SENSORIA Results Applied to the Case Studies

(≈ deliverable D8.8)

Maurice ter Beek

ISTI–CNR, Pisa, Italy

SENSORIA general meeting
Milan, 25 November 2009

Outline

Scope

Emphasize the central role of the case studies in *feeding* and *steering* the research in SENSORIA

- No description of the case studies
- No details of the techniques, methods and languages
- Not limited to the Finance case study, but consider applications also to the other four case studies from WP8:
  - Automotive, eUniversity, Telecommunications and Bowling Robot

Describe the experience/benefits of applying a particular technique, method or language to a case study scenario and *not* the technique, method or language itself, which is presented in the other chapters

- Deliverable D8.8 will be limited to the applications *after* M36, book chapter hence based on D8.5/D8.6/D8.7 from M36 and D8.8 from ~~M48~~ M54

Contributions

Call for contributions same for all deliverables D8.5/D8.6/D8.7/D8.8

Aim: Besides validating the technique, method or language against requirements of the case study, was there a specific *aim* that triggered the use of the case study?

Experience: Which were the *problems* (if any) faced when applying the technique, method or language to the case study and what are the *results* that have been obtained?

Benefits: What have been the *advantages* (from the engineering, scientific or business point of view) of applying the technique, method or language to the case study?

Feedback: Did the application of the technique, method or language to the case study lead to *improvements* of that technique, method or language?

Validation of SENSORIA techniques, methods and languages

www.sensoria-ist.eu

	Automotive	Finance	eUniversity	Bowling Robot	Telco
UML4SOA	√	√	√	√	√
SRML	√	√			
StPowla	√	√			√
COWS	√	√			
SOCK/Jolie	√	√			
(Mar)CaSPiS	√	√		√	
CC		√			
λ^{req}	√				
CaPiTo		√			
SRMC			√		
cc-pi		√			√
ADR	√				
Institutions			√		
SocL	√	√			
CMC-UMC	√	√		√	√
chorSLMC		√			
Flow logic		√			
SC-ESC	√	√			
OCPR	√	√			
WS-Engineer	√	√		√	
SoSL/MoSL		√		√	
PEPA toolkit	√	√	√	√	
Modes/Dino	√	√			
JCaSPiS		√			
MDD4SOA	√	√	√		√
VIATRA2	√	√			
GT		√			
MBTE		√			
Patterns	√	√	√		
SDE	√	√	√		

Information Society Technologies

Synthetic overview

- Clearly illustrates the central role of the industrial case studies from the Automotive and—in particular—Finance domain, and the more specific role of the academic eUniversity case study

- It moreover shows that only a few of the SENSORIA techniques, methods and languages have been applied to the Bowling Robot and Telecommunications case studies:

  1. Telecommunications case study has suffered from the fact that Telecom Italia has left the SENSORIA consortium by end of M18

  2. The Bowling Robot case study has been introduced around M36; initially as a game scenario to demonstrate SENSORIA's software engineering approach at hands-on demonstrations, after which it has evolved into an academic case study

Detailed analytic overviews

Theme 1: Linguistic Primitives (WP1, WP2 & WP5)

- Deals with *linguistic primitives* for services and their interaction and composition, developed both on an *architectural* level (e.g. UML4SOA, SRML) and on a *programming* level (e.g. COWS, SOCK, Jolie, CaSPiS, CC)

Theme 2: Qualitative and Quantitative Analysis (WP3 & WP4)

- Deals with *type systems* (e.g. λ^{req}), *logics* (e.g. SoCL, SoSL) and core calculi *extensions* (e.g. MarCaSPiS) to develop verification techniques (e.g. CMC/UMC, ChorSLMC, PEPA SW toolkit) to *analyse* behavioural, performance and QoS properties of services

Theme 3: Deployment and Development (WP6 & WP7)

- Deals with engineering aspects of services: *deployment* (e.g. Modes, Dino, JCaSPiS, VIATRA2), *reengineering* (e.g. graph transformations) and *model-driven development* (e.g. MDD4SOA, SDE)

