

Modeling Security Requirements Through Ownership, Permission and Delegation

Paolo Giorgini, Fabio Massacci, John Mylopoulos

University of Trento

Nicola Zannone

Eindhoven University of Technology

Delft University of Technology



John

14:05

Motivation

- In 2005, mainstream approaches to security design focused on identifying security requirements *after* system design; and even today an overwhelming focus is still on *system* security.
- Security solutions have to fit a pre-existing design
 - may not be able to accommodate them;
 - security requirements may conflict with functional and quality requirements;
- Social concepts are fundamental to building secure systems
 - Security is often compromised by exploiting vulnerabilities in the social part of a socio-technical system.

Once upon a time



Università degli Studi di
Trento

Modeling Security Requirements Through Ownership, Permission and Delegation

P. Giorgini F. Massacci J. Mylopoulos N. Zannone

www.troposproject.org



Some Ideas... we don' t like

- **Idea 1**
 - Add primitives/constraints to Tropos/Kaos/name-your-pet-RE-formalism for the various security requirements
 - Confidentiality, authentication, access controls or so on are security services and mechanisms NOT security requirements!
 - ACID Transactions are a DB service not a IS requirement...
- **Idea 2**
 - Model security requirements separately from functional requirements
 - This is exactly the problem everybody is ranting about
- **Idea 3**
 - Model the goals of the attacker
 - They are not the goals of the security engineer!



Some ideas... we like

- **Hunch 1**
 - **Security Requirements are social requirements**
- **Hunch 2**
 - **We must model at the same time Functional Requirements and Security Requirements**
 - So we can see the interplay of both and check one does not get in the way of the other
- **Occam's Razor**
 - Add **few** primitive constructs
 - Other security requirements as patterns, services, mechs

Nicola

14:09

We were not the first to address SRE

- Early RE models (SREs using “vanilla” RE)
 - [3] Anton, Privacy reqs with privacy goals. RE’02
 - [6] Crook+. SREs as anti-reqs. RE’02.
 - [19] Liu+. SREs with goal models. RE’03
 - [25] Toval+. Legal reqs. RE’02
- SRE specific graphical model but no logic
 - [9] Fredriksen+ CORAS for risk assessment. SAFE-COMP’02
 - [22] McDermott & Fox. Abuse Cases. ACSAC’99
 - [24] Sindre & Opdahl. Misuse Cases. TOOLS’ 2000
- Logic and tool for security but no model
 - [11] Gans+. Trust and Distrust in Agents. AOIS’01
 - [15] Jones & Sergot Formal Institutionalised Power. JGPL 1996.
 - [18] Li+. Trust-management Framework. IEEE SSP’02
- Logic, graphical model and tool but focus on system
 - [16] Jurjens. UMLSec. 2004.
 - [20] Lodderstedt+ SecureUML. UML’02
 - [26] van Lamsweerde+ Anti-Goals. RHAS’03

