On the Correlation between Process Model Metrics and Errors

Jan Mendling[1,2] **Gustaf Neumann**[2] **Wil van der Aalst**[3]

[1] BPM Cluster, Faculty of Information Technology
Queensland University of Technology
Email: j.mendling@qut.edu.au

[2] Institute of Information Systems and New Media
Vienna University of Economics and Business Administration,
Email: neumann@wu-wien.ac.at

[3] Department of Mathematics and Computer Science,
Eindhoven University of Technology,
Email: w.m.p.v.d.aalst@tue.nl

Abstract

Business process models play an important role for the management, design, and improvement of process organizations and process-aware information systems. Despite the extensive application of process modeling in practice there are hardly empirical results available on quality aspects of process models. This paper aims to advance the understanding of this matter by analyzing the connection between formal errors (such as deadlocks) and a set of metrics that capture various structural and behavioral aspects of a process model. In particular we discuss the hypothetical relationship between errors and metrics, and provide a validation of correlation based on an extensive sample of EPC process models from practice. The strong connection between metrics and errors has considerable consequences for the design of future modeling guidelines and modeling tools.

1 Introduction

Even though workflow and process modeling have been used extensively over the past 30 years, we know surprisingly little about the act of modeling and which factors contribute to a "good" process model in terms of error probability. This observation contrasts the large body of knowledge that is available for the formal analysis and verification of desirable properties, in particular for Petri nets. While conceptual work was conducted on guidelines and quality frameworks (see Lindland et al. 1994, Becker et al. 2000, Hoppenbrouwers et al. 2005, Krogstie et al. 2006), there is clearly a need for an empirical research agenda to acquire new insights on quality (see Moody 2005) and usage aspects (see Davies et al. 2006) of process modeling.

A recent study provides evidence that larger process models from practice tend to have more formal flaws (such as e.g. deadlocks) than smaller models (Mendling et al. 2006; 2007b). One obvious hypothesis related to this phenomenon would be that human modelers loose track of the interrelations of large and complex models due to their limited cognitive capabilities (see Simon 1996), and then introduce errors that they would not insert in a small model. Yet, there are further factors beyond simple count metrics such as the degrees of sequentiality, concurrency, or structuredness that need to be considered (Mendling 2007). Against this background, the paper provides the following two contributions. First, we introduce a tool-based approach for detecting errors as calculating metrics for Event-driven Process Chains (EPCs), a popular business process modeling language. Second, we utilize an extensive sample of EPC models from practice to analyze the statistical connection between errors and metrics. Both these contributions relate to the formal correctness of the process model as a design artifact. Validation aspects with respect to the content of a process model, human understandability issues, ease of use of the modeling language, and modeling pragmatics are also closely related to quality, but they are not considered here.

The remainder of the paper is structured as follows. In Section 2 we give a brief overview of the error detection techniques that we utilize for our analysis and which kind of metrics we calculate. In Section 3 we introduce a sample of 2003 EPCs from practice that we use to investigate the connection between errors and metrics. Moreover, we provide disaggregated descriptive statistics, and determine the correlation between errors and metrics. Section 4 discusses our findings in the light of related research before Section 5 concludes the paper.

2 Error Detection and Metrics Calculation

The Event-driven Process Chain (EPC) is a business process modeling language for representing temporal and logical dependencies of activities in a business process (see Keller et al. 1992). EPCs offer *function type* elements to capture activities of a process and *event type* elements describing pre- and post-conditions of functions. Furthermore, there are three kinds of *connector types* including AND (symbol $\wedge$), OR (symbol $\vee$), and XOR (symbol $\times$) for the definition of complex routing rules. Connectors have either multiple incoming and one outgoing arc (join connectors) or one incoming and multiple outgoing arcs (split connectors). The informal (or intended) semantics of an EPC can be described as follows. The AND-split activates all subsequent branches in concurrency. The XOR-split represents a choice between one of alternative branches. The OR-split triggers one, two or up to all of multiple branches based on conditions. In both cases of the XOR- and OR-split, the activation conditions are given in events subsequent to the connector. The AND-join waits for all incoming branches to complete, then it propagates control to the subsequent EPC element. The XOR-join merges alternative branches. The OR-join synchronizes all active incoming branches. This feature is called non-locality since the state of all transitive predecessor nodes has to be considered. Regarding the connectors EPCs are quite similar to BPMN (OMG, ed. 2006) and YAWL (Aalst and Hofstede 2005).

