Empirical Software Engineering, 7, 93–114, 2002.

Software Process Improvement Motivators: An Analysis using Multidimensional Scaling

NATHAN BADDOO
TRACY HALL
Department of Computer Science, University of Hertfordshire, UK

Abstract. In this paper we present an analysis of software practitioners' motivations for software process improvement (SPI). Our findings are based on an empirical study of SPI in 13 software companies where we conducted focus groups with nearly 200 software practitioners. Our aim is to better understand how companies can maximise practitioner support for SPI. This insight should help SPI managers establish more effective SPI implementation strategies. In this paper we introduce the use of multidimensional scaling (MDS) in SPI research. MDS is a social science data analysis technique designed to generate a rich visual understanding of human issues. By using MDS we found evidence to suggest distinct clusters of punitive and rewarding SPI motivators. Furthermore our analysis also suggests that different clusters of motivations exist for different staff groups.

Keywords: Software process improvement, multidimensional scaling, software practitioners, motivators, SSA.

1. Introduction

In this paper we report our findings into software practitioners' motivators for software process improvement (SPI). Our study uses data collected from focus groups in 13 UK companies encompassing almost 200 software practitioners. We looked separately at the motivations of developers, project managers and senior managers. We pilot the use of multidimensional scaling (MDS) (Shye et al., 1994) to analyse our data and we relate our findings to classic motivation theory.

The principal aim of our work is to understand the relationship between software practitioners' SPI motivators by showing how these motivators co-occur to each other. We use MDS to represent this co-occurrence. MDS is a research technique used increasingly by psychologists to uncover such attitudinal co-occurrence. Our results should help SPI managers develop a richer understanding of the association between the motivators of SPI for different staff groups of software practitioners. This improved understanding should enable SPI managers to implement SPI more effectively.

In previous work, we collected and refined sets of SPI motivators from developers, project managers and senior managers (Baddoo and Hall, to appear in 2002; Baddoo et al., 2000). In particular, we analysed the similarities and differences across practitioner groups. In this follow-up work, we use MDS to examine the relationships between the motivators derived in our previous work.

Our secondary aim in this paper is to demonstrate how MDS can be applied in software engineering to achieve a richer understanding of some of the human factors of this discipline.

SPI has become a popular approach to delivering improvements to the software product (Humphrey, 1989). Many companies have either a formal or informal SPI programme based on one of the existing SPI models. Despite this, many companies report low process maturity (Paulk and Chrissis, 2000). In this paper we explore the non-technical, people-management factors that may explain why companies are failing to achieve high process maturity.

Other empirical studies on SPI confirm the importance of people factors. Herbsleb and Goldenson's study (1996) of process maturity found a relationship between high process maturity and high staff morale. Kaltio and Kinnula's study (2000) of deploying defined software processes found that people factors were most important, in particular skills, motivation and time (Kaltio and Kinnula, 2000).

We suggest that SPI has a higher chance of success in companies where practitioners experience high motivation for it. We also suggest that motivations for SPI vary across practitioner groups and that this variation dilutes the effectiveness of single strand SPI implementation strategies. In presenting the relationship between different practitioners' motivators for SPI, we provide SPI managers with an opportunity to gain a richer insight into these motivators. We suggest that such insight may help managers to develop SPI implementation strategies that are more likely to maximise support for SPI across practitioner groups.

In Section 2 we introduce MDS. In Section 3 we describe our research methods. Section 4 presents results and discussions from the plots of practitioner motivators for SPI. In Section 5 we conclude and summarise this paper.

2. Multidimensional Scaling

Multidimensional Scaling (MDS) is a data analysis technique that represents the relationship between variables within geometric space. It is defined as:

> A set of procedures that allow for information contained in a set of data to be represented by a set of points in space, arranged in such a way that the distances between the points reflect the empirical relationship (Guttman, 1968).

Traditionally, MDS has been used in conjunction with facet theory (Guttman, 1968). Facet theory is an empirical research design and analysis technique that Guttman and Greenbaum (1998) describe as:

1. Having a definitional framework for a universe of observations in a study.
2. Establishing an empirical structure of observations within this framework.

3. Searching for correspondence between the definition framework and the empirical structure.

Facet theory proposes ways of exploring and understanding a concept, for example attitude, intelligence or motivation in this instance, by depicting the inter-relationship between the components of that concept (Shye et al., 1994). This is achieved through the empirical observation of many variables of this concept.

In facet theory, the researcher attempts to better understand a concept in terms of the inter-relationship between the components of the concept. For example, according to Herzberg's (1959) Motivation–Hygiene theory, motivation is made up of 'satisfiers' and 'dissatisfiers'. According to facet theory, in order to understand motivation better, one needs to understand the inter-relationship between the satisfiers components and the dissatisfiers components of motivation. Such understanding is achieved through empirical observation of several motivator variables.

MDS is one technique for representing and analysing the observed empirical variables of a concept.

In this study we apply the principles of facet theory to gain better insight into software practitioners' motivators for SPI. Our concept is motivation for SPI. Our components or facets are the classification of motivation according to classic motivation theory. Our variables are the motivators for SPI as cited by three groups of software practitioners (Baddoo and Hall, to appear in 2002). We use MDS to represent and analyse our variables.

