The Rosetta Stone Methodology – A Benefits Driven Approach to Software Process Improvement

Fionbarr McLoughlin and Ita Richardson

Lero – the Irish Software Engineering Research Centre
University of Limerick
Limerick
Ireland

Abstract. In response to the lack of a business-focused approach to software process improvement (SPI), the Rosetta Stone objective-driven SPI Methodology (RSM) has been developed which allows organizations to undertake SPI based on business-driven objectives using proven SPI methodologies. To demonstrate usefulness and practicality, the Rosetta Stone IGSI-ISM to CMMI Instance mapping (RS-ICMMI) is developed using a generic set of business objectives which are mapped to the CMMI (Staged) model using a modified version of GQM. This methodology and the RS-ICMMI instance have been validated by experts.

1 Introduction

In companies, a significant amount of capital expenditure and operating expenses are spent on Information and Communications Technology (ICT). In fact, according to the Organization for Economic Co-Operation and Development (OECD) [1], total worldwide spending on ICT was expected to reach $2.964 trillion in 2005 (the most recent OECD estimates). Therefore, it is important that the ICT maturing process continues to evolve. From a Software Process Improvement (SPI) perspective, there are several competing and, in some cases, complementary standards such as the Software Engineering Institute's CMMI for Development version 1.2 [2]; the International Standards Organization's (ISO) ISO15504 [3], formerly known as SPICE; the Trillium Model [4], developed originally in 1991 by Bell Canada; the ISO's 9000-3 standard [5] and the ISO 9001:2000 standard [6], a process-driven approach to define, establish and maintain software quality within an organization that will allow organizations to meet their business objectives [7].

Quite a deal of literature supports the hypothesis that implementation of the various SPI methodologies will result in benefits to organizations. However, they do so from an IT perspective. There are few, if any, methodologies which approach systems improvement from a business goals and objectives perspective. Our research has demonstrated that these benefits come about *as a result* of implementation of SPI which is IT-centric. In other words, ICT drives the business benefits. The Rosetta

Stone Methodology[1], developed and evaluated as part of our research and presented in this paper, consists of a methodology which allows businesses to undertake business- and organizational-driven goals and objectives.

Section 2 of this paper describes the reported benefits of implementing software process improvement, and our research method is described in section 3. In section 4, we present the development of the Rosetta Stone Methodology, its constituent elements and a specific implementation. This is followed by a discussion and conclusion in sections 5 and 6.

2 Benefits of SPI

Software and systems development methodologies have evolved to enable the development of ever larger and more complex solutions to real-world problems. However, there are concerns and while advances have been made there are still quite a few horror stories reported [8]. To get to where we are now has resulted from the gradual evolution of development processes. This evolution includes, but is not limited to, solutions such as software inspections [9], structured programming [10], software process improvement techniques and project management methodologies. We are cognisant of the work of Solon and Statz [11] and Zahran [12] when they discuss the difficulties of using benchmark SPI benefits in making business cases for the implementation of SPI. While at a high level benefits are categorized consistently in macro terms such as Return On Investment (ROI), Quality, Defect Density, and Reduced Cycle Times, upon more detailed review results are not normalized nor is there consistency in how benefits are defined. An additional problem is that much of the literature deals with the results of SPI from individual organizations. Also, there are benefits which, while of interest to the community as a whole, are mentioned in only a small minority of research reports.

We now present an overview of the reported benefits resulting from the implementation of SPI. Our intention here is not to provide an exhaustive review of all the reported benefits of SPI but merely to demonstrate that there is considerable evidence to support the view that SPI is beneficial to organizations.

2.1 Reported Benefits of SPI

Return On Investment (ROI) reviews often feature large companies. Humphrey et al. [13] described the Software Process Improvement initiative at Hughes Aircraft where, during a 4 year period (1987-1990) they progressed to CMM Level 3. From an ROI perspective, the assessments cost Hughes Aircraft $45,000 and a further $400,000 over the two-year program of improvements. Hughes estimated savings to be about $2

[1] The Rosetta Stone is an Egyptian stele found by the French in 1799 with three translations of a single passage in Hieroglyphics, Demotic, and classical Greek. It allowed scholars to translate between these three languages. The analogy is that the Rosetta Stone Methodology will allow the translation between business objectives and SPI methodologies.

million. The effects of a CMM-based SPI program at Software Systems Laboratory (SSL) within Raytheon Inc. are described in [14], [15]. Over 5 years this program cost $5million, and the organisation progressed from Level 1 to Level 3, and was working towards a Level 4 assessment. ROI had increased by a factor of 7.7 based on a sample of six projects. Boeing STS, a division of Boeing Inc. that supports space transportation programs for the Department of Defense and NASA, achieved a rating of CMM Level 5 in July 1996. Yamamura and Wigle [16] present an analysis of cost-to-benefit ratios citing a reduction in rework effort by 31% due to formal inspection alone - this translated into a 7.7:1 ROI. In reporting on the progression of the Oklahoma City Air Logistics Center (OC-ALC) from CMM Level 1 to CMM Level 4, Butler and Lipke [17] reported that, for an investment of $6 million, the OC-ALC calculated a reduction in cost of $50.5 million – an 8.4:1 ROI. To further support the argument that ROI increases as a result of implementation of CMMI, the SEI [18] reported an increased ROI of between 2:1 and 27.7:1%, with a median increase in ROI of 4.7:1, based on 16 separate data points. In addition to the individual reports outlined above, both El Emam and Briand [19] and Krasner [20] report summary evidence of the benefits of SPI.

