Determinants of Requirements Process Improvement Success

Aminah Zawedde[1] and Ddembe Williams[2]
School of Computing and Informatics Technology, Makerere University, Uganda[1]
Faculty of Computing and Information Management, KCA University, Kenya[2]
sazawedde@gmail.com[1]; dwilliams@kca.ac.ke[2]

Abstract. Improving the requirements process improvement (RPI) of software projects has become an important area of research and professional practice. This paper highlights the inefficiencies in RPI that results in poor quality, and escalating cost or schedule. The efficiency with which the changing processes are managed determines how successful a project will be in terms of attaining a satisfactory balance between quality, cost and schedule of the delivered software systems. A number of software development companies suffer from ineffective RPI; therefore there is a need for understanding the underlying structure and explaining the determinants of the RPI success. This facilitates RPI stakeholders in taking informed decisions that would lead to more successful RPI due to improved understanding of the underlying structure and feedback that exists among the RPI success factors. This calls for continuous improvement of the requirements processes by analyzing the relationships and the dynamics that exist amongst the RPI factors for cost effective RPI decisions. The paper presents a system dynamics RPI model validated by practitioners and discusses the insights generated from the model. The authors suggest that the resulting model and the insights generated through sensitivity analysis tests constitute significant contributions towards understanding the factors that determine RPI success.

Keywords: cost, schedule, quality, requirements process improvement, system dynamics

1. Introduction

Requirements Process Improvement (RPI) is a systematic approach taken by RPI stakeholders to identify, analyze and improve the specification of a requirements specification, and associated activities, in order to improve the efficiency in terms of quality, reduce costs and delivery within a specified schedule (Solemon *et al*., 2009). The results of a successful RPI are measured through increased productivity, improved customer satisfaction, reduction in costs of obtaining the requirements and improved quality of the requirements specification. The aim of RPI is to eventually lead to improved processes for software development.

Various researchers have attested to the fact that inefficient RPI of software projects results into either scope, cost or schedule creep (Cooper *et al*., 2009; Ferraira *et al*., 2009). In many software development projects, the requirements together with the processes that manage these requirements are assumed to be static (Abdel-Hamid and Madnick, 1991) and yet in reality these processes are interlinked and dynamic with feedback effects on each other (Beecham *et al*., 2005; Williams, 2003a). Therefore the efficiency with which one manages the changing processes determines how successful a project will be in terms of having a satisfactory balance between quality, cost and schedule of the delivered requirements specification.

Companies do not effectively evaluate the impact of process improvement because they mainly focus on evaluating the costs and do not have a systematic way of evaluating process improvement performance (Hall *et al*., 2002). It has been pointed out that a number of software development companies suffer from ineffective RPI due to the lack of access to timely and accurate information, delays in communication of errors and excessive rework (Ferreira *et al*., 2009; Nasir and Sahibuddin, 2011). In Hall *et al*. (2002) and Anliang *et al*. (2006), it is emphasized that during the requirements engineering phase, the various RPI stakeholders must be able to communicate in order to have an opportunity to share best practices and have feedback on process improvement concerns that may arise.

In order to attain cost effective decisions for RPI, there is a need for understanding the underlying structure and explaining the feedback interrelationships that exist amongst the RPI variables. This helps RPI stakeholders in taking informed decisions for a successful RPI when choosing amongst various alternatives for process improvement. However, once the impact of the feedback effect between requirement processes is not precisely understood and explained, one cannot control effectively the cost, schedule and quality of the process. This therefore calls for continuous improvement of the requirements processes by analyzing the relationships and the dynamics that exist amongst the RPI variables (Pfhal and Ruhe, 2003) for cost effective RPI decisions.

The most commonly used methods for RPI are unable to capture the dynamics and the interrelationships that exist amongst RPI variables because they mainly follow a static approach that is hierarchical and does not capture the dynamics (Beecham *et al*., 2005; Sommerville and Ransom, 2005; Zawedde *et al*., 2011). These methods do not capture the interdependence and feedback that exists amongst the processes which makes it difficult for RPI stakeholders to gain a common understanding of the emerging behavior resulting from the dynamics that exists amongst the RPI variables (Ferreira *et al*., 2009; Zawedde *et al*., 2011). The use of such RPI methods, may explain why most software projects fail due to scope, cost or schedule creep (Williams, 2003a; Ferreira *et al*., 2009). In order to address these shortcomings, a dynamic process improvement model based on the dynamic synthesis methodology that integrates the system dynamics methodology with case studies (Williams, 2003b; Rwashana *et al*., 2009) is developed in this paper. Integration of these methods facilitates understanding of the extent of the impact of a change in one variable on the other variables thus giving an explanation for the causes of the impact. The resulting model can support informed decision making through generation of insights in the RPI process (Williams and Kennedy, 2000; Pfhal and Ruhe, 2003; Zawedde *et al*., 2011).

1.1 Background Information

System dynamics (SD) is an approach for modeling and simulation of dynamic behavior of complex systems over time (Forrester, 1991; Harris and Williams, 2005). The complexity of a system is defined by feedback loops, non-linearity and time delays that often affect the system behavior. System dynamics models being a representation of real world situations are well suited to offer explanation and generate insights into the root causes of the behavior of complex systems. The insights generated can facilitate informed decision making before any improvements can be implemented (Clempner, 2010).

RPI is characterized as a complex system due to the interactions that exist amongst the processes and the dynamic behavior that results from the interactions (Sterman, 2000). Usually, the expected output of the process improvements differs from the desired outcomes even after RPI stakeholders have implemented realistic decisions that are based on process improvement goals because of the interactions that exist amongst the processes (Berard, 2010). This makes the decision making process of the RPI stakeholders difficult because it is affected by complex system structures and limitations on their cognitive skills (Berard, 2010). In this context, modeling approaches are necessary for making RPI issues better analyzed and understood (Williams 2003b; Berard, 2010). System dynamics is a suitable modelingand problem solving approach for RPI because the defined characteristics of problems addressed by the SD approach match the characteristics of RPI.

