Towards a Universal Requirements Engineering Process

Mohammed Odeh

Software Engineering Research Group,
Centre for Complex Cooperative Systems,
Bristol Institute of Technology,
University of West of England,
Bristol BS16 1QY,
UK
Mohammed.Odeh@uwe.ac.uk

Abstract

This paper introduces new a vision for a universal requirements engineering process, namely the TUREP process model with emphasis on the engineering dimension, but as well being business process driven, architecture centric, role driven, and relating to the activities of quality management, configuration management, system and software cost estimation, project management, and thus integrating with the overall software development process. In addition to other research projects that have been contributing to the evolution of the TUREP, this model is being evolved to establish a universal and integrated framework for the process of requirements engineering.

Keywords: requirements engineering, TUREP, architectural framework, business process modelling, requirements elicitation, analysis, specification, validation, cost estimation, project management, configuration management, quality management.

1 Introduction

One of the major challenges that continue to face the field of requirements engineering (RE) is the proper positioning and enactment of an associated requirements engineering process as part of the organisation's software or systems development life cycle model. In addition, a number of research attempts have been cited in the literature to bridge the gap between business processes (and associated models) and systems, for example, BPMSOA [9] as a generalised framework to bridge the gap between business process models and service oriented architectures and also the attempt in [14-17] to translate business process models to system models such as use-case and class models. Furthermore, the observation has been that the current RE processes and in particular the generic ones (for example the waterfall and spiral [12] based models) have a significant gap between their activities and the developing practice in systems and software engineering. This, however, may explain the widening of the gap between RE processes (and practices) and their positioning within the systems and software engineering life cycle model adopted by in associated specific organization..
In this regard, this paper sets a vision for a universal requirements engineering process that can adopt and be in line with the state of the art practices and technologies in relation to developing software systems. TUREP (Towards a Universal Requirements Engineering Process) has been inspired for the past few years by author of this paper reflecting on the above set vision while being taught as part of the requirement engineering postgraduate module of the Masters progarmme in Software Engineering and PhD modules [21] and related projects as the evolving building blocks for the inception of this model as briefly mentioned in Section 5.

In this paper, Section 2 sets the scene for the TUREP model and the basic incentives behind its inspiration followed by Section 3, which explains this model in detail. In Section 4, the relationship between requirements engineering as an engineering discipline and the TUREP model is explored with a newly stated definition for requirements engineering. Finally, the paper is concluded in Section 5 along with further future research directions.

2 Why TUREP?

Following the current generic requirements engineering process model [12], the activities of eliciting and modeling the associated business processes of associated systems are cemented within the requirements elicitation activity. Thus, one needs to ask the following questions:

(1) Can the current RE processes address the evolving needs of organisations' business processes? In other words, are these RE processes business process driven? Are they business process centric?
(2) Do the current RE processes address business process understanding as a core activity? Do they adopt formally the activity of business process modelling?
(3) What will be the nature and the means for requirements validation in view of the evolving business challenges and the associated implications on the associated

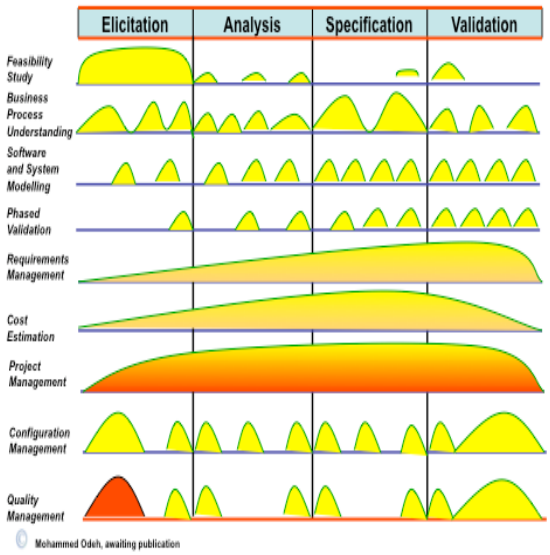

Figure 4: Towards a Universal Requirements Engineering Process (TUREP)