10 years later

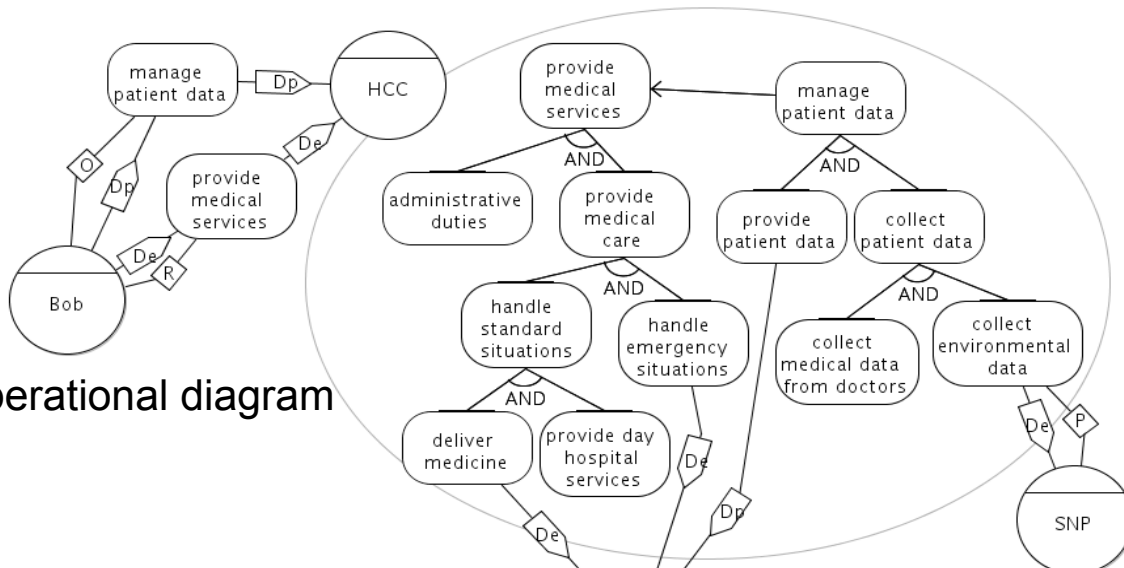
Contributions of our proposal

- The first work providing (at the same time):
 - a Security specific (and novel) **ontology** for talking about an important class of security requirements, namely ownership and permissions
 - a coherent **graphical** representation for both functional and security requirements
 - a **reasoning** technique for formal requirements analysis
- CASE tool support
- Cross communities
 - Requirements Engineering
 - Security

Security-specific Ontology

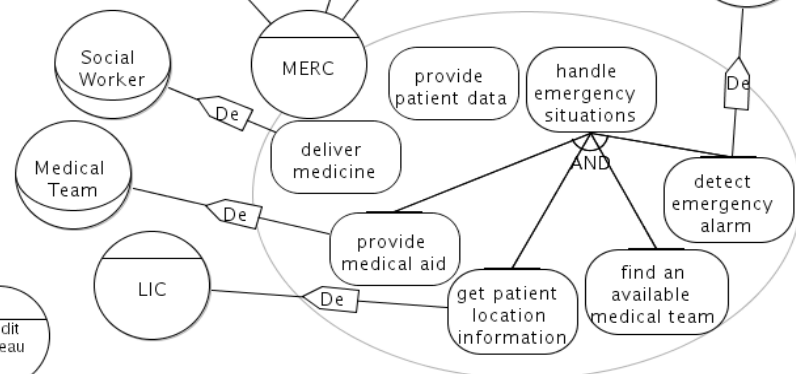
- **Permission** != **Execution**
 - Authority vs. Responsibility
- **Ownership** != **Provisioning**
 - Distinguishing who can fulfill a goal from who is entitled to decide who can fulfill a goal
- **Trust** relationship between actors
 - Distinguishing trusted actors from untrusted actors
- **Delegation** relationship between actors
 - Formal transfer of authority/responsibility

Graphical Representation

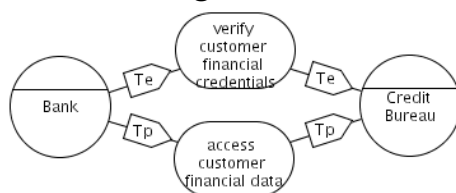


Operational diagram

- O: owns
- P: provide
- R: request
- Dp: permission delegation
- De: execution dependency
- Tp: trust of permission
- Te: trust of execution