The rest of the paper is structured as follows: Section 2 provides an overview of the behavior of the individual RPI variables over time; Section 3 is a discussion of the development of the system dynamics based RPI model; Section 4 explains the sensitivity analysis test for validation of the extent to which a change in the RPI variables impacts of the behavior of the RPI model; Section 5 discusses limitations of the study; and Section 6 discusses further research directions for enhancing RPI.

2. Behavior of Key RPI Factors

The key factors for RPI success (Zawedde *et al*., 2011) include:

a. Productivity of requirements engineers: the rate at which requirements engineers process errors in requirements specifications as agreed upon by the RPI stakeholders.

b. Process capability index: the potential of a process to meet its specifications given available technology support (Sommerville and Ransom, 2005).

c. Management commitment: the continuous support and involvement of executive strategic management based on acknowledged benefits in the implementation and maintenance of a system under development.

d. Process improvement cost: the total cost of resources in terms of wages, documentation, development, training technology and initial set-up costs to undertake process improvement.

e. Customer satisfaction: the degree to which customers expectations of a product or a service are met or surpassed.

f. Process rigor: the level of thoroughness that a process adheres to established standards when effecting or implementing process improvements.

g. Errors observed: defects identified by the requirements engineering stakeholders during the review process or during maintenance.

These variables are a mixture of hard (precisely measured) and soft (quantified but not precisely measured) variables (Rainer and Hall, 2002; Zawedde *et al*., 2011). Following the system dynamics based modeling approach described in Section 1.1; behavior over time (BOT) graphs were generated as graphical representations of mental models that fit the quantitative behavior of the variables as experienced in practice (Richardson and Lyneis, 1998; Williams, 2003b).

The behavior of the variables displayed by the BOT graphs is used as a basis for comparison with the resulting simulation model behavior presented in Section 5 which enables the researcher to create a level of confidence in the developed RPI model (Richardson and Lyneis, 1998). On the horizontal axis of the BOT graphs is the "Time" variable in weeks plotted against the vertical axis, the "behavior" (process performance index) of the RPI variables changes over time (Richardson and Lyneis, 1998). RPI experts emphasized the importance of setting the time as a clear scope for process improvement so that people are motivated to work within a specified period. Figures 1 and 2 illustrate the BOT of the seven (7) key RPI variables.

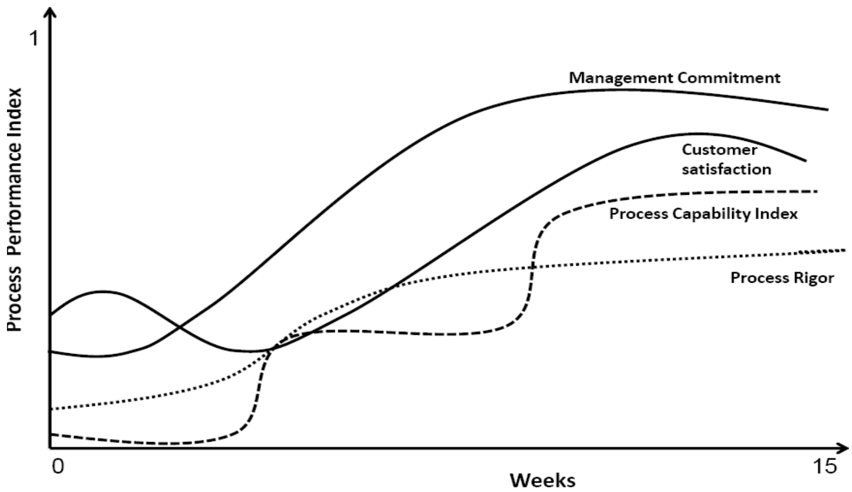

Fig. 1: Reference Modes for management commitment, process capability index, process rigor and customer satisfaction

Figure 1 shows a mixture of growth and declines in management commitment, process capability index, process rigor and customer satisfaction. These variables are dimensionless have their scale as unitless ranging from 0 to 1 on the process performance axis. Management commitment, process capability index, process rigor and customer satisfaction are among the factors that have a major impact on process improvement as examined by (Rainer and Hall, 2002). Increasing levels of customer satisfaction are attributed to higher management commitment and process rigor.

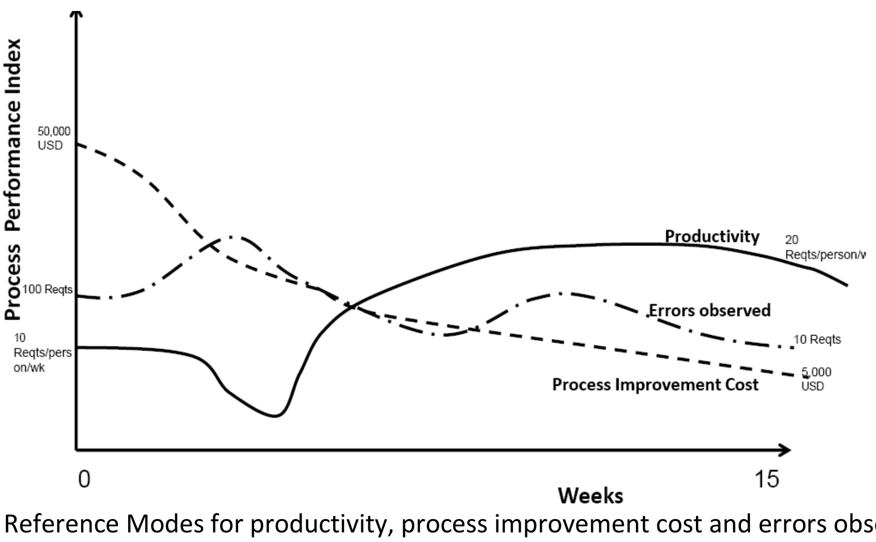

Fig. 2: Reference Modes for productivity, process improvement cost and errors observed

Figure 2 indicates a combination of variations and trends in process improvement costs, errors observed and productivity over a period of time. The productivity of the requirements engineers first declines at the beginning of the process improvement project because of a number of factors involved like training and acquiring of skills but it later goes up after the skills have been attained and the engineers are now familiar with the improvement process. This effect translates into lower errors observed and reduced process improvement costs. The variations in the behavior of the variables overtime presented in Figure 1 and 2 form a basis for effective understanding by RPI stakeholders through exploring the dynamics that exists amongst these variables (Zawedde *et al*., 2011).