2.1. Studies Using MDS

Meads' (1992) review of the development of MDS indicates that most initial use of the technique was by scientists working in psychophysics and sensory analysis. Today, MDS is predominately being used in investigative and occupational psychology. For example, Donald (1994) used MDS to explore the structure of office workers' experience of work environments. Donald's study used the non-parametric MDS technique of smallest space analysis (SSA) to analyse attitudinal data collected from 215 office workers. Donald's (1994) findings showed that in addition to functional aspects of the office environment other aspects of the environment were equally important, for example, group cohesion and a *spirit de corps*.

Other studies done with MDS in the UK have been in the field of crime research. For example, in interviews with sexual offenders, the motivations, excuses, reasons and justification given by the offenders for their actions form the variables of the analysis. Researchers then try to ascertain how these variables co-occur with each other in multidimensions and establish a criterion for grouping them together (Canter and Heritage, 1990).

The use of MDS in software engineering is rare. In this study we use MDS to analyse the association between motivators of SPI. The type of data we have collected in our study means that we have used SSA—a non-parametric MDS technique.

Table 1. Multivariate correlations: three variables.

	A	B	C
A	*	0.8	0.5
B		*	0.1
C			*

*Denotes correlation of variable with itself. This is invalid.

2.2. An Overview of MDS

A basic tenet of MDS is that the stronger the statistical relationship between variables, the closer they will appear in geometric space. Table 1 shows the statistical relationship between three variables, A, B and C.

Table 1 shows that variables A and B have the highest correlation (0.8). Variables A and C show a weaker correlation than between A and B but a stronger correlation than between B and C. These correlations can be represented in one-dimensional (1D) space as in Figure 1.

The limitation of this 1D representation becomes apparent if Table 1 is expanded to include a fourth variable. Table 2 shows a four variable correlation matrix. To represent the relationship between all four variables as distances will be impossible to do on a 1D scale. This is because it is impossible to place D anywhere in Figure 1 that truly reflects its statistical relationship with the other three variables. Representing the statistical relationships will also be difficult in two dimensions, i.e. in an area. However, conceptualising the space in which these variables occur in three dimensions (within a cube) makes it possible to adequately present this statistical relationship.

MDS comes from a need to represent multiple variables that are related to one another. As the examples in Tables 1 and 2 show, when there are more than three variables, this representation becomes difficult to show either on 1- or 2D scales. A

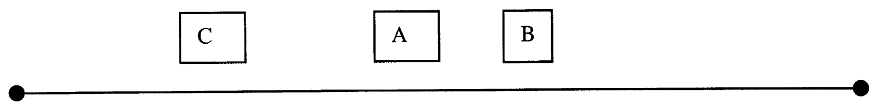


Figure 1. One dimensional representation of statistical relationship.

Table 2. Multivariate correlations: four variables.

	A	B	C	D
A	*	0.8	0.5	0.3
B		*	0.1	0.6
C			*	0.4
D				*

*Denotes correlation of variable with itself. This is invalid.

3D scale, therefore, becomes the best option of representing the statistical relation between several variables.

2.3. Smallest Space Analysis (SSA)

SSA is one of the three MDS techniques recommended by Guttman (1968). It is a non-parametric MDS technique, used for cases where the data being analysed is nominal or categorised ordinal data. The greater the similarity between two variables, the greater their proximity in the corresponding geometric space.

2.3.1. Overview of SSA

The technique is termed *smallest* space analysis because it uses the ranks of the correlation, as opposed to the actual correlation to plot the closeness of points in geometric space. This ensures that the smallest space possible is used in the depiction of relationships.

Traditionally, correlation between quantitative variables is measured by the degrees of change over cases. Correlation is measured in non-parametric data by the presence or absence of variables over a series of cases.

The following is a summary of the steps involved in SSA:

1. Establish a content category dictionary of all variables.
2. Create a data matrix based upon how many occurrences of the variables in the content dictionary appear in each 'case'.
3. Run the SSA tool to plot the variables and their relationships.
4. Partition the SSA plot into regions. The researcher should be able to argue the inclusion of variables in the region they appear.

When a SSA plot is produced, a coefficient of alienation is used to measure the 'fit' between the geometric representation and the original correlation matrix. A coefficient of 0 represents a perfect fit and anything upto 0.25 is considered good.

2.3.2. Illustrating SSA Using Capability Maturity Model Maturity Levels

We illustrate how a SSA plot is derived from non-parametric data, using process maturity levels as an example:

Consider the data in Table 3. Assume that these are the results of a survey conducted on six companies using the capability maturity model (CMM) (Paulk et al., 1994a). SPI managers in each of the six companies have indicated those key

Table 3. Example of a data matrix of non-parametric variables.

Company	KPA's			
	A	B	C	D
1	0	1	0	1
2	1	0	1	0
3	1	0	0	0
4	1	0	1	0
5	0	1	0	1
6	0	0	1	0

performance areas (KPAs) they are currently concentrating on. So that overall, four KPAs have been indicated between the six SPI managers. The similarity between the responses offered by these six SPI managers determines the relationship amongst the four KPAs. Using a SSA plot, such a relationship is depicted by the geometric representation of the KPAs. In Table 3, KPAs depicted similarly by the six respondents will appear closer to each other in a SSA plot.