Recently, EPC semantics have been formalized and there is tool support for the verification of EPC soundness (see Mendling and Aalst 2007). We will use two complementary tools to test whether an EPC is sound (has no errors) or unsound (has errors). In a first step, we use *xoEPC*, a batch program written in XOTcl (Neumann and Zdun 2000). It applies a set of reduction rules on the input EPC and calculates an extensive set of metrics as described in Mendling (2007). *xoEPC* loads all *.xml files from the current directory and checks whether they are XML files using the format of ARIS Toolset (IDS Scheer AG 2001). If yes, the XML is processed. For each EPC model that has at least one event and one function *xoEPC* checks syntactical correctness and applies the reduction algorithm. If errors are encountered they are recorded in an XML file called errorresults.xml. This file also records the processing time of the reduction, metadata of the model as well as the size of the original and the size of the reduced EPC. All EPCs that cannot be reduced completely are written to the reducedEPCs.epml file in EPML format (Mendling and Nüttgens 2006). Furthermore, the errorresults.xml file is transformed to an HTML table.

In a second step, the EPCs in the reducedEPCs file are further analyzed with the ProM framework (Verbeek et al. 2006). Since ProM can load EPML files it complements *xoEPC*. In ProM there is a conversion plug-in for calculating the reachability graph of an EPC as defined in Mendling (2007) and it reports whether the model is sound or not. The results of this analysis are added to the error results HTML table and the *hasErrors* column captures whether there are errors in the EPC or not. Finally, the table is stored with MS Excel since this format can be loaded by SPSS, the software package that we use for the statistical analysis.

Furthermore, we mentioned a set of process model metrics being calculated by *xoEPC*. We briefly describe them in the following list including their hypothetical connection with errors (+ for hypothetically positive connection, – for negative). For more formulas to calculate the metrics and related work on metrics see Mendling (2007).

Size refers to the number of nodes of the process model graph. An increase in $Size_N$ should imply an increase in error probability (+). We will use the following notation for count metrics of different node types: $Size_C$ for connectors, etc.

Diameter gives the length of the longest path from a start node to an end node in the process model. It is presumably positively connected with error probability (+).

Density relates the number of arcs to the maximum number of arcs between all nodes. We presume a positive connection (+).

Coefficient of Connectivity (CNC) gives the ratio of arcs to nodes (+).

Average Connector Degree (AvCDegree) gives the number of nodes a connector is in average connected to (+).

Maximum Connector Degree (MaxCDegree) captures the maximum degree over all connectors (+).

Separability relates the number of cut-vertices to the number of nodes. An increase in Separability should imply a decrease in error probability (–).

Sequentiality is the number of arcs between none-connector nodes divided by the overall number of arcs (–).

Structuredness of the process graph is one minus the number of nodes in structured blocks divided by the number of nodes (–).

Depth captures how deep nodes are nested between splits and joins (+).

Connector Mismatch (MM) gives the sum of mismatches for each connector type. The mismatch is the absolute sum of all input arcs minus output arcs over all connectors of a connector type (+).

Connector Heterogeneity (ConnHet) gives the type entropy of the connectors (+).

Control Flow Complexity (CFC) sums up all choice of a process (+).

Cyclicity relates nodes on cycles to all nodes (+).

Token Split sums up all concurrent threads that can be activated by AND- and OR-splits in the process (+).

3 Distribution of Errors and Metrics

This section describes the sample of EPC models. In particular we present descriptive statistics disaggregated by group and error as well as a correlation analysis. The sample includes four collections of EPCs with a total of 2003 process models. All EPCs of the four groups were developed by practitioners.

1. *SAP Reference Model:* The first collection of EPCs is the SAP Reference Model. Its development started in 1992 and first models were presented at CEBIT'93 (Keller and Teufel 1998, p.VII). We use the version from 2000 that includes 604 EPCs.

2. *Service Model:* The second collection of EPCs stems from a German process reengineering project in the service sector. The project was carried out in the late 1990s. The models of this project include 381 non-trivial EPCs.