The graphs provide insight into the underlying dynamics that exists in RPI. The interactions amongst the key RPI variables are described in (Zawedde *et al*., 2011). It is the interaction among these variables that is responsible for the emerging behavior of variables over time as a result of the improvement process.

3. Dynamics of RPI Factors

Validation of the relationships that exist amongst the RPI variables was done by practicing RPI experts in an iterative way using the Delphi method until a consensus was reached. The criteria for the selection of the RPI experts were based on (Colton and Hatcher, 2004). 15 experts were identified to participate in the validation process of the descriptive model for RPI of which 6 participated up to the end of the process. Some of the experts were identified with the help of the researcher's supervisors.

Messages of introduction were sent via LinkedIn to 15 experts who were identified as potential for the study. The message included the purpose of the invitation which was to participate in the study and a request for the participants email for further communication about the study. 8 positive responses were obtained, which made the response rate 53%. This is considered to be good for email responses (Day and Bobeva, 2005). Among the respondents, there were 2 requirements process improvement expert, 3 project managers, and 3 quality assurance managers. All the respondents had over 10 years working experience in the software development domain. To maintain professional integrity the identity of the experts and their respective organizations is not disclosed in this thesis due to non disclosure agreements.

The structure of the RPI model (conceptual model) was developed to enhance understanding of RPI's interrelated components. This was done by constructing causal loop diagrams (CLDs) for each one of the six RPI model sectors and then all the CLDs were merged into one complete RPI model structure. The CLDs qualitatively describe the structure of the RPI model showing the interrelationships and feedback that exist amongst the variables of RPI. This improved our understanding of the dynamics involved amongst RPI variables. In this section, for purposes of brevity we discuss one out of the six CLDs that were constructed.

3.1 CLD for the Productivity of Requirements Engineers Sector

Figure 3 presents the variables that make up the model structure of the productivity of requirements engineers sector. The variables and loops demonstrate the relationships and feedback that exists between the requirements engineer workforce and the effect of attaining skills on productivity. There are two balancing loops B1 and B2 in this sector as presented in Figure 3.

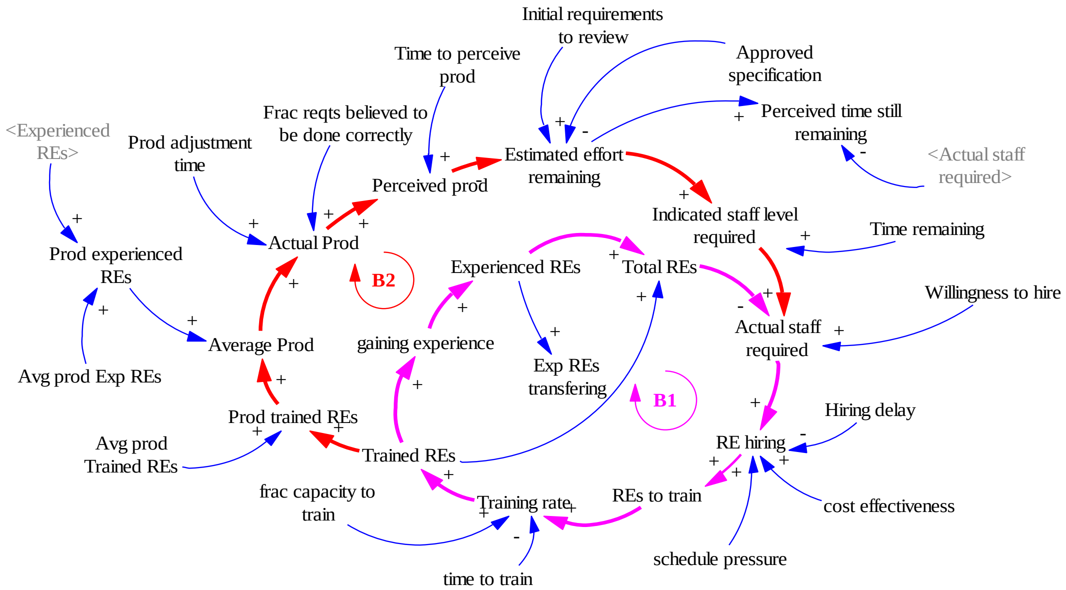

Fig. 3: The causal loop diagram for the productivity of REs sector

In (Zawedde and Williams, 2013), loop B1 demonstrates the requirements engineer workforce management on a RPI project. An increase in the actual number of staff level required to carry out RPI is triggered by the increase in willingness to hire and the indicated staff level required on the project. The increase in the staff level required increase the rate at which requirements engineers (REs) are hired. An increase in the number of requirements engineers that are hired on a project increases the number of requirements engineers training on the project for a certain period which in turn increases the number of trained engineers, hence the delay between RE hiring, RE training and trained REs. Increasing the trained engineers results into an increase of the number of experienced engineers which in turn increases the total number of requirements engineers. However, the increased number of experienced engineers is attained after a delay in gaining experience on the project. An increase in the total number of requirements engineers will feedback into a reduction in the number of staff level required on the project. The importance of having experienced engineers on process improvement projects is emphasized by (Hall *et al.*, 2002) who state that "*experienced people in process improvements are critically important for its success*", because process improvements are mostly successfully implemented by them.

On the other hand, an increase in the number of trained REs increases the total productivity of the trained Res in loop B2. This increases the average productivity of both the trained and experienced REs on the project resulting into increased actual productivity. An increase in actual productivity and the fraction of requirements believed to be done correctly increases the levels of work accomplished and as a result, there is a decrease in the rate at which errors are reworked in the rework and effort management sector.