The geometric representation in a SSA plot is achieved by using the data matrix—the 0's and 1's in, say Table 3—to derive a multivariate correlation matrix with quantitative coefficients between 0 and 1. This matrix is then plotted in MDS to show the likelihood of variables co-occurring to each other. In this illustration, a SSA plot is used to demonstrate the relationship between the four KPAs. That is, the relative likelihood of the four KPAs co-occurring to each other. Figure 2 represents this information geometrically.

Figure 2 shows two clustering of KPAs: A & C and B & D. A further analyses of Figure 2 reveals that the SSA plot has two broad regions in accordance with CMM levels and the closest KPAs happen to co-occur around each other within these regions. These broad regions are CMM level 3 and CMM level 4.

The plot in Figure 2 can now provide some insight into the relationship between the four KPAs. One such insight is that, according to the six SPI managers, companies focusing on *quantitative process management* (A) are more likely to be also focusing on *software quality management* (C) than they are *integrated software*

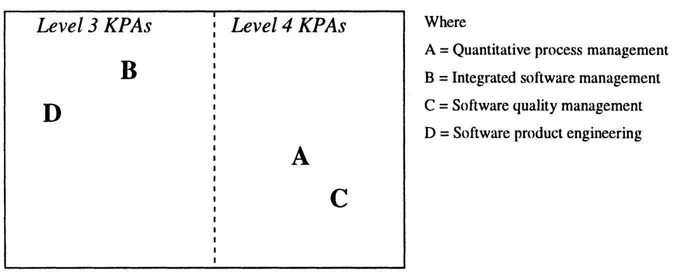


Figure 2. An illustration of a SSA plot using CMM KPAs.

management (B) or *software product engineering* (D). Of course we know this to be true, despite the fact that this is a contrived example. However, it is possible to see how the technique can be used to investigate relationships between variables.

2.3.3. Applying SSA in this Study

In this study, focus group sessions take the place of 'Companies' in the previous illustration and 'KPAs' are the motivators that practitioners cited in focus groups.

The partitioning of SSA's is done in relation to particular 'facets'. In this study our 'facets' are the motivation theories of Skinner and Herzberg (see Appendix C). We partition our SSA into regions defined by the stimulus response theory (SRT) of Skinner (1976) and motivation–hygiene theory of Herzberg et al. (1959).

The regions in a SSA plot are the components (i.e. the facets) of the concept under investigation. Partitioning takes place according to how the variables represent themselves in the plot. When partitioning a SSA, the researcher is trying to show how components of a concept are mapped out on the plot and how variables are represented within the components. Established theory defines the components of each concept and the variables within that component. It should therefore be possible to defend the inclusion of any variable in a particular region/component/facet of a plot with theory.

In this study we have chosen to use Skinner's theory to define the regions in the SSA plots of developers and project managers. However, for the SSA plot of senior managers, we have chosen the theory that defines motivators as either tangible or intangible. The choice of these theories is subjective. Indeed other researchers may choose to define the regions in the same SSA plots according to Maslow's hierarchy of needs theory (Maslow, 1954) or Hackman and Oldman's (1976) JCT. The choice is strictly arbitrary, but the partitioning should be done such that the inclusion of every variable in a particular region is supported by theory suggesting that the variable can be so defined.

3. Methodology

3.1. Data Collection—Focus Groups

We used focus groups as our main approach to collecting data on SPI motivators. Focus groups are a tried and tested technique in the social sciences. They involve assembling small groups of peers to discuss particular topics. Discussion is largely free flowing, but is discreetly orchestrated by a researcher. Focus groups have the advantage of being a flexible but sophisticated research technique that allows soft issues to be explored. Conventional software engineering research methods are usually ineffective at collecting the rich data that focus groups generate. Indeed focus groups have been described as 'a way to better understand how people feel and think about an issue' (Krueger and Casey, 2000). Also, Morgan and Krueger (1993) say that 'the comparisons participants make among each other's experiences and opinions are a valuable source of insights into complex behaviours and motivations'. Focus groups are, therefore, an ideal vehicle for exploring motivation for SPI.

Focus groups also elicit data that allows a better understanding of the differences between groups of people (Krueger and Casey, 2000). This makes them ideal in our study where we are interested in exploring the different motivations of practitioner groups.

From September 1999 to March 2000 we visited 13 software companies and conducted 49 focus groups. The groups comprised between four and six members and altogether 200 software practitioners took part in this study. Participating companies were selected from a larger sample of companies who responded to a detailed questionnaire giving broad information about their software development activities and company demographics. They were chosen to provide our research project with a cross-section of company maturity levels and software applications.

Each focus group lasted approximately 90 min. Based on our previous experiences of using focus groups (Hall and Wilson, 1997), we separated senior managers, project managers and developers into separate focus groups. Kitson and Masters (1993) in their study of software process assessment in 59 companies collected data from similar staff groups. They report that because they were collecting data from managers and practitioners who were confronting real issues on a daily basis, they had high confidence in the accuracy and validity of the data. In total, we conducted the following focus groups: 21 developer focus groups, 16 project manager focus groups, and 12 senior manager focus groups.
Two of the questions we asked focus groups to discuss were:

– What are the motivators for SPI in your company?

– What would motivate you to support SPI?

Practitioner responses to the questions were audio recorded and subsequently transcribed.

