Extending *i** to Fit with the Requirements World

James Lockerbie & Neil Maiden

Centre for HCI Design, City University London, UK
[N.A.M.Maiden@, J.Lockerbie@soi.]city.ac.uk

Abstract. Whilst the *i** approach has been applied to case studies for some time, its wider uptake in industrial requirements processes and projects necessitates integration with established methods and techniques. However, there has been little reported integration. This paper will report how *i** has been integrated into a wider scenario-based requirements process, and summarize industrial uses of *i** on a recent food traceability project. Each of these projects has necessitated process and tool extensions to *i** to enable its uptake and use. The paper will report these extensions.

1. Introduction

Whilst the *i** approach has been developed and applied to case studies for some time, its wider uptake in industrial requirements processes and projects necessitates integration with established methods such as the Rational Unified Process (RUP) and effective requirements techniques such as scenario walkthroughs. However, there has been little reported integration so far, and this lack of integration threatens future uptake and industry-based evaluation of *i** and its underlying concepts. In this short paper we summarize previous and current research to integrate the *i** approach with other reported requirements methods and techniques.

2. Objectives of the Research

The objectives of the reported research are to investigate and evaluate the integration of the *i** approach with established requirements methods and techniques. If these objectives are successfully met, the outcomes will include an agenda of future research and knowledge transfer for the wider uptake and effective use of *i** in requirements processes and projects. Three specific research questions to which we are currently seeking answers to achieve the objectives are:

Q1: Can the strengths of the *i** approach in large-scale requirements projects relative to other requirements techniques be identified?

Q2: Can the strengths of the *i** approach in large-scale requirements projects be harnessed in established requirements methods to deliver quantitative and qualitative benefits to these projects?

Q3: Can we develop new software-based tools and techniques with which to use the *i** approach successfully in large-scale requirements projects?

Scientific contributions that seek to provide answers to these 3 questions are summarized in the next section.

3. Scientific Contributions

We have sought to answer the 3 research questions by developing and evaluating new requirements methods, techniques and tools that exploit the *i** approach.

The RESCUE process is a concurrent engineering process in which different modeling and analysis processes, including use of *i*,* take place in parallel. Each stream has a unique and specific purpose in the specification of a socio-technical system:

1. Human activity modelling provides an understanding of how people work, in order to baseline possible changes to it;
2. System modelling enables the team to model the future system boundaries, actor dependencies and most important system goals;
3. Use case modelling and scenario-driven walkthroughs enable the team to communicate more effectively with stakeholders and acquire complete, precise and testable requirements from them;
4. Managing requirements enables the team to handle the outcomes of the other 3 streams effectively as well as impose quality checks on all aspects of the requirements document.

The RESCUE process was reported at length in [1]. It is supported with an *i** modelling tool called REDEPEND, which is designed to provide systems engineers with *i** modelling and analysis functions, coupled with additional functionality and reliability of Microsoft Visio. It provides a graphical palette from which systems engineers can drag-and-drop *i** concepts to develop Strategic Dependency (SD) and Rationale (SR) models. REDEPEND also provides systems engineers with simple model checking functions for SD and SR models. It implements modelling constraints that, if activated, forbid a user to add or change a model element that violates *i** model constraints. Usability has been enhanced by, for example, adding new check features to highlight and shade-out model elements using layers, to partition and mark up models during analysis and review tasks, and to support *i** model colour-coding, which highlights model features during walkthroughs. Most of these features emerged from feedback on REDEPEND use in large-scale requirements projects.

In contrast, new productivity features were added to REDEPEND as results of academic research. For example we researched simple patterns – recurring syntactic and semantic structures in the *i** models – that can be applied automatically to any SD model expressed in REDEPEND to generate textual requirement statements. Our patterns are not traditional in the design sense – a solution to a problem in context. Rather each pattern defines one or more desired properties (requirements) on the future system that must be satisfied for the SD model dependency to hold for the future system. As such, the SD model, which has been signed off as complete and correct, informs further discovery and specification of requirements statements. The concepts and patterns underlying this approach are described at length in [2] and an application of the approach is reported in [3].

Research undertaken with NATS, the UK's air traffic service, resulted in a new version of REDEPEND to support the specification of satisfaction arguments [4] for *i** means-end links and the procedure to analyse the impact of software requirements on system-wide goals and soft goals. The procedures extend existing *i** model propagation rules [5] with domain knowledge imported through the satisfaction arguments, thus enabling effective use of domain assumptions in such propagation techniques for the first time. These procedures also addressed a pressing industrial need in NATS by providing techniques to relate system-wide safety-related goals to functional requirements of new software systems. Further details are in [6]

We have been applying RESCUE and REDEPEND on large-scale requirements projects including air traffic management projects reported elsewhere. The most recent application of *i** and REDEPEND has been on the TRACEBACK project. Assuring the total traceability of food and feed along the whole chain from production to consumption is a cornerstone of EU policy on the quality and safety of food. This is a complex procedure involving identification, detection and processing of a vast amount of information. TRACEBACK is developing innovative solutions based on micro-devices and innovative service-based architectures to provide innovative new information services to actors from primary food producers to consumers and health authorities. Solutions, which will include new micro-devices and a service-oriented reference architecture for food traceability called RATIS, are to be trialled on two major product chains – feed/dairy and tomatoes.