business processes? How will such validation be managed? Will this be mainly phased-validation before the over all requirements specifications are signed off?
(4) How can the current practice in requirements engineering be transparently linked to the activities in the associated software and systems development life cycle models?
(5) In what way(s) do these RE processes relate to the software and system design process? Can these be considered as software architecture centric?
(6) The need to develop and validate service-oriented software engineering process models is increasingly becoming a necessity given the wider scale adoption of the service-oriented model of computing [19]. Thus, this raises the question about service-oriented requirements engineering, which will be addressed in Section 5 and our current state of the art work in this regard.
(7) What about the interrelationships of the requirements engineering process with the activities of project, configuration, and quality management? Are these RE process related and how they can relate to the overall project, configuration and quality management of the systems and software engineering life cycle modes?
(8) Software cost estimation is usually performed as part of the overall software development process and at the very early initial stage in the process of software development. However, this raises the concern here to reflect on this being part of the RE process given the mutability of software requirements and the implications of the rate of change on the cost.
(9) Given the above questions/challenges, this increases the range of stakeholders involved in the RE process and hence this raises the concern regarding the readiness of a given RE process to identify and manage the involvement of such wider spectrum of stakeholders.

3 The TUREP Model:

3.1 The Static Organisation of TUREP

The TUREP model has been inspired by Rational Unified Process [9] in terms of its static structure and dynamic aspects to suggest TUREP's provision as a state of the art model addressing the evolving practices in the development of requirements specifications for software systems. TUREP is composed of four phases (as depicted in Figure 1) that progress over the life span of the activities related to requirements engineering as part of the system and software engineering life cycle model adopted in a particular organisation. These phases are: (1) Elicitation, (2) Analysis, (3) Specification, and (4) Validation. And, hence structurally TUREP is similar to the generic requirement engineering process.

3.2 The Dynamic Organisation of TUREP

The execution of work in these activities/phases is behviourally controlled and managed by nine workflows that progress across all the above phases. The particular depth of the work involved in each workflow is relatively shown as the area shaded under the curves in each of these workflows. Three of these workflows both overlap and are related to the particular and associated developments in the overall software and systems development life cycle model.

The Feasibility Study WorkFlow

Initially and before any further work can continue when enacting the TUREP model, the feasibility of carrying out a project needs to be established as whether the anticipated user requirements can be met given the nature of the requirements, technology availability and readiness, and constraints that can be budgetary, expertise, and schedule related, etc.

This will have been established early enough in the elicitation phase as part of the feasibility study workflow. A short feasibility study report will have been written with work to continue further if the project is feasible to carry out and approved by the project management board and/or key decision makers.

The Business Process Understanding WorkFlow

As TUREP is business process driven, business process understanding is a fundamental workflow, which builds on various approaches to understanding the organisation's business processes. Approaches may include visiting the Business Process Architecture (BPA) [7] of the organisation involved (if the BPA exists), domain knowledge development, conducting interviews, etc. This will in no doubt lead to identifying the concerned business processes, specify them (or improve on if they do exist), and the associated business process and user requirements.

Process and System Modelling

Based on the gathered business processes and elicited user requirements using different techniques, system requirements will be incrementally developed with emphasis on the functional and non-functional aspects of requirements in satisfying the overall business process.

Typically, a business process model will be modelled using any of the modelling notations such as RAD (Role Activity Diagramming, [17], RAD and UML Activity Diagrams [15], and BPMN [18] (Business Process Modelling Notation.

Some initial system models will start to appear as part of the work carried out in the software and system modeling workflow with some initial architectural models including for example use-case and conceptual models such as ER/class models, etc.

The Configuration Management WorkFlow

Although requirements may not be changing significantly in the very early stages, there needs to be standards and procedures to deal with managing changes to requirements and this is expected to be largely influenced by the overall activity of configuration management as part of the overall software and systems life cycle model being utilised. But, what about specific policies for requirements management such as the policies associated with assessing the impact of changes to requirements? Hence, this is addressed in the next workflow.

Requirements Management

When and where will the above polices be identified and applied, respectively? Such questions will have to be addressed quite early in the first phase of TUREP with standards and policies will established in addition to tool support including policies for requirements traceability from/to the RE process.

The Cost Estimation WorkFlow

One may ask the question as why is the need to concentrate on cost estimation while in the process of requirements engineering? This may also be the argument as the cost of software development is usually estimated ahead of investing further in a given software development project using various techniques including estimation by analogy based on existing systems, algorithmic ones such as COMOMO II suite of models [5], using use-

case models to estimate complexity with varying levels of detail [8], etc. However, these estimates are unlikely to be relatively accurate and to remain relatively accurate given for example: the increased changes to requirements, the increased agility of business processes, and the evolution of the service-oriented model of software development, etc.