3.2. Data Analysis—SSA

We followed the SSA steps outlined in Section 2.3 to establish a content category dictionary of all SPI motivator variables. Inter-rater reliability was established for this and the dictionary is provided in Appendix A. We used this dictionary to produce data matrices for each of the three practitioner groups. The matrices are provided in Appendix B. The SSA tool used these matrices to calculate multivariate correlations between motivators. Finally, the SSA tool used these correlations to plot the geometric distances between motivators.

3.3. Study Limitations

We note the following issues in our study methods:

- The data we have collected characterises practitioners' perceptions of their SPI motivations. We have not verified these perceptions directly. It may be that what practitioners believe motivates them is not what actually motivates them. Furthermore, what people say motivates them may not be accurate.
- The data we collected does not show the strength of feeling about a motivation within individual focus groups. If a motivation came up in a group it was recorded. So one person may have cited a motivation and whether everyone agreed or no one else agreed it was recorded. We have measured the importance of a motivation by it occurring in multiple groups.
- Although nearly 200 practitioners were involved in this study our data collection points were the 49 focus groups. This means that our data set is relatively small. We chose not to scale up the data points to reflect participants rather than focus groups, but this means that it is sometimes difficult to generate significant relationships with such a relatively small data set.
- Although MDS is a technique to plot data in 3D we present 2D representations of these plots. This means that some of the relationships are difficult to see from our plots.

4. Results and Discussion

We present results of our study and discuss the implication of these results in this section.

4.1. Developer Motivators

We conducted 21 developer focus groups. Appendix D shows the SSA plot of the motivations cited in these focus groups. It shows that using our two motivation facets (see Appendix C) developer motivations can be partitioned into two main regions: punitive motivators and rewarding motivators. Punitive motivators are easier to implement, but their effect is less lasting, whereas rewarding motivators require more ingenuity to devise but do have a longer lasting effect.

The plot shows that there are only two motivators in the punitive region and these are closely associated: compulsory (7) and standardisation (8). When developers indicate that they are motivated by compulsory SPI practices (i.e., they will do SPI work only when they are forced to), they are also highly likely to suggest standardisation as a motivator. Standardisation is classified as punitive because, arguably, it does not reward individual flair and constricts practitioners to work in a particular way. It is interesting to note how closely these two motivators co-occur and is a result that merits future investigation.

The rewarding region occupies the vast majority of the plot. Within this region, the closest association between motivators is between top-down commitment (4) and process ownership (13). Both of these motivators are reported in the literature to be critical to SPI success (Paulk et al., 1994b). Our results provide corroboration that these factors are indeed important to developers in that they are highly likely to cite them in tandem. This result may suggest that developers relate top-down commitment to process ownership.

The plot also shows a close relationship between shared best practice (5) and job satisfaction (11). Although shared best practice and job satisfaction are directly related to the jobs developers actually do, each has a different basis—one is extrinsic and the other intrinsic to the job. However the relationship between these motivators is not too surprising and the importance of such factors to developers has been commented on in other studies (Khalil et al., 1997).

Two other very closely associated rewarding motivators for developers seem to be visible success (1) and communication (16). We suggest that because these two motivators relate to information sharing, developers are highly likely to cite them together. Hence the close association depicted in the plot. After all, providing evidence of SPI success and effectively communicating SPI procedures within the company seem to be very closely related issues.

Overall, the developer motivators are plotted closer together than the motivators in the other two SSA's. This suggests that there are closer relationships between the factors that motivate developers for SPI than there are for the other two practitioner groups.

The plot in Appendix D also shows that some of the more distant associations are those between:

– Eliminates bureaucracy (12) and reward schemes (14)

– Bottom-up initiative (3) and maintainable processes (19)

– Phased introduction (9) and eliminates bureaucracy (12)

4.2. Project Manager Motivators

We conducted 16 project manager focus groups. Appendix E shows the SSA plot of the motivations cited in these focus groups. It shows that project manager motivators can also be partitioned into punitive and rewarding regions. The plot shows that there are generally fewer close associations between the motivators cited by project managers.

The association between project managers' motivators is disparate. In fact the plot from project managers' motivators shows the weakest association between motivators than for either of the other two groups. This does not mean that project manager motivators are any less important, it simply indicates that association between the individual motivators is not as strong.

However, autonomy (8) and top-down commitment (13) do seem very closely related. The two motivators have exactly the same profile, hence end up as a perfect fit (though autonomy is shielded from view on the plot being directly behind top-down commitment).

This association is the same as that presented in the developers' plot between process ownership and top-down commitment. We suggest that autonomy to project managers is similar to process ownership to developers. So that in both cases, project managers and developers are citing a close relationship between the factors that give them the authority to conduct SPI and at the same time imbues them with the responsibility to carry out that work.

Other associations are between:

– Reward schemes (11) and process ownership (12)
– Resources (1), maintainable processes (2) and empowerment (4)
– Visible success (9), knowledgeable team leaders (10), top-down commitment (13) and autonomy (8)
– Easy processes (6) and communication (7)

External audits (5) is the only punitive motivator in the plot. It also does not share particularly strong association with the other motivators in the plot. We find this result not surprising as external audits tends to have a different basis to the other motivators.

The least closely associated project manager motivators were top–down commitment (13)/autonomy (8) and saleability (14).