The rest of the variables in the sector that are not part of the loops have an indirect influence on the variables within the loops. Their importance in the sector can therefore not be over looked. Coming up with the CLD of the productivity of REs sector together with all the CLDs of the various sectors was a way of capturing the metal models of the RPI experts. The CLDs served as a communication and unifying medium between the researchers and the RPI experts. This helped to clarify our knowledge and understanding of RPI. We also discovered ways in which the behavior of RPI can be improved through analyzing the loops and how they relate with each other. These benefits we attained are supported by (Albin, 1997).

3.2 Stock and Flow Diagram for the Productivity of REs Sector

In this section we develop a quantitative model using stock and flow diagrams (SFDs) based on the qualitative CLD. The SFDs contain mathematical equations that describe the relationships amongst the RPI variables as discussed in Section 3.1. Simulation runs of the mathematical model are made to generate insights into the dynamics of RPI and to determine the most influential variables for RPI by carrying out sensitivity analysis tests.

Productivity of Engineers Sector shown in Figure 4 is the human resource component of the project (Abdel-Hamid and Madnick, 1991). It shows the structure of integrating the requirements engineer work force with productivity. The stock and flow structure illustrated is transformed from the CLD for productivity of engineers in Figure 3. The human resource management of the requirements engineers is illustrated through a chain of stocks and flows that represent hiring the engineers, training them until they gain experience. The influence of this chain on productivity is illustrated through converters that are interrelated.

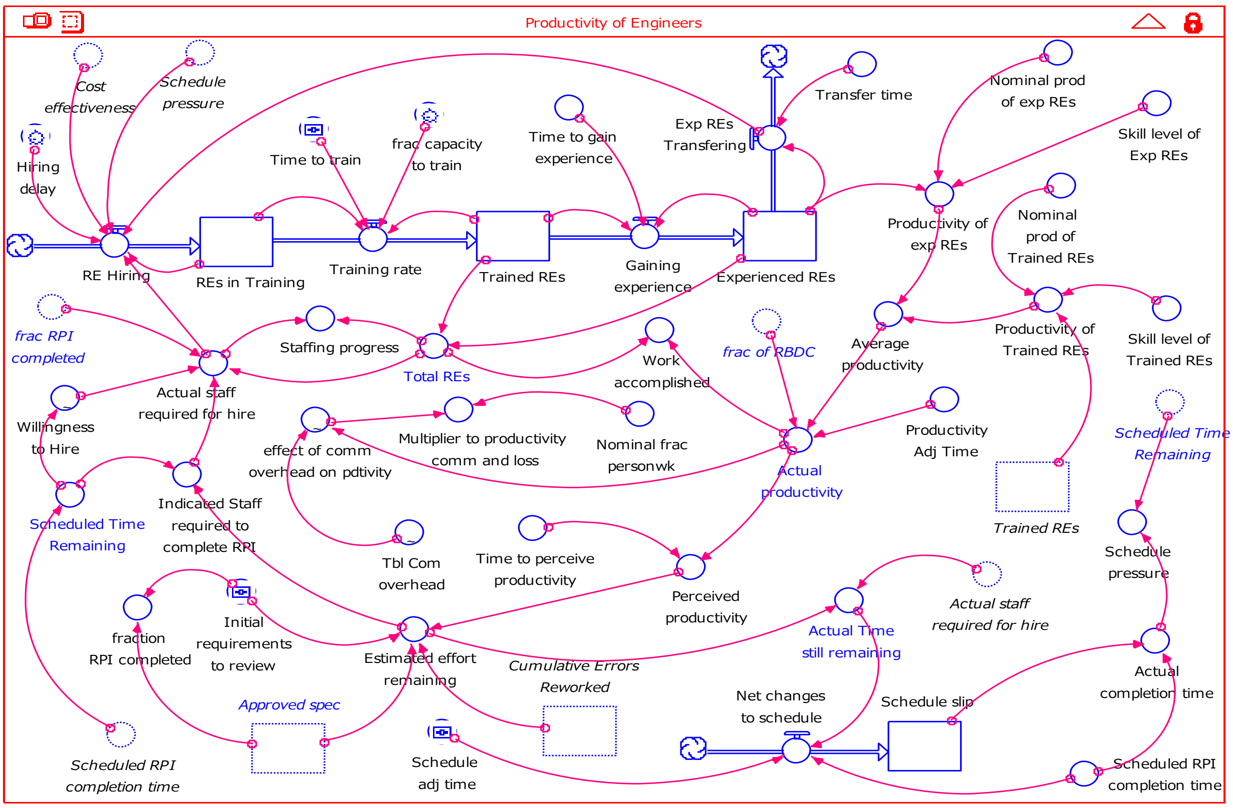

Fig. 4: The productivity of requirements engineers sector

In Figure 4, productivity depends heavily on labor (requirements engineers) and on the firm's work week. The work week and productivity sometimes themselves can be endogenous variables, dependent on factors such as schedule pressure, engineers experience and technical skills level. In this model, the work week and productivity represent averages. Sterman (2006) suggests that some workers are more productive than others or some put in more hours than others (Sterman, 2000). In this model, new engineers when recruited receive mentoring and on the job training often provided by experienced engineers and a few tool use short courses.