There are many studies which demonstrate that productivity increases as a result of software process improvement. Brodman and Johnson [21], [22], [23] investigated the effect of improving process capability in 33 companies who were at various levels of CMM maturity. They demonstrate increases in productivity ranging from 6.4% to 100%. A study of four projects was undertaken by Software Productivity Research Inc. of the benefits of SPI within Oklahoma City Air Logistics Center (OC-ALC) [24]. This determined that there was a 10 times increase in productivity from the baseline project to their most recent project (while OC-ALC was at CMM Level 2, working its way to Level 4). Dion [14], [15] also reported Productivity increases of a factor of 2.3 in a 5 year time period as a result of implementing CMM. Also reporting productivity increases as a result of implementation of CMM are Herbsleb et al.[25], [26]. Their report shows a productivity increase of between 9% and 67% over a wide range of maturity levels after implementing CMM, with the median increase being 35%.

Goldenson and Gibson [27] detail some preliminary results from the application of CMMI process improvement. In particular, they quoted a 30% increase in Productivity as a result of implementation of CMMI. In a follow-up to the initial 2003 report, the SEI [28] attributed productivity increases as a result of implementation of CMMI of between 9% and 255%, with a median value of 62%. Garmus and Iwanicki [29] report productivity increases of 132% (based on Function Point/Effort Month), and an effort reduction by 50%. NASA's (National Aeronautics and Space Administration) SEL (Software Engineering Laboratory) spent 10 years undertaking an SPI initiative at their Goddard Space Flight Center. Reporting on the SEL in 1994, Krasner, Pyles et al. [30] report that predicted costs were always within 10% of actual costs; only one deadline was missed in 10 years; maintenance cost of code was half that at other IBM software facilities; defects of 0.01 per thousand lines of source code (KSLOC); and an increased error detection rate of 95%. Krasner [31] further reported a reduction in error rates of 75% between 1985 and 1993, a reduction of software development costs by 55%, and an increase of reuse by 300%. He also notes that costs have become more predictable. Yamamura and Wigle [16], in their report on their implementation of CMM Level 5, report that their processes were finding 89% of the defects – thus leaving 11% still baked in. After implementing SPI, virtually 100% of all defects are found. Putnam and Myers [32] reported that quality improvements (by defect ratio) fell from just over 0.1 defects per 1 KSLOC to 0 defects per KSLOC.

The SEI [18], based on 20 separate data points, has attributed quality increases of between 7 and 132%, with a median of 50% to the successful implementation of CMMI. From a defect perspective, McLoone and Rohde [33] found a significant reduction in the hours/KSLOC metric and another reduction in the dollars/KSLOC cost while Garmus and Iwanicki [29] report a reduction in defect density of 75%, all through CMMI implementation. Liu [34] reports significant improvements as a result of Motorola's implementation of CMMI Level 5 in their sites in China. Between 2003 and 2006, Cost of Quality was reduced from approximately 35% to 25%, fewer defects were inserted into code and the faults per line of code was reduced by 13.01% from its pre-CMMI Level 5 level. Studies analysed demonstrate that as organizations implement more quality-oriented processes, the quality of code improves. Additionally, quality increases as process capability maturity levels increase. We also note that it becomes more difficult, and therefore more costly, to increase quality between higher maturity levels.

There is a note of caution, however, associated with these reported results. While there appears to be clear evidence of a correlation between increased ROI and implementation of various SPI initiatives, there also seems to be a trade-off between ROI and Quality, which would seem natural. In the case of SPI programs like CMM and CMMI, the higher an organization progresses up the maturity ladder, the more quality processes are put in place and therefore there is a tendency for quality to increase but, at the same time, ROI decreases [35]. As Fayad and Laitinen [36] note, "moving to levels 4 and 5 sounds worthwhile but there is little empirical evidence to support the move." In addition, while there is consistent evidence of increases in productivity coinciding with the implementation of CMM/CMMI, there is also evidence to suggest that the rate of increase in productivity is not uniformly higher as successive CMM/CMMI levels are implemented. In addition, some research suggests that at least part of the productivity increases relates to technological innovation as a result of process improvement.