Trust diagram



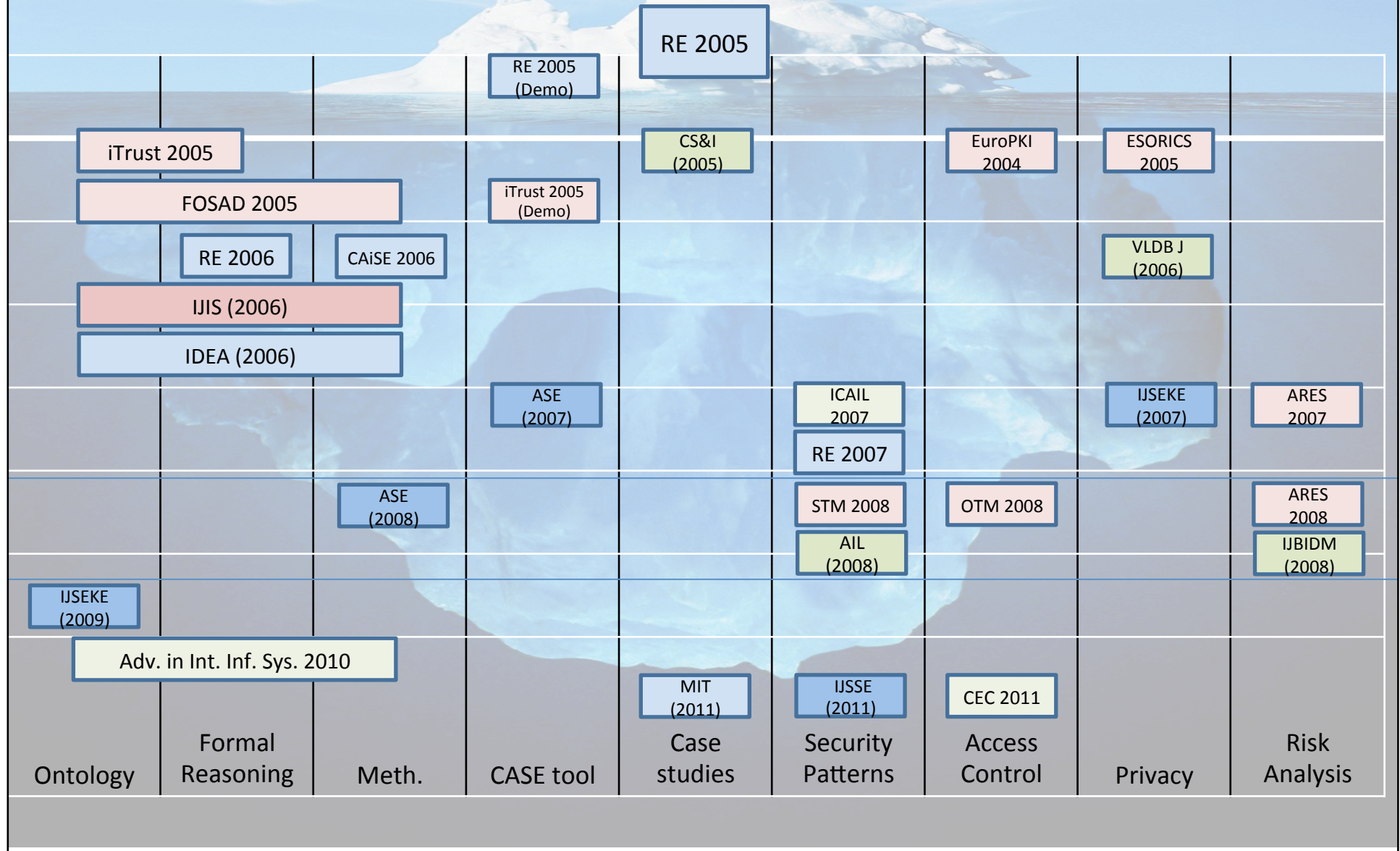
Formal Reasoning

- Formal Model
 - Answer Set Programming (aka Datalog-)
- Axioms
 - Intensional properties and rules
- Models (SI* diagrams)
 - Extensional properties of classes (and instances)
- Properties
 - Formulas that may be true or may not be true
 - Availability (3), Confidentiality (1), Authorization (3), Avail+Auth (3), Privacy (1)
 - eg Need-to-know: all actors which have been delegated a permission to fulfill a goal has also been delegated the execution of the goal (directly or indirectly)

FABIO

14:16

Beyond the tip of the iceberg



Involving Industry

- Several studies in joint research projects
 - SAP (SERENITY, MASTER, TAS3)
 - Thales (SECURECHANGE, ANIKETOS)
 - ATOS (MASTER, ANIKETOS)
 - Engineering SpA (SERENITY, MASTER)
 - British Telecom (MASTER)
 - DBlue (SERENITY, SECURECHANGE, ANIKETOS,
 - National GRID (SECONOMICS)
- A painful road to humility
- The realm of measurable value and the academically unexpected...



What is important in a tool for industry?

- Our Expectation [RE'05]
 - “In addition, the tool provides a user-friendly interface to the DLV system and permits a designer to select properties of each model and to specify additional security policies. The resulting Datalog specifications are automatically verified by the DLV system.”
- VM, former Air Traffic Controller, Expert in Human Factors for Safety, 35+ years of experience, CTO of small company
 - “Your tool has a bug. We were verifying a safety pattern and a window popped up with... you know that Windows error... Ax07F12”
 - Well, it was not actually a bug, the window presented a datalog formula showing how trusted delegation would not hold
- Still “debugging it” after 10 years
 - E.g. Formal method is there but has to be “transparent”

