Tropos[1] at the Age of Eight: On-going Research at FBK, UniTN and UT

Paolo Giorgini[1], John Mylopoulos[1,2], Loris Penserini[3], Anna Perini[3], Angelo Susi[3]

[1] University of Trento, Trento, Italy
{paolo.giorgini,jm}@disi.unitn.it

[2] University of Toronto, Toronto, Canada

[3] Fondazione Bruno Kessler, Povo, Trento, Italy
{perini,penserini,susi}@fbk.eu

Abstract. The Tropos project was launched in the Spring of 2000. Its aim has been to establish a methodology for building agent-oriented software systems. The methodology that has emerged is founded on the *i** modelling framework to support four phases of software development: early and late requirements, as well as architectural and detailed design. The purpose of this report is to offer an overview of on-going work on the project at Fondazione Bruno Kessler (FBK), the University of Trento (UniTN) and the University of Toronto (UT).

1 Introduction

The Tropos project was launched in the Spring of 2000 at the University of Toronto (hereafter UT), the University of Trento (UniTN) and the Fondazione Bruno Kessler (FBK) known as IRST back in those days. Its aim has been to establish a methodology for building agent-oriented software systems. The methodology that has emerged is founded on the *i** modelling framework to support four phases of software development: early and late requirements, as well as architectural and detailed design. Its initial contributors (… founding fathers and mothers) included at UT Jaelson Castro[2], Manuel Kolp and John Mylopoulos; at UniTN/FBK Paolo Bresciani, Paolo Giorgini, Fausto Giunchiglia, Anna Perini, Marco Pistore and Paolo Traverso.

The first major milestone of the project was to lay out a methodology for building agent-oriented software. This milestone was achieved within the first year with the help of two case studies, leading to the most cited publications of the Tropos project [Castro02], [Bresciani04]. The next milestones focused on developing formal reasoning techniques to support the Tropos methodology. One thread of research

[1] “tropos”, in Greek τροπος, is an ancient word. The very first words in Homer’s Odyssey are “Ανδρα μοι εννεπε μουσα πολυτροπον …” – “Muse, help me tell the story of the man of many ways” (…”the man” is Ulysses).

[2] On leave from the Federal University of Pernambuco (Brazil).

aimed to develop a tool that would enable verification of Tropos models through model checking. This work led to the Formal Tropos specification language and the T-tool [Fuxman04]. In parallel, the UniTN team developed formal reasoning techniques for goal (and softgoal) models, along with the G-tool that implemented these techniques [Giorgini03], [Sebastiani04]. Publications on and running versions of these tools can be found at http://www.troposproject.org/.

The one-and-only purpose of this report is to offer a guide to some of the research threads at FBK, UniTN and UT that followed the original milestones outlined above. There have been other significant threads of research at other universities, but they fall outside the scope of this report.

The rest of the document is structured as follows. Sections 2, 3 and 4 overview respectively on-going but reasonably mature research threads at the three institutions. Section 5 concludes and offers some hints on future directions for the project.

2 FBK

Research on Tropos is conducted within the Software Engineering (SE) unit at FBK.[3] More generally, the research carried out by the SE unit addresses the development of complex software systems, having large size, operating in a distributed environment, exhibiting autonomic behaviours, expected to fulfil high quality standards, and realized using innovative technologies and approaches. The SE unit focuses on two strategic areas of software development, namely *Requirements engineering* and *Code analysis and testing.* In the first area, the scientific challenges deal with the explicit representation of requirements for autonomic behaviours (e.g., those of self-adaptive systems), of the normative constraints and of the flows. Here, agent-oriented approaches seem particularly promising. In the area of software testing, the challenge is to automate the generation of the test cases and their execution.

Research results contributed to the extension of the agent-oriented modelling tool TAOM4E (http://sra.itc.it/tools/taom4e/). Advanced functionalities include test case derivation and execution (see the eCAT framework) and automated BDI code derivation [Morandini07a].

Normative i* modelling. A distinguishing feature of socio-technical organisations over ad hoc groups of interacting individuals is the existence of norms. Various types of norms exist in the real world, but those that are more relevant at requirements time are behavioural norms that impose actions to be performed, goals to be achieved, resources to be delivered or principles to be respected. We propose to use a goal-oriented approach, based on i*, for modelling such kind of norms and introduce a limited set of additional abstractions and diagrams for modelling norms. More specifically, our idea is to model contextually and homogeneously, but separately, the normative context of a domain and its stakeholders with their intentionality [Siena07a]. A recent application of normative i* modelling to a food-chain scenario gave promising results towards proving its effectiveness [Siena07b].

[3] More details on research activities, projects and collaboration at http://se.fbk.eu.

