESEREC an engineering oriented metrics system has been defined that supports the continuous monitoring of performance, dependability and security aspects of the system and allows for both fast cicatrization and hot reaction for system reconfiguration.

Day 2 / Session 3 – "*Resilience in Business Processes*"

Defining operational plans to provide dependability and security

G. López - UMU – DESEREC Project

Abstract

The aim of this contribution is to describe the current work being done in the DESEREC (DEpendability and Security by Enhanced REConfiguratiblity) project, about the definition, design and deployment of an Operational Plan for dependability and security infrastructures. This plan includes the set of Operational Configurations available to be used on a real system and the set of Detection and Reaction scenarios related to each configuration.

This contribution will briefly introduce the DESEREC modelling framework, based on the modelling of requirements, policies and configurations by means of a layered infrastructure. Then, it will be focused on the definition of the meta models for operational plans, which includes the definition of high and low level concepts: Operational Plans (OPs), Operational Configurations (OCs); and Detection and Reaction Scenarios.

4 Conclusions

The first DESEREC dissemination workshop, held in the 2nd ESFORS workshop "Trust, Security and Dependability in Service Oriented Infrastructures", was a successful event, both because of the role of the DESEREC project in the organization of the workshop and the number and the scientific and technical quality of the presentations provided by the DESEREC partners.

Moreover, the joint official presence of four European Projects has allowed, on the one hand, to show the DESEREC technical and scientific achievements to a broader audience and, on the other one, to compare, share and merge the DESEREC experience with other European projects on Trust, Security and Dependability, leading to a common view of the gap between the current approaches and solutions and the real needs of the ICT community, i.e. scientists, managers and users. In particular, see slides "Conclusions and plenary session" in Appendix and on the ESFORS website, the DESEREC approach has been explicitly indicated as the leading one in trying to fill this gap.

Appendix

This appendix collects all the presentations done by DESEREC speakers and all those done in the session chaired by L. Durante (IEIIT). Also the presentation “Conclusions and plenary session” has been added.

Some thoughts for future RTD in secure software systems and services

Keynote speech – A. Lioy

Some thoughts for future RTD in secure software systems and services

Workshop on

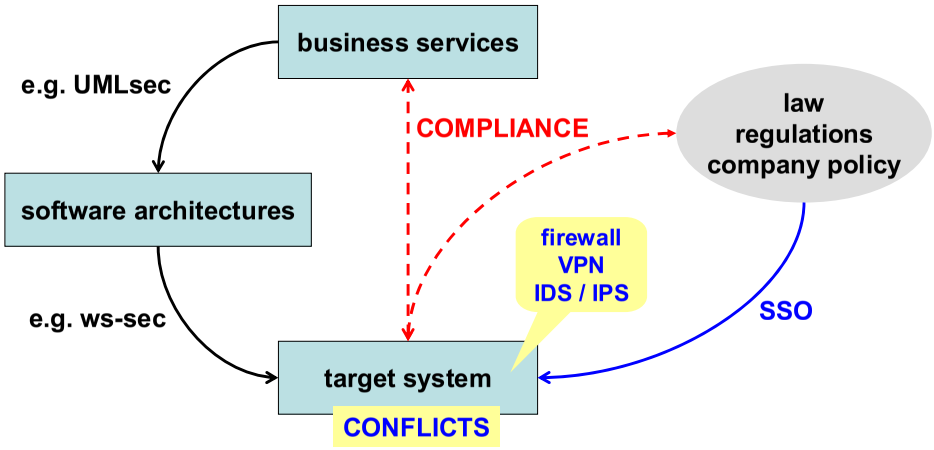
Software and Service Development, Security & Dependability

Antonio Lioy
Politecnico di Torino
Lioy@polito.it

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Services, software and hardware



Funded by EC contract FP6-027599

Conflict example #1

- ***my service is secure because it uses WS-security***
  - e.g. SOAP over HTTPS
  - e.g. XMLencryption
- … but the SSO says "any kind of encryption is forbidden on the Intranet because it prevents data inspection by the IDS"

Funded by EC contract FP6-027599

Conflict example #2

- ***this service is based on RPC and therefore it will use a random port in the range 1024-65535***
- ***please allow all these ports through the firewall***
- and the answer of the SSO is …
  - YES = no security
  - NO = no service

Funded by EC contract FP6-027599

Conflict example #3

- ***this service runs on port 80/tcp***
- ***please allow all this port through the firewall***
- what's the problem here?
  - packet filter is happy … but low security
  - application gateway wants to know application protocol (e.g. SOAP over HTTP)
  - semantic firewall and IDS wants to know authorized SOAP messages and users

Information Society Technologies
Funded by EC contract FP6-027599

Design methodologies and security

- design methodologies must be security-aware
- wrt the underlying system, sw architects must know:
  - its security capabilities
  - its security constraints
- automate design (and config and mgmt) as much as possible
  - e.g. automatic fw design (min config … max sec)

Information Society Technologies
Funded by EC contract FP6-027599

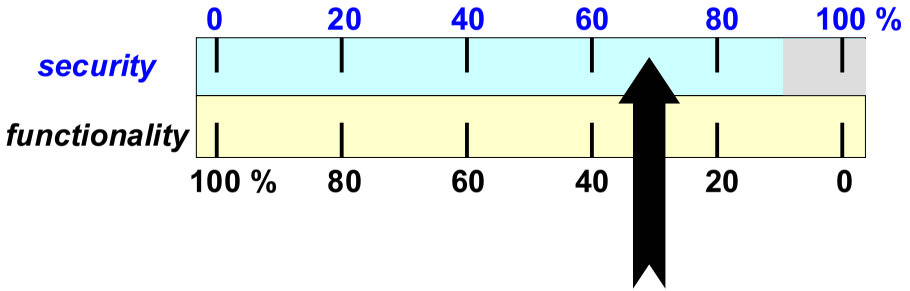
Automation

- complete specification is needed
  - business rule, policy, requirement, …
- conflicts are possible
  - must be solved (and solution recorded back into specification)
- not a top-down approach
  - rather a backtracking process (trial-and-conflict) ...
  - … to find a nearly optimal trade-off

Information Society Technologies
Funded by EC contract FP6-027599

Trade-off

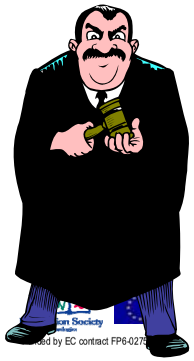
- be ready to give-up some functionality or performance for security



Information Society Technologies
Funded by EC contract FP6-027599

Compliance checking

- most often asked question today:
  "are you compliant with (*insert your favourite para-legal regulation here*)?"
- design, develop and manage an ICT service according to some regulation is not enough
- you must be able to prove it!



Funded by EC contract FP6-027599

Be ready

- a complete specification helps the auditor
  - e.g. auditing firewall rules (packet-level) with no knowledge of the business ruels
- an automatic refinement process helps to demonstrate the absence of errors (but for bugs in the process itself …)

Funded by EC contract FP6-027599

Dependability

- as for security, we'll never achieve 100% (even with a lot of redundancy)
- so what is it about?
- cost-benefit analysis (or investment done vs. prevented loss)
- ability to predict system behaviour
  - to demonstrate compliance
  - to prepare alternate conifigurations

Funded by EC contract FP6-027599

System behaviour prediction

- real system test
  - late!
- emulation
  - reduces complexity, introduces inaccuracy
- simulation
  - needs basic models, pruning of the infinite input space
- formal analysis
  - needs analytical model, must cope with large systems

Funded by EC contract FP6-027599

Computer scientist and engineer

- be a scientist:
  - new theory must explain known facts, must predict the results of new experiments
  - analyzer must reproduce known behaviour and predict unknown behaviour
- be an engineer:
  - can't design/control a thing that you can't measure
  - define a metrics (different from measure!)
  - do approximations, ignore details

Information Society Technologies
Funded by EC contract FP6-027599

Example

- MDA
  - model of system, service and business rules
  - refine, design, implement, manage … always linking to the business rules
- when problem P occurs, alternative configurations C1 and C2 are possible
  - (analysis of C1) 80% of customers working
  - (analysis of C2) 30% of customers working (but this includes 98% of premium users)

Information Society Technologies
Funded by EC contract FP6-027599

Questions?

- thanks for your attention

Politecnico di Torino

TORSEC group

POSITIF

DESEREC

Information Society Technologies
Funded by EC contract FP6-027599

An ICT for Trust and Security research project addressing the dependability of Information systems

Presentation of the IP project DESEREC – A. Lioy

DESEREC
Dependability and Security by Enhanced Reconfigurability

An *ICT for Trust and Security* research project addressing
the dependability of Information systems

speaker: Antonio Lioy (Politecnico di Torino)

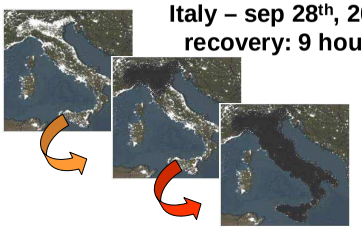
DESEREC
Dependability & Security by Enhanced Reconfigurability

Information Society Technologies

Dependability concerns

- the everyday life of European citizens relies on critical activities supported by networked Information Systems (I.S.):
  - **Communications** (telephone, Internet)
  - **Energy & fluids** (electricity, gas, water)
  - **Transportation** (railways, airlines, road)
  - **Health and emergency response**
  - ***e*-Government**

- so far, limited taken actions let these I.S.
  - not failure-proof enough to face:
    - **software & hardware faults**
    - **malicious actions: intrusion, virus**
  - with poor self-healing capability
    - **and therefore sensitive to cascading effects**
  - suffering long recovery time



Italy – sep 28th, 2003
recovery: 9 hours

- DESEREC aims to leverage those capabilities
  - in new and existing Information Systems

Why DESEREC?

The picture

- administrators are swamped by information of inappropriate level
- most of the decision are taken short-term, with poor mid-term capability to arbitrate between business services with different criticality
- no synthetic view on dependability is provided

The proposed approach

- provide information and interaction at *service* level instead of *component* level for day-to-day management
- create high-level management capabilities giving the ability to react appropriately upon errors/failures to maintain critical services
- support mid-term strategy with planning and simulation tools enabling a proactive management of performance and dependability

The 3-tiered approach proposed by DESEREC

First objective – ***Detect & Prevent***

- detect proactively incident and potential fault
- keep as much as possible every failure local
  - contain the incident: isolate the compromised area

containment

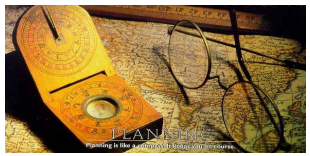
Second objective - ***React***

- Sustain or quickly resume the critical applications
- Reallocate resources used by less critical ones

reconfiguration

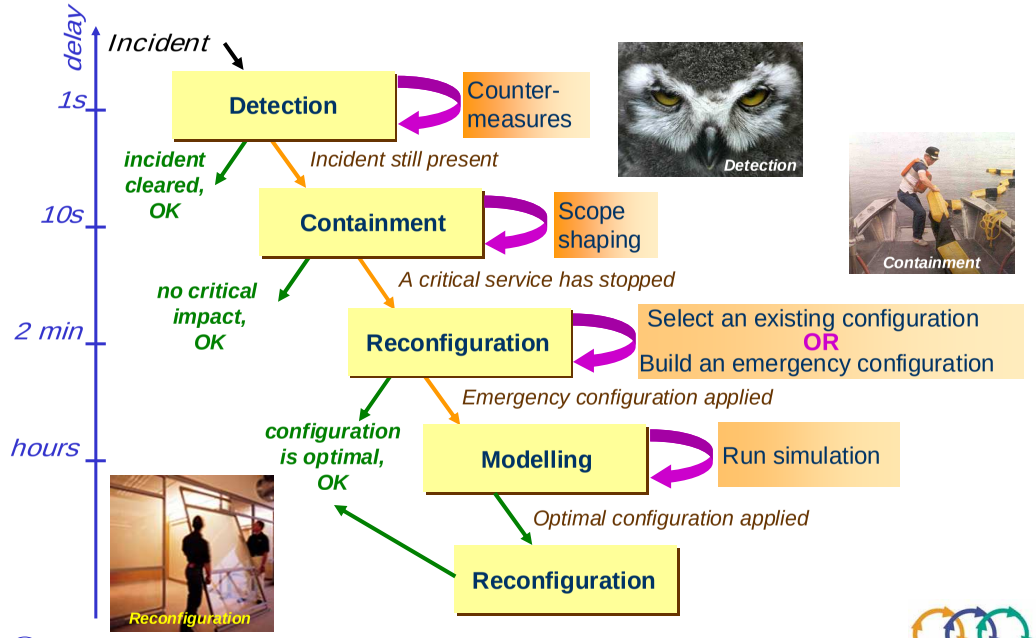
Third objective – ***Plan***

- Reallocate optimally the resources to recover the full range of services
- Validate the configurations by simulation



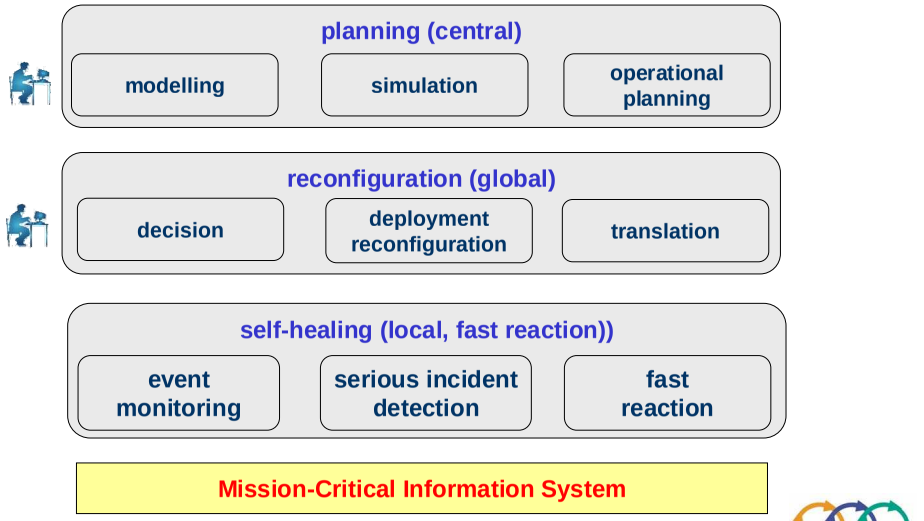
planning

DESEREC - A multi-tiered response

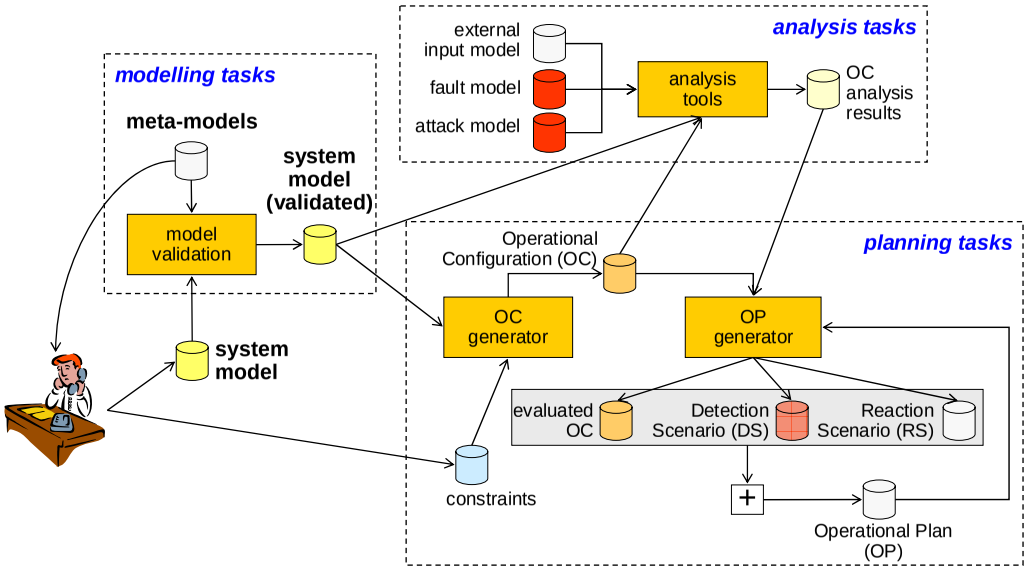


High level functional blocks

Management of mission-critical CIS via a model-based solution organised around three-tier reaction loop