10 years later

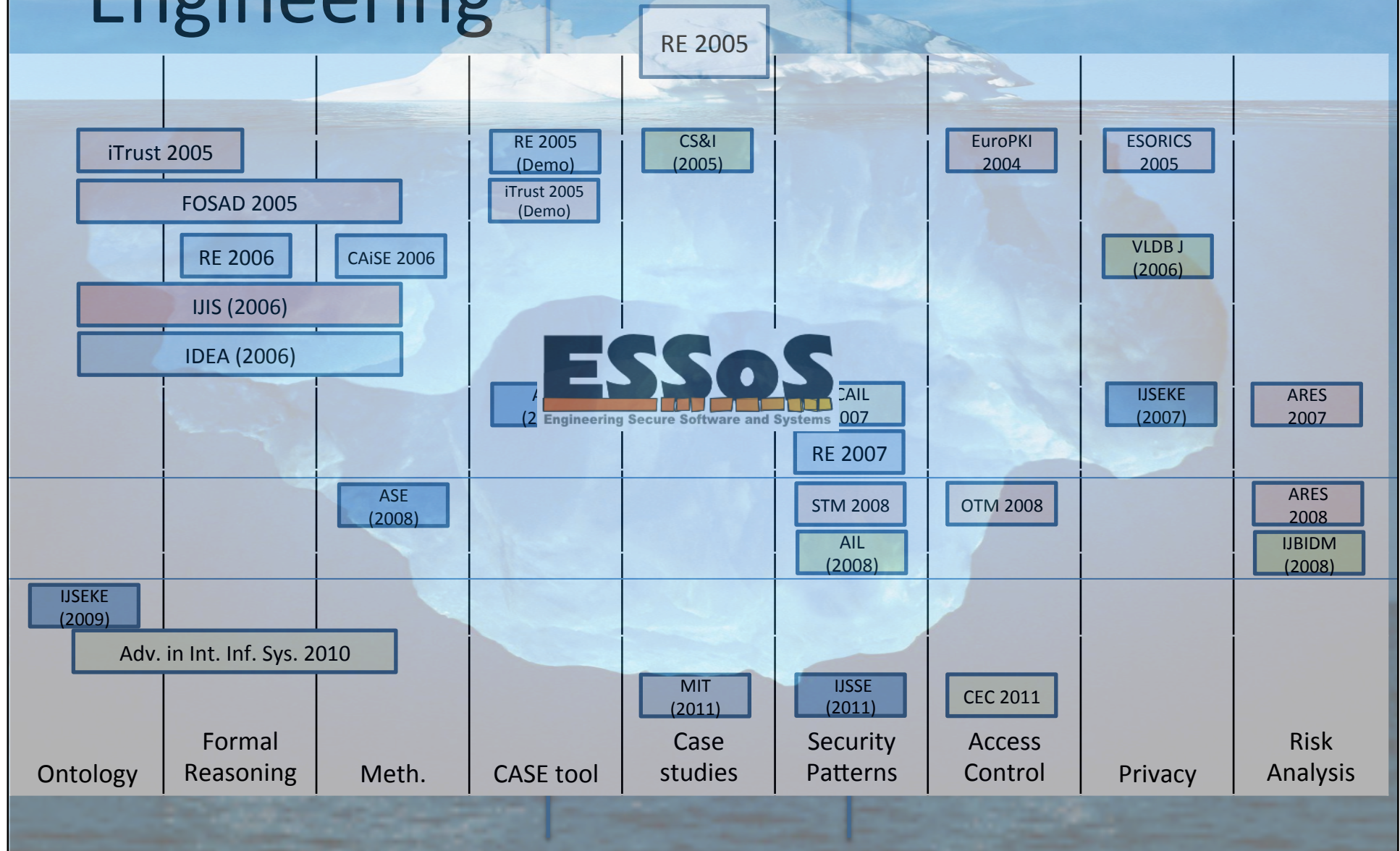
tool != model stencil

- Meta-Models not just Graphics
 - Different Industries → different graphics convention
 - Air Traffic Management vs Business Processes
 - Must have a flexible meta model for plugging different models
- Interface with Reasoning Capabilities
 - Different applications → different reasoning reqs
 - untangle trust relationships vs compute risk values
 - Interface with different reasoners might be needed
- Process Support
 - Main lesson from eRISE Challenge
 - evaluating SRE methods with professionals and students
 - You can't just leave dudes figuring out what to do next and whether they wrote is a 'model' or a 'pile of gibberish'
- Automatic report generation in a pdf
 - Yes, that's measurable value (writing reports is expensive!)