For this workflow, the area under the curve may get higher when we proceed from the elicitation to the validation phase. This is anticipated to be the case as more information starts to appear to be utilised in updating the cost implications because of the emergence of such information that may be largely attributed to requirements changes. Hence, the total project cost become gradually more accurate.

The Quality Management WorkFlow

Following the TUREP model, the activity of quality management is related to ensuring that the activities of the requirements engineering process are carried out according to the pre-defined standards and procedures established for the overall software development life cycle. However, by having such a dedicated workflow running across all phases of the requirements engineering process necessitates that standards and RE process related quality aspects are adhered. This strengthens the conjecture that the approach adopted in TUREP will yield not only quality requirements, but as well enhance the overall quality of the developed product. Clearly, there is an overlap between TUREP and the overall software development project life cycle and hence this entails close coordination with the overall quality management team of the project.

The Phased Validation WorkFlow

And, finally is the phased validation workflow. To help understand what this workflow entails, let's first study the philosophy behind the division of work in the four phases of TUREP to help us understand what we need to validate against before exiting a particular phase.

It is apparent that progress throughout TUREP's phases proceeds as part of the work carried out in the respective workflows. But, is it possible to iterate over the workflows, how, and for how long? As the overall software development process is practically iterative and incremental, the RE process is no exception. Hence, having at least two iterations over the whole workflows within a phase is anticipated to be at least the case with more iterations to take place depending on the nature of the problem domain, the agility of requirements and associated business process, etc. Hence, the phased validation seems to be naturally appearing with iterations within a phase.

But, let's first test our understanding of how we iterate across the workflows as discussed in the next section.

4 Understanding TUREP's Philosophy

4.1 TUREP's Philosophy in Elicitation

In this phase, the requirements engineer is concerned with arriving at some satisfying answers to the following question:

"Have a good level of understanding of the organisation's business processes and associated user requirements been reached?"

Typically, one would like to conclude that these have been done with some relatively high degree of accuracy, that some good cost estimates have been made based on elicited requirements, that some initial business process model is in place, that some initial conceptual data models are emerging, that the requirements engineering team are aware of the overall project plans, and that policies and standards for requirements management have been agreed and followed in relation to both the key activities of configuration and quality management.

4.2 TUREP's Philosophy in Analysis

Moving across to the next phase after the elicitation phase, one would ask a number of questions regarding the completion of this phase:

"Are there any further feasibility issues emerging? Have all aspects of the business processes been captured and modelled? How can we validate or ascertain "all" aspects? Is there a need to improve these business processes? Have all requirements been

identified? How can we quantify or ascertain "all"? Have all functional requirements been examined in relation to stakeholders' needs? Have non-functional requirements been identified, and quantified (where possible)? Are there any possible implications of non-functional requirements on functional ones? Have trade-off measures been taken in adhering to non-functional requirements in line with requirements' priorities? Is there a need to do further requirements understanding and definition? Have conflicts between stakeholders been discovered and resolved? Have inconsistency between requirements been surveyed, identified and resolved, if any? Will any change to cost estimation due to requirements changes or further analysis of business processes and requirements be considered? Have all procedures and standards of configuration and quality management been adopted and adhered to? Is the work done so far in line with the schedule, cost, environment specific constraints, etc? Are any updates required to project plans? "

The above questions relate not only to the analysis aspects of the user requirements but all the associated implications of carrying out analysis of business processes and system requirements in order to comply with the key goals of the requirements analysis phase.

4.3 TUREP's Philosophy in Specification

Moving across to the next phase after analysis, one would ask:

Has the business process architecture and associated business processes been modelled and specified using the agreed the language or notation for business process modelling?

Have all identified functional and non-functional requirements been specified?

Has the system architecture been specified using some form of organisation specific architectural views? (e.g. 4+1 views of the system architecture [13])

Have the necessary system models been modelled along with relationships between business, structural and behavioural models.

Have all the functional and non-functional requirements been specified using the organisation specific standards and notations taking into consideration requirements for business and mission critical systems?

Have system level requirements been specified in preparation for the detailed design stage of the system and software development life cycle model being used?

Are there any necessary updates for the cost estimation document?

Have all requirements changes been specified following the standards and procedures of configuration and project management?