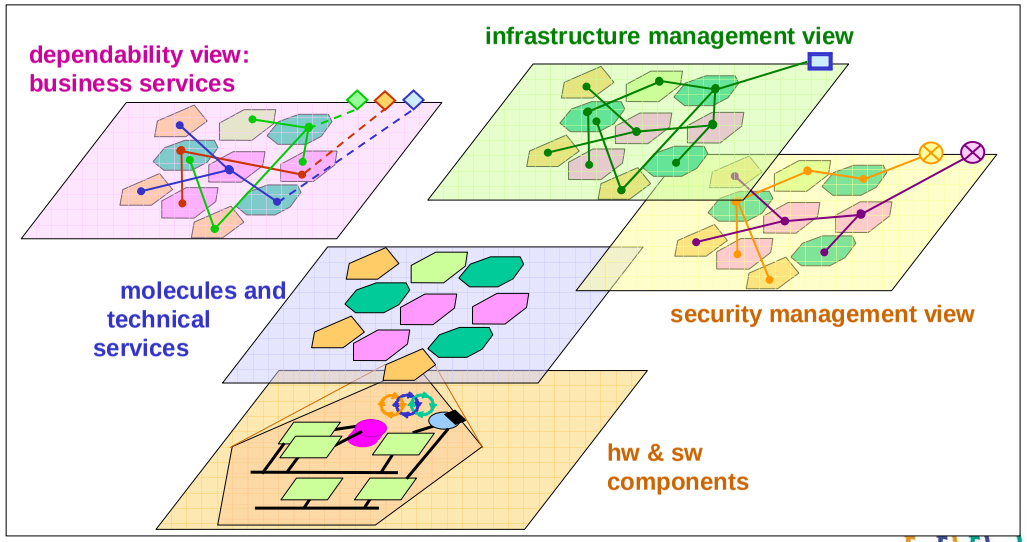


Generation of operational scenarios

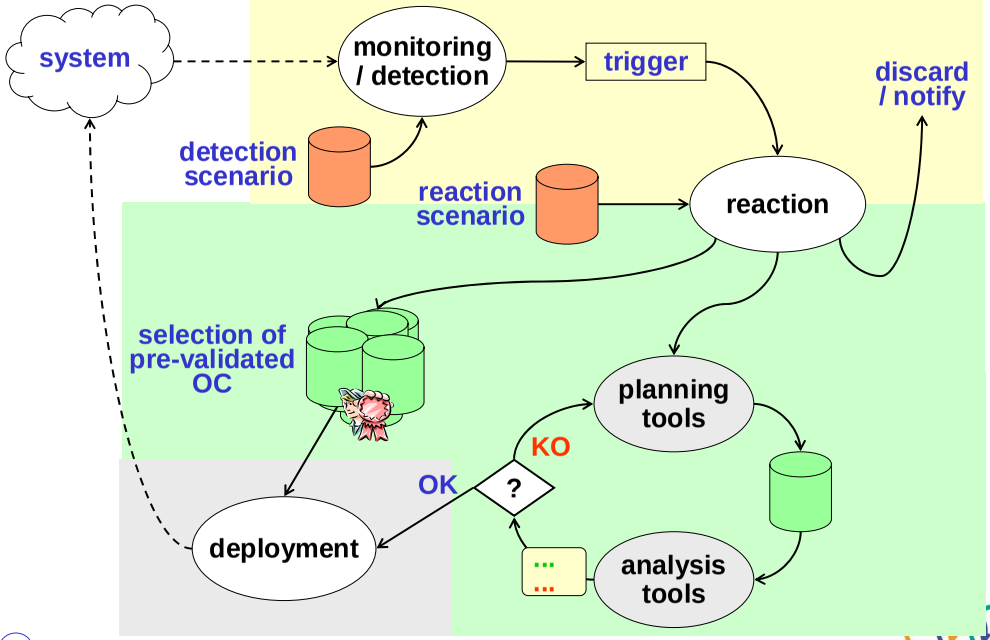


CIS seen as a cluster of molecules

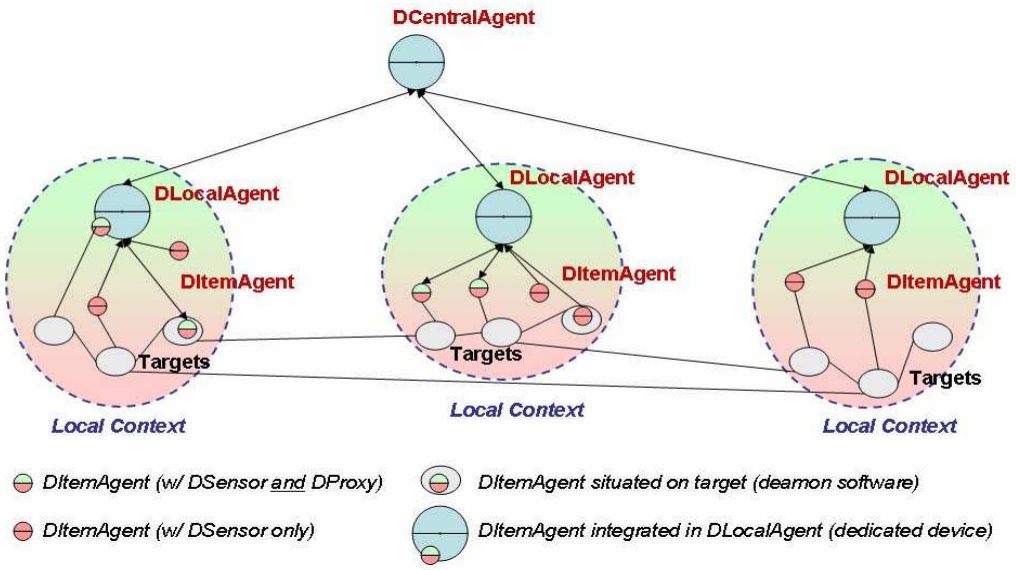
- system decomposition into sub-systems (=molecule) hosting technical services



Workflow – reaction



Architecture for Self-healing and Reconfiguration



Engineering Dynamic & ad-hoc Service Coalitions Design and Operational (run-time) TSD aspects

Day 1 / Session 1 chair presentation – L. Durante

Session 1 “Engineering Dynamic & ad-hoc Service Coalitions”

Design and Operational (run-time) TSD aspects

Workshop on

Software and Service Development, Security & Dependability

Luca Durante IEIIT/CNR
luca.durante@polito.it

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Design and Operational TSD aspects

- Design
  - Security engineering
  - Design tools
    - Modeling languages
    - Model-driven development of adaptive structures
  - Formal methods
- Operational (run-time) aspects of SW systems and services
  - Web services discovery

Funded by EC contract FP6-027599

Design and Operational TSD aspects

- The target is to perform gap analysis
  - What has been done
  - Under which assumptions (simplifying assumptions, isolation, …)
    - To reduce complexity (of design, analysis, run-time management)
    - To enable the use of available tools and technologies

Funded by EC contract FP6-027599

Design and Operational TSD aspects

- The target is to perform gap analysis
  - Future directions
    - Costs
      - Open research problems
      - Development of new technologies
      - Integration of existing technologies
      - Development of new tools
      - Integration of existing tools
    - New emerging requirements and needs

Funded by EC contract FP6-027599

Design and Operational TSD aspects

- Agenda of this session
  - Presentations about experiences coming from TSD Europeans projects
    - SEINIT
    - SERENITY
    - DESEREC
  - Lively Discussion

Funded by EC contract FP6-027599

Design and Operational TSD aspects

- Presentations
  - Design and Methodology
    - "Serenity Framework Rules" (P. Soria - ATOS Research)
    - "Modeling services for trust and security assurance" (M. Aime - Politecnico di Torino)
    - "Model-driven development of adaptive structures" (G. Csertán - BUTE)

Funded by EC contract FP6-027599

Design and Operational TSD aspects

- Presentations
  - Dynamic and Operational Aspects of SW Systems & Services
    - "Web Services Discovery" (J. C. Pazzaglia - SAP)
    - "Security wrappers" (A. Waller - THALES)
  - Formal Methods
    - "Formal methods for the analysis of wide systems providing business services" (L. Durante - IEIIT/CNR)

Funded by EC contract FP6-027599

Design and Operational TSD aspects

Thank you for your attention

Funded by EC contract FP6-027599

Serenity Framework Rules

P. Soria

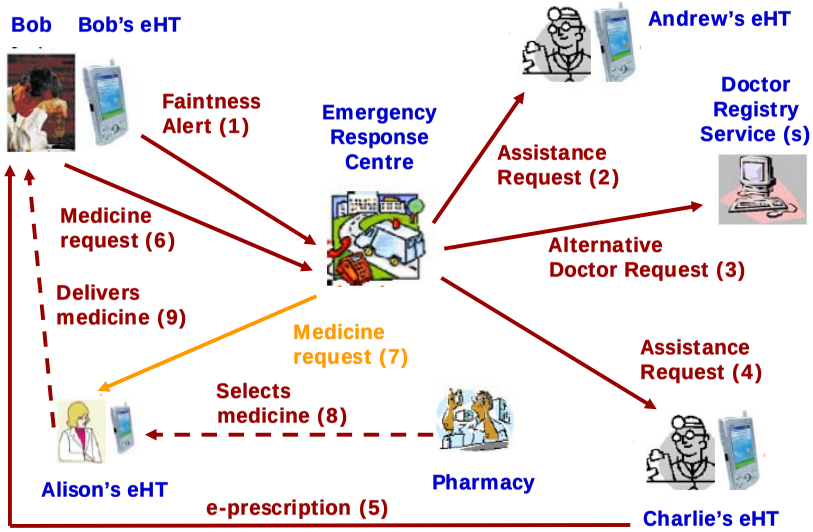
A. Maña ( )

Modelling of Security & Dependability Solutions

General approach and how this contributes to S&D in AmI

- Modelling of S&D Solutions
  - The goal of this presentation is to provide an overview of
    - The need to model S&D solutions in AmI
    - The artifacts used
    - The relations and usefulness of these artifacts

Example: Medicine request



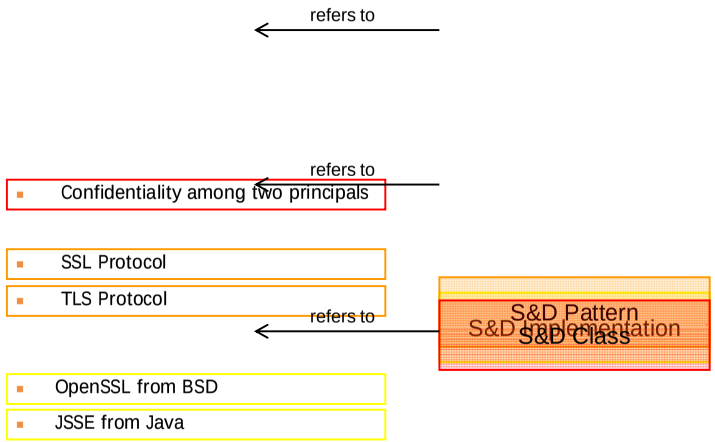
Real World Solutions

- The **scenario**:
  - Bob feels giddy and sends via his e-health terminal a request for assistance to ERC.
  - ERC receives the request and, since Bob's doctor is in vacation, redirects it to Charlie.
  - Charlie analyses Bob's medical data and history and sends to Bob an e-prescription.
  - Bob requests ERC for a medicine delivery.
  - ERC selects Alison to execute this task, sends a message to her to which she promptly acknowledge receiving then back the data for accomplishing this activity.
- Prescription transferred to Alison's eHT includes Bob´s personal information.
  - The developers of the eHealth System identified this requirement:
    - Confidentiality among two principals
  - Some solutions seem to provide it:
    - SSL Protocol
    - TLS Protocol
  - Alison's device and the ERC negotiates and SSL is the selected protocol. Now, several implementations of this solution are available:
    - OpenSSL from BSD
    - JSSE from Java

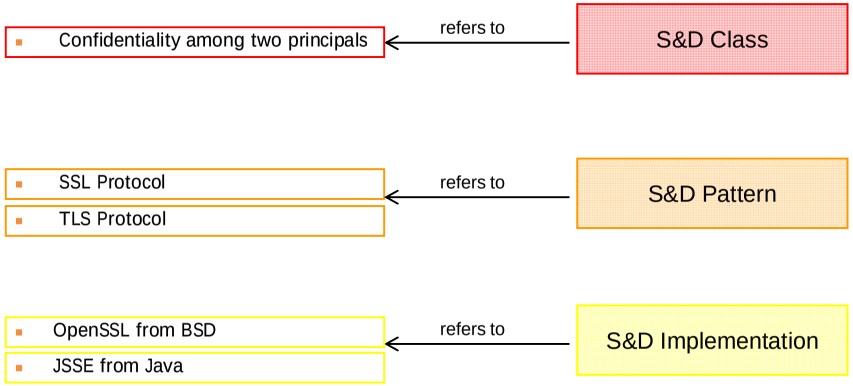
Real World Solutions

- The **scenario**:
  - Bob feels giddy and sends via his e-health terminal a request for assistance to ERC.
  - ERC receives the request and, since Bob's doctor is in vacation, redirects it to Charlie.
  - Charlie analyses Bob's medical data and history and sends to Bob an e-prescription.
  - Bob requests ERC for a medicine delivery.
  - ERC selects Alison to execute this task, sends a message to her to which she promptly acknowledge receiving then back the data for accomplishing this activity.
- Prescription transferred to Alison's eHT includes Bob´s personal information.
  - The developers of the eHealth System identified this requirement:
    - Confidentiality among two principals
  - Some solutions seem to provide it:
    - SSL Protocol
    - TLS Protocol
  - Alison's device and the ERC negotiates and SSL is the selected protocol. Now, several implementations of this solution are available:
    - OpenSSL from BSD
    - JSSE from Java

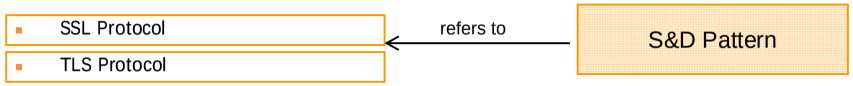
Real World Terms vs. Serenity Artefacts



S&D Patterns

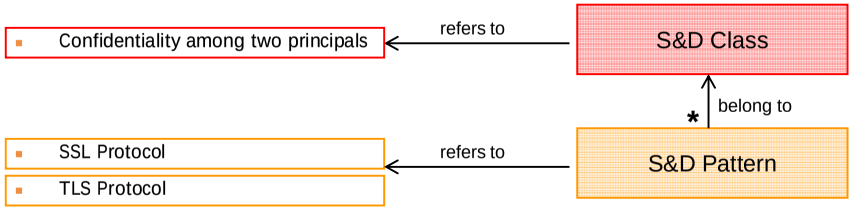


S&D Patterns



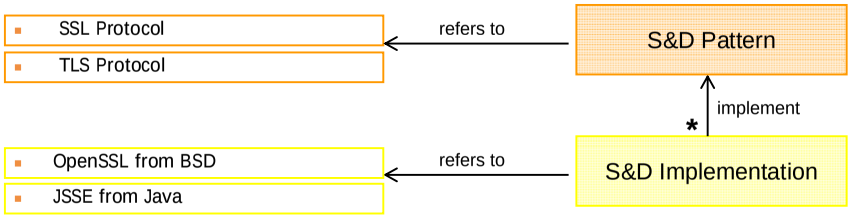
- S&D Patterns are detailed *descriptions of* abstract *S&D Solutions* (well-defined mechanisms that provide *S&D Properties*: confidentiality, integrity, availability, etc.)
- These descriptions must contain:
  - Information necessary for the selection, instantiation and *dynamic application of the solution* described in the S&D Pattern
  - Results of any *static analysis* performed on them
  - Support for the dynamic validation of the behaviour of the described solutions by means of *monitoring mechanisms*
- S&D Patterns represent not only simple solutions, but also complex ones:
  - An special type of S&D Patterns, called **Integration Scheme**, is used to represent solutions that are built by combining other S&D Patterns

S&D Classes



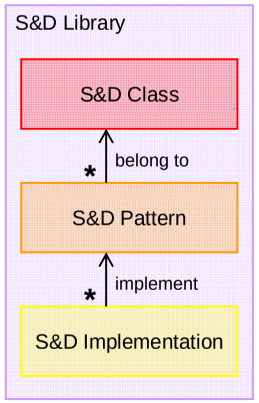
- An S&D Pattern belongs to One or Many S&D Classes given that:
  - **S&D Classes** represent abstractions of a set of S&D Patterns characterized for:
    - providing the same S&D Properties and,
    - Having compatible interfaces
- As this artefact defines a high-level interface, it is possible for developers to create an application bound to a specific S&D Class
- At runtime all S&D Patterns belonging to this S&D Class can be selected by the SERENITY Runtime Framework (depending on context)
- Thus, S&D Classes facilitate the dynamic substitution of the S&D solutions at runtime while facilitating the development process and supporting interoperability

S&D Implementations



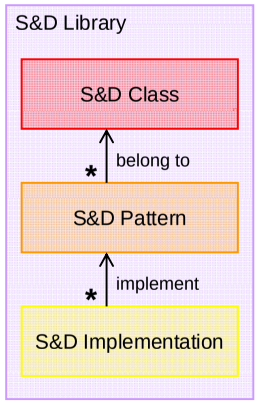
- **S&D Implementations** represent the realizations the S&D Solution
- All S&D Implementations of an S&D Pattern:
  - Must conform directly to the interface, monitoring capabilities, and any other characteristic described in the S&D Pattern
  - They may have differences, such as the specific context conditions that must be met before deploying it, their performance, target platform, programming language, etc...
- Note that S&D Implementations are **not** the actual realizations but their representation/description.