High-Variability Design for Software Agents: Extending Tropos. High-variability design has been proposed to generate generic software solutions and to support self-configuration in autonomic software. Complementing research developed in UT, we focused on designing software agents [Penserini07, Morandini07b]. We extended the Tropos methodology, enhancing its ability to support high variability design, through the explicit modelling of alternatives, by adopting an extended notion of agent capability. A tool-supported process founded on the Model-Driven Architecture (MDA) framework and standards, supports goal-oriented analysis of requirements of self-configuring software and the derivation of BDI agent code which realizes them.

Goal-Oriented Testing. Goal-oriented specifications are particularly appropriate for distributed, concurrent systems, which communicate by means of messages and have been designed to behave autonomously (like agents). Testing of these kinds of systems remains an unexplored area, of great importance for their adoption in SE practice. We are studying testing techniques for goal-oriented systems. In particular we address the problem of automating test case generation as well as their execution.

Main results of this research include a goal-oriented testing methodology that complements Tropos analysis and design [Nguyen07a]. Test cases are derived directly from the goal-oriented specification of the system under test; a novel testing framework, called eCAT[4], which integrates manual and automated test cases generation techniques, so that it can generate and evolve test cases automatically, and run them continuously [Nguyen07b,c].

3 UniTN

At UniTN, research on Tropos is done within the Software Engineering and Formal Methods research group[5]. Three are the most relevant research activities: Security Modelling and Analysis, Goal-based Risk Analysis and Automated Design.

Security Modelling and Analysis

Managing high-level user requirements is a key issue for the successful and cost effective development of IT systems, but managing security requirements is almost completely ignored. We propose a requirements engineering methodology, Secure Tropos [Giorigni05a, Giorgini05b, Giorgini06c], to support IT designers in the capture of high-level security and trust requirements and their implementation. In particular, we have extended and refined the i*/Tropos methodology with basic primitives suitable for capturing security aspects of organizations. In particular, we introduced primitives for modelling entitlements of actors and making explicit their capabilities. Moreover, the notions of delegation and (dis)trust are used to model the transfer of entitlements and responsibilities between actors, and the expectation of an actor about the behaviour of other actors. Once the security and trust model has been captured, our purpose is to automatically verify security and trust requirements [Giorgini06a]. To provide automated reasoning support with a quick prototyping lifecycle we use Datalog. In this setting, each concept/relation occurring in graphical

[4] See http://sra.fbk.eu/people/cunduy/ecat/. eCAT has been integrated with TAOM4E.

[5] More details about the group can be found at http://dit.unitn.it/research/rp.xml?rpid=3

diagrams is represented as a Datalog predicate. The collection of these predicates represents the extensional description of the system. The formal framework is comprised of rules that define the semantics of primitive concepts and are used to make explicit the information that are necessary for the verification of security requirements. Such information is then used to define constraints whose violation points out inconsistencies in the system [Giorgini06b]. These constraints are essentially in form of patterns that represent system vulnerabilities.

Goal-based risk analysis
Goal models have been proved to be useful to model and analyze stakeholder objectives to elicit requirements of information systems. However, a goal model also needs to anticipate uncertain circumstance that can affect the achievement of stakeholder objectives. Therefore, Goal-Risk Framework [Asnar06a, Asnar06b] are introduced extending Tropos goal model with 3 layers of conceptual analysis: goal, event, and treatment layer. Goal layer is meant to analyze strategic interest of stakeholders, event layer analyzes the impact of uncertain events to the goal layer (i.e., a risk is uncertain event with negative impact), and treatment layer analyzes a course of actions that are meant to treat uncertain events (e.g., mitigate risks). Using this framework, an analyst can model and reason about IS requirements that have encompassed risks and their mitigation besides stakeholder objectives [Asnar07a]. The framework has been implemented and enhanced for analyzing safety critical systems (e.g., Air Traffic Management [Asnar07b]) and goal deliberation process of autonomous agent systems [Asnar07c].

Automated Design
The focus of the work is on exploring the space of alternative choices during requirements analysis and design of information systems. Namely, the problem is in how to find an optimal/good-enough set of delegations and assignments of goals (to be fulfilled by a system) to the system actors. The approach taken consists of two parts: generating alternative design structures with the help of AI (Artificial Intelligence) planning techniques, and evaluating the generated alternatives with respect to the local strategies of system actors [Bryl06b]. The problem of constructing a design structure that guarantees the fulfilment of system goals is framed and formalized as a planning problem. An off-the-shelf planning tool is used to generate an alternative design structure, which is then evaluated, amended and finally adopted [Bryl06a]. Evaluation schema is inspired by game-theoretic ideas; basically, system actors are seen as self-interested and rational players that are trying to maximize their local utilities, i.e. the benefit they could gain from the adopted alternative. The prototype tool (P-Tool) implements the approach, and is supposed to support the designer in selecting good-enough alternative design structures. The described planning-and-evaluation approach has a number of applications, e.g. it was applied to the problem of self-configuring systems [Bryl06c], which change their structure in response to internal or/and environmental changes.