Have all updates in relation to project management aspects been documented with all aspects of planned and actual expenditure addressed?

The above issues relate to the specification aspects that are not only related to user requirements, but that are also associated with all other aspects of specifying business processes, system requirements, changes to requirements, project plans, etc? Are these in line with the key objectives of the specification phase?

4.4 TUREP's Philosophy in Validation

The next phase is the overall validation of the requirements engineering process, where in the simplest form phased validation is executed as discussed in this respective workflow. In addition, exiting this phase implies the end of the requirements engineering process with a sign-off of the requirements specifications document by the respective procurers. However, as a result of executing the tasks implied by the phased validation workflow, and in successive iterations in this phase, the key objectives of each of the above discussed workflows will have to be assessed as whether they have been satisfied.

5 Discussion

TUREP is anticipated to evolve as a universal requirements engineering process that can be easily adaptable to suite the orgnisation's specific software development processes in

relation to requirements engineering. Moreover, TUREP is considered to be in line with the engineering discipline in association with the activities of requirements elicitation, analysis, specifications, and validation. Despite that the requirements engineering literature does not seem to provide very close engineering based definition of these activities; and in this regard, I present below a definition for the field of requirements engineering with further reflection on TUREP's adherence to this definition:

"The use of domain knowledge and art to elicit, analyse, specify, validate, and manage end user and system requirements, with timeliness, economy and elegance, to generate a validated requirements document that is usable by all stakeholders involved in the process of systems development."

While composing this definition, I was very inspired not only by the more than two decades of experience in the field, but also by the very well-articulated definition of software engineering by Brown et. al. [4] in which they defined the field of software engineering as:

"The science and art of specifying, designing, implementing, and evolving, with economy, timeliness and elegance, programs, documentation and operating procedures whereby computers can be made useful to humanity."

Brown et al. were as well inspired by the definition of engineering by the Institute of Structural Engineers and their definition of engineering *"The application of science and mathematics to the design and construction of artefacts, which are useful to humanity"* [7].

Clearly (as implied by the above definition), the notion of engineering relates to the application of domain knowledge in the form of science, (2) an association with a process to carry out activities, (3) creativity, (4) project management, (5) the resulting product, and (6) the usefulness of the product to humanity. And, in particular the above newly stated definition of requirements engineering implies the use of ***"domain knowledge and art"*** with respect to business, scientific knowledge depending on the nature of the application strengthened by the experience and skills of stakeholders in related fields, tools, etc. Also, the phrase ***"to elicit, analyse, specify, validate, and manage"*** implies a set of structured activities with some ordering which collectively refer to a process. Furthermore, the phrase ***"with timeliness, economy"*** relates to project management, whereas ***"elegance"*** is associated with well-identified and specified requirements but the phrase "well .." is subjective to evaluate stemming from the above stated "elegance". The last key phrases shape the key goals of the requirements engineering process when achieved as ***"a validated requirements document"*** that is agreed by all stakeholders involved, and ***"involved inthe process of systems development"*** to imply the proper relationship and positioning of the RE process with the other phases or activities in the overall systems development lifecycle model.

In conclusion, the above segmentation of the newly stated definition of requirements engineering reflects on the adherence of TUREP to the engineering dimension by enacting a process-based approach supported by nine workflows that are iterated over in each phase of this process to achieve the respective objectives.

5 Conclusion

This paper has introduced a new vision to pave the grounds for a universal requirements engineering process through the introduction of the TUREP model with its key characteristics as adhering to the engineering dimension, being business-process driven, architecture centric, and role-driven, but relating to quality management, configuration management, system and software cost estimation, project management to achieve integration with the overall associated software engineering process model. In addition, this work is in line with other research projects that have been attempted to bridge the gap between business processes and system models and requirements specifications such as generating use-case models from business process models [15, 16], BPMSOA [9] (in bridging gap between business process models and service-oriented architectures), BPMOntoSOA [20] (semantic-based approach to generate service-oriented requirements specifications from business process models), OntoREM [10,11] (ontology or knowledge driven requirements engineering methodology applied in the aerospace sector), OntoRAT [3] (Ontology driven requirements analysis tool), generating conceptual data models from knowledge

models such as OWL-DL models [6], business process improvement through business process modeling applied in the healthcare sector [1,2], early anticipation of software cost estimation using varied levels of complexity of use-case models [9]. Therefore, these research projects are considered as part of the building blocks to develop an integrated but generalized framework with the TUREP model at the centre in attempting to evolve a universal integrated framework for requirements engineering.