2.2 SPI Challenges

There are several challenges associated with the interpretation and use of the research we have reported in the previous sections. Firstly, there is a lack of uniformity in the definition and interpretation of the metrics/indicators used as evidence of the benefits of SPI. Different researchers and practitioners use the same metric to mean different things. Secondly, for various reasons, not all companies, even when using standard industry definitions for metrics, use the same metrics in their studies. The effect of this is that, while there may be quite a lot of research, it is sometimes difficult to find like-metrics upon which to base comparisons. Thirdly, companies may be reluctant to divulge information for commercial reasons, particularly if the results of their SPI effort paint them in a worse light than their peers. Therefore, it is difficult to find studies which report negatively on process improvements.

However, to say that SPI in itself is the silver bullet for the software development process would be less than disingenuous. Nothing in life is free and SPI is no exception to this rule. Various criticisms such as high cost, rigidity in approach, and the increased administrative overhead associated with SPI have all been levelled at SPI – or more particularly at SPI models such as CMMI or ISO15504 [37], [38]. These have been legitimate criticisms. However, it is up to individual organizations to balance the increased costs of assessment and accreditation, the increased size and overhead associated with the SPI model, and any issues arising from rigidity in application of the model with the benefits to the organization as a whole.

In summary, there is a lot of evidence in the literature to show that there are definite benefits to be realized from implementing SPI. However, we have noted little evidence to show that implementation of particular process improvements have a particular effect on the business requirement.

2.3 Bridging the Gap between SPI and the Business

As noted in section 1, there are several SPI methodologies currently available for organizations to use in order to improve their software processes. These methodologies are software centric and are often not tightly linked to an organization's business goals and objectives. In fact, Debou and Kuntzmann-Combelles [39] contend that the major bottleneck to the success of SPI initiatives is the lack of business orientation in how the program is run. Specifically with regard to CMMI, Liu et al. [40] state that there exists a disconnect between business goals and maturity levels. The RSM bridges these gaps by adding two weapons to the practitioner's arsenal. Firstly, it provides a generic methodology, based on a modified version of GQM [41], [42], which can be used to couple a generic benefits model to an arbitrary SPI. Secondly, and perhaps more importantly from a practitioner's perspective, it provides an implementation of the RSM using a for-profit benefits model tied to an industry-standard SPI methodology which has been validated by industry peers and modified based on their feedback.

3 Research Methodology

The research commenced with a literature review and initial interviews with academics and industry personnel. No approach was identified which supported businesses in deciding which software processes to improve to gain specific business benefits. Therefore, the purpose of our research is to create an objectives-driven approach whose use should allow this. It is expected to save organizations both time and resources by allowing them to focus only on those process areas which have a direct bearing on the business objectives they are trying to achieve.

The first step was to create a generic methodology, the Rosetta Stone Methodology (RSM). This was done by creating a meta-model of all the elements involved in an SPI implementation (see LHS of Fig. 1). After this, a step-by-step approach was developed which guides practitioners in using an SPI methodology and benefits model to define a mapping between business-focused benefits and individual SPI process areas. In essence, this process allows practitioners to substitute the meta-model with a concrete implementation instance of the model (see RHS of Fig. 1). This mapping is then used as the basis to answer various questions regarding which process areas should be implemented to achieve specific business benefits and in what particular order.

To demonstrate the implementation of the RSM in a specific instance, we investigated available return on investment models which did not deal exclusively with software process improvement, but with which existing SPI models could be combined. We chose to work with the IGSI-ISM Benefits Model [43] and CMMI Version 1.2 [2]. This is done as follows:

1. Determine which benefits model and which SPI model which is to be used;
2. Define the mapping (relationships) between objectives/benefits and software processes;
3. Answer the questions that are relevant to the individual organization.

The initial methodology, meta-model and implementation instance were developed as described and were then reviewed by a small group of peers for validity. For triangulation purposes, they were validated through a Delphi review of 17 people with an average work experience in the software industry of 19 years along with an average of 11 years of SPI experience. Additionally, to validate the implementation instance, a group of experts was interviewed about each relationship within the RS-ICMMI model. Out of a pool of ten experts, two experts were randomly selected to review a set of IGSI-ISM Benefit/CMMI Level 2 combinations. They discussed whether they agreed with the relationship presented and where they had seen these relationships work in practice. This process was repeated until all combinations had been reviewed. In some cases, the RS-ICMMI was modified as a result of these interviews.

4 Rosetta Stone Methodology

While there are many reported benefits from SPI projects, our observation is that the SPI agenda has been undertaken to improve particular processes for the process-sake, rather than organizational benefits as the primary objective. This is typically not the way the commercial world works. Therefore, to achieve a business-oriented focus, the outcome from our research will allow organizations to achieve organization-specific objectives through improving their software process.