3. *Finance Model:* The third model collection contains EPCs of a process documentation project in the Austrian financial industry. It includes 935 EPCs.

4. *Consulting Model:* The fourth collection covers 83 EPCs from three consulting companies.

3.1 Disaggregation by Group

In this section we characterize the overall EPC sample and its four sub-groups by the help of mean values μ and standard deviation σ for each metric. Several of the disaggregated mean values are quite close to each other, but in particular the Finance Model shows a striking differences: it has the highest mean in structuredness and sequentiality. Figures 1 and 2 illustrates the distribution of both these metrics as box plots disaggregated by group. In this type of diagram invented by Tukey (1977) the median is depicted as a horizontal line in a box that represents the interval between lower and upper quartile, i.e. the EPCs ranked by the metric from 25% to 75%. The upper and lower wicks define a one and a half multiple of the respective quartile. Values outside these two intervals are drawn as individual points and are considered to be outliers. From this observation on structuredness and sequentiality we might conclude that the Finance Model contains the more structured EPCs and thus might have less error models.

Table 1: Errors in the sample models

Parameter	all	SAP Model	Service Model	Finance Model	Cons. Model
xoEPC errors	154	90	28	26	10
ProM error	115	75	16	7	17
EPCs (errors)	215	126	37	31	21
EPCs (total)	2003	604	381	935	83
Error ratio	10.7%	20.9%	9.7%	3.3%	25.3%

There is some evidence for such a hypothesis when we look at the number of errors in each of the four groups. Table 1 gives a respective overview. It can be seen that there are 2003 EPCs in the overall sample and 215 of them have at least one error. Accordingly, there is an overall error ratio of 10.7%. 154 of the 215 errors were found by *xoEPC*. 156 EPCs could not be reduced completely and were analyzed with ProM. This analysis revealed that 115 of the unreduced EPCs still had errors. Please note that there are EPCs for which both *xoEPC* and ProM found errors. Therefore, the number of EPCs with errors is less than the sum of EPCs with *xoEPC* and ProM errors. The comparison of the groups shows that the error ratio is quite different. In the previous paragraph we hypothesized that the finance model group might have less errors since its models are more structured. This suggests that metrics could be able to explain the low error ratio of only 3.3 %. We search gather further evidence in the next section.

3.2 Descriptive Statistics Disaggregated by hasErrors

In this section we discuss the distribution of the different metrics disaggregated by the variable *hasErrors*. Table 2 shows that there are quite large differences in the mean values of the sub-samples with and without errors. It is interesting to note that the error mean μ_e is higher than the non-error mean μ_n for most metrics where we assumed a positive connection with error probability in Section 2 and smaller for those metrics with a presumably negative connection. The only case where this does not hold is the density metric; it seems that it more accurately works as a counter-indicator for size than as an indicator for the density of connections in the model. The two columns on the right hand side of Table 2 might provide the basis for proposing potential error thresholds. The first of these columns gives a double σ_n deviation upwards from the non-error mean μ_n. Given a normal distribution only 2.5% of the population can be expected to have a metric value greater than this. The comparison of this value to the mean μ_e of the error EPCs gives an idea how good the two subparts of the sample can be separated by the metric. In several cases the mean μ_e is outside the double σ_n interval around μ_n. The box plots in Figures 3 and 4 illustrate the different distributions. It can be seen that correct EPCs tend to have much higher structuredness values and lower connector heterogeneity values. The next section investigates this observation with correlation statistics.

3.3 Correlation Analysis

This section approaches the connection between error probability and metrics with a correlation analysis. We use the Spearman rank correlation coefficient for ordinal data. As a confirmation of the previous observation all variables have the expected direction of influence besides the density metric. Table 3 presents the Spearman correlation between *hasErrors* and the metrics ordered by strength of correlation. It can be seen that several correlations are quite considerable with absolute values between 0.30 and 0.50. The significance of all correlations is better than 99%.