In this sector RE hiring, which is the weekly rate at which requirements engineers are recruited on the project, is driven by the difference between the actual staff needed to complete the project and the requirements engineers who are to undertake training (Richardson and Pugh, 1981). RE hiring is also determined by increased cost effectiveness, schedule pressure, and the number of requirements engineers who are transferred to work on other projects. RE hiring is done over a time period referred to as the hiring delay in this model. We assume that each process improvement project is an individual project, and therefore the engineers hired on the new project undergo training to obtain knowledge and ideas, and to meet the requirements of the on-going project. The newly recruited requirements engineers undergo training for a specified duration referred to as time to train; however, the number of engineers to be trained depends on the fraction of the capacity of engineers that can be trained at a time. The trained engineers gain experience and become experienced engineers by working on the project over a period "time to gain experience."
When the number of experienced engineers on the project increases, some of them are transferred via an outflow on the Experienced REs stock. The assumption made is that the requirements engineers do not resign but are transferred to other projects within the organization after process improvement is accomplished. The level of productivity of the engineers on the project is determined by the total number of the trained and experienced engineers on the project, the skill levels, and nominal productivity of the respective categories of engineers. The experienced requirements engineers are a key driver of RPI success in this sector. The higher number of experienced engineers that there are on a project implies that there will be increased productivity (Rainer and Hall, 2002). Based on the level of productivity at the current time on the project and the comparison between the approved specification and the initial requirements document, the estimated effort remaining to complete the project is computed. The estimated effort remaining helps us to determine the engineers required to complete the project. However, the engineers that are actually needed depends on two aspects namely: the willingness by management to hire new engineers based on the time remaining, and the total number of engineers on the project at the time given the fraction of RPI that has been completed so far.

Some of the critical equations of the described variables that show how they mathematically relate to other variables in the productivity of requirements engineers sector are illustrated.

i. The equation for RE hiring whose measure is {person/wk} is derived as follows:

RE_hiring (t) = {(A_SRFH - R_IT) * E_RT * C_effect * S_pressure} / {H_delay}

Where:
A_SRFH is the actual staff required for hire.
R_IT are the REs in training.
E_RT are the experienced REs transferring.
C_effect is the cost effectiveness.
S_pressure is the schedule pressure.
H_delay is the hiring delay.

RE hiring is the weekly rate at which requirements engineers are recruited on the project. RE hiring is driven by the difference between the actual staff needed to complete the project and the requirements engineers who are undergoing training (Richardson and Pugh, 1981). RE hiring is also triggered by the increase in the number of requirements engineers who are transferred to work on other projects, increased cost effectiveness, and increased schedule pressure.

ii. Actual productivity (Actual_prod) whose unit of measure is {Requirements/person/wk}, has its equation derived as follows:

Actual_prod = SMTH1(Avg_prod * frac_RBDC, Prod_AT)

Where:
SMTH1 is a smooth function; smooth(x, y) that takes on the following variables:
x= Avg_prod * frac_RBDC
Avg_prod is the average productivity.
frac_RBDC is the fraction of requirements believed to be done correctly.
y = Prod_AT is the productivity adjustment time which is the time it takes the requirements engineers to have adjustments made in the actual productivity.

Actual productivity is the average productivity of the requirements engineers given the fraction of requirements believed to be done correctly. The higher the fraction of requirements believed to be done correctly, the higher the actual productivity will be.

iii. Estimated effort remaining (Effort_rm) has its unit of measure as {person-wk}. The equation for this variable is:

Effort_rm = {Reqts_review - App_spec} / {perceived_prod}

Where:
Reqts_review are the total number of requirements in the initial requirements specification document that are to be reviewed.
App_spec are the cumulative errors reworked.
perceived_prod is perceived productivity of the requirements engineers.

Estimated effort remaining helps us to plan for adjustments in the total number of requirements engineers required to complete the process improvement project given the perceived productivity of the engineers on the project. The successful process improvement project has the trend of a continuously declining estimated effort remaining or almost stable estimated effort throughout the duration of the process improvement (Royce, 1998).

This trend results from increases in the number of approved requirements over the period of the process improvement project.

3.3 RPI Model Behavior

The RPI model was gradually developed in consultation with the RPI stakeholders that included requirements engineers, process improvement experts, quality assurance managers, project managers and customers (Zawedde *et al.*, 2011). Stakeholders were presented with graphical and tabular output from the RPI model which they checked for correctness and insights generated by the model at agreed upon meetings every after two months. This was done to ensure that the researcher and the RPI stakeholders understood how the model structure relates to its behavior. In this section we present the base case model that illustrates the initial behavior of the model. We compared this behavior with that of the BOT graphs for the key process improvement variables in Section 2. Figure 5 is an illustration of the behavior of the RPI model that reflects the behavior of 5 out of the 7 key variables of RPI as observed from literature and the field studies.

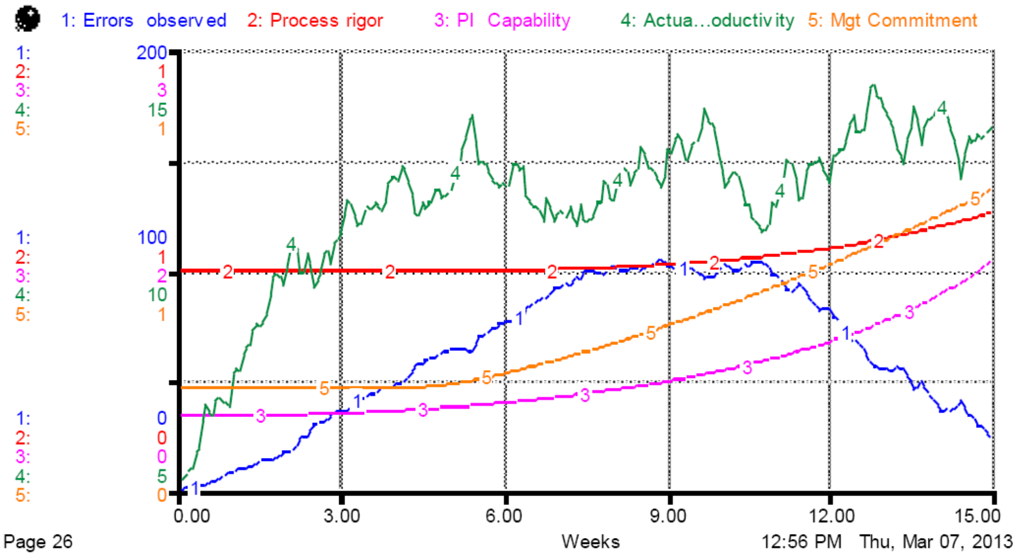

Fig. 5: Initial output of the simulation model

The behavior of the RPI variables displayed in Figure 5 are errors observed (1), process rigor (2), process improvement capability (3), productivity (4), and management commitment (5). On the horizontal axis we have the simulation time in weeks and on the vertical axis we have the scale marked 1 to 5 for each one of the 5 key variables.