In the first instance we have developed the Rosetta Stone Meta Model. The meta-model is, in essence, an entity-relationship model which relates together all the major elements within any SPI initiative – business objectives desired, returns associated with achieving the business benefits, process areas, costs of implementing the process areas, and the metrics/indicators to determine progress/regression towards the objectives (see LHS of Fig. 1). The Rosetta Stone Methodology (RSM) consists of using the Rosetta Stone Meta Model to create a concrete instance of the meta model. The main benefits of using this methodology are that users are able to:

- Achieve specific business objectives by targeting particular software processes to improve in order to achieve business benefits

- Understand what benefits may be derived from the improvement of which particular software process
- Given a set of existing metrics and values, determine what processes may be more readily and quickly implemented than others.

4.1 Objectives, Process Areas, and Indicators

The most important element in RSM is the *set of business objectives* or *benefits* which an organization wishes to achieve. If possible, these should be hierarchical so that the achievement of one should lead to the achievement of others. For example, if on-time delivery of projects is achieved (one possible business objective) then this should result in better customer satisfaction (another possible business objective). Each benefit should have some form of *return* associated with it – some way of determining, frequently quantifiable but sometimes qualitative, the value of the benefit. Returns are meaningful to the business and, as such, are typically not SPI-type metrics such as defects/KSLOC or defects/function point – unless, of course, the business is primarily focused on software development. For example, if productivity were the objective, it might be possible to say that, for an x% increase in productivity, there should be an increase in profits of y%. For each objective, there is at least one *indicator* - a set of metric(s) that are an indication that a particular benefit has occurred. In other words, a set of indicators that can prove (or disprove) that progress is being made towards a specific benefit – a way to measure a benefit. *Process areas* are those processes which are being improved during the SPI program, and would include, for example requirements management, risk management and project planning. Each process area has a *cost* associated with it – costs associated with implementing the improvement.

In order to make a concrete instance of the model, the practitioner must first choose which objectives are most relevant to their business and then choose which SPI model is most appropriate for their organization. These two entities then drive the choice of costs, returns, and indicators. In addition, it is important to define the relationships between the objectives/benefits model and the software processes. This can be done using specific instances of the model.

4.2 Return, Costs and ROI

For the majority of organizations, where profit is a primary goal, benefits should ultimately lead to a monetary impact on the organization. One of the main advantages of RSM is that it is now possible to tie software process improvements to specific benefits due to the fact that the benefits defined in RSM are very granular. It must be recognized, however, that in some cases it is difficult to measure the monetary value of a benefit – for example, how can a dollar value be put on increased team morale? In the case of RSM, the return on the SPI is compared to the cost of improving the specific software process. Great care must be taken, therefore, to not only capture the monetary equivalents that accrue from the benefits of process improvement but also the cost of implementation.

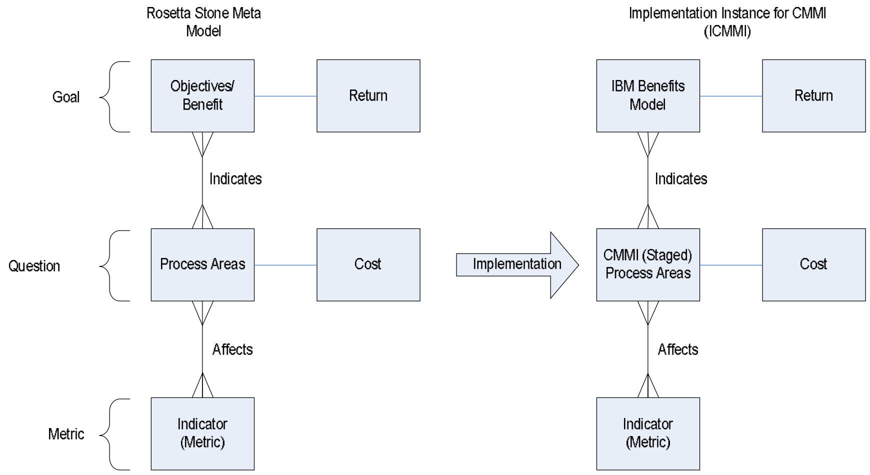

Fig. 1. Rosetta Stone objective-driven software process improvement Model (RSM)

5 Rosetta Stone Methodology: CMMI Implementation Instance

We demonstrate the implementation of the RSM through mapping the CMMI (staged) model to a benefits model developed by IBM Global Services, the IGSI-ISM Benefits Model [43]. The implementation of this instance is illustrated in the RHS of Fig. 1 and the final output is the RS-ICMMI.