Table 3: Spearman correlation between hasError and metrics ordered by absolute correlation

	hasError		hasError
ConnHet	0.46	Sequentiality	-0.35
$Size_C$	0.43	Depth	0.34
MM	0.42	MaxCDegree	0.33
CFC	0.39	Cyclicity	0.30
$Size_A$	0.38	Diameter	0.30
Token Split	0.38	Separability	-0.29
$Size_N$	0.38	CNC	0.28
$Size_E$	0.38	AvCDegree	0.23
Density	-0.37	$Size_F$	0.19
Structuredness	-0.36		

The ability of a metric to separate error from non-error models by ranking is illustrated in Figures 5

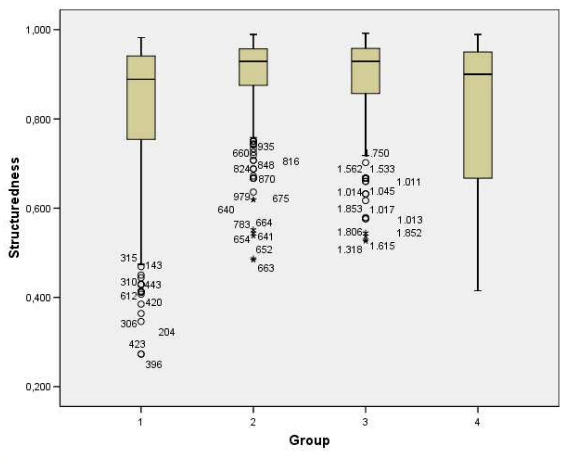

Figure 1: Box plot for structuredness disaggregated by group (1=SAP, 2=Service, 3=Finance, and 4=Consulting)

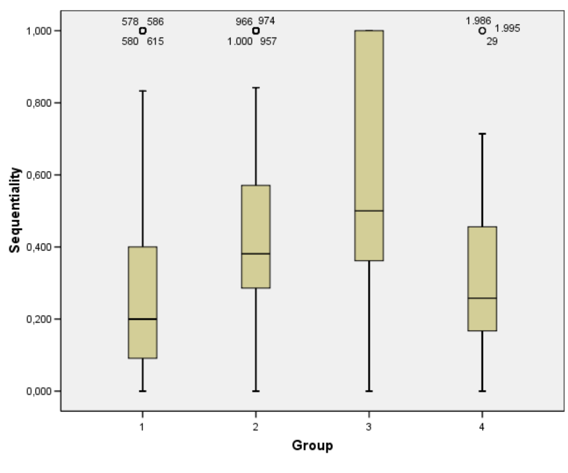

Figure 2: Box plot for sequentiality disaggregated by group (1=SAP, 2=Service, 3=Finance, and 4=Consulting)

Table 2: Mean and Standard Deviation of the sample models disaggregated by error

Parameter	Complete Sample		Non-Error EPCs		Error EPCs		2 σ dev. up		2 σ dev. down	
	μ	σ	μ_n	σ_n	μ_e	σ_e	$\mu_n + 2\sigma_n$		$\mu_n - 2\sigma_n$	
Size$_N$	20.71	16.84	18.04	13.48	42.97	24.08	44.99	$\approx \mu_e$		
Size$_E$	10.47	8.66	9.06	6.69	22.17	13.19	22.45	$\approx \mu_e$		
Size$_F$	5.98	4.94	5.67	4.65	8.53	6.33	14.97			
Size$_C$	4.27	5.01	3.30	3.47	12.26	7.89	10.24	$< \mu_e$		
Size$_A$	21.11	18.87	18.14	15.20	45.79	26.78	48.54	$\approx \mu_e$		
Diameter	11.45	8.21	10.63	7.71	18.25	9.01	26.06			
Density	0.09	0.07	0.09	0.07	0.03	0.02	0.23			
CNC	0.96	0.13	0.95	0.13	1.05	0.08	1.21			
AvCDegree	3.56	2.40	2.80	1.66	3.57	0.68	6.11			
MaxCDegree	2.88	1.60	3.31	2.28	5.64	2.41	7.87			
Separability	0.56	0.27	0.59	0.27	0.35	0.13			0.06	
Sequentiality	0.46	0.31	0.49	0.30	0.18	0.14			-0.12	
Structuredness	0.88	0.11	0.90	0.09	0.70	0.16			0.72	$> \mu_e$
Depth	0.70	0.74	0.61	0.69	1.45	0.73	1.98			
MM	3.31	4.55	2.54	3.45	9.71	6.92	9.44	$< \mu_e$		
ConnHet	0.28	0.35	0.22	0.32	0.75	0.19	0.85			
CFC	382.62	8849.48	202.19	6306.23	1883.17	19950.26	12814.64			
Cyclicity	0.01	0.08	0.01	0.06	0.07	0.17	0.12			
Token Split	1.82	3.53	1.28	2.46	6.26	6.62	6.20	$< \mu_e$		