4.3. Senior Manager Motivators

We conducted 12 senior manager focus groups. Appendix F shows the SSA plot of the motivators cited in these focus groups. It shows that, with the exception of one outlier (task forces (7)) senior manager motivators partitioned into tangible and intangible regions to show the relationship between motivators within and across these regions. Tangible motivators are usually extrinsic to the job that practitioners do, whereas intangible motivators are intrinsic to practitioners' jobs and satisfying. Overall, the plot indicates that there are only few close associations between senior managers' motivators. These close associations appear amongst the intangible motivators. Most of senior managers' motivators however, are only loosely associated to each other.

The plot shows that taller hierarchies (6) and career prospects (8) have a close association. In fact the plot indicates that they are a near perfect fit—suggesting that they have very similar profiles. This suggests that senior managers are highly likely to cite these two motivators in tandem.

In fact, in our previous work we found that senior managers inaccurately predict these two as factors that will motivate developers (Baddoo and Hall, to appear in 2002). We suggest that this strong association being depicted by the SSA plot is one that may be derived from the fact that these two motivators are very similar because they are both distinctively dissimilar to any of the other motivators. Furthermore, other studies have found that developers are less motivated by such factors than other professionals (Nicholson et al., 1995).

The plot also shows a close association between maintainable processes (12) and feedback (11). We suggest that senior managers may be closely associating these two motivators on the basis that processes can only be truly maintainable with feedback.

The association of resources (10) with the other tangible motivators seems removed. This suggests that senior managers do not envisage a close relationship between resources and the other motivators. We speculate that this result may relate to the fact that senior managers have a different relationship to resources as reported previously (Baddoo and Hall, to appear in 2002).

Overall, there is a closer association between the motivators of senior managers than that of project managers.

4.4. Motivators and Process Maturity

In this study, although we considered process maturity levels in conjunction with practitioner motivators we found the sample size too small to generate any significant results. In future studies, using a larger sample size, we intend to explore the relationship between motivations based upon process maturity levels.

5. Conclusion

Overall we have shown that the application of MDS in analysing the motivations of practitioners for SPI is useful. Not only does the technique work well with the data but also it generates novel and sophisticated insights into the relationships between motivators.

Our most important finding, is that there is a strong relationship between many of the motivators cited by developers. This finding should give SPI managers better insight into how to amalgamate factors in order to effectively motivate developers for SPI.

Overall, our analysis of the geometric association of motivators provides insight into the relationship between the factors that actually motivate developers, project managers and senior managers to participate in SPI. Although our analysis is not exhaustive, it suggests relationships that SPI managers should consider when designing SPI implementation strategies. Although some of our findings confirm the critical SPI success factors cited in the literature, we also present a novel analysis of some motivators. We present our findings as guidelines for SPI managers.

5.1. Guidelines for SPI Managers

5.1.1. Communication

Complex relationships seem to exist between motivators related to communication. Feedback relates to maintaining processes; communication relates to visible success; shared best practice relates to job satisfaction. The relationship between communication and improvement is strong throughout the three groups, but particularly in developer and senior manager groups. SPI managers should use the effectiveness of good communication to improve SPI implementation.

5.1.2. Empowerment

Many of the motivations cited by developers and project managers relate to aspects of empowerment. Developers relate top-down commitment to process ownership; project managers relate top-down commitment to autonomy. These findings suggest that issues of empowerment are motivators that underpin many aspects of SPI. Again, the widespread relevance of this set of motivators means that SPI managers need to address these in SPI implementation strategies.

5.1.3. Career Motivations

A number of career-orientated senior manager motivators are clustered. This suggests that senior managers focus on career motivators. These motivators relate not only to themselves as senior managers but are also projected onto the other two staff groups by senior managers. SPI managers should use these motivators when 'selling' SPI to senior managers.

5.1.4. Resources and External Audits

Although resources and external audits appear as SPI motivators they are not clearly associated with other motivators. Such isolated motivators are probably more difficult to use effectively as motivators than others that are more inter-related. SPI managers should therefore focus attention on other motivators in the first instance.

These insights into the motivations of practitioners should allow SPI managers to more easily identify effective approaches to the implementation of SPI. Our data shows close relationships between motivations that SPI managers should address in their implementation plans. Our data also suggests that there are different likelihoods of co-occurrence of motivators for different practitioner groups. This suggests that SPI implementation plans should have several tailored strands to account for these differences.

Acknowledgments

We are sincerely grateful to all the companies and practitioners (who, for reasons of confidentiality, must remain anonymous) for their participation in this project. We are also grateful to David Wilson from the University of Technology, Sydney, Austen Rainer and Sarah Beecham from the University of Hertfordshire for their contribution to this project. The project is funded by the UK's Engineering and Physical Science Research Council, under grant number EPSRC GR/L91962.

Appendix A

Content Category Dictionary

Plot codes			Motivator	Definition
D18			Automation	Tools to eliminate paper work.
	P8		Autonomy	Enables practitioners carry on present roles without prescribing specific roles for them.
D3			Bottom-up initiatives	Developers have input into the design and planning of SPI.
		S8	Career prospects	Improves career prospects.
D16	P7		Communication	Improved communication about SPI.
D7			Compulsory	SPI practice is made mandatory.
		S3	Cost beneficial	Favourable cost/revenue ratio through SPI.
D10			Critical mass	The presence of sufficient number of people who want to see SPI happen.
	P6		Easy processes	Processes that are easy to understand and follow.