Rationale for the artefacts (I)



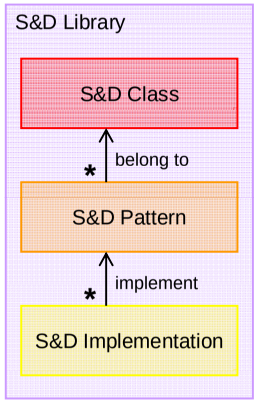
- From the point of view of the **type of information** they contain:
  - S&D Patterns can be verified: the results of the verifications concern only the abstract solution
  - S&D Classes can not be verified: they only provide the minimum amount of information required at development time
  - S&D Implementations: in general, it is not possible to verify software implementations

Rationale for the artefacts (II)



- From the point of view of their **role**:
  - S&D Classes: mainly interface definitions
  - S&D Pattern's role is to capture important information that will be used to ensure the correct usage of the solution at runtime
  - S&D Implementations: they also capture information but the represented knowledge is related to the specific realization

Rationale for the artefacts (III)



- Regarding the **producers** of the different specifications:
  - S&D Classes will be defined by entities mainly interested in interoperability (e.g. industry associations, standardization bodies)
  - S&D Patterns will be produced by independent entities interested in security and dependability (e.g. S&D Companies and Experts)
  - S&D Implementations will be produced by entities interested in the creation of working solutions (e.g. commercial solution providers)

Integration Schemes as S&D Patterns

- Interface specifications in S&D Patterns **describe** the **available functionality** and how other external entities would interact with the realizations
- They **do not describe** the **expected security behaviour and effect expected** from a composition of several solutions

- A **pattern** describes the semantics of a **simple solution**, it is not the description of the solution itself
- An **IS** describes the semantics of a **combined solution**, it is not the description of how this combination is done

- The implementation of a **pattern** is a **realization of the solution**
- The implementation of an **IS** is a **realization of the combination**

SERENITY
System Engineering for Security & Dependability

Thank you

Information Society Technologies

Modeling services for trust and security assurance

M. Aime

Modelling services for trust and security assurance

Workshop on

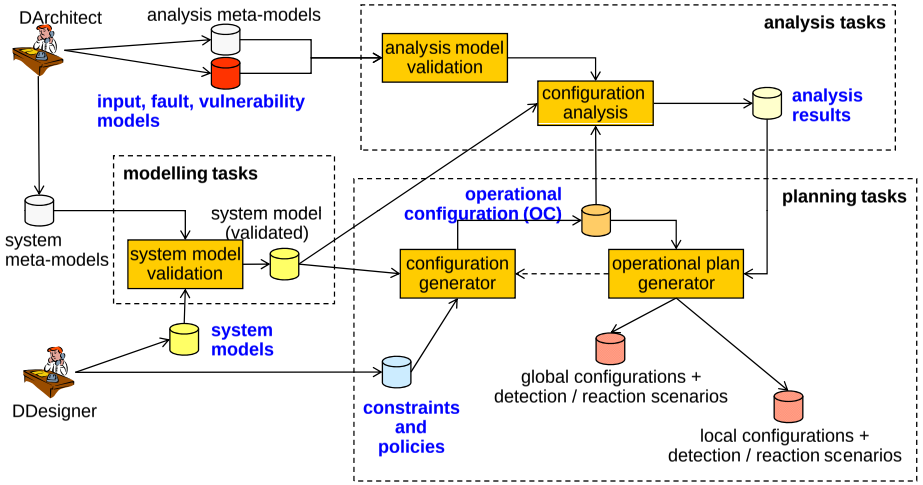
Software and Service Development, Security & Dependability

Marco Aime
m.aime@polito.it
POLITECNICO DI TORINO
DESEREC PROJECT

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

DESEREC planning tools



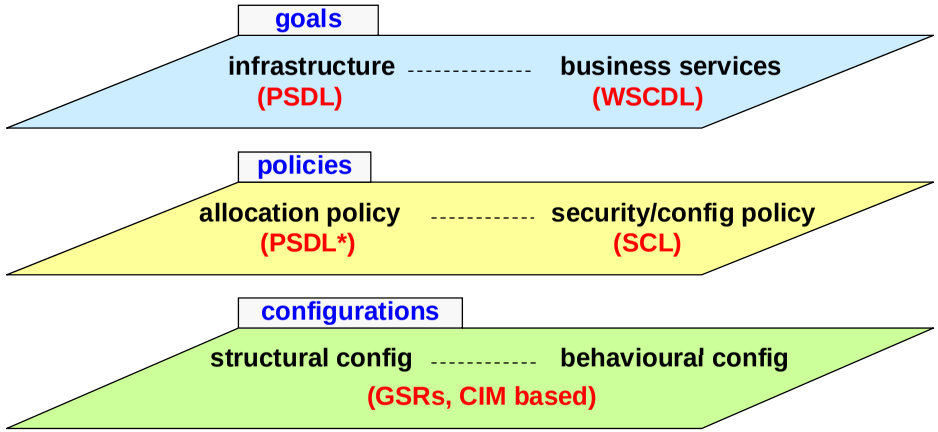
Funded by EC contract FP6-027599

Rationale

- designed for critical infrastructure protection
  - centralised approach
  - policy oriented
- some key traits can be extended towards ad-hoc service composition
  - modelling framework
  - evaluation metrics

Funded by EC contract FP6-027599

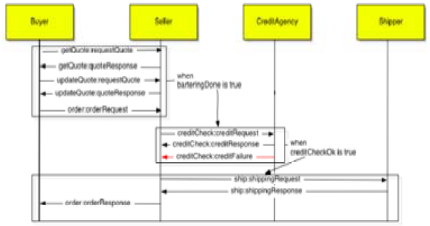
Modelling framework



Funded by EC contract FP6-027599

W3C's Web Service Choreography Description Language

- *Roles, Behaviours, Relationships*
  - define the logical view of the system
  - relationship between **two** roles (behaviours)
- *Participants, Channels*
  - group some roles, define the way in which participants talk
  - channel (instantiated as variables) can be passed through interactions
- *Choreographies, Interactions*
  - describe a (multi-participant) protocol
- *Variables*
  - information exchange (forwarding), states, channels, exceptions
- *Tokens, Token locators*
  - pieces of variables (locate somewhere), relevant for the system functions

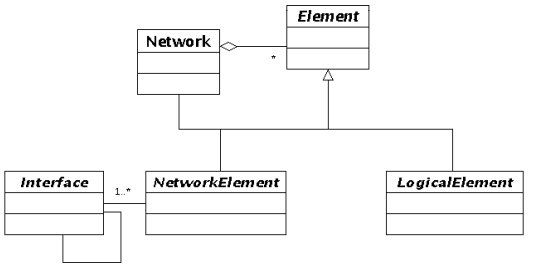


http://www.w3.org/ws/chor

Funded by EC contract FP6-027599

Positif's System Description Language

- Main elements
  - network elements: networks, hosts, hardware platforms
  - logical elements: software, technical services
  - connections between devices, hosts, networks



http://www.positif.org

Funded by EC contract FP6-027599

Policy model

- Allocation
  - constraints on the structural part of the configuration
  - **add:** additional resources added to the infrastructure (e.g. software, technical services)
  - **mapping:** maps roles in the service model to elements in the resource model
- Security and configuration policies
  - Language for modelling general-purpose service constraints: **SCL** (Services Constraints Language).
  - two types of rules: those which configure specific services, and those related to security which span all the system

Funded by EC contract FP6-027599

Configuration

- Generic Service Rulesets (GSRs)
  - merge the structural and behavioural part of the configuration
  - transform them into (abstract) element configuration rules
  - based on a subset of DMTF's **Common Information Model (CIM)**

http://www.dmtf.org

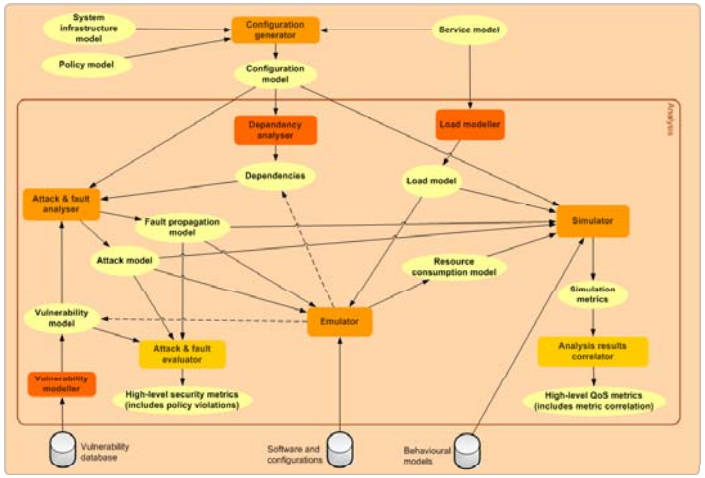
Funded by EC contract FP6-027599

Model-based configuration analysis

- Input models
  - system and configuration models (services, resources, policies, configurations)
  - vulnerability model, fault model
  - load model, resource consumption model
- Output models
  - low level analysis results
    - attack graph,
    - error propagation,
    - simulation events and statistics
  - high level metrics
    - service capacity, service performance,
    - protection level, policy violations,
    - observability , accountability

Funded by EC contract FP6-027599

Analysis framework (initial design)



Funded by EC contract FP6-027599

Toward ad-hoc service composition

- *What can be used for ad-hoc service composition?*
  - **system models**
    - layered approach: service workflow vs platform structure
  - **policy model**
    - can be extended for modelling Service Level Agreements
  - **metrics**
    - service capacity, service performance,
    - protection level, constraints violations,
    - observability, accountability
- *What is missing for ad-hoc service composition?*
  - **SLA modelling and validation**
  - **protocols to exchange models**
  - **distributed monitoring infrastructure**

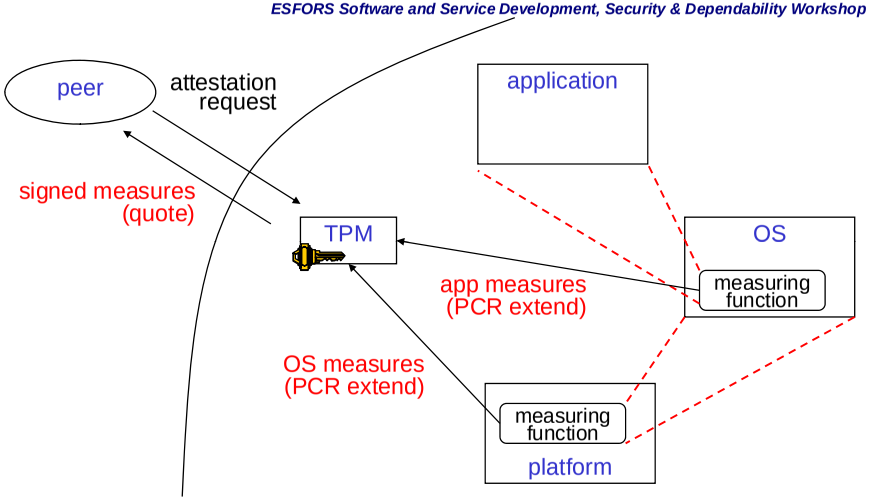
Funded by EC contract FP6-027599

Seeking assurance

- We need monitoring strategies and techniques
  - for monitoring SLA violations
  - for accounting
- Trusted computing may allow us to measure what is typically difficult to measure (e.g. security)
  - how to be assured that "the service will store user identity data in a protected way, e.g. encrypted"?

Funded by EC contract FP6-027599

TC's remote attestation



https://www.trustedcomputinggroup.org

Funded by EC contract FP6-027599

Toward ad-hoc service composition

- *What can be used for ad-hoc service composition?*
  - **trust** is having assurance of someonelse's behaviour (under some context)
  - **remote attestation** can give us some assurance in the structure and behaviour of our peers
  - from this information we can evaluate peers against our policies
  - both application and platform layers are essentials
- *What is missing for ad-hoc service composition?*
  - **attestation protocols**
    - support attestation within standard application protocols (e.g. TLS)
  - **model-based attestation**
    - only binary attestation now supported
    - property-based attestation, but practical examples lack
    - attesting more structured and formal data may solve the problem

Funded by EC contract FP6-027599

Model-driven development of adaptive structures

G. Csértan

ESFORS
European Security Forum for Web Services, Software and Systems

Model Driven Development of Adaptive Structures

Workshop on

Software and Service Development, Security & Dependability

György Csertán
csertan@mit.bme.hu

10-11 July 2007, Maribor

Budapest University of Technology and Economics

Funded by EC contract FP6-027599

Results of work on MDD tool-chain

DECOS

DECOS

Dependable Embedded Components and Systems

Funded by EC contract FP6-027599

MDD on example of embedded systems

But:

- resource mapping
- execution scheduling
- network / communication design
- wrapper generation / synthetisation
- code generation
- middleware mapping
- reconfiguration

are problems in SOA too.

Funded by EC contract FP6-027599

DECOS

- Facts:
  - IP6 project, 36 months, cca. 9 Mio Euro support, 19 partners
- Domain:
  - safety-critical, dependable systems
    - automobile
    - aerospace
    - industrial control
  - time-triggered communication platform

Funded by EC contract FP6-027599

DECOS (contd.)

- Partners
  - Industrial Partners:
    - Airbus, Thales, EADS, Audi,Infineon, Esterel, TTTech, Fiat, Profactor, Hella, Liebherr
  - Research Centers:
    - ARC Seibersdorf, SP Swedish Test & Res. Institute
  - Universities:
    - TU Vienna, TU Darmstadt, TU Hamburg, Uni Kassel, Uni Kiel, Budapest University

Funded by EC contract FP6-027599

ESFORS

European Security Forum for Web Services, Software and Systems

DECOS development tool-chain

Workshop on

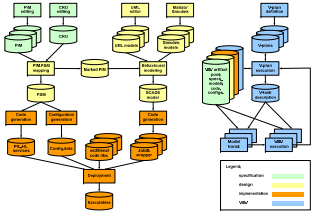
Software and Service Development, Security & Dependability

10-11 July 2007, Maribor

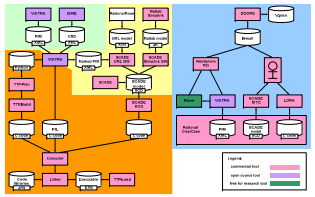
Funded by EC contract FP6-027599

Objectives of work

- Provide **methods & tools**
  - specification,
  - design,
  - implementation,
  - validation & verification.

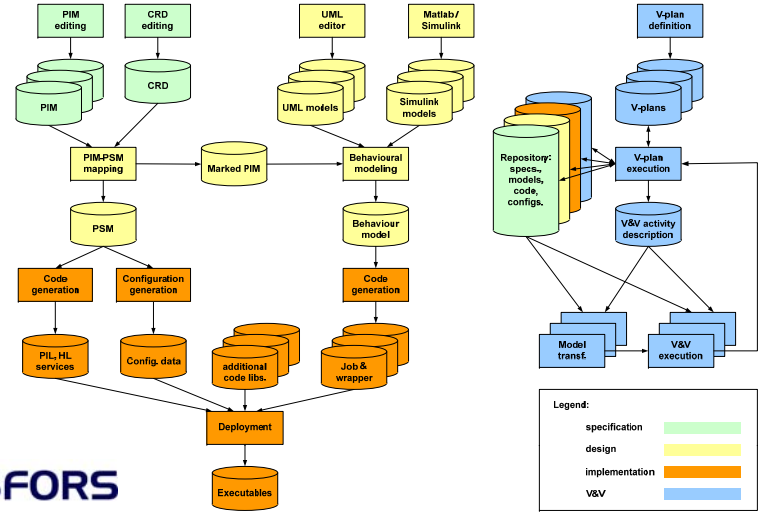


- Integration into a **tool-chain**
  - open,
  - extensible,
  - standards compliant.



