Empirical Processes for Object-Oriented Reengineering

Y. Wang, I. Court, M. Ross, G. Staples and G. King

Research Centre for Systems Engineering
Southampton Institute, Southampton SO14 0YN, UK
Tel: +44 1703 319773, Fax: +44 1703 334441
Email: wang_s@solent.ac.uk or yingxu.wang@comlab.oxford.ac.uk

Abstract: To seek the practical foundation of base process activities (BPAs) in the software industry and to support the research in modelling the software processes, a survey has been designed and conducted worldwide. Seven processes with 63 BPAs for object-oriented reengineering have been modelled and analyzed as a subset of the survey. This paper reports the survey findings on the BPAs for OO reengineering processes. Each BPA in the reengineering processes is benchmarked on the attributes of mean importance and the ratios of significance, practice and effectiveness. Based on the benchmark data, the OO reengineering processes in a software organization can be diagnosed and evaluated quantitatively, and process improvement opportunities can be identified and prioritized based on the importance and effectiveness of the BPAs within the processes.

Key words: Software engineering, OO reengineering, software process, base process activities, survey, SPRM

1. Introduction

The processes for OO reengineering have been modelled in a software process reference model (SPRM) [1-3]. A current worldwide survey of base process activities (BPAs) towards software process excellence [4-5] has been conducted by the Research Centre for Systems Engineering at Southampton Institute UK, in collaboration with the European Software Institute (ESI), IVF Centre for Software Engineering and BCS QSig. Based on the SPRM and the survey, this paper reports the findings and benchmarks on OO reengineering processes. The paper contains detailed and quantitative evaluation on a set of 63 BPAs in seven OO reengineering processes. For each BPA, a set of practical attributes, such as the mean weighted importance in process and the ratios of significance, practice and effectiveness, are derived and benchmarked.

2. Model of OO reengineering processes

Based on the structure of the SPRM, a summary of seven empirical processes with 63 BPAs are identified for OO reengineering in the software industry. These OO reengineering processes are: a) requirement review, b) maintenance reengineering, c) process database/library, d) configuration control, e) change control, f) defect control, and g) project risk avoidance.

The process structure for OO reengineering is modelled in Table 1. The numbers of BPAs within a reengineering process are listed in the column labelled by BPA. Further refinement of all the BPAs will be development in Section 3.

Table 1. Structure of the OO reengineering processes

No.	Process	BPA
1	Requirement review	9
2	Reengineering maintenance	10
3	Process database/library	6
4	Configuration control	8
5	Change control	9
6	Defect control	10
7	Project risk avoidance	11
Total	7	63

3. Benchmark of the reengineering processes

For the seven OO reengineering processes modelled in Section 2, the 63 BPAs and their benchmarked practical attributes are given in Table 2 based on the survey [4-5].

Table 2 shows, for each BPA, the mean importance weighting (*W*), the percentage of organizations rating the BPA highly (i.e. weighting ≥ 3) significant (r_w), the percentage of organizations that used the BPA (r_p), and the percentage that rated it as effective (r_e). The final column φ, the characteristic value, uses a combination of these three percentages to provide a combined indication of the BPA's significance, practical use and effectiveness as defined in Formula (1).

$$\varphi = [(r_w*100) * (r_p*100) * (r_e*100)] * 100\% \qquad (1)$$

The higher the value of φ, the more important and effective the BPA in practice; and vice versa. Therefore φ can be used to index the importance and effectiveness of a BPA in practice.

Table 2. Benchmark of OO reengineering processes

No.	BPAs	W [0 .. 5]	r_w (%)	r_p (%)	r_e (%)	φ (%)
1	Requirement review					
1.1	Specification verification	4.2	94.1	64.7	93.3	56.8
1.2	Formal review requirements	4.0	93.8	81.3	92.9	70.7
1.3	Review statutory requirements	3.2	83.3	61.5	81.8	42.0
1.4	Customer acceptance specifications	3.9	92.9	64.3	81.8	48.8
1.5	Adopt specification verification tools	2.5	66.7	13.3	80.0	7.1
1.6	Update requirements for next iteration	3.8	81.3	80.0	80.0	52.0
1.7	Agree on requirements	4.3	93.8	93.8	93.3	82.0
1.8	Manage requirements changes	4.1	94.1	75.0	87.5	61.8
1.9	Maintain requirements traceability	3.8	93.8	62.5	80.0	46.9
2	Reengineering maintenance					
2.1	Determine maintenance requirements	4.0	92.9	78.6	92.3	67.3