The IGSI-ISM model (see Fig. 2) shows the relationships between the various benefits which culminate in the ultimate benefit for the organisation – *increased revenues/profits*. This benefit can be achieved through relationships between 21 separate identifiable benefit areas. These include benefits such as lower time-to-market, better risk management and competitive proposals. In addition, the model is a hierarchy of benefits – higher level benefits are derived from elements that are lower in the benefit tree. For example, *better product quality* leads to *increased productivity*. Similarly, *increasing the understanding of customer needs* leads to *setting right customer expectations*, thus to *improved predictability* and to *more competitive proposals*. Both *increased productivity* and *more competitive proposals* lead to an *improved image* which feeds directly to *increased revenues/profits*

The RSM requires us to map the IGSI-ISM Benefits model to the software processes whose improvements will provide these benefits. To do this, each the generic goal, specific goal and specific practice of each CMMI process area was reviewed, determining which ones have particular relevance to the IGSI-ISM benefit model. To define the mapping between objectives/benefits and software processes, a modified approach to Basili's Goal-Question-Metric approach [41], [41] is used. A *reverse mapping* between process areas (Questions) and business objectives (Goals) is created by asking what process areas (Questions) impact what business objectives (Goal). In effect, the reverse lookup asks "What objectives does this process area fulfil?"

We note here that not all benefits are equal and the RSM differentiates between primary and secondary benefits. A *primary* benefit of a process area is one that is brought about as a direct result of implementation of that process area where the cause and effect relationship between the process area implementation and the benefit is very strong. *Secondary* benefits are those benefits which are not primary benefits and include *derived* benefits. A *derived* benefit is a benefit which is a hierarchical ancestor of either a primary or secondary benefit. As we shall see later, the benefit classification is used to determine the recommended order of process areas to be implemented.

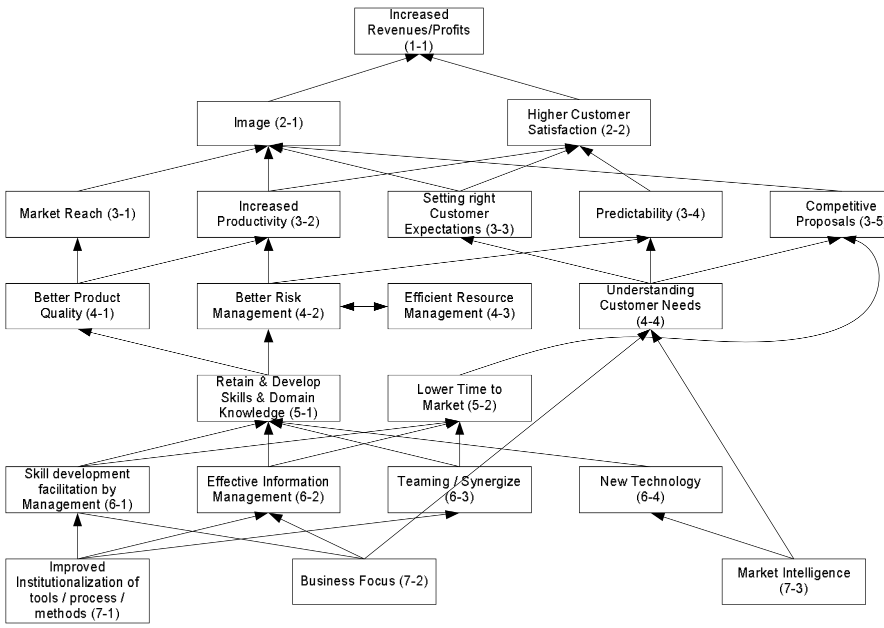

Fig. 2. IGSI-ISM Benefits Model

5.1 Examples of CMMI Level 2 Process Area to Benefit Mappings

Requirements Management. The Requirements Management (REQM) process area contains 1 Specific Goal (SG) which in turn consists of 5 Specific Practices (SP). The goal is that “requirements are managed and inconsistencies with project plans and work products are indentified”, maintaining a current approved set of requirements over the life of the project. REQM requires the implementation of the obtaining of an understanding of requirements (SP 1.1-1), the obtaining of a commitment to requirements (SP 1.2-2), the management of requirements (SP 1.3-1), and the identification of inconsistencies between project work and requirements.

Based on the specifications of the REQM as defined by the SEI, the following are the expected primary benefits of implementing REQM:

- *Better Risk Management*: By managing requirements and identifying inconsistencies, we are better able to identify alternative strategies and avoid building software that isn't part of a customer's requirements.
- *Understanding Customer Needs*: Proper management of requirements forces us to consistently review those requirements and thus focus on understanding customer needs. By indentifying inconsistencies between requirements, plans, and work products we are constantly ensuring that the customer's needs are always foremost.
- *Lower Time to Market*: by identifying inconsistencies up front, we will spend less time working on items that are not required by customers or that are inconsistent with customers' needs and expectations. As a result, less time will be spent on rework, thus saving resources and reducing time to market.

Configuration Management. The purpose of Configuration Management (CM) is to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits [44], [45]. CM consists of 3 SGs – SG1 (the establishment of baselines), SG2 (the tracking and control of changes), and SG3 (the establishment of integrity). In software projects it is absolutely essential that all artefacts are correctly baselined and tracked. Without this baselining and tracking, there is no guarantee that code, requirements or any other project artifact will be consistent with each other, thus increasing risk and reducing quality. In fact, the opposite is true, effective configuration management is essential for increasing quality and reducing risk. In addition, as [28] note, "configuration management, and in particular version control, plays a role in supporting to work of teams" and that "software configuration management serves as a mechanism for communication, change management and reproducibility."