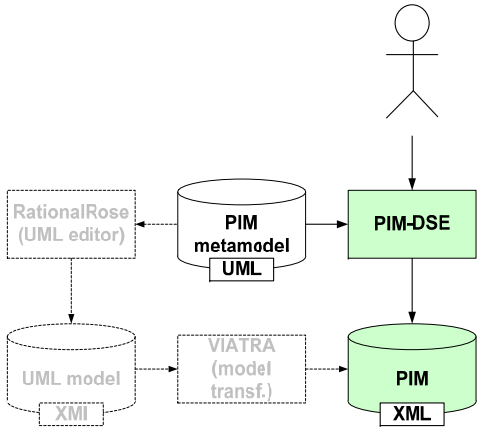
Funded by EC contract FP6-027599

Tool-chain functionality



1. Specification (DAS)

- PIM editing
  - functionality
  - dependability
  - performance



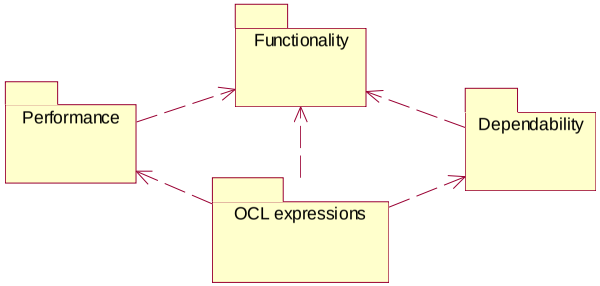
Funded by EC contract FP6-027599

PIM metamodel

- DAS, operating modes, jobs, resources (sensors/actuators)
- Interfaces, ports, communication
- Messages, state variables
- Assertions, defined states,
- Dependability
  - reliability, availability, redundancy degree, SIL
- Performance
  - WCET, interarrival time, latency
- SysML compliant, seems to be AADL compliant

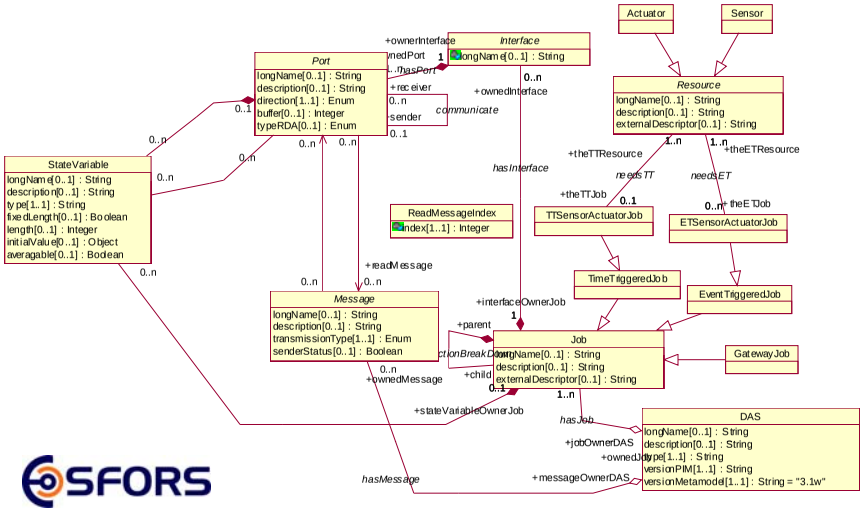
Funded by EC contract FP6-027599

PIM metamodel main package



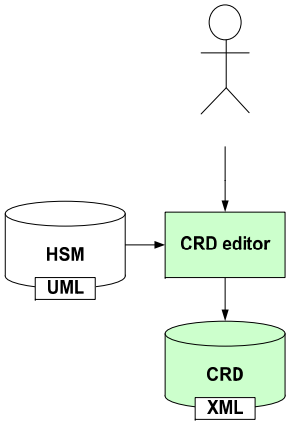
Funded by EC contract FP6-027599

Functionality package (part)



1. Specification (HW)

- CRD editing
  - cluster definition
  - network definition
  - resource definition
  - sensor / actuator definition



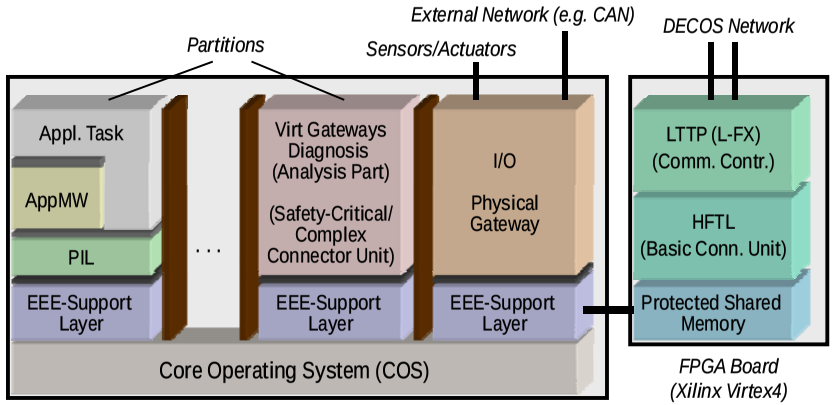
Funded by EC contract FP6-027599

HSM (CRD metamodel)

- Cluster, SubSystem, Component, Appl.Comp.
- ConnectorUnit, BCU
- Resources
  - CommunicationInterface, CommunicationController, Connector, Feature, FPGA, HardwareElement, HWProperty, Memory, Processor, Resource, NonVolMemory, VolMemory
- Network
  - LegacyNetwork, CoreNetwork, ConnectorNetwork, Fieldbus, PhysicalNetwork, PINetwork

Funded by EC contract FP6-027599

DECOS HW platform



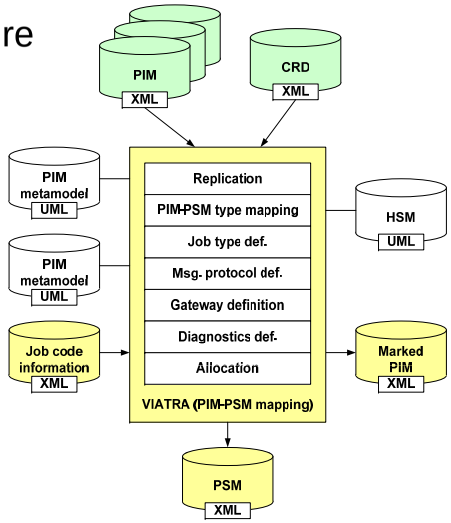
Encapsulated Execution Environment 'EEE' (TC 1796)

EEE-Support Layer: oFTL + SIL

2. Design (architecture)

Model Driven Architecture

- PIMs, PMs
- marking
- mapping
- PSM



PIM-PSM mapping

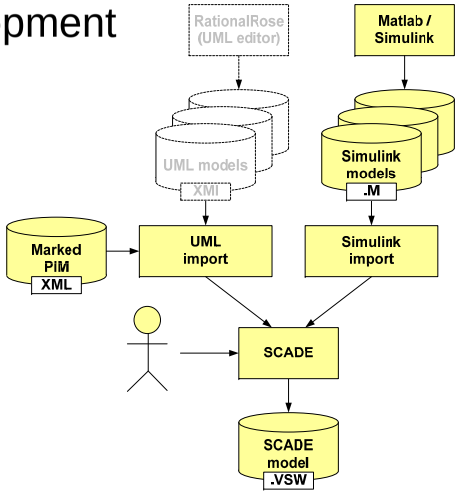
- Import: PIMs, CRDs
- Replication
- PIM-oFTL type mapping
- Interface protocol definition
- Job type marking
- Non-DECOS job allocation
- DAS interconnection mapping
- Resource allocation
- Job compatibility definition
- Job allocation
- PIL code and configuration generation
- Scheduling input generation

Funded by EC contract FP6-027599

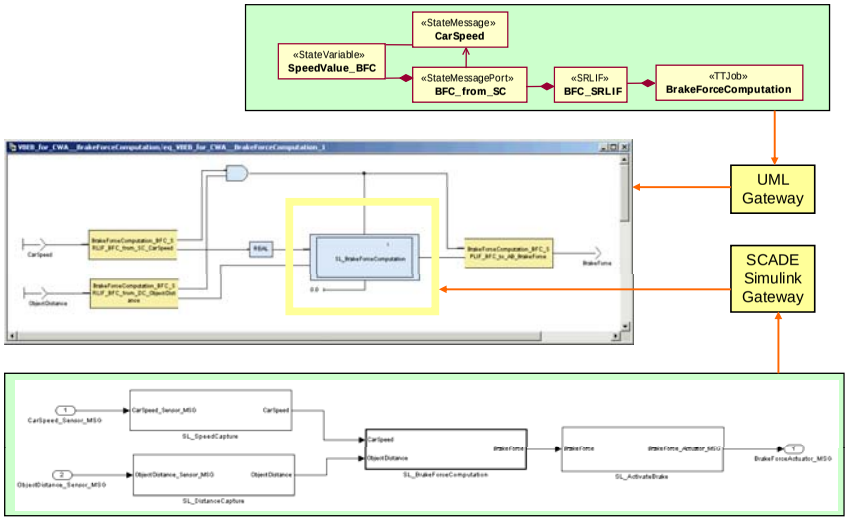
2. Design (behaviour)

Model-Driven Development

- Matlab support
- UML support
- Formally precise modeling

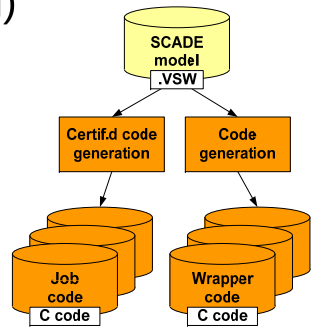


Scade modeling



3. Implementation (code generation)

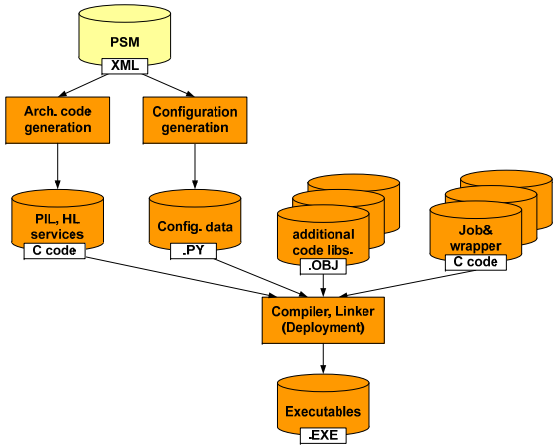
- Job code (behaviour)
- Wrapper code (interfacing)



Funded by EC contract FP6-027599

3. Implementation (config. & code gen.)

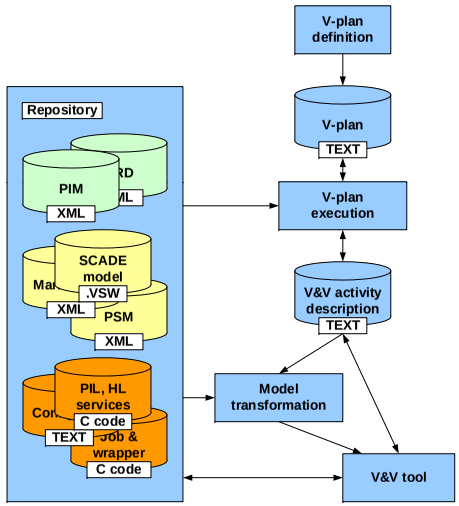
- PIL
- scheduling
- deployment



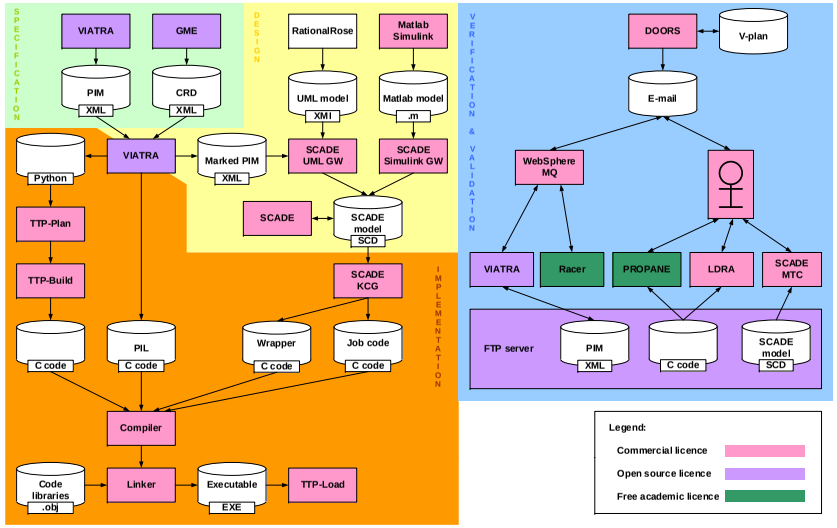
Funded by EC contract FP6-027599

4. Verification & Validation

- Consistency & completeness checking (Racer)
- Model checking (SCADE MTC)
- Source code analysis (LDRA)
- Simulation (SCADE)
- SWIFI (PROPANE)



Tool-chain: tools



Current limitations

- Tools are not fully integrated
- No end-to-end requirement tracing
- No central design artefact repository
- Result visualization often ad-hoc

Funded by EC contract FP6-027599

The DECOS PIM-PSM mapping editor

Workshop on

Software and Service Development, Security & Dependability

10-11 July 2007, Maribor

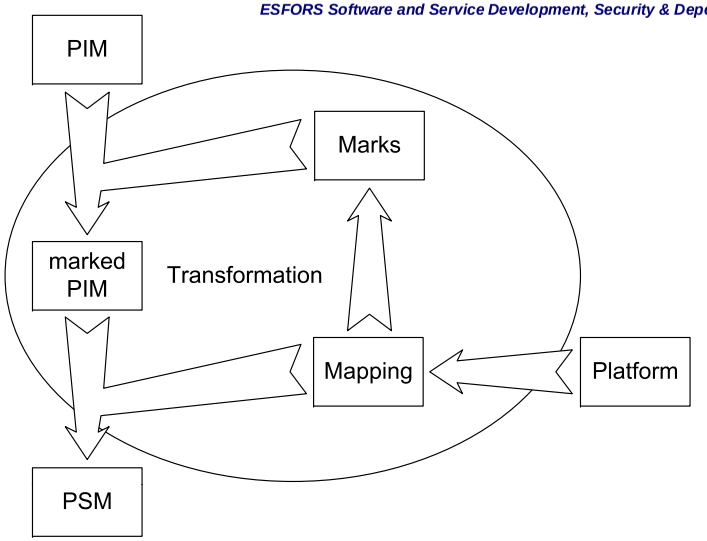
Funded by EC contract FP6-027599

Objective

- A tool for MDA support
- From platform independent to platform dependent
- Platform independent: DECOS PIM
- Platform: DECOS platform of TTTech
  - TTP core network / FlexRay core network
  - TriCore board
  - EEE
  - C language

Funded by EC contract FP6-027599

MDA



Demonstration

Funded by EC contract FP6-027599

V&V Test Bench: Tool Integration Framework

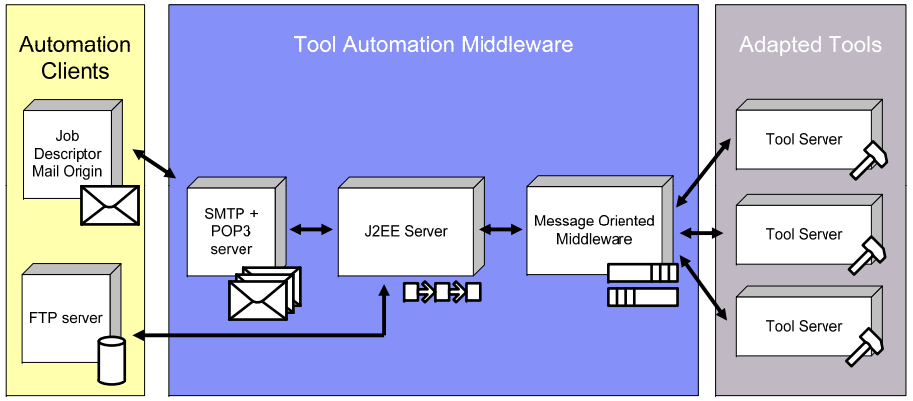
Workshop on

Software and Service Development, Security & Dependability

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Architecture



Funded by EC contract FP6-027599

Example: PIM checking with RACER

- VIATRA2
  - PIM to RACER transformation
- RACER
  - Ontology based consistency and completeness check
  - *„The redundancyDegree of a Resource should be equal with the redundancyDegree of the corresponding SensorActuatorJob."*
- The two integrated into a chain:
  - Send e-mail to samplepimcheck@daniel.mit.bme.hu

Funded by EC contract FP6-027599

Prototype Implementation

Funded by EC contract FP6-027599

Discovery, the Final Frontier

J. C. Pazzaglia

SAP RESEARCH

Discovery, the Final Frontier

Workshop on

Software and Service Development, Security & Dependability

Dr Jean-Christophe Pazzaglia

SAP Research - Security & Trust - France

Results are based on our collaboration with Slim Trabelsi and Yves Roudier - Eurecom Institute (see references)

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Outline

- Service Discovery in Pervasive Environment
- Discovery Insights
- Discovery challenges
- Revisiting Security Threats
- Discovery Policy Contents
- Secure Service Discovery Model
  - Registry Based Solution
  - P2P Based Solution
- Conclusion and Future Works

Discovery in Pervasive Environment

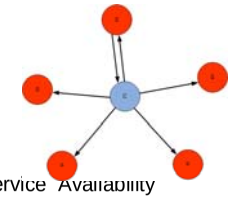
*"**Pervasive computing** is the trend towards increasingly ubiquitous (another name for the movement is ubiquitous computing), **connected computing devices** in the environment, a trend being brought about by a convergence of advanced electronic - and particularly, wireless - technologies and the Internet."*
http://searchnetworking.techtarget.com/sDefinition/0,,sid7_gci759337,00.html

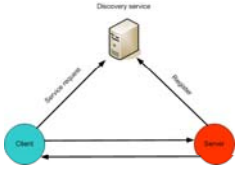
- Challenges
  - Self configuration
  - Adaptation to dynamic environment
  - Unknown (and maybe hostile) peers
- Pervasive computing often based on SOA
  - Smart devices often modeled as Services

Service Discovery is the enabler of Pervasive Computing

Discovery Insights

- Main actors
  - Service Requester
  - Consumer
  - Registry [optional]
- Two models
  - Advert model
    - Provider broadcasts Service Availability
  - Lookup model
    - Consumer broadcast Service Needs
- Main risk for the initiator
  - No control over the entities that receive the discovery message





Discovery: first exchange of information between potential partners

Discovery Challenges

- How to build comprehensive advertisement ?
  - What is the scope of service description ?
    - Functional and non functional (WSDL + metadata)
    - Security, privacy, dependability, SLA, QoS, …
  - How to include that in the engineering process ?
  - Design AND Run time problem
- How to initiate a secure lookup ?
  - PKI is in general not a solution (unknown counterpart)
- How to assess advertised information ?
  - Third party, reputation model, …
- How to build trustworthy federation ?
  - Composition of policies …

Revisiting Security Threats for Discovery (1/2)

- Initiator specific threats
  - Confidentiality
    - Unknown counterpart
      - Relying on PKI based solution is futile !
      - First message as clear text
    - Door open to a *man-in-the-middle* attack
  - Privacy
    - *Lookup* model: information disclosed reveal the intentions of the consumer
    - *Advert* Model: commercial competitor or *malware* may gather too easily information
    - Relying on X509 identity certificates is even worst !