4 UT

We present three mature research threads.

Variability in Goal Models. Goal models describe a set of alternative ways for fulfilling a requirement. We are interested here in making the design of such models more systematic by identifying the origins of variability. For example, variability may arise from a choice of the agent assigned to fulfil a goal, the medium to be used, or the time of the fulfilment [Liaskos06]. Once variability is identified, it can be used to support personalization [Liaskos05].

Goal-oriented design. Goal-oriented design is characterized by an explicit consideration of design alternatives, and a selection based on non-functional requirements (a.k.a. softgoals). However, the space of design alternatives is based partly of the solution space for the problem-at-hand (dealt with by goal models) and partly on the nature of the artifact-to-be. We have been exploring two threads of research on this.

Lei Jiang, Alex Borgida and Thodoros Topaloglou have been exploring goal-oriented database design. Here, the idea is to start from stakeholder goals, identify plans for fulfilling them, pinpoint information needs for these plans, and design a database on that basis. Variability is an important parameter here: there are many possible designs for a given set of stakeholder informational goals. So are data quality considerations that can make-or-break an information system [Jiang07].

Along a different path, Alexei Lapouchnian is developing a methodology for design that starts from stakeholder requirements expressed as goal models and refines them to generate business process designs [Lapouchnian07]. The proposed methodology exploits the variability inherent in goal models to generate business process designs that that can fulfil root level goals in multiple ways.

5 The Future

Future trends for the Tropos project are largely dictated by the emerging focus on run-time software behaviour. This trend is manifested under different buzzwords: autonomic, adaptive, dynamic, etc. Independently of the buzzword, the theme is the same: software in the future will have to self-manage itself and adapt to changes in its environment through monitoring, diagnosis and compensation components.

The other major trend influencing Tropos is the broadened scope of modelling, analysis and design techniques to support not just software systems through their lifecycle, but also the organizational environment within which they live and operate.

Acknowledgements. The Tropos project benefited tremendously from generous funding provided by the Provincial Government of Trentino through the STAMPS (a.k.a SMTPPS) project and the Natural Sciences and Engineering Research Council (NSERC) of Canada.

References

[Asnar06a] Y. Asnar and P. Giorgini. Ensuring Dependability in Socio-Technical System by

Risk Analysis. In *Proceedings of the 6th European Dependable Computing Conference*, 2006.

[Asnar06b] Y. Asnar and P. Giorgini. Modelling Risk and Identifying Countermeasures in Organizations. In *Proceedings of 1st International Workshop on Critical Information Infrastructures Security* (CRITIS '06), 2006.

[Asnar07a] Y. Asnar, P. Giorgini, F. Massacci, and N. Zannone, From Trust to Dependability through Risk Analysis. In *Proceedings of the Second International Conference on Availability, Reliability and Security* (AReS 2007), Vienna, April 10th-13th, 2007

[Asnar07b] Y. Asnar, R. Bonato, P. Giorgini, F. Massacci, V. Meduri, C. Riccucci, and A. Saidane, Secure and Dependable Patterns in Organizations: An Empirical Approach. In *Proceedings of the 15th IEEE International Requirements Engineering Conference* (RE '07), India Habitat Center, New Delhi, India, October 15-19th, 2007

[Asnar07c] Y. Asnar, P. Giorgini, and N. Zannone, Reasoning about Risk in Agents Deliberation Process: a Jadex Implementation. In *Proceeding of 8th International Workshop on Agent Oriented Software Engineering* (AOSE '07), Honolulu, Hawai'i, May 14th, 2007

[Bresciani04] Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J., and Perini, A., "Tropos: An agent-oriented software development methodology", *Journal of Autonomous agents and Multiagent Systems* (JAAMAS), Kluwer, 203–236, 2004.

[Bryl06a] Bryl, V., Massacci, F., Mylopoulos, J., Zannone, N.: Designing security requirements models through planning. In: *CAiSE'06*, Sprinter (2006) 33–47.

[Bryl06b] Bryl, V., Giorgini, P.: Designing cooperative IS: Exploring and evaluating alternatives. In: *CoopIS*, Springer (2006) 533-550.

[Bryl06c] Bryl V., Giorgini P.: Self-configuring socio-technical systems: Redesign at runtime. *ITSSA*, 2(1):31-40, 2006.

[Castro02] Castro, J., Kolp, M., and Mylopoulos, M., "Towards Requirements-Driven Information Systems Engineering: The Tropos Project", *Information Systems (27),* Elsevier, Amsterdam, The Netherlands, 365–389, 2002.