Configuration management allows projects to properly track the various parts that make up their products. By instituting CM, multiple teams will be able to edit/modify code without stepping over each others' toes. In addition, CM allows project teams to map changes back to specific issues or requirements, this increasing product quality and managing risk. Therefore in the RS-CMMI, CM results in the primary benefits: *Better Quality Product, Better Risk Management, Teaming / Synergize.*

5.2 Achieving Specific Business Objectives

In the exemplar we have demonstrated that organisations may (normally) require increased revenues/profits, and are not particularly interested in which software process improvement methodology or software process area is used to deliver the business benefits. The process to determine which process areas to execute in order to achieve specific business benefits is as follows:

1. Determine which of the IGSI-ISM objectives that we wish to achieve. This is normally determined from outside the software process improvement group, possibly from either external clients or senior management. We will use *lower time-to-market* as the objective in this example.

2. Using the IGSI-ISM model (Fig. 2), determine which other objectives, if any, contribute to achieving our primary objective. We observe that *skill development facilitation by management, effective information management, teaming/synergize, improved institutionalization of tools/process/methods* and *business focus* all contribute to *lower time-to-market.*
3. Using the implementation mapping developed during the creation of the implementation instance of the RSM methodology, establish which process areas contribute to both the primary and secondary objectives of the selected business benefits. For illustrative purposes we will use *lower time-to-market* (node 5-2 in Fig. 2) as the example benefit we wish to achieve and have provided a reduced version of the RS-ICMMI mapping in Table 1 which contains only those process areas which have *lower time-to-market* as either a primary or secondary benefit.
4. Rank the PAs in order of relevance and implementation. There are quite a few PAs which have an effect on lower time-to-market. Most organizations have finite resources and therefore will need to prioritise their implementation. There are many different ways to rank them. More consideration should be given to those PAs that *primarily* satisfy a particularly objective. In the case of lower time-to-market, we would implement Process and Product Quality Assurance (PPQA) before implementing Configuration Management (CM) as PPQA primarily satisfies lower time-to-market while CM only secondarily satisfies lower time-to-market (see Table 1). Additionally, we should observe the software process model we are using. Although within RS-ICMMI both Requirements Management (RM) and Requirements Development (RD) directly satisfy *lower time-to-market*, as, within the CMMI staged model, RM is a Level 2 PA, it should be undertaken before RD. Using these principles, the first three process areas that we propose implementing to *lower time-to-market* from the Level 2 Process Areas would be Requirements Management, Supplier Agreement Management, Measurement and Analysis and Process and Product Quality Assurance. Configuration Management would not be implemented until later as lower-time-time market is only a secondary benefit. By ordering implementation based on relevance, as above, we ensure that those process areas are implemented which have most impact on the business objective. As a result, we implement those process areas up front which provide biggest bang for the buck for the business objective desired.

We must recognize, however, that this proposed methodology is not without its limitations. From a practical perspective, while both the methodology and the RS-ICMMI implementation instance have been reviewed at length by practitioners, it has not yet been actually put into practice. Another challenge is that this research has taken place over several years and one of the challenges is to keep the RS-ICMMI model up to date with the latest version of CMMI. Finally, while the IGSI-ISM Benefits model is a good generic business objectives model there are many organizations out there which do not follow a for-profit business model such as represented by the IGSI-ISM model. Further research may be appropriate to bring in other types of benefits model.