Revisiting Security Threats for Discovery (2/2)

- Classical threats
  - Access control
    - No proper authentication
      => No access control during discovery
    - No restricted services
      => How to advert only to potential users ?
  - Availability
    - Service descriptions exposure
      => Attackers may exploit vulnerabilities
    - Registry based architecture
      => DoS against registries

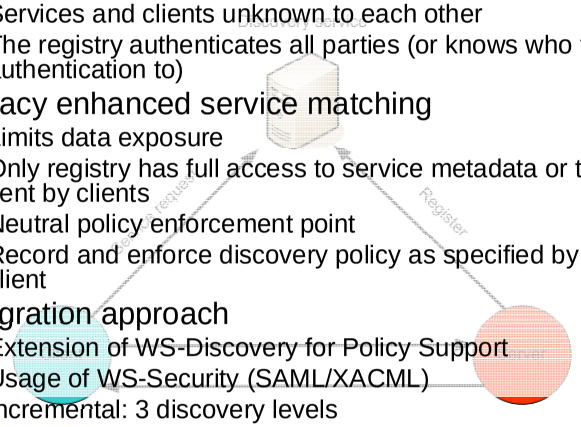
Discovery Policy Expressiveness

- **Authentication**: Each entity may require the other entities to authenticate during the registry/discovery phase
- **Access Control**: Server resources are protected against an unauthorized discovery
- **Confidentiality**: Using an encryption infrastructure to protect the discovery phase [between the registries and other entities]
- **Privacy**: Each entity can selectively expose its services to a restricted set of entities
- **Non Repudiation**: The possibility to use secure tokens to prove that a client discovered the system legitimately [to log it / to provide anonymous access].

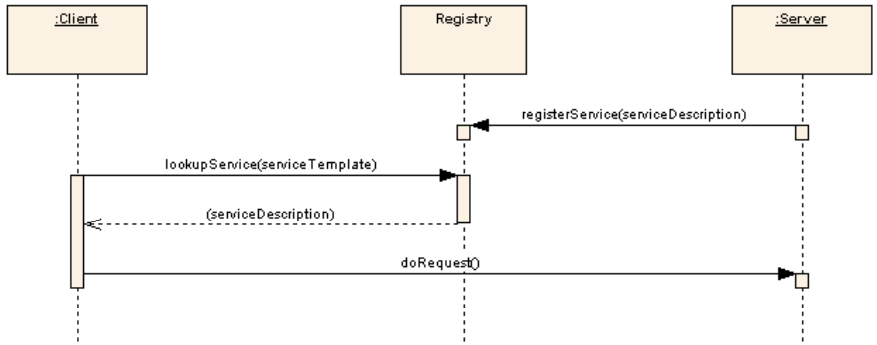
Backup slides

Registry Based Solution Sketch

- Reference for trust establishment
  - Services and clients unknown to each other
  - The registry authenticates all parties (or knows who to delegate authentication to)
- Privacy enhanced service matching
  - Limits data exposure
  - Only registry has full access to service metadata or to attributes sent by clients
  - Neutral policy enforcement point
  - Record and enforce discovery policy as specified by service and client
- Integration approach
  - Extension of WS-Discovery for Policy Support
  - Usage of WS-Security (SAML/XACML)
  - Incremental: 3 discovery levels

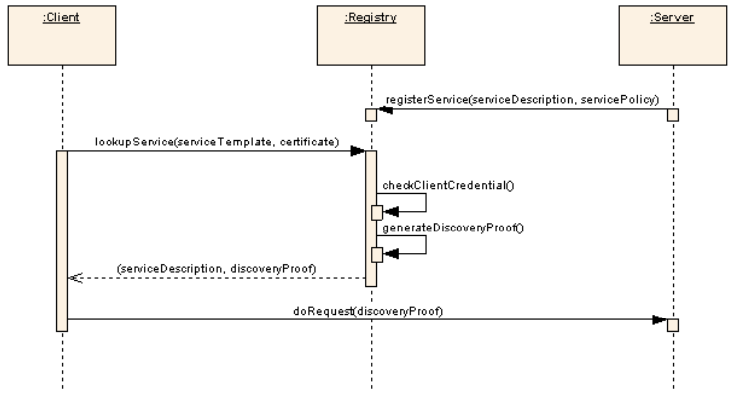


Secure Service Discovery Model Level 0



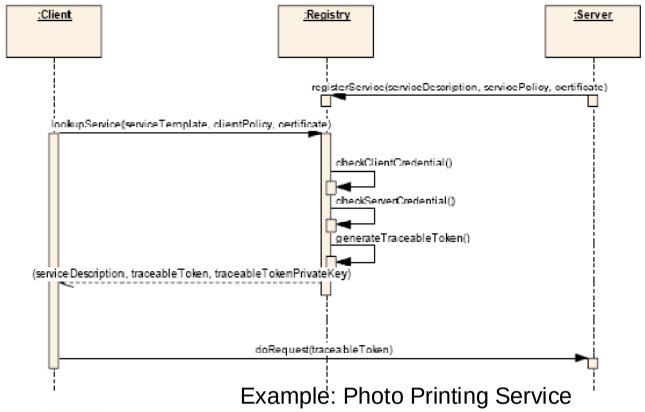
Example: Printing Service

Secure Service Discovery Model Level 1



Example: Color Printing Service

Secure Service Discovery Model Level 2



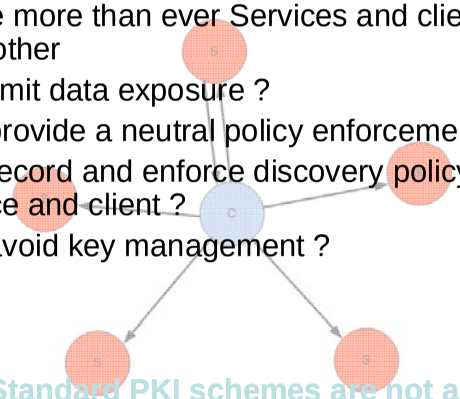
Example: Photo Printing Service

Limitations

- Secure Discovery Service
  - Rely on existing PKI infrastructure
  - Trust establishment between clients and services
  - Server resources protected against unauthorized discovery
  - Client's service request is accessible only to authorized servers
  - Privacy for service and service requestor.
  - Dynamic configuration of security mechanisms
  - Mechanism adapted to standard SOA/ESA mechanism
- Limitations
  - Solution is strongly dependent on the existence of a common trusted authority
  - Registry may be a bottleneck / DOS attack
  - Model not applicable to P2P discovery
  - Do not cop well with pervasive and highly mobile computing

Main Issues in P2P discovery

- No trustworthy trusted third party
  therefore more than ever Services and clients unknown to each other
- How to limit data exposure ?
- How to provide a neutral policy enforcement point ?
- How to record and enforce discovery policy as specified by service and client ?
- How to avoid key management ?



Standard PKI schemes are not adapted

P2P Solution Sketch

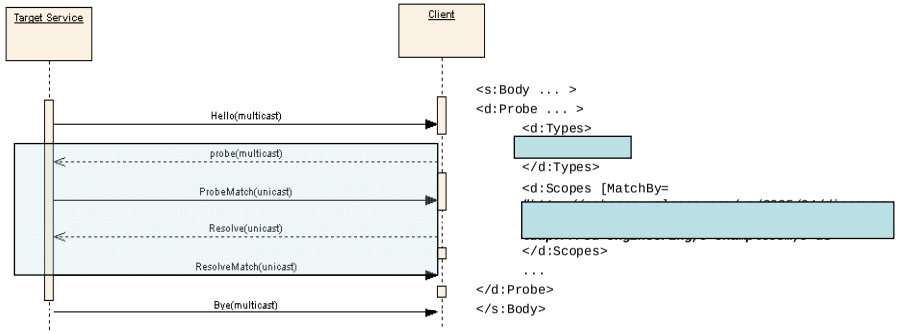
- To use an Identity Based Encryption (IBE) [Bonneh & Franklin 2001] variant
  - Standard IBE enables to encrypt a message knowing the name of the recipient
  - *Attribute Based Encryption* enables to encrypt a message knowing the recipient attributes
- Lookup message encrypted using desired service properties

  => only a recipient proving to fulfill the requirements will be able to decrypt it
- Hybrid architectue (SPKI alike certification mechanism may be used to provide the credential to retrieve the private key)
- By construction the Matching Mechanism is the neutral enforcement point
  - functional properties {printer}
  - non functional {printer | certified by SAP| auditable}

Server Protection

Client/Server protection

WS-Discovery based Prototype



The Good and the Evil

- Solution well adapted to P2P (but also worked for registry)
- Significant *albeit* “realistic” extra processing time (early experiment < 1s)
- Suffer from IBE limitation
  - Recipient should be able to connect to a PKG
  - Revokation / non-repudiation issue
- Matching capabilities
  - Rely on a share understanding of attributes
  - Relatively basic matching capabilities

Conclusion & Future Works

- Service Discovery fundamental component of dynamic world
- Securing Discovery Service is problematic
- To summarize:
  - Registry based solution
    - Pragmatic solution relying on PKI
    - Trust establishment between clients and services
    - Server resources protected against unauthorized discovery
    - Client's service request is accessible only to authorized servers
    - Privacy for service and service requestor.
    - Dynamic configuration of security mechanisms
  - Mechanism adapted to standard SOA/ESA mechanism
  - P2P/ABE based solution
    - More promising solution based on new encryption scheme
    - Same advantage than the first solution without known ttp
    - More research / experiment still needed
      - Time/CPU usage
      - Protocol / Encryption scheme to reduce the overhead for multi-criteria matching
      - How to handle complex matching mechanism ? [PBC Bagga & Molva 2005]
  - Mechanism able to cop also with pervasive computing
- Policy for Securing Service Discovery

To know more

2006

Pazzaglia, Jean-Christophe; Crosta, Stefano
MADSig: enhancing digital signature
to capture secure document processing requirements
ISSE 2006, Information Security Solutions Europe

Trabelsi, Slim;Pazzaglia, Jean-Christophe; Roudier, Yves;
Secure Web service discovery: overcoming challenges of ubiquitous computing
ECOWS 2006, Zurich, December 2006

Trabelsi, Slim; Pazzaglia, Jean-Christophe; Roudier, Yves
Enabling secure discovery in a pervasive environment
SPC 2006, 3rd International Conference on Security in Pervasive Computing, April 2006, York, UK

2005

Crosta, Stefano; Montagut, Frédéric; Pazzaglia, Jean-Christophe; Reznichenko, Yevgen; Rits, Maarten;Schaad, Andreas
A secure public sector workflow management system
ACSA 2005, 21st Annual Computer Security Applications Conference, December 2005, Tucson, USA

Crosta, Stefano; Pazzaglia, Jean-Christophe; Schöttle, Hendrik
Modelling and securing European justice workflows
ISSE 2005, Information Security Solutions Europe, 27–29 September 2005, Budapest, Hungary

Questions

Bibliography

- WSDL specifications http://www.w3.org/TR/wsdl
- D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing", Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology, pages 213–229. Springer-Verlag, 2001.
- A. Duffy and T. Dowling, "An Object Oriented Approach to an Identity Based Encryption Cryptosystem", 8th IASTED International Conference on Software Engineering and Applications, 2004.
- S. Trabelsi, J C. Pazzaglia and Y. Roudier "Enabling Secure Discovery in a Pervasive Environment" 3rd International Conference on Security in Pervasive Computing (SPC 2006) – York – UK – April 2006
- F. Zhu, M. Mutka, and L. Ni, "Splendor: A secure, private, and location-aware service discovery protocol supporting mobile services", in Proceedings of the First IEEE International Conference on Pervasive Computing and Communications (Percom'03). IEEE Computer Society, Mar. 2003, pp. 235–242.
- F. Zhu, M. Mutka, L. Ni "Prudent exposure: A private and user centric service discovery protocol" Proceedings of the 2nd IEEE International Conference on Pervasive Computing and Communications (PerCom'04) Orlando, USA, 2004
- A. Sahai and B. Waters, "Fuzzy Identity-Based Encryption", Advances in Cryptology-Eurocrypt'05.LNCS 3494, pp. 457-473, Springer, 2005.
- A. Duffy and T. Dowling, "Java Card Key Generation for Identity Based Systems", NUI Maynooth Department of Computer Science, Technical Report Series, 2005.
- M. Ghader et al "Secure resource and service discovery in personal networks" Wireless World Research Forum Meeting #12, Canada, 4-5 Nov 2004
- W. Bagga, R. Molva, "Policy-based cryptography and applications", FC' 2005, 9th International Conference on Financial Cryptography and Data Security, 28 February-03 March 2005, Roseau, The Commonwealth of Dominica - Also published in LNCS Volume 3570

Security wrappers

A. Waller

Security Wrappers

Workshop on

Software and Service Development, Security & Dependability

Adrian Waller
adrian.waller@thalesgroup.com

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Background

- Performed a gap analysis of security needs for future composable systems
  - From Thales internal initiative and UK MoD funded "MOSA – Modular Open Systems Architecture" work
- Includes many topics already covered, but also "security wrappers"
  - Idea and terminology comes from work in the safety community on "safety wrappers"

Funded by EC contract FP6-027599

The problem

- When creating an ad hoc, dynamic system-of-systems from component services one should do the following:
  - Perform a security requirements analysis for the whole system
  - Select individual services to meet the system requirements (including security requirements)
- However:
  - Difficult to discover the security properties of component services (covered in other presentations)
  - Component services that meet the security requirements may not be available
  - Even if they are available, they (and their operators) may not be fully trusted

Funded by EC contract FP6-027599

A potential solution

- A "security wrapper" around each individual component service could be part of the solution:
  - Monitor the compliance of partially trusted component services with the security policy/requirements
  - Enforce or otherwise compensate for security requirements that are not met by component services
- Security wrappers would need to be tailored to individual services, and, ideally, could be created at runtime
  - Could select appropriate monitoring and enforcement technologies from a "toolbox"
  - Security patterns may be part of the solution for creating such a toolbox
- Is a potential long-term research challenge candidate
  - A small amount of research has been done in this area and in related technologies, but it appears to be an area where a lot more research would need be done

Funded by EC contract FP6-027599

Formal methods for the analysis of wide systems providing business services

L. Durante

Formal methods for the analysis of wide distributed systems

Workshop on

Software and Service Development, Security & Dependability

Luca Durante - IEIIT/CNR, Italy
luca.durante@polito.it

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Outline

- Formal methods: why and how
- Experiences on the use of formal methods in the design and analysis of wide systems providing high level business services
  - security analysis
  - dependability analysis
  - experiments with a tool
- Open issues and conclusions

Funded by EC contract FP6-027599

Formal methods: why and how

- Formal methods allow to perform exhaustive* analysis with mathematical rigor on systems
  - the system must be formally described
  - the analysis is aimed at verifying the fullfillment of formally described properties
- Non-exhaustive analysis are
  - simulation
  - testing

* It deals with all the system states, i.e. all possible behaviors

Funded by EC contract FP6-027599

Formal methods: why and how

- It is carried out off-line, at the design level
- The exhaustive analisys of large systems leads to
  - the state space explosion problem
  - untractable computational complexity
- Simplyfing assumptions are needed
  - abstract, high level descriptions, where useless details are hidden or removed

Funded by EC contract FP6-027599

Formal methods: why and how

- Results computed by formal analysis are inputs to
  - the system designer(s)
  - the system manager(s)
- and integrate results coming from other analysis tools (simulation, testing, run-time monitoring)
- Some semantic gaps must be filled
  - common descriptions shared by all tools may partially help

Funded by EC contract FP6-027599

Experience on the use of formal methods

- Two kinds of analysis
  - security analysis: which security properties of a system are altered
    - vulnerability analysis
  - dependability analysis: checks dependencies among components of a network providing high level business services and shows the impacts of fault[s]
    - fault propagation analysis

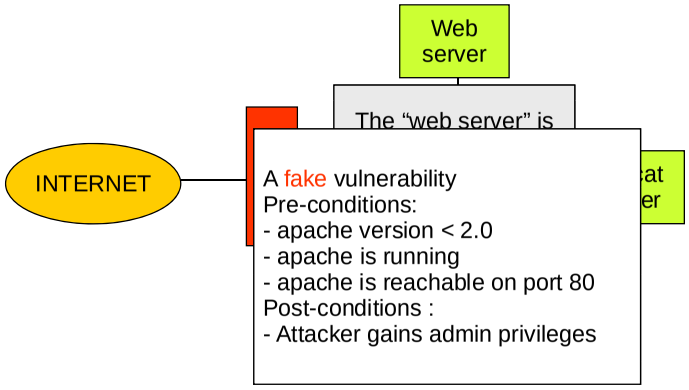
Funded by EC contract FP6-027599

Vulnerability analysis

- Deals with vulnerabilities
  - pre-conditions
    - vulnerable programs, specific configuration, sufficient privileges, connectivity conditions
  - post-conditions
    - privileges escalation, increased connectivity, new trust relationships
- Analysis
  - computes chains of exploitable vulnerabilities, i.e. the steps enabling an attacker to reach his target