(continued)

Appendix A. (*continued*)

Plot codes			Motivator	Definition
D12			Eliminates bureaucracy	Eliminates spending time on bureaucratic processes.
	P4		Empowerment	Practices within the SPI programme that empower staff to take decisions on changing processes.
	P5		External audits	Stipulation by some external body to maintain SPI practices. For example, certification bodies.
D2		S11	Feedback	Feedback, both from management and from customers.
D11			Job satisfaction	Practitioners get job satisfaction from producing good quality process and high quality products.
		S1	Justifiable benefits	The ability to justify the long-term benefits of SPI.
	P10		Knowledgeable team leaders	Having team leaders who know about software engineering. i.e. possess technical backgrounds.
D19	P2	S12	Maintainable processes	Processes that are changeable and maintainable.
		S5	Meeting targets	SPI practice doesn't deviate the company from meeting commercial and project goals.
D9			Phased introduction	SPI is introduced through small and incremental implementation.
D13	P12	S9	Process ownership	Practitioners own and therefore are able to change processes.
	P3		Reduced admin.	SPI leads to reduced administration.
D17	P1	S10	Resources	Sufficient time and resources allocated to SPI.
D14	P11	S4	Reward schemes	Practitioners are rewarded for SPI work.
	P14		Saleability	The perception that SPI will lead to more saleable job market skills.
D5			Shared best practice	Best practice is shared across teams and departments in companies.
D15			SPI forum	Creating a forum where SPI ideas can be discussed.
D8			Standardisation	SPI makes practitioners work in a standardised way.
		S6	Taller hierarchy	Taller company hierarchies which create more opportunity for promotion.
		S7	Task forces	Using task forces to drive improvement.
D4	P13		Top-down commitment	Visible senior management support for SPI.
D6			Training	Training provided to practitioners in SPI practices.
D1	P9	S2	Visible success	Evidence of the benefits of SPI.

Appendix B

Data Matrices

Motivators Cited by Developers

Company	Group	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19
A1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
A3a	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0
	2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A3b	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A4	1	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0
A5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A6	1	1	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
	2	0	0	1	1	1	1	0	0	0	0	0	0	0	1	1	1	1	0	0
A7	1	1	1	1	0	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0
	2	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0
A8	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A9	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A10	1	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A11	1	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	1
A12	1	0	1	1	1	1	0	0	0	1	0	0	0	1	0	1	1	1	0	0
	2	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0

Motivators Cited by Project Managers

Company	Group	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14
A1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A2	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0
A3a	1	1	0	0	0	0	0	0	0	0	1	0	1	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A3b	1	0	0	1	1	0	0	0	0	0	0	1	1	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A5	1	0	0	0	0	1	1	1	0	1	0	0	0	0	0
A6	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0
A7	1	1	0	0	1	0	0	1	0	0	0	0	1	0	0
	2	1	1	0	0	0	0	0	0	0	0	1	0	0	0
A8	1	0	0	0	0	0	0	0	1	1	0	1	0	1	0
	2	0	1	0	0	0	1	0	0	0	0	0	0	0	1
A9	1	1	0	0	0	0	0	0	0	1	1	0	0	0	0
A11	1	0	0	0	0	0	0	0	0	1	1	0	1	0	0
A12	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Motivators Cited by Senior Managers

Company	Group	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
A1	1	0	0	0	0	0	0	0	0	0	0	0	0
A2	1	0	0	0	0	0	0	0	0	0	0	0	0
A3a	1	0	0	0	0	0	0	0	0	0	0	0	0
A3b	1	0	0	0	0	0	0	0	0	0	0	0	0
A4	1	0	0	0	0	1	0	0	0	0	1	0	0
A5	1	0	1	0	0	0	0	1	0	0	0	0	0
A6	1	0	0	0	0	0	0	0	0	0	0	0	0
A7	1	1	1	1	1	1	0	0	0	0	0	0	0
A8	1	0	1	0	0	0	1	0	1	1	0	0	0
A9	1	0	0	1	0	0	0	0	0	0	1	0	0
A10	1	0	0	0	0	1	0	0	0	1	0	1	1
A12	1	0	0	0	1	0	0	0	0	0	0	0	0

Appendix C

Motivation Facets

Stimulus response theory (SRT) examines the punitive and rewarding stimuli that modify behaviour (Skinner, 1976). Punitive stimuli are easier to apply and produce responses in the short term. Rewarding stimuli are more difficult to apply and require more ingenuity to devise, but have a longer term effect. Our interest in SRT is to understand how SPI managers can implement SPI to generate long-term support from practitioners.

Motivation-hygiene theory (Herzberg et al., 1959) recognises the existence of extrinsic and intrinsic factors in work. Extrinsic factors are the conditions surrounding the job, like company policy and peer relationships—these are known as maintenance factors. Intrinsic factors are those that determine the nature of the job itself and how individuals feel about the job, like responsibility and achievement—these are known as motivation factors.