2.2	Analyse user problem and enhancements	4.0	100	92.3	100	92.3
2.3	Determine modifications for next upgrade	3.7	83.3	75.0	90.9	56.8
2.4	Implement/test modifications	3.7	90.9	72.7	80.0	52.9
2.5	Update user system	3.9	90.9	50.0	77.8	35.4
2.6	Maintenance consistency with specifications	3.9	100	54.5	90.9	49.6
2.7	Maintain nonconforming products	3.5	76.9	58.3	80.0	35.9
2.8	Record nonconformance treatment	3.9	92.3	75.0	90.9	62.9
2.9	Adopt regression testing tools	3.5	84.6	33.3	81.8	23.1
2.10	Conduct regression testing	4.1	85.7	50.0	90.9	39.0
3	Process database/library					
3.1	Establish organisation's process library	3.1	68.8	40.0	78.6	21.6
3.2	Establish organisation's process database	3.1	73.3	28.6	72.7	15.2
3.3	Establish software reuse library	3.3	60.0	33.3	80.0	16.0
3.4	Establish organisation's metrics database	3.6	70.6	43.8	80.0	24.7
3.5	Establish operation manual library	3.1	73.3	61.5	92.3	41.7
3.6	Establish practice benchmark database	2.3	50.0	0	58.3	0
4	Configuration control					
4.1	Establish configuration management library	3.8	84.2	77.8	94.4	61.9
4.2	Adopt configuration management tools	3.8	93.3	53.3	84.6	42.1
4.3	Identify product's configuration	4.2	100	82.4	88.2	72.7
4.4	Maintain configuration item descriptions	3.9	93.3	71.4	78.6	52.4
4.5	Control change requests	4.4	100	88.2	100	88.2
4.6	Release control	4.3	100	81.3	87.5	71.1
4.7	Maintain configuration item history	3.9	94.1	68.8	80.0	51.8
4.8	Report configuration status	3.6	81.3	73.3	86.7	51.6
5	Change control					
5.1	Establish change requests/approval system	4.0	100	76.9	100	76.9
5.2	Control requirement change	4.1	100	71.4	85.7	61.2
5.3	Control design change	3.9	100	71.4	92.9	66.3
5.4	Control code change	3.8	93.3	78.6	92.9	68.1
5.5	Control test data change	3.3	73.3	57.1	84.6	35.5
5.6	Control environment change	3.0	78.6	53.8	81.8	34.6
5.7	Control schedule change	3.6	84.6	66.7	100	56.4
5.8	Control configuration change	3.8	82.4	73.3	86.7	52.3
4.9	Adopt change control tools	2.9	60.0	35.7	76.9	16.5
6	Defect control					
6.1	Plan defect prevention	3.8	83.3	52.9	78.6	34.7
6.2	Defect reporting and record	4.2	88.9	76.5	93.8	63.7
6.3	Defect causal analysis	4.1	95.0	80.0	94.7	72.0
6.4	Propose process change for defect prevention	4.2	89.5	61.1	80.0	43.7
6.5	Track problem report	4.3	100	86.7	100	86.7
6.6	Prioritise problems	3.9	83.3	76.5	93.8	59.7
6.7	Determine resolutions	3.8	88.2	81.3	93.3	66.9
6.8	Correct defects	4.4	100	100	100	100
6.9	Review defect corrections	3.6	77.8	62.5	78.6	38.2
6.10	Distribute correction results	3.6	82.4	76.5	82.4	51.9
7	Project risk avoidance					
7.1	Identify project reengineering risks	3.8	88.2	50.0	86.7	38.2

7.2	Establish risk management scope	3.3	78.6	30.8	66.7	16.1
7.3	Identify unstable specification related risks	3.3	81.3	43.8	69.2	24.6
7.4	Identify process change related risks	3.1	73.3	28.6	63.6	13.3
7.5	Identify market related risks	3.8	93.3	64.3	83.3	50.0
7.6	Analyse and prioritise risks	3.4	73.3	40.0	71.4	21.0
7.7	Develop mitigation strategies	3.1	73.3	40.0	58.3	17.1
7.8	Define risk metrics for probability/impact	2.9	75.0	20.0	61.5	9.2
7.9	Implement mitigation strategies	3.1	78.6	28.6	63.6	14.3
7.10	Assess risk mitigation activities	2.9	68.6	33.3	58.3	13.4
7.11	Take corrective actions for identified risk	4.0	93.3	73.3	73.3	50.2

4. Analysis of the reengineering processes

As shown in Table 3 and Fig.1, it is interesting to find in the OO reengineering processes, that:

❑ On the mean weighted importance of the BPAs in the OO reengineering processes, 92.1% of the BPAs are heavily weighted with 63.5% at weight scale 3.0-3.99 and 28.6% at weight scale 4.0-4.99. There are only about 1/13 BPAs perceived to be not very important;

❑ On the ratio of significance of the BPAs, 42.9% of the BPAs are weighted extremely significant, 47.6% are very significant, 9.5% are fairly significant, and no BPAs are weighted not significant;

❑ On the ratio of practice of the BPAs, 4.8% BPAs have got extremely high application rate, 39.9% BPAs have very high application rate, and 30.2% have fairly high application rate. It is noteworthy that 25.4% BPAs are less practised; and

❑ On the ratio of effectiveness of the BPAs, 34.9% of the BPAs are weighted extremely effective, 52.4% are very effective, and 12.7% are fairly effective. No BPAs in the set are found not effective.

Table 3. General statistics of the survey findings on OO reengineering processes

	Mean weighted importance (W)						**Ratio of significance (r_w)**				**Ratio of Practice (r_p)**				**Ratio of effectiveness (r_e)**			
	0	1	2	3	4	5	E	V	F	N	E	V	F	N	E	V	F	N
No. of BPAs	0	0	5	40	18	0	27	30	6	0	3	25	19	16	22	33	8	0
Ratio (%)	0	0	7.9	63.5	28.6	0	42.9	47.6	9.5	0	4.8	39.9	30.2	25.4	34.9	52.4	12.7	0

Note: E - Extremely (⇩90%), V - very (70-89%), F - fairly (50-69%), and N - not (<50%)

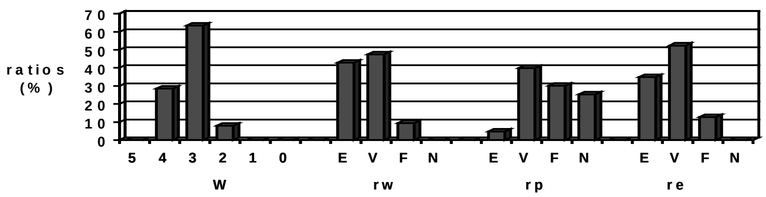

Fig.1 Overview of the survey findings on OO reengineering process

5. Conclusions

This paper reports the survey findings on a set of empirical software OO reengineering processes based on the benchmark of the worldwide survey on BPAs towards software process excellence [4-5]. In this paper, a model of the OO reengineering processes is developed with regard to the SPRM [1]. Based on the survey, detailed benchmark of the 63 BPAs in the seven OO reengineering processes has been obtained for the practical attributes, such as the mean weighted importance, and ratios of significant, practice and effectiveness in the OO reengineering processes.

The empirical model and survey results on the OO reengineering processes provide a set of valuable statistical data. The benchmark is useful for modelling and feature-identifying the fundamental software process activities in OO reengineering practices; for planning a specific OO reengineering project; and for identifying process improvement opportunities for the existing OO reengineering processes.

Acknowledgements

The authors would like to acknowledge the support of the European Software Institute, IVF Centre for software Engineering, and BCS QSig.

References

[1] Wang, Y., Court, I., Ross, M., Staples, G. and King, G. [1996], Towards a Software Process Reference Model (SPRM*), Proceedings of International Conference on Software Process Improvement (SPI'96),* Brighton UK, pp.145-166.

[2] Wang, Y., Court, I., Ross, M., Staples, G., King, G. and Dorling, A. [1997], Quantitative Analysis of Compatibility and Correlation of the Current SPA/SPI Models, Proceedings of the 3rd IEEE International Symposium on Software Engineering Standards (ISESS'97), IEEE Computer Society Press, USA, pp.36-56.

[3] Wang, Y., Court, I., Ross, M., Staples, G., King, G. and Dorling, A. [1997], Quantitative Evaluation of the SPICE, CMM, ISO 9000 and BOOTSTRAP, Proceedings of the 3rd IEEE International Symposium on Software Engineering Standards (ISESS'97), IEEE Computer Society Press, USA, pp.57-68.

[4] Wang, Y., Court, I., Ross, M., Staples, G., King, G. and A. Dorling [1997], Design of the Questionnaires for the Survey of Base Process Activities Towards Software Process Excellence, *Technical Report SI-RCSE-WANG96-DATA03*, pp.1-36.

[5] Wang, Y., Court, I., Ross, M., Staples, G., King, G. and A. Dorling [1997], Analysis Report of the Survey of Base Process Activities Towards Software Process Excellence, *Technical Report SI-RCSE-WANG97-PROC260*, pp.1-27.