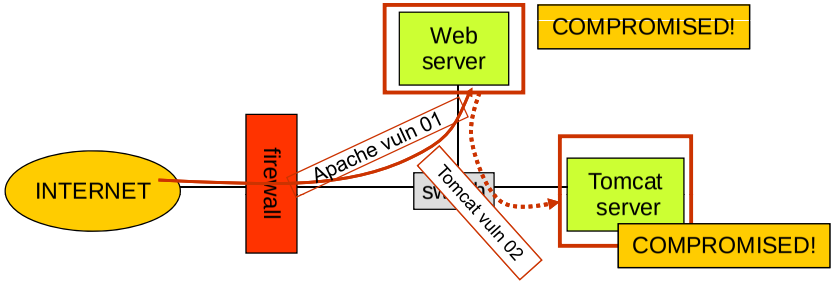
Funded by EC contract FP6-027599

Vulnerability analysis



Funded by EC contract FP6-027599

Vulnerability analysis



(trivial) CHAIN OF VULNERABILITIES

Apache vuln 01 → Tomcat vuln 02 → Web server is compromised, Tomcat server is compromised

Funded by EC contract FP6-027599

Vulnerability analysis

- Scalability
  - model checking experiments (small networks only)
  - monotonicity assumption
  - logic programming

Funded by EC contract FP6-027599

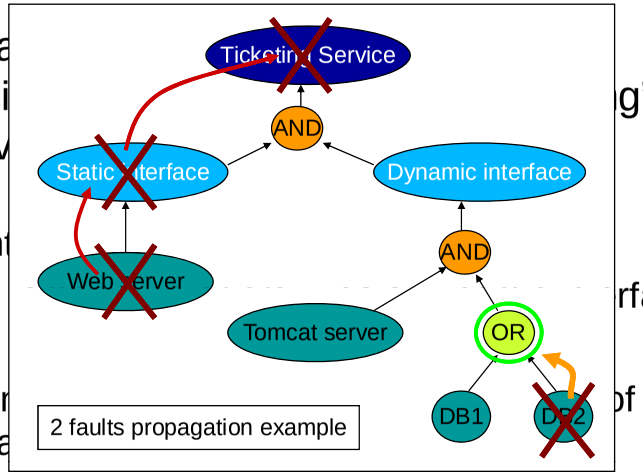
Fault propagation analysis

- Deals with all the logical and physical system dependencies
  - among all the system components, devices, software and business services
- Analysis
  - Computes how a fault can affect the system critical services and resources

Funded by EC contract FP6-027599

Fault propagation analysis

- As a busi… g" serv…
  - A int…
  - A …rface
  - A
  - An … of the da…



Funded by EC contract FP6-027599

Fault propagation analysis

- Formal analysis
  - qualitative: computationally more feasible
  - quantitative: more sophisticated - values, indexes
- Scalability in very large networks
  - qualitative impact analysis
- Kind of faulty states
  - binary (on/off)
  - complex states (performance indexes, % of availability, …)

Funded by EC contract FP6-027599

Common points

- Models of the system components
- Dependency models
  - relationship described by means of functions
- Scalability

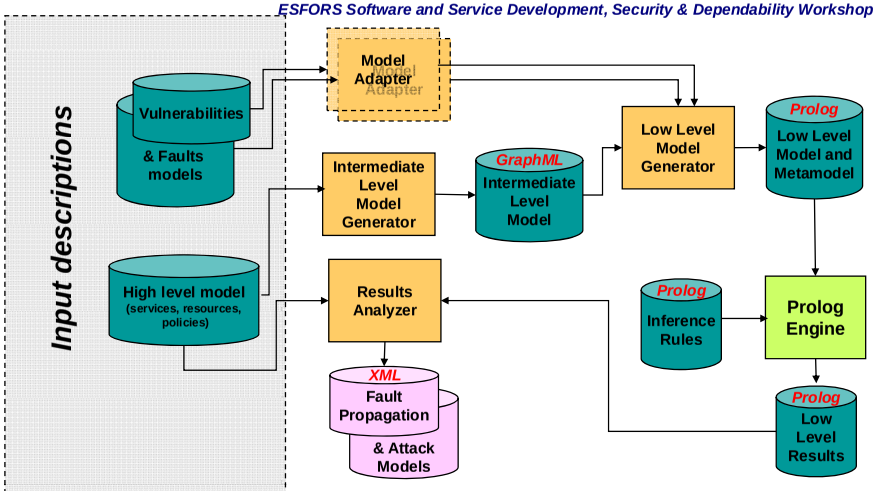
Funded by EC contract FP6-027599

Formal analysis tool

- Vulnerability analysis
  - logic programming approach
  - monotonicity assumption
- Fault propagation analysis
  - binary faulty states
- Combined analysis
  - attacks as faults

Funded by EC contract FP6-027599

Formal analysis tool



Funded by EC contract FP6-027599

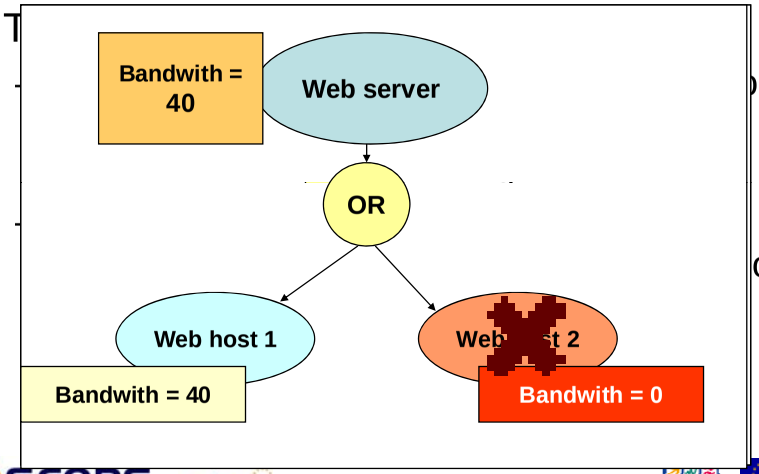
Resource model

- Description of hosts, firewalls, routers …
  - operating system, installed software, available services, firewall rules …
- Missing details are *assumed*
  - worst case analysis.

Funded by EC contract FP6-027599

Service and dependency model

- T



Funded by EC contract FP6-027599

Service and dependency model

- Services like “DB service” “web service” are high level services. The tool can however work with *low* level services
  - for instance, a host can depends upon a working hard disk and a working network card
  - the “hard disk” itself can be thought as a “macro-service”. For instance two raid-1 hard disks are bounded by an “AND” relationship

Funded by EC contract FP6-027599

Vulnerability model

- Contains the set of known vulnerabilities
- Represents part of the initial knowledge of the attacker
- Vulnerabilities are related to single network nodes

Funded by EC contract FP6-027599

Vulnerability model

- The vulnerability meta-model has been designed as an extension of the OVAL language (http://oval.mitre.org)
  - for each vulnerability it describes
    - pre-conditions (extension of the OVAL language)
    - post-conditions (effects of the exploitation of vulnerability, *new feature* not present in OVAL)

Funded by EC contract FP6-027599

Reasoning engine (vulnerability analysis)

- Accumulates *Facts* associated to entitities
- Monotonicity assumption
  - leads to polynomial complexity
- Attack graph as the output of the analysis
  - a lot of details available in order to allow a complex post-analysis

Funded by EC contract FP6-027599

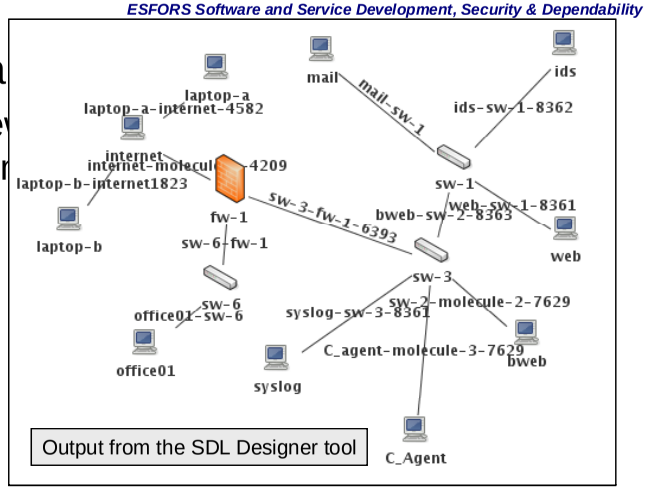
Reasoning engine (fault propagation)

- A fault in a service is propagated following the kind of dependencies among components
- Injecting faults will result in a list of “damaged” services
- Attacks can be considered as faults, thus the two kind of analysis are merged

Funded by EC contract FP6-027599

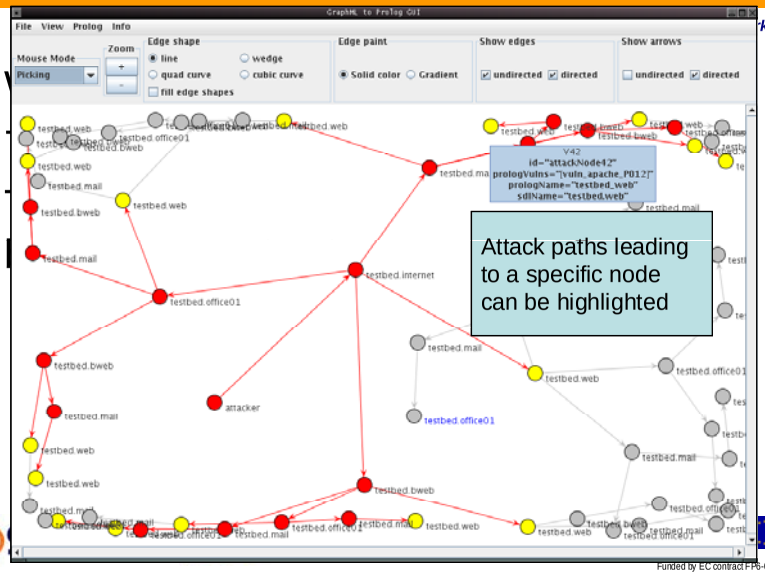
Small example (vulnerability analysis)

- Sma
  - Fe
    cor

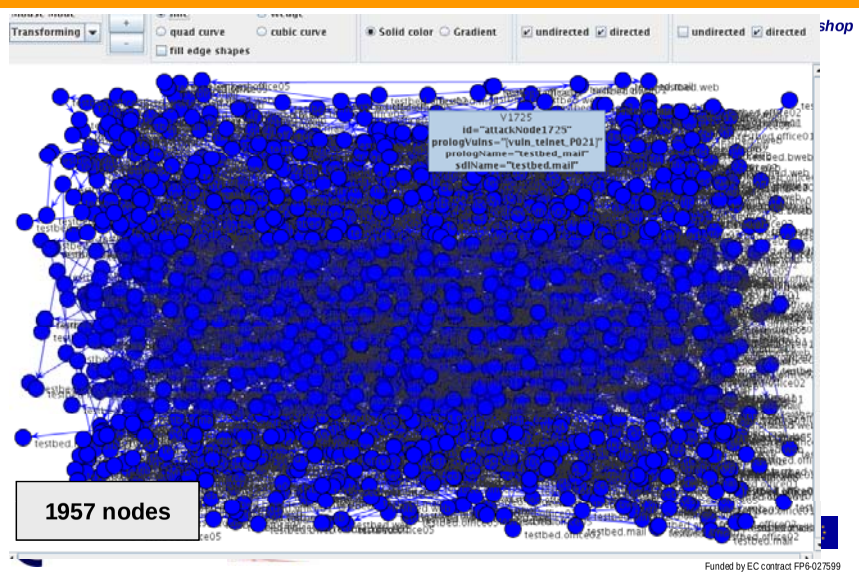


Funded by EC contract FP6-027599

Small example (vulnerability analysis)



Larger example (vulnerability analysis)



Open issues (modeling side)

- Today do not exist standard languages able to describe the whole system in an integrated way
- Vulnerabilities collected on publicly available repositories can't be automatically gathered and processed
- Models are written by hand
  - huge systems can not be modeled
  - standard scanners required

Open issues (analysis side)

- Root cause analysis
  - fault propagation can give the effects of a fault. But what, if we want to know which set of faults can cause those effects ?
    - computational complexity may explode
- Dynamic elements
  - do not exist simple models for dynamic elements like firewalls and routers

Open issues (analysis side)

- Analysis results are expressed in term of the abstract system models
  - in order they precisely help the designer, they have to be translated back to the “input descriptions”
  - huge attack graphs are not easily browsable
    - pruning / compression algorithms are required

Funded by EC contract FP6-027599

Conclusions

- Vulnerability and fault propagation analysis is feasible on networks providing high level business services if
  - simple vulnerability, fault and dependency models are used
  - monotonicity assumption holds
  - otherwise computational complexity explodes

Funded by EC contract FP6-027599

Conclusions

- Lack of integration among the other useful tools in the design and run-time phase
  - simulation, emulation, testing, monitoring
- Other open issues mainly depends on the lack of suitable languages and standards for representing all the involved information

Funded by EC contract FP6-027599

Thank you for your attention

Funded by EC contract FP6-027599

Dependability and Security Metrics

G. Csertán

Dependability & Security Metrics

Workshop on

Software and Service Development, Security & Dependability

György Csertán
csertan@mit.bme.hu

10-11 July 2007, Maribor

Budapest University of Technology and Economics

Funded by EC contract FP6-027599

Ongoing results of the „Metrics" WG

DESEREC

Dependability and Security by Enhanced Reconfigurability

Funded by EC contract FP6-027599

DESEREC

Idea: „*DESEREC aims at providing methods and tools to analyse, design, model, simulate, and plan, the optimised configurations of resilient information systems supporting critical activities*"

Architecture: DItemAgent / Element --- DLocalAgent / Molecule --- DGlobalAgent / System)

Basic functioning: Planning, Fast Reaction (local ...), Hot Reconfiguration

Main models: WS-CDL / behaviour --- SDL / structure --- OP Policy / configuration

Funded by EC contract FP6-027599

Overview

Definition of Metrics
Classification of Metrics
Relation of Metrics to Faults & Attacks
Fault Metamodel
Metrics Metamodel
Usage of Metrics in DESEREC
DESEREC Metrics
Computation of Metrics
Visualization of Metrics

Funded by EC contract FP6-027599

Definition of Metrics

- Definition:
  - Textual: „A metric is a precisely defined method to associate a number with an attribute."
  - Formal: „A metric is a function that assigns values to a system from an ordered set."
- Usage:
  - In an IT system the metric is used to assess the system against non-functional requirements.

Funded by EC contract FP6-027599

Definition of Metrics (contd.)

- DESEREC Metrics:
  - should be computable from observation(s) and knowledge of the system;
  - its possible values should be precisely defined;
  - a (total) ordering relation should be defined on the set of possible values;
  - should have a clearly defined temporal and entity scope;
  - its semantics should be clear, with diverse interpretations in various contexts allowed.

Funded by EC contract FP6-027599

Definition of Metrics (contd.)

- Calculating top-level metrics from low-level metrics in non-trivial
  - Problem: e.g. DB server computer performance falls to 50%, ticket selling service response time does not necessarily increases to 200%
  - However: dependency between metrics can be defined easily
- Time scope
  - Metrics for long term observation
  - Events for immediate reaction

Funded by EC contract FP6-027599

Metrics System for DESEREC

- Metrics hierarchy (3-level):
  - service (e.g. Train ticket selling)
  - service component (e.g. web service, DNS service, network service)
  - resource (e.g. computer, switch)
- Metrics definition (top-down chain of):
  - goal (e.g. provide on-line impression for the user)
  - question (e.g. how long a user has to wait for a reaction from the system)
  - metric (e.g. what is the current response time of the service)

Funded by EC contract FP6-027599

Metrics System for DESEREC (contd.)

- Metric is defined by:
  - name
  - meaning / explanation
  - interfaces
  - measurement method / evaluation algorithm

Funded by EC contract FP6-027599

Classification of Metrics

- Objective vs. subjective metrics
  - Whether measurement depends on the viewpoint of the person who performs the measurement.
  - Informal, unstable, immature objects decline to be involved in subjective metrics.
- Qualitative vs. quantitative metrics
  - Metrical range is split into regions. Question: granularity?
  - Real values of the range. Question: precision?
- Analitical vs. empirical metrics
  - Analytical metrics are based on a model of the system. Main usage in the planning phase.
  - Simulation falls in-between analytical and empirical metrics.

Funded by EC contract FP6-027599

Classification of Metrics (contd.)

- Direct vs. indirect metrics
  - Direct: measurement on the system.
  - Indirect: measurement of an effect of one system onto another.
  - Example: database response time as a metric for computer performance.

Funded by EC contract FP6-027599

Relation of Metrics, Faults and Attacks

- Dependability metrics are defined for erroneous opstate or failure of an element
  - E.g.: availability of Apache is computed over the states
    - Operational / Overbooked / Down
- Security metrics are defined for attack or exploitation type
  - E.g.: number of files in /bin file system without proper access controls associated
    - E.g.: number of unencrypted files under a Domino server

Funded by EC contract FP6-027599

Relation of Metrics ... (contd.)