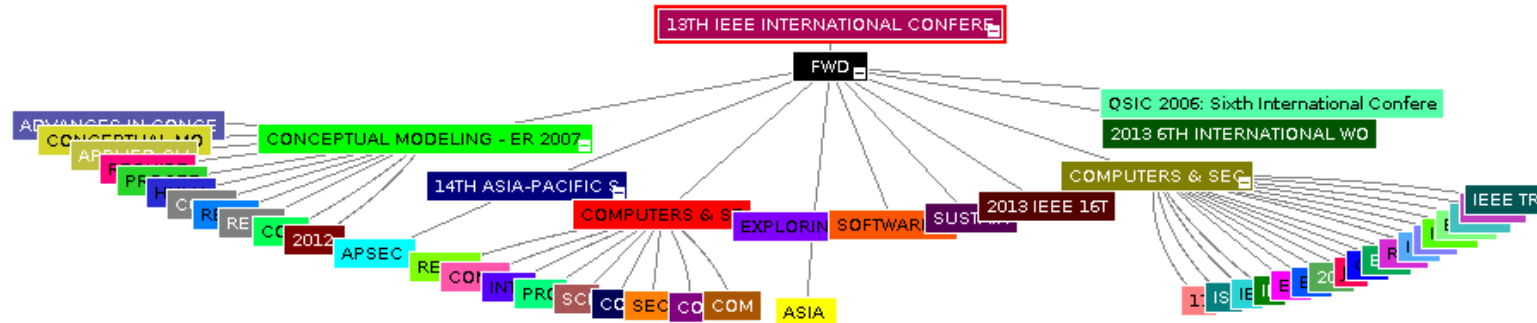
PAOLO

14:21

Software Engineering Behind the tip of the Iceberg



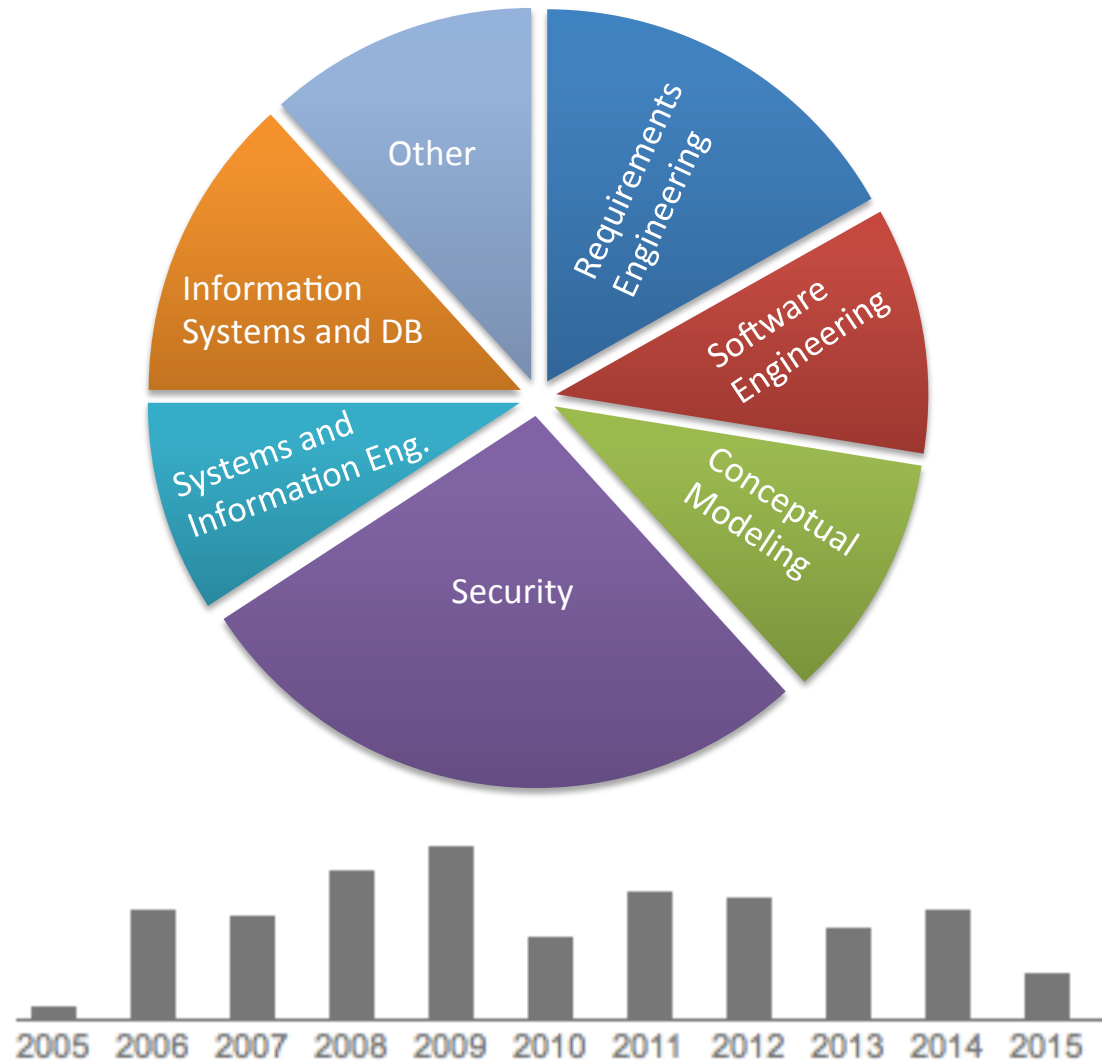
Impact of this work on others



- **Legal requirements:** our ontology provided a baseline for the definition of ontologies for modeling legal requirements