Table 1. Example Objective - Lower Time to Market

Staged Level	Process Area	Expected Primary IGSI ROI Benefits	Expected Secondary IGSI ROI Benefits
2 - Managed	Requirements Management	4-2, 4-4, 5-2	7-3, 5-1, 4-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Supplier Agreement Management	5-2, 4-1, 4-2, 3-4	3-1, 3-2, 3-3, 3-4, 2-1, 2-2, 1-1
	Measurement and Analysis	5-2, 4-1, 4-2, 3-4	3-1, 3-2, 3-3, 3-4, 2-1, 2-2, 1-1
	Process and Product Quality Assurance	7-1, 5-2, 4-1, 4-2, 3-4	6-1, 6-2, 6-3, 5-1, 5-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Configuration Management	6-3, 4-2, 4-1	5-1, 5-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
3 - Defined	Requirements Development	5-2, 4-1, 4-2, 4-4,	4-3, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Organizational Process Focus	7-1, 7-2, 7-3	6-1, 6-2, 6-3, 6-4, 5-1, 5-2, 4-1, 4-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Organizational Process Definition	7-1, 6-2	6-1, 6-3, 5-1, 5-2, 4-1, 4-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Organizational Training	6-1, 6-3	5-1, 5-2, 4-1, 4-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Integrated Project Management	6-3, 4-1, 4-2, 4-3, 4-4	5-1, 5-2, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Integrated Project Management for IPPD	6-3, 4-1, 4-2, 4-3, 4-4	5-1, 5-2, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Integrated Teaming	6-3, 6-2, 4-1, 4-3, 4-2, 4-4	5-1, 5-2, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Integrated Supplier Management	7-2, 7-3, 6-3, 6-4, 5-2, 4-1, 4-2, 3-4	5-1, 5-2, 4-3, 3-1, 3-2, 3-3, 3-5, 2-1, 2-2, 1-1
	Organizational Environment for Integration	7-1, 7-2, 6-2, 6-3, 5-2	6-1, 6-2, 6-4, 5-1, 5-2, 4-1, 4-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
4 - Quantitatively Managed	Organizational Process Performance	7-1, 7-2, 7-3, 6-2	6-1, 6-3, 5-1, 5-2, 4-1, 4-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Quantitative Project Management	5-2, 4-1, 4-2, 4-3	4-3, 4-4, 2-1, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
5 - Optimizing	Organizational Innovation and Deployment	7-1, 7-2, 7-3	6-1, 6-2, 6-3, 6-4, 5-1, 5-2, 4-1, 4-2, 4-3, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 2-2, 1-1
	Causal Analysis and Resolution	7-1, 6-2, 5-2, 4-1, 4-2, 4-3	6-1, 6-3, 5-1, 4-4, 3-1, 3-2, 3-3, 3-4, 3-5, 2-1, 3, 1-1

6 Conclusion

The purpose of this research was to develop a generic methodology that allows organizations to achieve specific business-focused objectives by implementing various existing and proven SPIs. While a business-driven approach to SPI is research-worthy in itself, in order for such a model to be successful in the real world it should be flexible enough to be able to support the sometimes vastly different organizational objectives of various types of business – government organizations, non-government organizations (NGOs), the military, and for-profit commercial companies to name but a few. Not only should it be flexible enough to support these various organization types, but it should also be customizable so that individual organizations are able to customize benefit models. In addition, as an enormous amount of effort has been spent on SPI and SPI research, any proposed model should leverage existing work as much as possible. In order to meet these objectives the Rosetta Stone methodology was developed. It is a generic benefits-driven methodology which, in its essence, allows practitioners to map from a benefits model which is appropriate to an organization to a proven SPI methodology. In addition, is it fully customizable and allows organizations to make adjustments to the model where they feel it appropriate.

This research has brought together business focus and SPI. Two business-focused SPI models are presented – the RSM meta-model which maps from arbitrary benefits models to arbitrary SPI models and the RS-CMMI model which maps from the IGSI-ISM benefits model to the CMMI (Staged) model. We are currently evaluating both models through case study research with software process practitioners.

Acknowledgement

This research was supported by the Science Foundation Ireland funded projects, Global Software Development in Small to Medium Sized Enterprises (GSD for SMEs) grant number 03/IN3/1408C and B4-Step grant number 02/IN.1/108 within Lero - the Irish Software Engineering Research Centre (http://www.lero.ie).

References

1. Organization for Economic Co-Operation and Development: OECD Information Technology Outlook, 2006. OECD (2006)
2. Software Engineering Institute: CMMI for Development Version 1.2. SEI (2006)
3. International Standards Organization: ISO/IEC 15504-2:2003 Information technology - Process assessment - Part 2: Performing an assessment. (2003)
4. Coallier, F.: Trillium Reference Manual. (1994)
5. International Standards Organization: ISO 9000-3. International Standards Organization (1994)
6. International Standards Organization: ISO 9001:2000. International Standards Organization (2000)
7. Hailey, V.A.: ISO 9001: A Tool for Systematic Software Process Improvement. In: Hunter, R., Thayer, R.H. (eds.): Software Process Improvement. IEEE Computer Society (2001) 291-309
8. Standish Group: The Chaos Report. The Standish Group (2009)
9. Fagan, M.E.: Design and Code inspections to reduce errors in program development. IBM Systems Journal 15 (1976) 182 - 211
10. Dijkstra, E.: Go to statement considered harmful. Classics in Software Engineering. Yourdon Press (1979) 27 - 33
11. Solon, R., Jr., Statz, J.: Benchmarking the ROI for Software Process Improvement (SPI). Software Tech News 5 (2002) 6 - 11
12. Zahran, S.: Business and Cost Justification of Software Process Improvement - "ROI from SPI". International Software Process Association Conference, Brighton, England (1996)
13. Humphrey, W.S., Snyder, T.R., Willis, D.R.: Software Process Improvement at Hughes Aircraft. IEEE Software 8 (1991) 11-23
14. Dion, R.: Elements of a Process-Improvement Program. IEEE Software 9 (1992) 83-85