Process rigor (2) maintains a constant level from week 0 to week 9 because during this period, the newly recruited requirements engineers on the project who are undergoing training maintain the current standards in the organization. This results into a gradual increase of the errors observed (1) as the trained engineers have the zeal to review the requirements specification using the newly acquired skills. Upon completion of training, new organizational standards are set to match the current industrial practices hence increasing the level of process rigor from week 9 to week 10. Errors observed maintain the same level during this period since the engineers are adopting the newly set standards. Beyond week 10, the engineers have gained the experience to adhere to the new standards and to easily identify errors. This results into further increases the level of process rigor and a continuous decline in the errors observed.

Actual productivity of the engineers (4) slowly increases while the level of management commitment and process improvement capability (3) remain constant from week 0 to week 3. This behavior is attributed to the fact that during this period, management is determining the value of return on investment for this project and training engineers is one of its priorities as a means to increasing the level of productivity. The level of process improvement capability is constant because during this period, the organizational processes are not streamlined and therefore inefficient. After week 3, the engineers have completed training and therefore their productivity increases with the new skills acquired and management now foresees the potential of high returns on investment in this project. Management invests more time and resources into this project which increases the levels of management commitment and actual productivity of the engineers. With more investment of resources into the project, the processes become more streamlined and efficient which increases the level of process improvement capability.

When we compare the behavior of this to the BOTGs in Section 2 you find that much as the behavior of the graphs of the BOT graphs does not exactly compare with the behavior of the RPI model, there is a similarity in the trend of the behavior of the variables to a certain extent. The difference in the behavior is attributed to the fact that the BOTGs are captured for individual variables, whereas the simulation results capture the holistic behavior of all the variables in relation to each other over time. The feedback and time delays amongst the variables result in the difference in behavior over time. We go ahead to test the validity and stability of the developed RPI model in order to determine which variables affect the behavior of the key RPI variables (Hekimoglu and Barlas, 2010).

4 Sensitivity Analysis

A sensitivity analysis test is a validation test used to assess the impact of a change in a parameter on the model behavior (Maani, 2000; Hekimoglu and Barlas, 2010). The test also helps to determine the key variables that drive the model's results. The sensitivity analysis test carried out for this research was aimed at minimizing the changes amongst the cost, quality and schedule as specified in Section 1. The outputs from sensitivity analysis enabled the researchers to come up with various scenarios that increased the level of confidence in the model (Hekimoglu and Barlas, 2010).

In this paper, the sensitivity analysis was done by varying the RPI model input parameters by plus or minus 10% (Maani, 2000; Sterman, 2000) and examining the impact of these changes on the model output results. In order to achieve 10% decrease and increase, each value in the model was multiplied by 0.9 to attain the 10% decrease and 1.1 to attain the 10% increase (Williams, 2003; Rwashana *et al.*, 2009). The variables that affect the RPI model behavior significantly when changed were identified and the justification of the behavioral changes was analyzed in the context of the literature and field findings. A summary of the sensitivity analysis tests and their impact on RPI performance due to the changes in the input parameters of the RPI model is shown in Table 1. The key used to indicate how sensitive the key RPI variables were to the 10% decrease and 10% increase of the input parameters or graphical relationships was adapted from (Maani, 2000).

- * = Sensitive (5%-14%)
- ** = Very sensitive (15%-34%)
- *** = Highly sensitive (35% and above)

An analysis of the sensitivity tests reveals that among the key RPI variables, errors observed and process rigor were highly sensitive to changes in almost all the input parameters. In Table 1 the blank cells indicate that the variables were not sensitive to the 10% increase or decrease in the input parameters of the RPI model. An analysis of the sensitivity test results, indicates that process improvement costs and management commitment are not sensitive to any changes in the values of the selected input RPI model variables. Based on these results, we may conclude that process improvement costs and management commitment are not important drivers for the success of RPI as was established from the literature and the field studies that we carried out. The behavior of customer satisfaction is driven by the acceptable quality objective, the table of experience on rigor and the table of schedule adjustment time. Customer satisfaction is only sensitive to decreases made to the acceptable quality objective, implying that customers react to declines in the acceptable quality objective. Customer satisfaction is also only sensitive to increases in the table of experience on rigor and the table of schedule adjustment time. This implies that customers react to the technical rigor in the specification and completion of the project within the scheduled time.

Table 1: Summary of Sensitivity Analysis Tests

	Selected Results of the 10% Decrease and Increase in Sensitivity Tests on Key Variables													
	PI Costs		Productivity		Errors Observed		Process rigor		PI Capability		Cust Satisfaction		Mgt Comitment	
Base Value	13734 {USD}		14 {Rqts/p/wk}		32 {Reqts}		0.81 {unitless}		2 {unitless}		0.43 {unitless}		0.79 {unitless}	
	-10%	+10%	-10%	+10%	-10%	+10%	-10%	+10%	-10%	+10%	-10%	+10%	-10%	+10%
Requirements to review			*	*	**	**	*	*	*	*				
Average review time			*	**	**	**	*	*	*	*				
Time to train			*	*	*			*	*	***				
Frac capacity to train			*	*	**	***			*					
Delay in tech progress			**	**	***	***	**		**					
Acceptable qlty objective			*	*	**	*	*	*	*	*	*			
Time mgt to respond to qlty			*		***	*	*	**	**	**				
Delay customer skills			*		***	**		*	*	*				
Table experience on rigor			*	**	**	**	*	**	*	**		*		
Table schedule adj time				*	**	**	**	**	**	**		*		

Productivity is not sensitive to decreases in the parameters of the table of schedule adjustment time; increases in the time for management to respond to quality; and increases in the time to acquire skills for customer product value addition. However, productivity is sensitive to all the other changes made to all the input variables of the developed RPI model. It is particularly very sensitive to changes made in the delay in gaining technological progress; and increases in both the average review time and table of experience on rigor. All the input variables influence the behavior of the productivity of requirements engineers.