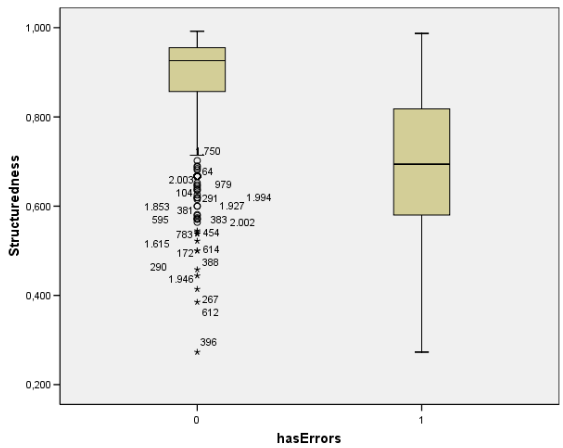

Figure 3: Box plot for structuredness disaggregated by error

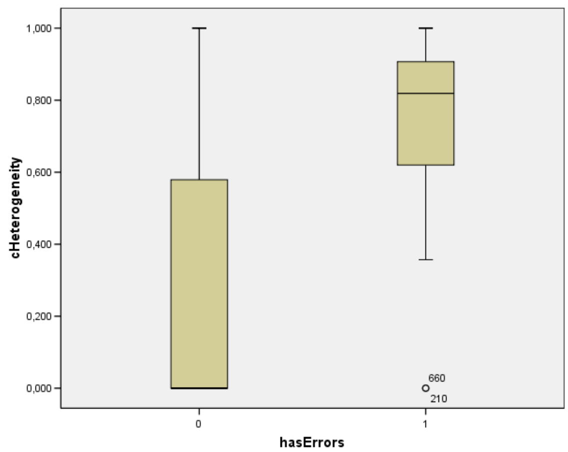

Figure 4: Box plot for connector heterogeneity disaggregated by error

and 6. For Figure 5 all models are ranked according to their size. A point (x, y) in the graph relates a size x to the relative frequency of error models in a subset of models that have at least size x, i.e. $y = |\{\frac{|errorEPCs|}{|allEPCs|} \mid Size_N(EPC) > x\}|$. It can be seen that the relative frequency of error EPCs increases by increasing the minimum number of nodes. In particular, the relative frequency of error EPCs is higher than 50% for all EPCs of at least 48 nodes. In Figure 5 all models are ranked according to their structuredness and (x, y) relates the structuredness x to the subset of models that have at most structuredness x. Here, the graph decreases and drops below 50% at a structuredness value of 0.80. Similar observations can be made for some of the other metrics, too. The values could be used as candidate thresholds. Altogether the relative frequency of error models above 50% is reached if

- Size$_N > 48$
- Size$_A > 62$
- Size$_C > 8$
- Size$_E > 22$
- Size$_F > 40$
- Token Split > 7
- Connector Mismatch $MM > 9$
- Structuredness < 0.8

4 Related Work

There are basically two main streams of research related to our work in the conceptual modeling area: top-down quality frameworks and bottom-up metrics that relate to quality aspects. For related work on Petri net verification refer to Reisig and Rozenberg (1998) and on EPCs to Mendling (2007).

One prominent *top-down quality framework* is the SEQUAL framework (Lindland et al. 1994, Krogstie et al. 2006). It builds on semiotic theory and defines several quality aspects based on relationships between a model, a body of knowledge, a domain, a modeling language, and the activities of learning, taking action, and modeling. Its usefulness was confirmed in an experiment (Moody et al. 2002). The Guidelines of Modeling (GoM) (Becker et al. 2000) define an alternative quality framework that is inspired by general accounting principles. The guidelines include the six principles of correctness, clarity, relevance, comparability, economic efficiency, and systematic design. This framework was operationalized for EPCs and also tested in experiments (Becker et al. 2000). Furthermore, e.g. Moody (2005) advocates a specification of a quality framework for conceptual modeling in compliance with the ISO 9126 standard for software quality (ISO 1991). A respective adaptation to business process modeling is reported in Gücegliogluand Demirörs (2005). Our research complements these approaches regarding semantical correctness. While the frameworks tend to be rather abstract, we find strong