- **Trust management:** inspired methods for elicitation, specification and analysis of trust requirements
- **Security patterns:** inspired the definition of security patterns at organizational level

10 years later

Google Scholar (200 cit.)

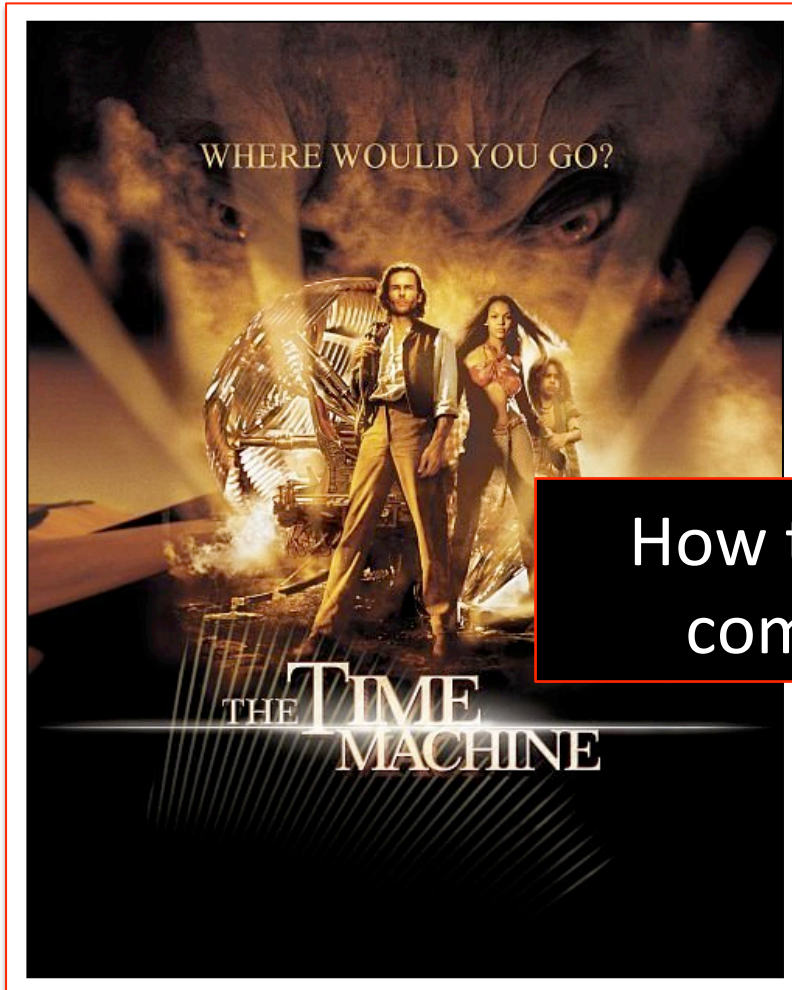


10 years later

Sample Citation Venues (>2cit)



What happened next?