15. Dion, R.: Process Improvement and the Corporate Balance Sheet. IEEE Software, Vol. 10 (1993) 28-35
16. Yamamura, G., Wigle, G.B.: SEI CMM Level 5: For the Right Reasons. CrossTalk - The Journal of Defense Software Engineering (1997)
17. Butler, K.L., Lipke, W.: Software Process Achievement at Tinker Air force Base, Oklahma. Software Engineering Institute (2000) 58
18. Software Engineering Institute: CMMI Performance Results - 2005. Vol. 2008 (2005)
19. El Emam, K., Briand, L.: Cost and Benefits of Software Process Improvement. Fraunhofer IESE (1997) 27
20. Krasner, H.: Accumulating the body of Evidence for the Payoff of Software Process Improvement. In: Hunter, R., Thayer, R.H. (eds.): Software Process Improvement. IEEE (2001) 519 - 539
21. Brodman, J.G., Johnson, D.L.: Return on Investment (ROI) from Software Process Improvement as Measured by US Industry. Software Process: Improvement and Practice 1 (1995) 35-47
22. Brodman, J.G., Johnson, D.L.: Return on Investment from Software Process Improvement as Measured by U.S. Industry. CrossTalk - The Journal of Defense Software Engineering (1996)
23. Brodman, J.G., Johnson, D.L.: Realities and Rewards of Software Process Improvement. IEEE Software 13 (1996) 99-101
24. Butler, K.L.: The Economic Benefits of Software Process Improvement. CrossTalk - The Journal of Defense Software Engineering 1995 (1995)
25. Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., Zubrow, D.: Benefits of CMM-Based Software Process Improvement: Initial Results. Software Engineering Institute, Carnegie-Mellon University (1994)
26. Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., Zubrow, D.: Benefits of CMM-Based Software Process Improvement: Executive Summary of Results. Software Engineering Institute (1994) 16
27. Goldenson, D.R., Gibson, D.L.: Demonstrating the Impact and Benefits of CMMI®:An Update and Preliminary Results. The Software Engineering Institute (2003) 55
28. Burzcuk, S.P., Appleton, B.: Software Configuration Management Patterns: Effective Teamwork, Practical Integration. Addison Wesley (2002)
29. Garmus, D., Iwanicki, S.: Improved Performance Should Be Expected from Process Improvement. Software Tech News 10 (2007) 14 - 17
30. Krasner, H., Pyles, J., Wohlwend, H.: A Case History of the Space Shuttle Onboard Systems Project. (1994)
31. Krasner, H.: A Case History of Process Improvements at the NASA Software Engineering Laboratory. (1995)
32. Putnam, L.H., Myers, W.: How Solved is the Cost Estimation Problem. IEEE Software 14 (1997) 105-107
33. McLoone, P.J., Rohde, S.L.: Performance Outcomes of CMMI-Based Process Improvements. Software Tech News 10 (2007) 5 - 9
34. Liu, A.Q.: Motorola Software Group's China Center: Value Added by CMMI. Software Tech News 10 (2007) 18 - 23

35. O'Neill, D.: Determining Return on Investment Using Software Inspections. CrossTalk - The Journal of Defense Software Engineering (2003)
36. Fayad, M.E., Laitinen, M.: Process Assessment Considered Wasteful. Communications of the ACM 40 (1997) 125 - 128
37. Jones, C.: The economics of software process improvement. IEEE Computer 29 (1996) 95-97
38. Reifer, D.: The CMMI: it's formidable. The Journal of Systems and Software 50 (2000) 97 - 98
39. Debou, C., Kuntzmann-Combelles, A.: Linking Software Process Improvement to Business Strategies: Experiences from Industry. Software Process: Improvement and Practice 5 (2000) 55 - 64
40. Liu, X.F., Sun, Y., Kane, G., Kyoya, Y., Noguchi, K.: Business-oriented Software Process Improvement Based on CMM using QFD. Software Process Improvement and Practice 11 (2006) 573 - 589
41. Basili, V.: Software Modeling and Measurement: The Goal/Question/Metric Paradigm. University of Maryland (1992)
42. Basili, V., Caldiera, G., Rombach, H.D.: The Goal Question Metric Approach. In: Marciniak, J.J. (ed.): Encyclopedia of Software Engineering, Vol. 1. John Wiley & Sons Inc (1994)
43. Goyal, A., Kanungo, S., Muthu, V., Jayadevan, S.: ROI for SPI: Lessons from Initiatives at IBM Global Services India. SEPG 2001 (2001)
44. Chrissis, M.B., Konrad, M., Shrum, S.: CMMI : Guidelines for Process Integration and Product Improvement. Addison-Wesley, Boston (2003)
45. Software Engineering Institute: CMMI for Software Engineering, Version 1.1, Staged Representation (CMMI-SW, V1.1, Staged) - CMU/SEI-2002-TR-029. (2002)