REFEENCES

1. AbuelRub F, Odeh M and Beeson.I. (2007) Modelling Non-Functional Requirements of Business Processes. Journal of Information and Software Technology, Vol. 49(11-12), pp1073-1172, Elsevier Publishers, ISSN: 0950-5849.

2. AbuelRub F, Odeh M and Beeson.I. (2008) Modelling Healthcare Processes Using Role Activity Diagramming. International Journal of Modeling and Simulation, Acta Press, ISSN: 0228-6203.

3. Al-Hroub Y, Kossmann M, and Odeh. (2009). OntoRAT: Ontology-driven Requirements Analysis (2009) International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2009), London Metropolitan University,

4. Brown, A. and Earl, A. and J McDermid (1992) Software Engineering Environments, Automated Support for Software Engineering, McGraw-Hill.

5. Boehm, et al. Software Cost Estimation with COCOMO II (2000). Englewood Cliffs, NJ:Prentice-Hall. ISBN 0-13-026692-2.

6. El-Ghalayini H, Odeh M and McClatchey R (2007). Engineering Conceptual Data Models from Domain Ontology: A Critical Evaluation. Journal of Information Technology and Web Engineering, Vol2(1), Idea Group Publishers, ISSN: 1554-1045.

7. Green S, Beeson I, and Kamm R. (2007) Process Architectures and Process Models: Opportunities for Reuse, 8th International Workshop on Business Process Modelling, Development, and Support, Trondheim.

8. Issa A, Odeh, M and Coward, D (2006) Using Use-Case Patterns to Estimate Reusability in Software Systems. Journal of Information and Software Technology, Vol. 48(19), pp 836-845, Elsevier Publishers, ISSN: 0950-5849.

9. Khan Z A, Odeh M and McClatchey R. (2006) Digital Libraries: From Process Modelling to Grid-based Service Oriented Architecture. Proceedings the 2nd Int. Conference on Information & Communication Technologies from Theory to Application (ICTTA'06) ISBN: 0-7803-9522-0

10. Kossmann M, Odeh M, Gillies A, and Wong R. (2008) From Process Driven to Knowledge Driven Requirement Engineering using Domain Ontology (2008) Proceedings of the INCOSE International Symposium Amsterdam.

11. Kossmann M, Odeh M, Gillies A, and Watts S. (2009) Ontology-driven Requirements Engineering with Reference to the Aerospace Industry. Proceedings of the 2nd International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2009), London Metropolitan University,

12. Kotonya G and Sommerville I. (2000) Requirements Engineering, Wiley.

13. Krutchen, P. (2000). Rational Unified Process, Addison-Wesley.

14. Odeh, M, Beeson, I, Green, S and Sa, J. (2002). Process Modelling Using RADs and UML Activity Diagrams: An Exploratory Study, The 3rd International Arab Conference on Information Technology - ACIT2002.

15. Odeh, M, Hauer T, McClatchey R and Solomonides T. (2003). A Use-Case Driven Approach in Requirements Engineering: The MammoGrid Project, Proceedings of the 2003 IASTED's Software Engineering Applications (SEA).

16. Odeh M, Kamm. R. (2003) Bridging the Gap between Business Processes and System Models. International Journal of Information and Software Technology, Vol. 45(15), pp.1053-1060, Elsevier Publishers, ISSN: 0950-5849.

17. Ould M A, (1995) Business Processes – modelling and analysis for re-engineering and improvement, John Wiley and Sons, Chichester.

18. Polančič G and Rozman T Business Process Modelling Notation (BPMN) Poster, http://bpmn.itposter.net. Accessed August 31, 2009 at 18:30h.

19. Yousef R, Odeh M, Coward D, and Al-Shariah A. (2009). BPAOntoSOA: A New Framework to Derving Software Service Oriented Models from Business Process Architectures. Proceedings of the 2nd International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2009), London Metropolitan University,

20. Yousef R, Odeh M, Coward D, and Al-Shariah A. (2009). Mapping RAD-based Business Process Models to BPMN Models (2009) Proceedings of the 2nd International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2009), London Metropolitan University,

21. http://courses.uwe.ac.uk/g70012/2009, accessed August 31, 2009 15:12h.