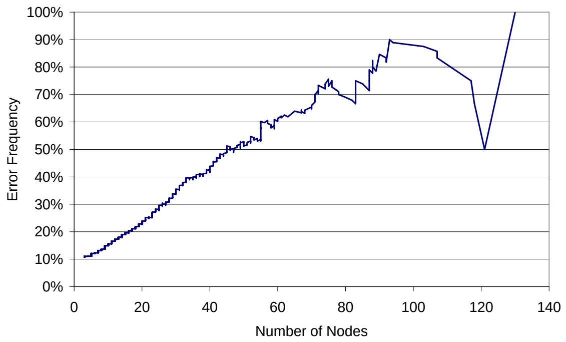

Figure 5: Error frequency to ordered number of nodes

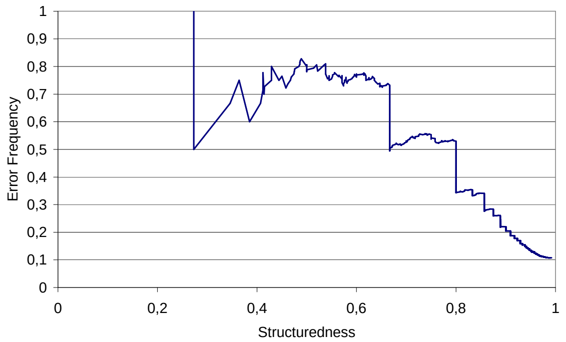

Figure 6: Error frequency to ordered structuredness

support for operational recommendations like using structured building blocks and limiting the number of nodes in a single process model.

Much work has been done related to *bottom-up metrics that relate to quality aspects* of process models, stemming from different research and partially isolated from each other (Lee and Yoon 1992, Nissen 1998, Morasca 1999, Reijers and Vanderfeesten 2004, Cardoso 2005, Balasubramanian and Gupta 2005, Canfora et al. 2005, Aguilar et al. 2006, Laue and Gruhn 2006, Mendling and Neumann 2007), for an overview see Mendling (2007). Several of these contributions are theoretic without empirical validation. Most authors doing experiments focus on the relationship between metrics and quality aspects: Canfora et al. (2005) study the connection between mainly count metrics for e.g. activities or routing elements and maintainability of software process models; (Cardoso 2006) validates the correlation between control flow complexity and perceived complexity; and Mendling and Neumann (2007), Mendling et al. (2007b) use metrics to predict control flow errors such as deadlocks in process models. The results of this research confirm the negative connection between size and quality aspects. In particular, it extends this stream of research with a validation of correlation based on an extensive sample of process models from practice.

Finally, there are further surveys that investigate the maturity (Rosemann et al. 2006), usability (Agarwal and Sinha 2003), and understandability of business process modeling languages (Sarshar and Loos 2005) and of models (Mendling et al. 2007a). They also relate to quality aspects of process models, but not directly to the connection of errors and metrics.

5 Summary

In this section we have conducted a correlation analysis related to a hypothetical connection between metrics and error probability. The results strongly confirm the hypotheses since the mean difference between error and non-error models as well as the correlation coefficients confirm the hypothetical impact direction of all metrics except the density metric. This metric had a strongly negative correlation with size in the sample which explains this exception.

These results have strong implications for the quality of business process modeling. First, the connection of the metrics with error probability provides a theoretical and empirical basis for defining process modeling principles and guidelines. The analysis reveals that in particular structured models are less error prone. Second, the established connection builds a foundation for a measurement-based management approach for the process of business process modeling. Different design alternatives can be discussed more objectively based on the metric values. Third, the design of future business process modeling tools can benefit from these findings by providing immediate feedback to the modeler when a certain metric passes an error threshold. Fourth, it has also some implications on the level of the process modeling language. Considering that the connector heterogeneity has an impact on error probability it might be a good idea to restrict modeling to the two connector types AND and XOR, and use OR-connectors only in structured blocks. Furthermore, there was a strong correlation between the number of start and end events with error probability. This fact suggests to restrict the use of multiple starts and ends. Modelers seem to loose track of the allowed combinations of these elements quite fast. In the reduced set of EPCs there are several EPCs for which no combination of start events guarantees a proper execution. Finally, the results have implications for the teaching of business process modeling. On the one hand the large number of errors suggest that practitioners frequently have problems to understand the behavioral implications of their design. On the other hand the metrics are a good starting point to teach patterns that are unlikely to result in errors.