During the application of the RESCUE process a team of 3 analysts, all experienced with *i** and REDEPEND, produced *i** SD and SR models describing actors in the diary food chain, and the introduction of RATIS and micro-devices into this food chain. The models were developed using information from descriptions of current processes and workflows in the dairy food chains in Europe, one-on-one interviews with stakeholders who fulfil modelled actor roles in these food chains, *i** modelling workshops at project partner sites, and electronic distribution of SD and SR models to stakeholders for comment and feedback. Overall the process lasted 6 months. Key results are reported in *i** models.

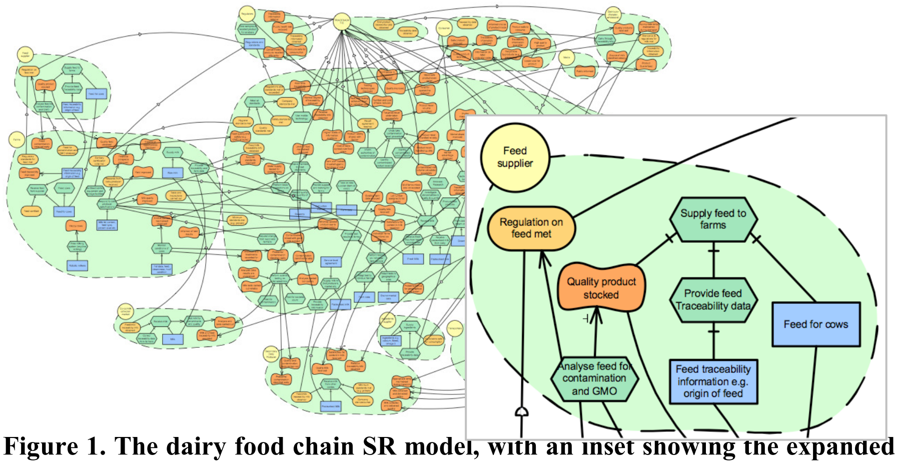
Feed supplier
Regulation on feed met
Supply feed to farms
Quality product stocked
Provide feed Traceability data
Feed for cows
Analyse feed for contamination and GMO
Feed traceability information e.g. origin of feed

Figure 1. The dairy food chain SR model, with an inset showing the expanded food supplier actor

The SR model for the dairy food chain is depicted in Figure 1. The model specifies 14 actors, 251 different process elements and 257 different associations between these elements. The inset demonstrates part of the model – the *Feed supplier* actor – in a readable form. Using the requirements generation functionality of REDEPEND, we generated a set of requirements prompts directly from the SR model which we reviewed and refined. These prompts were integrated into the use case modelling and scenario-driven walkthrough phase of the RESCUE process. For example, the requirement prompts were used to create additional “what if” questions for the ARTSCENE scenario walkthroughs [7] and were also used by the facilitators as background reference material to aid the walkthrough facilitation.

4. Conclusions

The research is not complete, and we still need to refine the use and hence effectiveness of REDEPEND features including pattern-based requirements generation and refining *i** means-end links with satisfaction arguments. Another ongoing research challenge is to understand the trade-off between the simplicity and usability of the *i** notation, to understand the number and types of *i** modelling elements that requirements analysts can model and analyse effectively on requirements projects.

5. Future Research

Future research will continue to seek answers to the 3 research questions, in particular by trying to developing RESCUE and REDEPEND for effective use in large-scale requirements projects. If successful we will make both available to new exponents of *i** for use in their own requirements projects.

References

1. Maiden N.A.M.,, Jones S.V., Manning S., Greenwood J. & Renou L., 2004, ‘Model-Driven Requirements Engineering: Synchronising Models in an Air Traffic Management Case Study’, Proceedings CaiSE’2004, Springer-Verlag LNCS 3084, 368-383
2. Maiden N.A.M., Manning S., Jones S. & Greenwood J., 2005, ‘Generating Requirements from Systems Models using Patterns: A Case Study’, Requirements Engineering Journal 10(4), 276-288.
3. Ncube C., Lockerbie J. & Maiden N.A.M., 2007, ‘Automatically Generating Requirements from i* Models: A Case Study with a Complex Airport Operations System’, Proceedings 13th International Working Conference, REFSQ’2007, Trondheim Norway, Springer-Verlag Lecture Notes on Computer Science LNCS 4542, 33-47.
4. Hammond J., Rawlings R. & Hall A., 2001, ‘Will It Work?’, Proceedings 5th IEEE International Symposium on Requirements Engineering, IEEE Computer Society, 102-109.
5. Horkoff J. Yu E. & Lui L., 2006’, ‘Analysing Trust in Technology Strategies’, Proceedings Privacy, Security, Trust Conference, Toronto, Canada, October 2006.
6. Maiden N.A.M., Lockerbie J., Randall D., Jones S. & Bush D., ‘Using Satisfaction Arguments to Enhance i* Modelling of an Air Traffic Management System’, Proceedings 15th IEEE International Conference on Requirements Engineering, IEEE Computer Society Press, 49-52.
7. Mavin A. & Maiden N.A.M., 2003, ‘Determining Socio-Technical Systems Requirements: Experiences with Generating and Walking Through Scenarios’, Proceedings 11th International Conference on Requirements Engineering, IEEE Computer Society Press, 213-222