- Metrics dependency derived from error-propagation graph and attack graph.
  - BUTE: error-propagation graph computation tool
  - IEIIT: attack graph computation tool
- Metrics computation in the planning phase relies on a fault assumption.
  - E.g.: failure rate of a NIC, storage

Funded by EC contract FP6-027599

Fault model

„A fault model is an ***engineering model*** of something that could go wrong in the construction or operation of a piece of equipment. From the model, the designer or user can then predict the ***consequences of*** this particular ***fault***. Fault models can be used in almost all branches of engineering.” – Wikipedia

Funded by EC contract FP6-027599

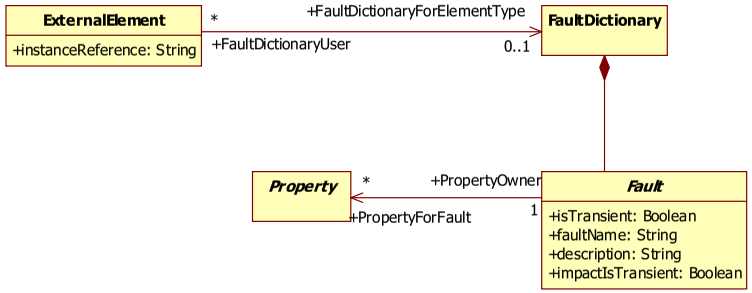
Fault metamodel

- Prepared by BUTE
  - Metamodel is a MOF model
    - XML concrete syntax
  - Model itself can reside inside DESEREC models (e.g. SDL)
  - It is derived from
    - Common domain knowledge
    - Practical engineering experience
- Metamodel packages
  - DependabilityCore
  - DSExtension

Funded by EC contract FP6-027599

Fault metamodel (contd.)

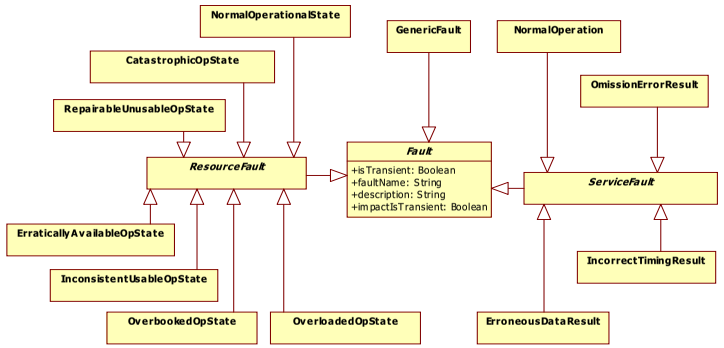
- DependabilityCore:
  Description of elements (possible faults and their properties)



Funded by EC contract FP6-027599

Fault metamodel (contd.)

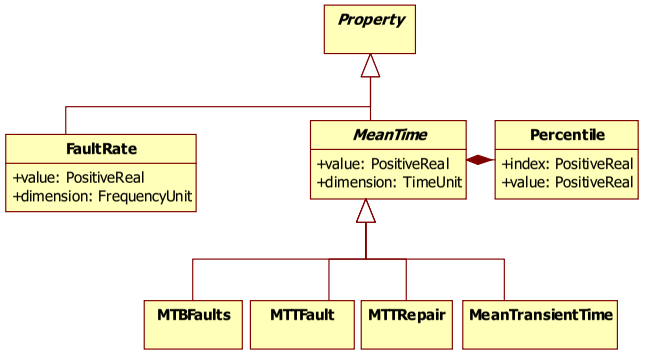
- DSExtension: DESEREC specific engineering classifications for
  - Faults: effect-based (error/failure) classification



Funded by EC contract FP6-027599

Fault metamodel (contd.)

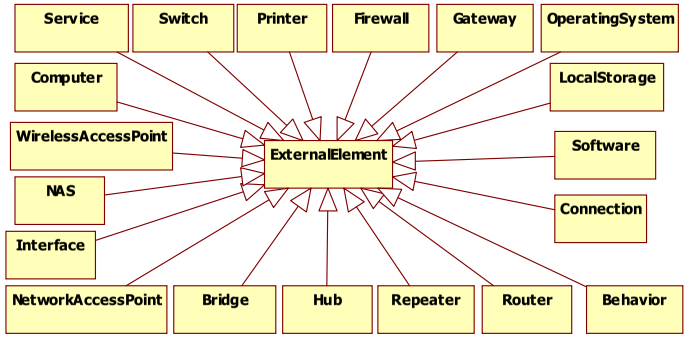
- Fault properties



Funded by EC contract FP6-027599

Fault metamodel (contd.)

- Element types: SDL metamodel-based



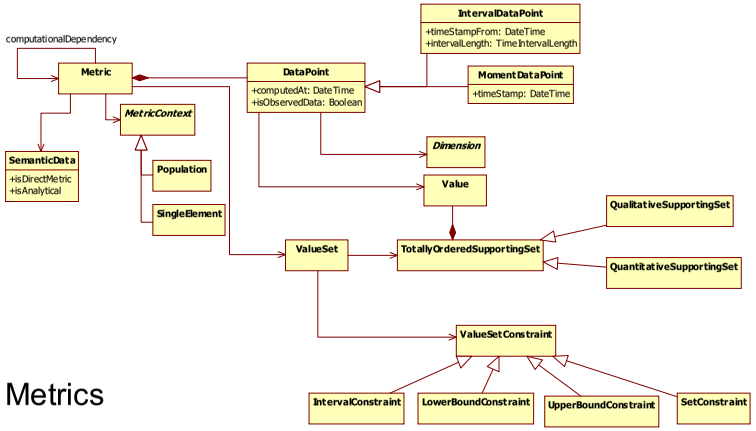
Funded by EC contract FP6-027599

Metrics metamodel

- Why model metrics?
  - metric involvement in planning needs formal representation
  - interoperability between components of the metric computation chain
    - DSensor
    - DItemAgent
    - DLocalAgent
    - DGlobalAgent
  - analyzability of a metric-supported supervisory configuration
  - rigorously describing metric computational hierarchy
  - model-based visualization support
- MOF based metamodel is under construction

Funded by EC contract FP6-027599

Metrics metamodel (contd.)

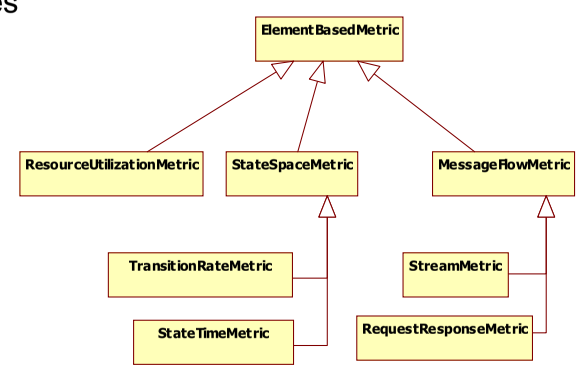


- Metrics

Funded by EC contract FP6-027599

Metrics metamodel (contd.)

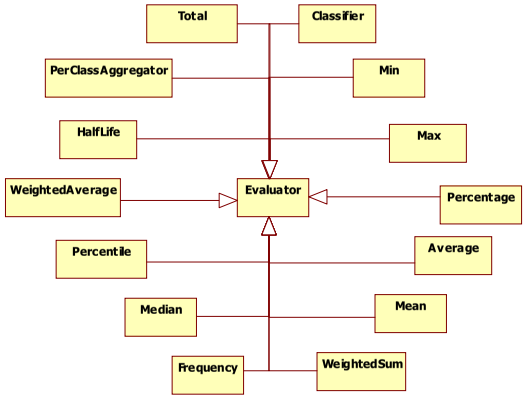
- Metric types



Funded by EC contract FP6-027599

Metrics metamodel (contd.)

- Evaluator types



Funded by EC contract FP6-027599

Usage of Metrics in DESEREC

- Planning phase
  - Fault assumption / system design & configuration
  - Analysis / simulation
  - Metrics computation
  - Metrics evaluation
  - Back to new fault assumption / system design or defining of SLA/alert level
  - Configuration generation for measure collection and metrics computation

Funded by EC contract FP6-027599

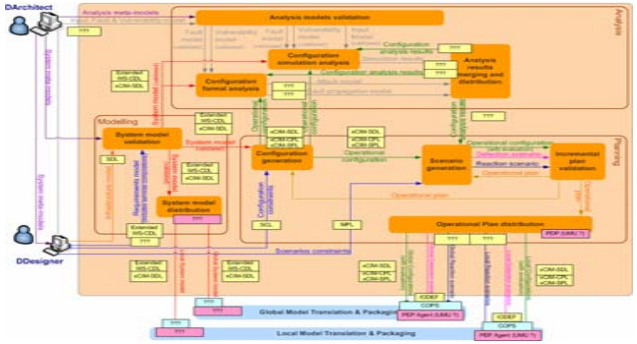
Usage of Metrics in DESEREC (contd.)

- Operation
  - DItemAgent makes local measurement & evaluation
  - Measures are sent to DLocalAgent
  - DLocalAgent makes metrics computation & SLA violation check
  - Fast reaction or sending metrics/measures to DGlobalAgent
  - DGlobalAgent makes metrics computation & SLA violation check
  - Hot reconfiguration
  - Metrics driven configuration selection

Funded by EC contract FP6-027599

Usage of Metrics in DESEREC (contd.)

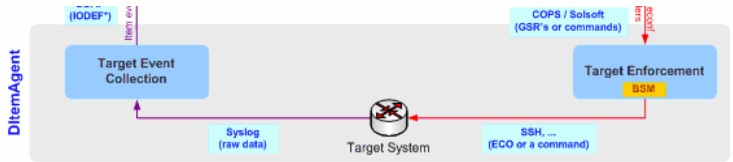
- Planning
  - plan „metrics configuration"
  - define limits on metrics (SLs)



Funded by EC contract FP6-027599

Usage of Metrics in DESEREC (contd.)

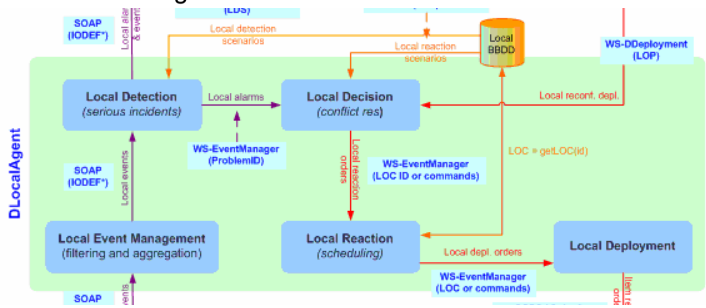
- DItemAgent (element level)
  - collect information
  - calculate metrics (SLO)
  - report to DLocalAgent



Funded by EC contract FP6-027599

Usage of Metrics in DESEREC (contd.)

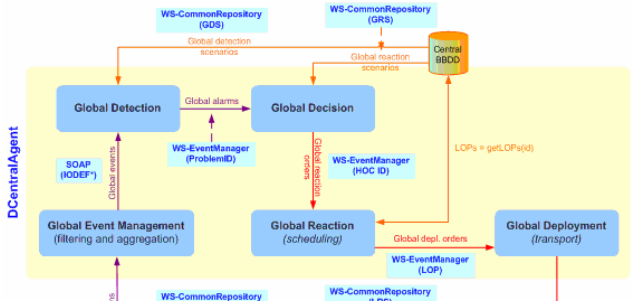
- DLocalAgent (molecule level)
  - calculate metrics (sub-SLA)
  - check violation & generate local events (for LocalDecision)
  - report to DGlobalAgent



Funded by EC contract FP6-027599

Usage of Metrics in DESEREC (contd.)

- DGlobalAgent
  - Calculate metrics (SLA)
  - check violation & generate global events (for GlobalDecision)
  - visualize metrics



Funded by EC contract FP6-027599

DESEREC Metrics

- Three categories for DESEREC metrics:
  - Dependability
  - Performance
  - Security
- A first, basic set of concrete metrics have been defined
  - Dependability
    - 17 elements / 47 metrics
  - Performance
    - 17 elements / 40 metrics
  - Security
    - 15 elements / 26 metrics

Funded by EC contract FP6-027599

DESEREC Metrics Example

Example: Apache availability (state based)

- Apache availability
  - software element (SDL model)
- Definition
  - P(Normal Operational State OR Overloaded Operational State)
- States (according to fault model)
  - Normal OS: The apache process is running.
  - Overloaded OS: 503 Service temporarily unavailable
  - Repairable UOS: The test request returns http error code.
  - Unrepairable UOS: The test request returns http error code, but the error condition cannot be repaired by restarting the server, non-trivial reconfiguration or reinstallation is necessary.

Funded by EC contract FP6-027599

DESEREC Metrics Example (contd.)

- Measuring
  - Plaform based (internal)
    - OS tool based – e.g. ps
  - Service Access Point based (external):
    - HTTP based – wget index.html
- Dimension and range
  - %
  - real (0.0-100.0)
  - time range
  - sampling frequency

Funded by EC contract FP6-027599

Computation of Metrics

- Metric computation is based on:
  - measurement
  - metric evaluator
  - metric propagation
- Metric evaluator
  - A metric evaluator is a generalized analytical algorithm that evaluates a particular category of metric data against an SLO.
  - Metric evaluators are classified by how they process data and the end result that they produce.
- Metric propagation
  - measures and low-level metrics should be propagated „upwards”
  - transmission mechanism is needed
  - semantical interfaces needed

Funded by EC contract FP6-027599

Computation of Metrics (contd.)

- Evaluator types
  - Total
  - Classification based aggregator
  - Classifier
  - Min
  - Max
  - Percentage
  - Average, mean, median, percentile
  - Weighted average, weighted sum
  - Time-in-state, time-to-state
  - Frequency
  - Time series
  - Half-life

Funded by EC contract FP6-027599

Computation of Metrics (contd.)

Metrics & evaluator types

- Availability
  - Percent of time in state
    - Average
  - State transition
    - Min/Max/Average
  - Transaction availability
    - Transaction success
- Performance
  - Weighted average
    - Average
  - Total performance
  - Max

Funded by EC contract FP6-027599

Computation of Metrics (contd.)

- Utilization
- Incident and Change Request
  - Counts
    - Min/Max/Average
  - Percentages
  - Time in state
  - Time to state

Funded by EC contract FP6-027599

Visualization of Metrics

- Why visualization important?
  - Similar metrics should appear in a similar fashion on the dashboard
  - A simple (but good) picture is worth of 1000 words
- Design principles for effective visualization [Jaquith07]
  - It is about the data, not the design
    - data should speak for itselt and not the „dressing up”
  - Say no to 3D graphics and cutesy chart junk
    - usually having 3D does not add additional information
  - Do not rely on the wizard
    - wizards can be helpful, but they do not know the semantics of data

Funded by EC contract FP6-027599

Visualization of Metrics (contd.)

  - Erase, erase, erase
    - if you do not need it, erase it (e.g tick marks, grid lines, any distracting design)
  - Reconsider technicolor
    - monochromatic palette / muted (unsaturated) colors
  - Label honestly and without contortions
    - use clear texts, no abbreviations, no slanting, simple fonts

Funded by EC contract FP6-027599

Defining operational plans to provide dependability and security

G. López

Defining operational plans to provide dependability and security

Workshop on

Software and Service Development, Security & Dependability

Gabriel López
gabilm@um.es

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Agenda

- Introduction to the DESEREC architecture
- Modelling framework
- Operational Plan definition
- Conclusions

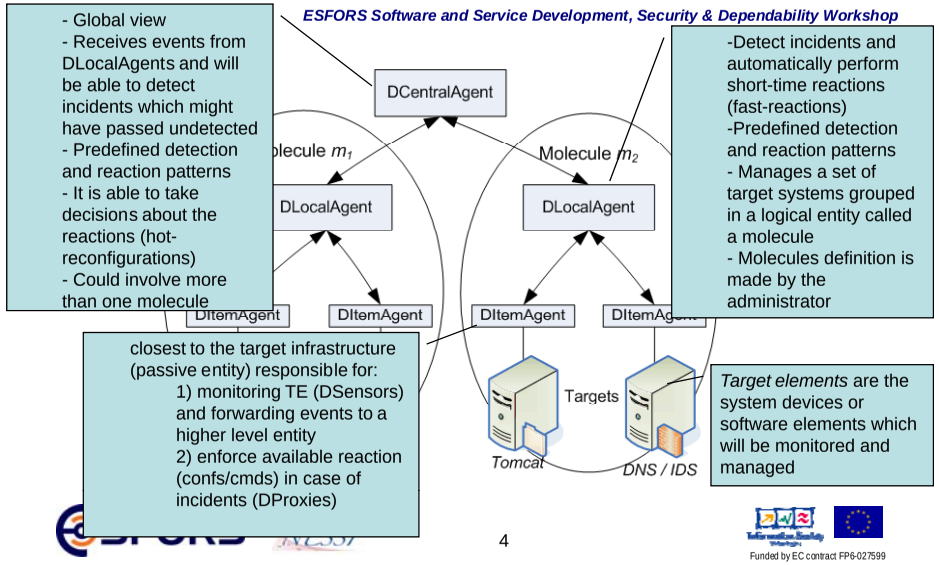
Funded by EC contract FP6-027599

The DESEREC architecture

- Objective: to define a framework able to manage critical infrastructures whilst offering dependability and security in an easy and scalable way
- Framework:
  - *detection* and *prevention* of incidents and potentials faults
  - *reaction* in case of incidents to reduce the time of unavailability or insecurity
  - *plan* the system management by modelling the particular scenarios and actions to allow the system to decide the best-fitted system configuration in case of those incidents

Funded by EC contract FP6-027599

The DESEREC architecture



Funded by EC contract FP6-027599

The DESEREC architecture

- Example:
  - HTTP server misconfigured or it becomes down
    - It could be detected by the upper *DLocalAgent* based on events from the *DItemAgent*.
    - *DLocalAgent* could also apply the reaction process to bring the server up again
  - The server is attacked
    - This incident might not be detected by the *DLocalAgent* due a lack of information
    - It could be detected by the *DCentralAgent* using both the information received from the HTTP server itself, and from an IDS deployed in another molecule
- Layered approach which allows distributing the framework responsibilities
  - Molecules are as autonomous as possible and only need to rely on the central node when strictly necessary
  - Requires a model of the whole infrastructure in order to provide the set of available configurations of the target system including:
    - **detection** and **reaction** patterns to both *DLocalAgents* and the *DCentralAgent*
    - **valid configurations** to be applied on the infrastructure.