[Fuxman04] Fuxman, A., Liu, L., Mylopoulos, J., Pistore, M., , Roveri, M., Traverso, P., "Specifying and Analyzing Early Requirements in Tropos", *Requirements Engineering Journal 9(2)*, Springer-Verlag, 132-150, July 2004.

[Giorgini03] Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R., "Formal Reasoning Techniques for Goal Models", *Journal of Data Semantics 1(1)*, Springer LNCS 2800, 1-20, 2003.

[Giorgina05a] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone. Modeling Security Requirements Through Ownership, Permission and Delegation. *In Proc. Of RE'05*, pages 167–176. IEEE Press, 2005.

[Giorgini05b] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone. Modelling Social and Individual Trust in Requirements Engineering Methodologies. In *Proc. of iTrust'05*, volume 3477 of LNCS, pages 161–176. Springer, 2005.

[Giorgini06a] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone. Detecting Conflicts of Interest. In Proc. of RE'06, pages 315–318. IEEE Press, 2006.

[Giorgini06b] P. Giorgini, F. Massacci, J. Mylopoulos, and N. Zannone. Requirements Engineering for Trust Management: Model, Methodology, and Reasoning. Int. *J. of Inform. Sec*., 5(4):257–274, 2006.

[Giorgini06c] P. Giorgini, H. Mouratidis, and N. Zannone. Modelling Security and Trust with Secure Tropos. In *Integrating Security and Software Engineering: Advances and Future Vision*, chapter VIII. IDEA Group, 2006.

[Jiang07] Jiang, L., Topaloglou, T., Borgida, A., Mylopoulos, J., "Goal-Oriented Conceptual Database Design", 16th Int. IEEE Conf. on Requirements Engineering (RE'07), Delhi, October 2007.

[Lapouchnian07] Lapouchnian, A., Yu, Y., Mylopoulos, J., "Requirements-Driven Design and Configuration Management of Business Processes", 5th International Conference on Business Process Management (BPM'07), Brisbane, September 2007.

[Liaskos05] Liaskos, S., Lapouchnian, A., Wang, Y. Yu, Y. and Easterbrook, S., "Configuring Common Personal Software: A Requirements-Driven Approach", 13th IEEE International Conference on Requirements Engineering, August 2005, Paris.

[Liaskos06] Liaskos, S., Lapouchnian, A., Yu, Y., Yu, E., Mylopoulos, J., "On Goal-Based Variability Acquisition and Analysis", 14th International IEEE Conference on Requirements Engineering, September 2006, Minneapolis/St. Paul.

[Morandini07a] M. Morandini, D. C. Nguyen, A. Perini, A. Siena and A. Susi, *Tool-supported Development with Tropos: the Conference Management System Case Study*, AOSE-07 Postproceedings LNCS 4951.

[Morandini07b] M. Morandini, L. Penserini, A. Perini, and A. Susi, *Refining Goal Models by Evaluating System Behaviour*, AOSE-07 Postproceedings LNCS 4951.

[Nguyen07a] Nguyen C. D., Perini A., Tonella P. "A Goal-Oriented Software Testing Methodology", AOSE-07 Postproceedings LNCS vol. 4951.

[Nguyen07b] Nguyen C. D., Perini A., Tonella P., "Automated Continuous Testing of Multi-Agent Systems", Fifth European Workshop on Multi-Agent Systems (EUMAS), December 2007.

[Nguyen07c] Nguyen C. D., Perini A., Tonella P., "Ontology-based Test Generation for Multi-Agent Systems", short paper at AAMAS 2008.

[Penserini07] Penserini, L., Perini, A., Susi, A., Mylopoulos, J., "High-Variability Design for Software Agents: Extending Tropos", *ACM Transactions on Autonomous and Adaptive Systems (TAAS) 2(4),* ACM Press, November 2007.

[Sebastiani04] Sebastiani, R., Giorgini, P., Mylopoulos, J., "Simple and Minimal-Cost Satisfiability for Goal Models", Sixteenth International Conference on Advanced Information Systems Engineering (CAiSE'04), Riga, Springer-Verlag LNCS 2003, June 2004.

[Siena07a] Siena, A., "Engineering Normative Requirements.", In Proceedings of the First International Conference on Research challenges in Information Science (RCIS). Ouarzazate, Morocco. 22-26 April, 2007.

[Siena07b] Siena, A., Maiden, N., Lockerbie, J., Karlsen, K., Perini, A., and Susi, A., "Exploring the Effectiveness of Normative i* Modelling: Results from a Case Study on Food Chain Traceability", Tech. Rep. FBK-IRST1207.