RE'05 - Modeling Security Requirements

What to do with elicited SRs?

How to manage complexity?

How to close the gap to design?

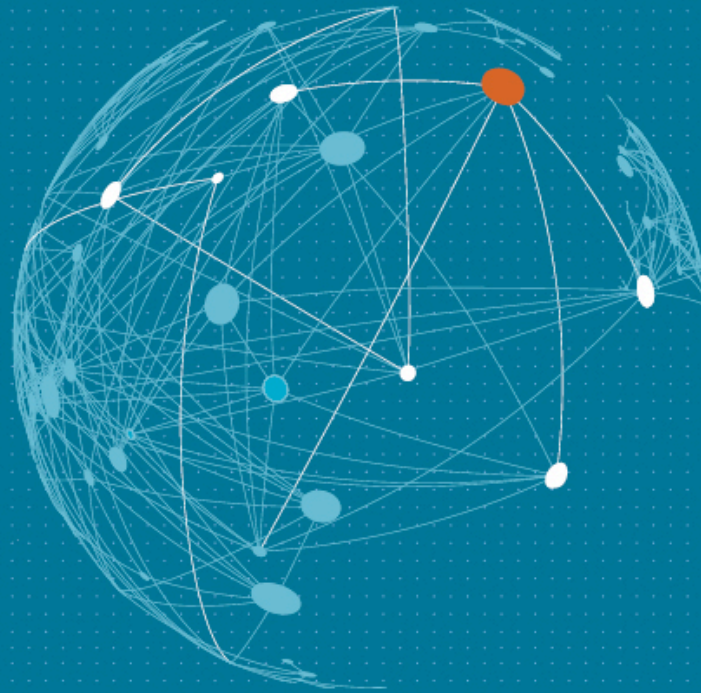
Does it really work in practice?

A MIT Press Book

Security Requirements Engineering

Designing Secure Socio-Technical Systems

Fabiano Dalpiaz,
Elda Paja,
and Paolo Giorgini



Nicola

14:25

10 years later

You elicited, so what?

- Realization of security and trust requirements
- Two inspiring follow-ups:
 - (2008) A Model-Driven Approach for the Specification and Analysis of Access Control Policies
 - (2006) Hierarchical Hippocratic databases with minimal disclosure for virtual organizations
- Ongoing research directions
 - Access control for distributed and collaborative systems
 - automotive, cloud, smart grid, social networks, systems of systems
 - Privacy compliance
 - Anomaly detection and analysis
 - Trust management
 - Credential-based, reputation-based



Paolo

14:27

10 years later

How to manage complexity?

- You can't just plug everything into a model
- Multi-view Socio-Technical Security (STS)
 - Social, Information and Authorization views
- From STS specification down to BP design and security enforcement
 - Security requirements refinement
- Visual Privacy (EU H2020 Project)
 - Visual models for information owners



SecBPMN



John

14:29

10 years later

And the gap towards architecture?

- Vanilla security analysis focuses on the system level; in our RE'05 paper we focused on the social level.
- But attacks often strike at the weakest link of a socio-technical system, social, system or infrastructure, and nowadays are often composite.
- Tong Li (PhD student, UniTN) is developing a holistic security analysis framework that supports analysis across all three levels.
- His analysis uses anti-goals and attack patterns from public domain repositories.



Fabio

14:31

10 years later

Does it really work?

- Full fledge security requirements engineering is often too costly (Industry paper at ESEM'14)
 - We need empirical protocols to evaluate RE models & methods, and understand what works, what doesn't work and why → K. Labunets (PhD @ UNITN)
- Is Process is more important than graphics? (NordSec'12)
- Is Perception everything?
 - Graphical SRE method are systematically *perceived* as superior to tabular SRE methods (ESEM'13,EMPIRE'14)
 - But there is *no diff in actual result* when industry people evaluate the final outcome (EMPIRE'14)
 - What about comprehension? (Watch this space)
- Do catalogues make a difference? (REFSQ'15)



Fabio

14:33

Take Away Messages

- Security & functional reqs can be elicited together
- Social models have a place in Security RE
- Representation, Reasoning, Running Code
 - You need all three to make an impact
 - Just adding the label “S” for “Security” to your pet RE method doesn’t make it a SRE
- Industrial evaluation & deployment is not the ‘last mile’ is the ‘last light year’