References

W.M.P. van der Aalst and A.H.M. ter Hofstede. YAWL: Yet Another Workflow Language. *Information Systems*, 30(4):245–275, 2005.

R. Agarwal and A.P. Sinha. Object-oriented modeling with UML: a study of developers' perceptions. *Communications of the ACM*, 46(9):248–256, 2003.

E. Rolón Aguilar, F. Ruiz, F. García, and M. Piattini. Towards a Suite of Metrics for Business Process Models in BPMN. In Y. Manolopoulos, J. Filipe, P. Constantopoulos, and J. Cordeiro, editors, *ICEIS 2006, Proceedings (III)*, pages 440–443, 2006.

S. Balasubramanian and M. Gupta. Structural metrics for goal based business process design and evaluation. *Business Process Management Journal*, 11 (6):680–694, 2005.

J. Becker, M. Rosemann, and C. von Uthmann. Guidelines of Business Process Modeling. In W.M.P. van der Aalst, J. Desel, and A. Oberweis, editors, *Business Process Management. Models, Techniques, and Empirical Studies*, pages 30–49. Springer, Berlin et al., 2000.

G. Canfora, F. García, M. Piattini, F. Ruiz, and C.A. Visaggio. A family of experiments to validate metrics for software process models. *Journal of Systems and Software*, 77(2):113–129, 2005.

J. Cardoso. *Workflow Handbook 2005*, chapter Evaluating Workflows and Web Process Complexity, pages 284–290, 2005.

J. Cardoso. Process control-flow complexity metric: An empirical validation. In *Proceedings of IEEE International Conference on Services Computing (IEEE SCC 06)*, pages 167–173, 2006.

I. Davies, P. Green, M. Rosemann, M. Indulska, and S. Gallo. How do practitioners use conceptual modeling in practice? *Data & Knowledge Engineering*, 58(3):358–380, 2006.

A. Selçuk Güceglioglu and O. Demirörs. Using software quality characteristics to measure business process quality. In W.M.P. van der Aalst, B. Benatallah, F. Casati, and F. Curbera, editors, *BPM 2005, Proc.*, LNCS 3649, pages 374–379, 2005.

S.J.B.A. Hoppenbrouwers, H.A. Proper, and Th.P. van der Weide. A Fundamental View on the Process of Conceptual Modeling. In *Conceptual Modeling – ER 2005*, LNCS 3716, pages 128–143, 2005.

IDS Scheer AG. *XML-Export und -Import mit ARIS 5.0, January 2001*, 2001.

International Standards Organisation ISO. Information technology - software product evaluation - quality characteristics and guide lines for their use. ISO/IEC IS 9126, 1991.

G. Keller and T. Teufel. *SAP(R) R/3 Process Oriented Implementation: Iterative Process Prototyping*. Addison-Wesley, 1998.

G. Keller, M. Nüttgens, and A.-W. Scheer. Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)". Heft 89, Institut für Wirtschaftsinformatik, Saarbrücken, Germany, 1992.

J. Krogstie, G. Sindre, and H.D. Jørgensen. Process models representing knowledge for action: a revised quality framework. *European Journal of Information Systems*, 15(1):91–102, 2006.

R. Laue and V. Gruhn. Complexity metrics for business process models. In W. Abramowicz and H.C. Mayr, editors, *9th Int. Conf. on Business Information Systems (BIS 2006)*, LNI 85, pages 1–12, 2006.

G.S. Lee and J.-M. Yoon. An empirical study on the complexity metrics of petri nets. *Microelectronics and Reliability*, 32(3):323–329, 1992.

O.I. Lindland, G. Sindre, and A. Sølvberg. Understanding quality in conceptual modeling. *IEEE Software*, 11(2):42–49, 1994.