Funded by EC contract FP6-027599

Agenda

- Introduction to the DESEREC architecture
- Modelling framework
- Operational Plan definition
- Conclusions

Funded by EC contract FP6-027599

Modelling framework

- Takes high-level descriptions and requirements as inputs:
  - Generates the needed configuration for the elements
  - Must comply with how the administrator wants services to operate
  - Respect any dependencies or constraints that might be applicable
- Set of high-level information which needs to be modelled:
  - Set of services that the system is intended to provide
    - W3C's *Web Services Choreography Description Language* (WS-CDL)
  - Physical infrastructure which is available to run such services
    - *System Description Language* (SDL): Developed within the POSITIF project, allows describing physical network infrastructure (computer systems, software capabilities, etc)
  - Constraints which might apply, such as system-wide policies or impositions
    - *Services Constraints Language* (SCL): Notation developed in DESEREC, allows specifying high-level configuration constraints for services (web, DNS, firewall …)
- It is necessary to transform them into more specific data → Operational Plan
  - DESEREC takes as a basis the *Common Information Model* (CIM), an initiative by DMTF
  - Actually, an XML mapping of the CIM (xCIM) classes is used

Funded by EC contract FP6-027599

Agenda

- Introduction to the DESEREC architecture
- Modelling framework
- Operational Plan definition
- Conclusions

Funded by EC contract FP6-027599

Operational Plans

- *Operational plan:* model including information needed by the system in order to:
  - Allocate the services
  - Configure them to run properly
  - To react automatically when incidents appear
- Includes:
  - One or more *operational configurations*: all admissible ways to set up the system
    - Allocating the technical services onto system elements,
    - Configure them properly taking into account business reqs., services dependencies, etc.
    - Security requirements (such as configuring a firewall according to organization policies)
  - One *detection scenario*: which incidents we are interested in, and how to detect them
  - One *reaction scenario:* ways to react when such incidents are detected
    - The actual reaction will be worked out by a decision engine
    - Reactions consists of switching to a different op. configuration that fixes the incident detected
- Defines a graph-like model
  - nodes are operational configurations
  - links are reactions to incidents

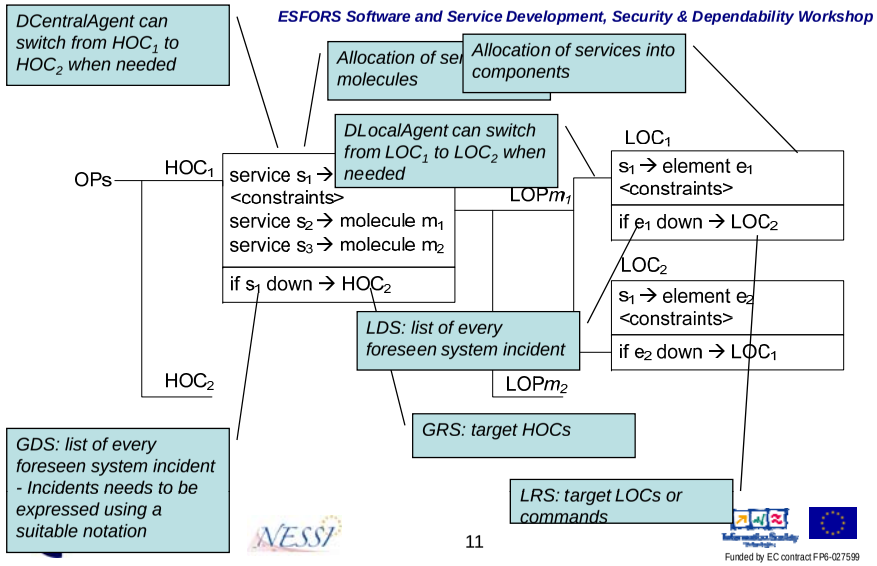
Funded by EC contract FP6-027599

Operational Plans

- Meta-models for operational *plans are defined* both for the global (high) and the *local* (low) levels
  - Local incidents may be resolved by changing some local configuration
  - System-wide ones may require a complete reconfiguration
- HOP = { { HOC1, HOC2, …, HOCn }, GDS, GRS }
- LOPi = { { LOCi1, LOCi2, …, LOCim }, LDSi, LRSi }
- Where for global level:
  - HOP: High-level Operational Plan
  - HOC: High-level Operational Configuration
  - GDS: Global Detection Scenario
  - GRS: Global Reaction Scenario
- Where for local level:
  - LOP: Low-level Operational Plan
  - LOC: Low-level Operational Configuration
  - LDS: Local Detection Scenario
  - LRS: Local Reaction Scenario
- *N:* number of high-level allocations generated by the framework, *1<=i<=n*
- M: number of molecules in the managed system.

Funded by EC contract FP6-027599

Operational Plans



Funded by EC contract FP6-027599

Operational Plans

- Allocation of services onto software elements makes use of an XML implementation of the CIM data model, extended for the purposes of DESEREC
  - xCIM-SDL (*System Description Language*): the system model
  - xCIM-CPL (*Configuration Policies Language*): configuration for services
    - "the web server should listen on port 80 and serve the '*booking.html*' page"
  - xCIM-SPL (*Security Policies Language*): system-wide security policies
    - "all firewalls in the system must allow only connections to port TCP/80"
- From these instances, it is possible to derive configuration data of specific services on specific nodes
  - Generic Service Rulesets (GSR's): describes the full desired configuration for a specific technical service, and is targeted to a specific element in a molecule
  - Includes all the needed information to generate the final configuration files for services

Funded by EC contract FP6-027599

Operational Plans


```
<hoc:HOC Id="HOP1.HOC1">
  <Mapping>
    <Service idRef="BookingWeb"/>
    <Molecule idRef="network.molecule-1"/>
  </Mapping>
  <Mapping>
    <Service idRef="DnsService"/>
    <Molecule idRef="network.molecule-2"/>
  </Mapping>
  <scl:SCL>
    <scl:ConstraintsSet>
      <scl:Service idRef="BookingWeb"/>
      <web:WebServiceConstraints>
        <web:Connector>
          <web:Port>80</web:Port>
          <web:Protocol>tcp</web:Protocol>
        </web:Connector>
        <web:PathMapping>
          <web:VirtualPath>/</web:VirtualPath>
          <web:LocalPath>/usr/local/apache/htdocs</web:LocalPath>
        </web:PathMapping>
      </web:WebServiceConstraints>
    </scl:ConstraintsSet>
    <scl:ConstraintsSet>
      <scl:Service idRef="DnsService"/>
            ...
      </dns:DnsServiceConstraints>
    </scl:ConstraintsSet>
  </scl:SCL>
</hoc:HOC>
```

```
<gds:GDS>
  <Entry>
    <ProblemId>WebDown</ProblemId>
    <cond:Condition>
      <RunningAllocation> HOP1.HOC1  </RunningAllocation>
      <iodef:Incident purpose="traceback">
        <iodef:IncidentID name="String"> event IODEF description
        </iodef:IncidentID>
        <iodef:ReportTime> 2006-12-17:30:47 </iodef:ReportTime>
            ...
      </iodef:Incident>
    </cond:Condition>
  </Entry>
</gds:GDS>

<grs:GRS>
  <Entry>
    <ProblemId>WebDown</ProblemId>
    <react:Reaction>
      <SwitchTo>HOP1.HOC2</SwitchTo>
    </react:Reaction>
  </Entry>
</grs:GRS>
```


Funded by EC contract FP6-027599

Operational Plans


```
<loc:LOC Id="HOP1.HOC1.LOP1.LOC1">
  <allocations>
    <participant idRef="BookingWeb">
      <host idRef="network.molecule-1.web.apache"/>
    </participant>
  </allocations>
  <scl:SCL>
    <scl:ConstraintsSet>
      <scl:Service idRef="BookingWeb"/>
      <web:WebServiceConstraints>
        <web:Connector>
          <web:Port>80</web:Port>
          <web:Protocol>tcp</web:Protocol>
        </web:Connector>
        <web:PathMapping>
          <web:VirtualPath>/</web:VirtualPath>
          <web:LocalPath>
            /usr/local/apache/htdocs
          </web:LocalPath>
        </web:PathMapping>
      </web:WebServiceConstraints>
    </scl:ConstraintsSet>
  </scl:SCL>
  <gsr:GSRBASE>
    <!—snip…
  </gsr:GSRBASE>
</loc:LOC>
```

```
<lds:LDS>
  <Entry>
    <ProblemId>HTML-IntegrityFailure</ProblemId>
    <cond:Condition>
      <iodef:Incident purpose="traceback">
        <iodef:IncidentID name="String">
          event IODEF description
        </iodef:IncidentID>
        <iodef:ReportTime> 2006-12-17:30:47
        </iodef:ReportTime>
            ...
      </iodef:Incident>
    </cond:Condition>
  </Entry>
</lds:LDS>

<lrs:LRS>
  <Entry>
    <ProblemId>HTML-IntegrityFailure</ProblemId>
    <react:Reaction>
      <Execute>
        <Target idRef="network.molecule-1.web.apache"/>
        <Command>/usr/sbin/restore_backup.sh</Command>
      </Execute>
    </react:Reaction>
  </Entry>
</lrs:LRS>
```


Funded by EC contract FP6-027599

Agenda

- Introduction to the DESEREC architecture
- Modelling framework
- Operational Plan definition
- Conclusions

Funded by EC contract FP6-027599

Conclusions

- DESEREC architecture is based on the modelling of the requirs. for target systems
- The framework works by modelling requirements such as underlying equipment and business services from a high level point of view
  - Translating them into the set of valid configurations for the target system
  - The modelling of dependability and security requirements also includes the definition of:
    - Detection scenarios: known issues that can affect the system
    - Reaction scenarios: known available solutions for those specific issues
- The presented modelling framework would be incomplete without the detection / decision / reaction engine
  - Enables the framework to give an automated response to the incidents which may arise
- We are working in:
  - Modelling event descriptions
  - The dynamic transformation from the high level requirements to the final configuration data
  - The development of a complete AI-based decision engine
  - Analysis and conflict resolution tasks over the resulting configurations

Funded by EC contract FP6-027599

Conclusions and plenary session

Session Chairs and Rapporteurs

Conclusions and plenary session

Workshop on

Software and Service Development, Security & Dependability

10-11 July 2007, Maribor

Funded by EC contract FP6-027599

Day 1

- Keynote speakers
  - Conflicts between security policy and business
  - Compliance consequences
  - Predicting system behavior still the largest unsolved problem
  - Potential solution: mix of formal and empirical analysis
  - 4 pillars: proactive failure management, service-oriented OS, fault tolerant services, self-* methods

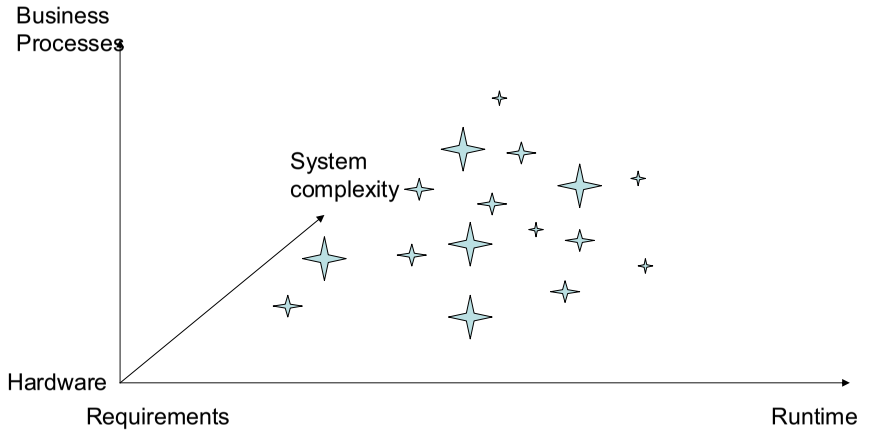
Funded by EC contract FP6-027599

Day 1

- Engineering dynamic and ad-hoc service coalitions
  - Jean Christophe Pazzaglia
  - Luca Durante

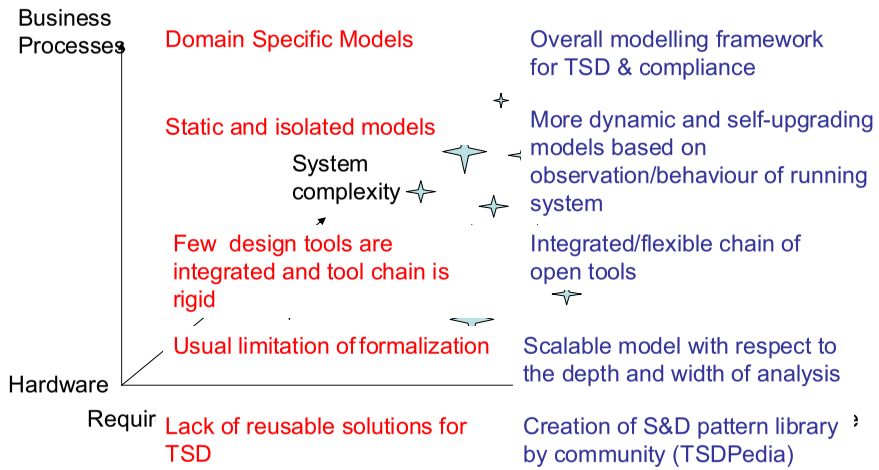
Funded by EC contract FP6-027599

Engineering Dynamic & ad-hoc Service Coalitions



Funded by EC contract FP6-027599

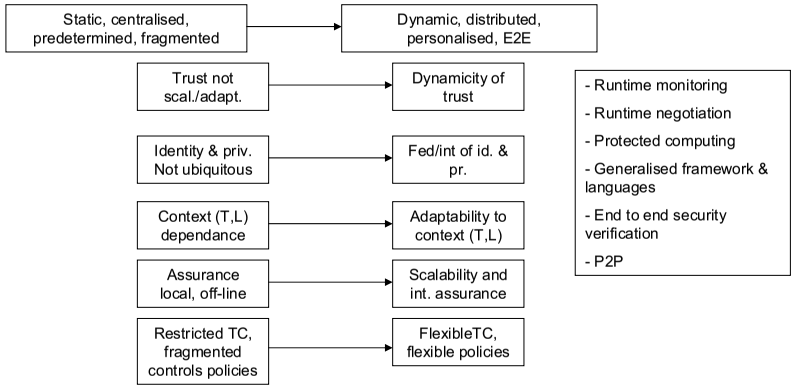
Engineering Dynamic & ad-hoc Service Coalitions



Funded by EC contract FP6-027599

Day 1

- Scalable and adaptive service infrastructures



Funded by EC contract FP6-027599

Day 2

- Keynote speakers
  - Current assurance standards and management practises being insufficient for SOA: need to upgrade SLA´s
  - Need for closer link between trust (as seen by users) and trustworthiness (of infrastructure and components)
  - Scalable trustworthiness: a scalable and resilient platform for resilient SO computing
  - Some basic ingredients: P2P, scalable publish/subscribe etc

Funded by EC contract FP6-027599

Day 2

- Resilience in infrastructures, systems and services
  - Pedro Carvalho
  - Sandy Johnston

Funded by EC contract FP6-027599

State of the art

- People and Process Quality (Assurance)
- Ad hoc (Fault tolerance)
- Distinction between Trust & Trustworthiness is blurred (Ad hoc deployment of enforcements)
- Technical solutions valid for limited period/no plan for fundamental changes

Funded by EC contract FP6-027599

Vision

- Model-based automation for dynamic reaction to change
- Best practice routinely used
- Reactive coverage of faults
- Infrastructures operations should be trustworthy
- Automatic Security & Dependability
- Aware people (reports) to make best Security & Dependability
- Trustworthy coverage/use knowledge formal methods

Funded by EC contract FP6-027599

Outlook Research Agenda

- Trusted infrastructures
- Model synthesis (rigorous models real world)
- Design patterns for fault tolerance
- Promote Intrusion Tolerance (to complement Intrusion Prevention - firewalls etc)
- Visualization of Security & Dependability issues (decision support)
- Models for human abstractions/reasoning
- Hybrid system models (formal proofs)

Funded by EC contract FP6-027599

Day 2

- Resilience in business processes
  - Domenico Presenza
  - Luca Save

Funded by EC contract FP6-027599

Main conclusion from BPR panel

- Resilience of an organisations is different from resilience of its (ICT) infrastructure.
- There is not the ambition to remove humans from the loop. Automation instead has to be read as support.
- It appears there is a gap to be filled
- Software Engineering community is starting to investigate how to extend their approaches to fill the gap. (DESEREC team)

Funded by EC contract FP6-027599

Next steps

- Presentations published on www.esfors.org
- Analysis of results: actions and roadmapping
- Report by Oct 2007

Funded by EC contract FP6-027599