Errors observed is highly sensitive to: changes made to the delay in gaining technological progress; increases made to the fraction of capacity to train; and decreases made both to the time for management to respond to quality and the time to acquire customer skills for customer product value addition. Errors observed are not sensitive to increases in the time to train requirements engineers. Like productivity, errors observed are sensitive to changes made to all the key input variables of the developed RPI model.

Process rigor and process improvement capability are not sensitive to increases made to both the fraction of capacity to train and delay in technological progress. Process rigor is also not sensitive to decreases in the time to train, fraction of capacity to train, and delay in acquiring customer skills. Process rigor and process improvement capability are sensitive to changes made to the rest of the input variables. Process improvement capability, however, is highly sensitive to increases in the time to train.

These findings reveal that a small change in the sensitive RPI variables causes instability in the model results. Therefore one may conclude that these variables are the most important drivers for RPI. Both researchers and managers should closely monitor these variables during RPI since they may have significant implications on the RPI results.

The conclusion we further draw from the sensitive variables to changes in the input variables is that they are critical determinants of the quality of process improvement (Williams, 2003a; Forrester and Senge, 1980). The sensitive variables are errors observed, productivity of requirements engineers, process improvement capability, process rigor, and customer satisfaction. Among the sensitive variables, productivity of requirements engineers, errors observed and process rigor are a true reflection of the critical success factors that influence process improvement in low maturity organizations as emphasized by (Rainer and Hall, 2002). The authors argue that the critical success factors for process improvement in low maturity organizations begin with training and reviews, followed by developing standards and procedures (process rigor) (Rainer and Hall, 2002). Our results depict that process improvement capability and customer satisfaction are also critical success factors for process improvement in low maturity organizations.

In (Rainer and Hall, 2002), it is claimed that the high maturity organizations, which have a better understanding of process improvement, consider more variables as critical for the success of process improvements in addition to the variables recognized as critical by low maturity organizations. The additional variables in high maturity organizations include management commitment and process improvement capability (Rainer and Hall, 2002). In Table 1, our results reveal that management commitment is not among the critical success factors for RPI in low maturity organizations but process improvement capability is. Furthermore, (Trienekens *et al*., 2007) argue that at whatever level of maturity an organization is, its success will also depend on customer satisfaction and the process improvement costs involved (Trienekens *et al*., 2007). However, our results reveal that process improvement costs are not a critical variable for the success of RPI in low maturity organizations but true for customer organizations.

All the seven key success variables for RPI described in Section 2 may be important for process improvement at whatever level of maturity the organization is, however some are more critical when the organization is at a higher maturity level (Rainer and Hall, 2002; Zawedde *et al*., 2011). The relationship amongst the key variables for successful process improvement has been demonstrated and the effects of each of these variables can be explained through the insights generated by the developed RPI model (Rainer and Hall, 2002; Zawedde *et al*., 2011) as part of our future work.

5 Limitations of the Study

The RPI tool developed representing the dynamics of key RPI variables provided insights and understanding of the dynamic behavior emerging from the feedback structure presented in Section 3. Several contribution were made, however, there were some limitations that need to be addressed to improve the RPI tool further.

Data Collection: One of the limitations of the developed requirements process improvement model was data used to populate the model. Unreliability of some data was as a result of the data not being readily available and some figures like the cost figures were estimations since the actual figures could not be readily obtained from the RPI cases as a result of confidentiality. Data for some variables like average industrial salaries and costs was captured from technical and survey reports on software project management. This was considered to be accurate data since the reports were company reports or published in peer reviewed journals.

Model Verification: There were six RPI experts who participated in the validation and verification of the model during the development stage (that is verification of the causal loop diagrams and the stock and flow diagrams) and the results were attained using the Delphi method. The most appropriate method to be used for the validation could have been a focus group discussion but given the difference in the time zones and the busy nature of their work, the verification was done individually. This could possibly have affected the verification results since group verification would have produced the best results.

Model size and complexity: The size of the developed RPI model and the resulting complexity tend to increase as the model is subjected to more variables. Therefore large volumes of data may obscure the insights generated by key variables of interest for studying the dynamics of RPI.

Based on the limitations of this research, this paper recommends further work that may be pursued to improve the requirements process improvement model.

6 Conclusion and Future Work

In this paper we have discussed the development of a system dynamics based RPI model. Using the developed RPI model, a number of feedback structures variables on process improvement. It is recommended that the feedback structures are investigated further and validated to ensure that their importance is generic to other process improvement domains besides the business systems domain that was considered for this research. The following were also identified as potential future research work:

Integrating the developed SD RPI model with Statistical Process Control (SPC) could further enhance process improvement decision making. SPC determines the stability or instability of a process by dynamically determining an upper and lower control limit of acceptable process performance variability (Baldassare *et al*., 2004). These control limits act as signals or decision rules, and would provide RPI stakeholders information about the process and its state of control (Baldassare *et al*., 2004).

Use of group model building (GMB) to improve on the data that was collected for this research. Data was collected from RPI experts on an individual basis due to global location and busy schedule. GMB has an advantage of group communication, collaboration, conflict management, and decision making in which diverse strengths and expertise of the group members can yield a greater number of alternatives that are of higher quality than the individual (Dwyer and Stave, 2008).