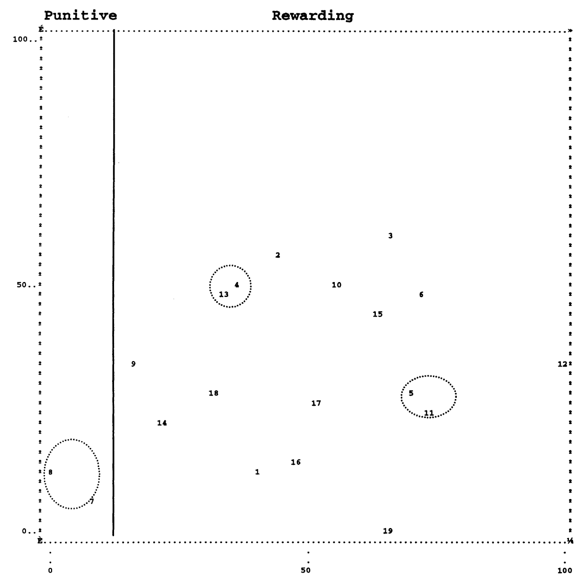
The essence of Herzberg's theory is that an individual's satisfaction can only be attained when there are motivational conditions at work. Hence the extrinsic conditions in a job alone do not make the worker satisfied. They only make them '*not dissatisfied*' (Couger and Zawacki, 1980). For practitioners to be really satisfied, there also needs to be intrinsic conditions or motivational factors.

Appendix D

SSA Plot of Developers Motivators

This SSA of developers' motivators for SPI has been partitioned according to Skinner's SRT which defines motivators as either rewarding or punitive.

The partition separates the punitive region where the two punitive motivators: compulsory (7) and standardisation (8) co-occur from the remaining motivators which are rewarding.



Code	Motivator
D1	Visible success
D2	Feedback
D3	Bottom-up initiatives
D4	Top-down commitment
D5	Shared best practice
D6	Training
D7	Compulsory
D8	Standardisation
D9	Phased introduction
D10	Critical mass
D11	Job satisfaction
D12	Eliminates bureaucracy
D13	Process ownership
D14	Reward schemes
D15	SPI forum
D16	Communication
D17	Resources
D18	Automation
D19	Maintainable processes

The plot shows the relationship between individual motivators within their regions/components/facets and also the relationship across regions.

Appendix E

SSA Plot of Project Manager Motivators

This SSA of project managers' motivators for SPI has been partitioned according to Skinner's SRT which defines each motivator as either rewarding or punitive.

The partition separates the punitive region where the one punitive motivator: external audits (5) occurs from the remaining motivators which are rewarding.

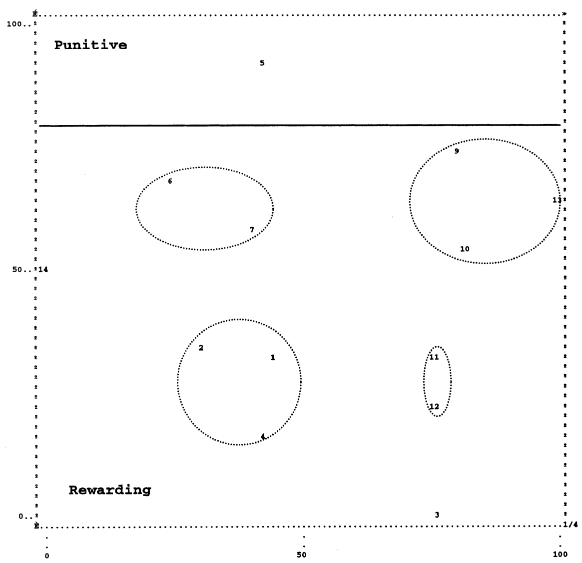
The plot shows the relationship between individual motivators within their regions/components/facets and also the relationship across regions.

Appendix F

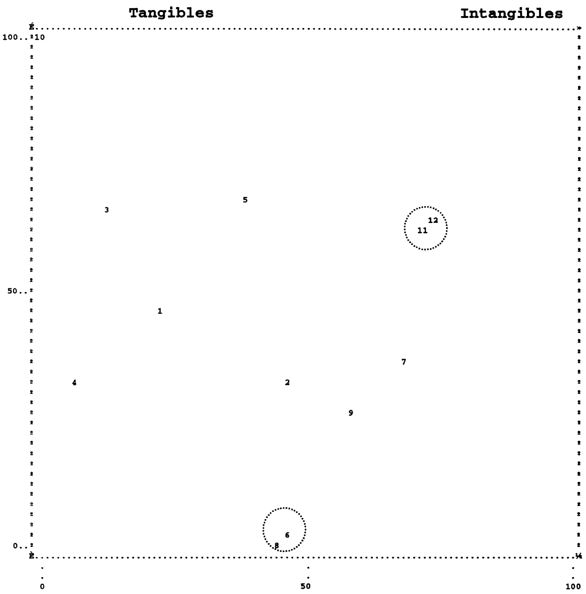
SSA Plot of Senior Manager Motivators

This SSA of senior managers' motivators for SPI has been partitioned according to the concept of defining motivators are either tangible or intangible.

The partition separates the motivators that are tangible: justifiable benefits (1), visible success (2), cost beneficial (3).