J. Mendling. *Detection and Prediction of Errors in EPC Business Process Models*. PhD thesis, Vienna Univ. of Economics and Business Admin., 2007.

J. Mendling and W.M.P. van der Aalst. Formalization and Verification of EPCs with OR-Joins Based on State and Context. In J. Krogstie, A.L. Opdahl, and G. Sindre, editors, *Proceedings of CAiSE 2007*, LNCS 4495, pages 439–453, 2007.

J. Mendling and G. Neumann. Error metrics for business process models. In J. Eder, S.L. Tomassen, A.L. Opdahl, and G. Sindre, editors, *Proceedings of CAiSE Forum 2007*, pages 53–56, 2007.

J. Mendling and M. Nüttgens. EPC Markup Language (EPML) - An XML-Based Interchange Format for Event-Driven Process Chains (EPC). *Information Systems and e-Business Management*, 4 (3):245 – 263, 2006.

J. Mendling, M. Moser, G. Neumann, H.M.W. Verbeek, B.F. van Dongen, and W.M.P. van der Aalst. Faulty EPCs in the SAP Reference Model. In J.L. Fiadeiro S. Dustdar and A. Sheth, editors, *Proceedings of BPM 2006*, LNCS 4102, page 451-457, 2006.

J. Mendling, H.A. Reijers, and J. Cardoso. What makes process models understandable? In *Proceedings of BPM 2007*, LNCS 4714, pages 48–63, 2007a.

J. Mendling, H.M.W. Verbeek, B.F. van Dongen, W.M.P. van der Aalst, and G. Neumann. Detection and Prediction of Errors in EPCs of the SAP Reference Model. *Data & Knowledge Engineering*, 2007b. accepted for publication.

D.L. Moody. Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. *Data & Knowledge Engineering*, 55(3):243–276, 2005.

D.L. Moody, G. Sindre, T. Brasethvik, and Arne Sølvberg. Evaluating the quality of process models: Empirical testing of a quality framework. In Stefano Spaccapietra, Salvatore T. March, and Yahiko Kambayashi, editors, *Conceptual Modeling - ER 2002, Proceedings*, LNCS 2503, pages 380–396, 2002.

S. Morasca. Measuring attributes of concurrent software specifications in petri nets. In *METRICS '99: Proceedings of the 6th International Symposium on Software Metrics*, pages 100–110, 1999.

G. Neumann and U. Zdun. XOTcl, an Object-Oriented Scripting Language. In *Proc. of the 7th USENIX Tcl/Tk Conference, Austin, USA*, 2000.

M.E. Nissen. Redesigning reengineering through measurement-driven inference. *MIS Quarterly*, 22 (4):509–534, 1998.

OMG, ed. Business Process Modeling Notation (BPMN) Specification. Final Adopted Specification, dtc/06-02-01, Object Management Group, February 2006.

H.A. Reijers and I.T.P. Vanderfeesten. Cohesion and coupling metrics for workflow process design. In J. Desel, B. Pernici, and M. Weske, editors, *BPM 2004, Proceedings*, LNCS 3080, pages 290–305, 2004.

W. Reisig and G. Rozenberg, editors. *Lectures on Petri Nets I: Basic Models*, LNCS 1491, 1998.

M. Rosemann, J. Recker, M. Indulska, and P. Green. A study of the evolution of the representational capabilities of process modeling grammars. In E. Dubois and K. Pohl, editors, *CAiSE 2006, Proceedings*, LNCS 4001, pages 447–461, 2006.

K. Sarshar and P. Loos. Comparing the control-flow of epc and petri net from the end-user perspective. In W.M.P. van der Aalst, B. Benatallah, F. Casati, and F. Curbera, editors, *BPM 2005, Proceedings*, LNCS 3649, pages 434–439, 2005.

H.A. Simon. *Sciences of the Artificial*. The MIT Press, 3rd edition, 1996.

J.W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 1977.

H.M.W. Verbeek, B.F. van Dongen, J. Mendling, and W.M.P. van der Aalst. Interoperability in the ProM Framework. In T. Latour and M. Petit, editors, *Proceedings of the CAiSE Workshops*, pages 619–630, 2006.