Theme 1: experience/benefits of SENSORIA techniques, methods and languages

www.sensoria-ist.eu

Theme 1	Automotive	Finance	eUniversity	Bowling Robot	Telco
UML4SOA	Test its usefulness in practice; Provide models for formal verification				
SRML	Validate primitives,3-layer approach,def.SLA				
StPowla		Validate approach; Now aut. derives policy templates			Assess impact business process design
COWS	Feasibility mechanisms & primitives for SOA				
SOCK	Validate primitives & faults/compensations modelling; Verify dynamic handler & aut. fault notification; Now improved handlers				
Jolie	Test service-oriented programming; Found new SOA patterns				
(Mar)CaSPiS	Test effectiveness & Markovian extension			Feasibility	
CC		Type to verify key properties			
λ^{req}	Orchestrate with policies & contracts				
CaPiTo		Model SOA especially when security by crypto protocols			
SRMC			Test expressivity; Novel analysis approach created		
ADR	Disambiguate informal spec. by formal model reconfigurations & constraints				
Institutions			Validate approach; Separate behavioral description services from choreography		

Information Society Technologies

Theme 2: experience/benefits of SENSORIA techniques, methods and languages

www.sensoria-ist.eu

Theme 2	Automotive	Finance	eUniversity	Bowling Robot	Telco
StPowla	Detection of conflicts by theorem proving; Needed extension Appel semantics	Detection of conflicts by UMC; Needed definition of correspondence Appel-UML			
SocL	Test expressivity for SOA properties; Defined patterns of service properties				
CMC	Test & fine-tune model checker; Verify properties; Feasibility type checking; Now automated translation from UML4SOA to COWS				
UMC	Test & fine-tune model checker; Verify properties; Now automated translation from UML4SOA into UMC			Test & fine-tune model checker; Verify properties	
CaSPiS		Type system to check progress property; Control flow analysis to detect & prevent misuses			
chorSLMC		Verify multiparty interaction; Proved protocols of interaction well-defined & deadlock-free			
SC-ESC	Experiment, evaluate & reason about long running transactions				
Flow logic		Proved authenticity & confidentiality			
OCPR	Defined compositional notion of service equivalence; Verified service equivalence & replaceability				
WS-Engineer	Analyzed correctness & consistency of service compositions; Helped to further develop the tool			Test & further develop tool	

continued on next slide...

Information Society Technologies

Theme 2: experience/benefits of SENSORIA techniques, methods and languages – cont'd

Theme 2	Automotive	Finance	eUniversity	Bowling Robot	Telco
SoSL		Test expressivity for SOC features; Verify workload, reactivity & performance		Show methodology provides intuitive & effective estimations of robot behaviour	
MoSL	Validate expressivity dependability & performance properties of services: difficult & error-prone in natural language				
λ^{req}	Use output model checker to highlight design flaws, suggesting how to revise orchestration & security policies				
PEPA toolkit	Passage-end, response-time & safety analysis; Development environment now more user-friendly	Sensitivity & response-time analysis to identify bottleneck activity; technically particularly challenging	Show main developments PEPA wrt deterministic semantics; Key benefit of PEPA: both a deterministic & a stochastic semantics	Passage-end analysis: precisely quantified probability of expected outcome	
SRMC		Test software tool suite; Verify scalability in presence of uncertainties			

Theme 3: experience/benefits of SENSORIA techniques, methods and languages

Theme 3	Automotive	Finance	eUniversity	Telco
Modes	Test modelling orchestration & choreography requirements; Created UML2 profile; Extended analysis & WS-Engineer			
Dino	Provide requirements, guide design & validate Dino; Now improved & extended design			
JCaSPiS		Implement real-world scenario; Preserve analysis modelling to implementation level; Now new functionalities		
MDD4SOA	Validate practicability UML4SOA & usefulness MDD4SOA transformers in practice (by full transformation from UML4SOA to actual code & execution); Developed hand in hand with UML4SOA; Transformers now provide more readable & precise error reporting			
VIATRA2	Test method; Support incremental service development & re-usability; XML descriptors created automatically			
GT		Untangle business & presentation logic; New rule set; Architecture migration now automated		
MBTE		Test method; Steered better user guiding		
Patterns	Accompanied by examples from the case study application that identified pattern; Created pattern catalogue documenting (dis)advantages & feedback of approaches			
SDE	Orchestration features employed to combine integrated tools for development & analysis of case studies; Now contains all SENSORIA tools; Now improved SDE interfaces & graphical orchestration mechanism			

Information Society Technologies

Conclusions

- If your technique, method, or language is missing: please send me your contribution asap...
- Overviews available for review:
  1. Present synthetic overview to show central role of case studies
  2. Present analytic overview only of SENSORIA's "main" (which?) techniques, methods and languages
- Thanks to all the SENSORIA partners for their contributions!!!