Code	Motivator
P1	Resources
P2	Maintainable processes
P3	Reduced admin
P4	Empowerment
P5	External audits
P6	Easy processes
P7	Communication
P8	Autonomy
P9	Visible success
P10	Knowledgeable team leaders
P11	Reward schemes
P12	Process ownership
P13	Top down commitment
P14	Saleability



Code	Motivator
S1	Justifiable benefits
S2	Visible success
S3	Cost beneficial
S4	Reward schemes
S5	Meeting targets
S6	Taller hierarchy
S7	Task forces
S8	Career prospects
S9	Process ownership
S10	Resources
S11	Feedback
S12	Maintainable Processes

References

Baddoo, N., and Hall, T. to appear in 2002. Motivators of software process improvement: an analysis of practitioners' views. *Journal of Systems and Software*.

Baddoo, N., Hall T., and Wilson, D. 2000. Implementing a people focused SPI programme. *Proceedings of the 11th European Software Control And Metrics Conference and the Third SCOPE Conference On Software Product Quality*, Munich, 373–381, April 18–20.

Canter, D., and Heritage, R. 1990. A multivariate model of sexual offence behaviour: developments in 'offence profiling'. *Journal of Forensic Psychiatry* 1: 185–212.

Couger, J. D., and Zawacki, R. A. 1980. *Motivating and Managing Computer Personnel*. New York: John Wiley and Sons.

Donald, I. 1994. The structure of office worker's experience of organisational environments. *Journal of Occupational and Organisational Psychology* 67(3): 241–258.

Guttman, L. 1968. A general nonmetric technique for finding the smallest coordinate space for a configuration of points. *Psychometrika* 33: 469–506.

Guttman, R., and Greenbaum, C. W. 1998. Facet theory: it's development and current status. *European Psychologist*, 3(1): 13–36.

Hackman, J. R., and Oldman G. R. 1976. *Motivation Through the Design of Work*: *Test of a Theory*. New York: Academic Press.

Hall, T., and Wilson, D. 1997. Views of software quality: a field report. *IEE Proceedings on Software Engineering* 114(2): 111–118.

Herbsleb, J. D., and Goldenson, D. R. 1996. A systematic survey of CMM experience and results. *18th International Conference on Software Engineering — ICSE*. Berlin, Germany, 323–330, 25–29 March.

Herzberg, F., Mausner, B., and Snyderman, B. B. 1959. *The Motivation to Work*. New York. John Wiley and Sons Inc.

Humphrey, W.S. 1989. *Managing the Software Process*. Reading, MA: Addison-Wesley.

Kaltio, T., and Kinnula, A. 2000. Deploying the defined software process. *Software Process—Improvement and Practice* **5**: 65–83.

Khalil, O. E. M., Zawacki, R. A., Zawacki, P. A., and Selim A. 1997. *What Motivates Egyptian IS Managers and Personnel: Some Preliminary Results*. SIGCPR 97.

Kitson, D. H., and Masters, S. M. 1993. *An Analysis of SEI Software Process Assessment Results: 1987–1991*. 15th International Conference on Software Engineering. Baltimore, Maryland, May 17–21.

Krueger, R. A., and Casey, M. A. 2000. *Focus Groups: A Practical Guide for Applied Research*. (3rd Ed.) CA, London: Sage Publications.

Maslow, A. 1954. *Motivation and Personality*. New York: Harper & Row.

Mead, A. 1992. Review of the development of multidimensional scaling methods. *The Statistician* 41(1): 27–39.

Morgan, D. L. and Krueger, R. A. 1993. When to use focus groups and why. In: D. L. Morgan (ed.): *Successful Focus Groups: Advancing the State of the Art*. CA, London: Sage. pp. 3–19.

Nicholson, I., Holmes, D., Richards, J., and Warden, R. 1995. Motivation of quality practitioners. In: M.M. Ross, C.A. Brebbia, G. Staples, and J. Stapleton (eds.): *Quality Management III*. Vol. 1. Southampton, UK Computational Mechanics Publications. pp. 347–357.

Paulk, M., Chrissis, M., Curtis, B., and Weber, C. 1994a. The capability maturity model: guidelines for improving the software process. Reading, MA: Addison-Wesley.

Paulk, M. C., and Chrissis, M. B. 2000. *The November 1999 High Maturity Workshop*. Software Engineering Institute, Carnegie Mellon University.

Paulk, M., Weber, C., Curtis, B., and Chrissis, M. 1994b. A high maturity example: space shuttle onboard software, the capability maturity model: guidelines for improving software process. Reading, MA: Addison–Wesley.

Shye, S., Elizur, D., and Hoffman, M. 1994. *Introduction to Facet Theory*. Thousand Oaks, CA: Sage.

Skinner, B. F. 1976. *Walden Two*. New York: Macmillan.

Nathan Baddoo is a Ph.D. researcher in the Department of Computer Science, University of Hertfordshire, U.K. Nathan spent a year in the information systems department of Nissan Manufacturing, U.K. as a systems developer and subsequently conducted an investigation into the use of CASE technology in Nissan. His general interests are in the areas of software quality and software process improvement. He has published and presented material at seminars, workshops and conferences, both in the U.K. and in the U.S.

Tracy Hall is a principal lecturer at the Department of Computer Science, University of Hertfordshire, U.K. Tracy is an active researcher in the area of software measurement and software process improvement. She has many publications in this area. She holds a Ph.D. from the Centre for Software Reliability, City University, London.