References

1. Abdel-Hamid, T., Madnick, S.: Software Project Dynamics. Prentice-Hall (1991)
2. Albin, S.: Building a System Dynamics Model Part 1: Conceptualization. Technical report, Massachusetts Institute of Technology (1997)
3. Anliang, N., Hong, H., Qingyi, H., Bin, Y., Kegang, H.: Requirements engineering processes improvement: A systematic view. Lecture Notes in Computer Science 3840/2006 (2006) 151-163
4. Baldassare, T., B_oli, N., Caivano, D., Visaggio, G.: Managing Software Process Improvement (SPI) through Statistical Process Control (SPC). In Bomarius, F., Lida, H., eds.: PROFES. (2004) 30-34
5. Beecham, S., Hall, T., Rainer, A.: De_ning a Requirements Process Improvement Model. Software Quality Journal 13 (2005) 247-279
6. Berard, C.: Group Model Building Using System Dynamics: An Analysis of Methodological Framework. The Electronic Journal of Business Research Methodology 8(1) (2010) 35-45
7. Clempner, J.: A Hierarchical Decomposition of Decision Process Petri nets for Modeling Complex Systems. Internationa Jounal of Applied Mathematics and Computer Science 20(2) (2010) 349-366
8. Colton, S., Hatcher, T.: The Web-based Delphi Research Technique as a Method for Content Validation in HRD and Adult Education Research. (2004)
9. Cooper, J., Lee, S., Gandhi, R., Gotel, O.: Requirements Engineering Visualization: A Survey on the State-of-the-Art. In: Fourth International Workshop on Requirements Engineering Visualization (rev'09). (2009)
10. Day, J., Bobeva, M.: A Generic Toolkit for Successful Management of Delphi Studies. The Electronic Journal of Business Research Methodology 3(2) (2005) 103-116
11. Dwyer, M., Stave, K.: Group Model Building Wins: The results of a comparative analysis. In: Proceedings of the 26th International Conference of the System Dynamics Society. (July 2008)
12. Ferreira, S., Collofello, J., Shunk, D., Mackulak, G.: Understanding the effects of requirements volatility in software engineering by using analytical modeling and software process simulation. Journal of Systems and Software 82(10) (2009) 1568-1577
13. Forrester, J.: System Dynamics and a lesson for 35 years. Sloan School of Management Massachusetts Institute of Technology, Cambridge, MA, USA (1991)
14. Forrester, J., Senge, P.: Tests for building Confidence in System Dynamics models. In Legasto, A.J., Forrester, J., Lyneis, T., eds.: Multimodal Intelligent Information Presentation. System Dynamics, New York: Elsevier North-Holland (1980) 209-228
15. Foulds, L., West, M.: The productivity of large business information system development. Journal International Journal of Business Information Systems archive 2(2) (February 2007) 162-181
16. Hall, T., Rainer, A., Baddoo, N.: Implementing Software Process Improvement: An Empirical Study. Software Process Improvement and Practice 7 (2002) 3-15
17. Harris, B., Williams, B.: System Dynamics Methodology. W.K. Kellogg Foundation (2005)
18. Hekimoglu, M., Barlas, Y.: Sensitivity Analysis of System Dynamics Models by Behavior Pattern Measures. In: Proceedings of the 28th System Dynamics Conference. (2010)

19. Maani, K.: Group Model Building for Consensus Building and Team Learning – A Case Study. In: Proceedings of the 18th International Conference of the System Dynamics Society, Norway (August 2000)
20. Mohapatra, S., Gupta, K.: Finding Factors Impacting Productivity in Software Development Project Using Structured Equation Modelling. International Journal of Information Processing and Management 2(1) (January 2011) 90-100
21. Nasir, M., Sahibuddin, S.: Critical success factors for software projects. International Journal of Physical Sciences 6 (2011) 2174-2186
22. Pfhal, D., Ruhe, G.: IMMoS: A Methodology for Integrated Measurement, Modelling, and Simulation. International Journal of Software Process Improvement and Practice 7 (2003) 189-210
23. Rainer, A., Hall, T.: Key success factors for implementing software process improvement: a maturity-based analysis. Journal of Systems and Software 67(2) (2002) 71-84
24. Richardson, G., Lyneis, D.: Getting Started with Behavior Over Time Graphs: Four Curriculum Examples. Creative Learning Exchange. (May 1998)
25. Richardson, G., Pugh, A.: Introduction to System Dynamics Modeling with DYNAMO. Productivity Press, Portland, Oregon, USA (1981)
26. Royce, W.: Software Project Management: A Unified Framework. Addison Wesley (1998)
27. Rwashana, A., Williams, D., Neema, S.: System Dynamics Approach to Immunisation Healthcare issues in Developing Countries: A Case Study of Uganda. Health Informatics Journal 15(2) (2009) 95-107
28. Solemon, B., Shahibuddin, S., Abd Ghani, A.: Re-defining the Requirements Engineering Process Improvement Model. In: Proceedings of the 2009 16th Asia-Pacific Software Engineering Conference, Penang, Malaysia (December 2009) 87-89
29. Sommerville, I., Ransom, J.: An Empirical Study of Industrial Requirements Engineering Process Assessment and Improvement. ACM Transaction on Software Engineering and Methodology 14(1) (2005) 85-117
30. Sterman, J.: Business Dynamics: System thinking and modeling for a complex world. Irwin Professional Publications, New York, USA (2000)
31. Trienekens, J.M., Kusters, R., van Genuchten, M.I., Aerts, H.: Targets, drivers and metrics in software process improvement: Results of a survey in a multinational organization. Software Quality Journal 15 (2007) 135-153
32. Williams, D.: Challenges of System Dynamics to Deliver RE Projects: faster, better and cheaper. In: Proceedings of the 21st International Conference of the System Dynamics Society. (2003a)
33. Williams, D.: Integrating Systems Dynamics Modelingand Case Study Research Method: A theoretical framework for process improvement. In: System Dynamics Society. (2003b)
34. Williams, D., Kennedy, M.: Towards a Model of Decision Making for Systems Requirements Engineering Process Management. Software Quality Journal 8 (2000) 1-5
35. Zawedde, A., Williams, D.: Dynamics of Software Systems projects during the Requirements Process Improvement. In Press: International Journal of Simulation and Process Modeling(2013)
36. Zawedde, A., Klabbers, M., Williams, D., Brand, M.v.d.: Understanding the dynamics of requirements process improvement: a new approach. In D. Caivano, M. Oivu, M.B..G.V., ed.: Product-Focused Software Process Improvement (12th International Conference, PROFES 2011, Torre Canne, Italy, June 20-22, 2011. Proceedings). Volume 6759., Berlin: Springer, Lecture Notes in Computer Science (June 2011) 276-290