Cardoso, J., "Process control-flow complexity metric: An empirical validation", IEEE International Conference on Services Computing (IEEE SCC 06), Chicago, USA, September 18-22, 2006. pp. 167-173, IEEE Computer Society. ISBN: 0-7695-2670-5.

Process control-flow complexity metric: An empirical validation

Jorge Cardoso

Department of Mathematics and Engineering
University of Madeira, 9000-390 Funchal, Portugal
jcardoso@uma.pt

Abstract

Organizations are increasingly faced with the challenge of managing business processes, workflows, and, recently, Web processes. One important aspect of processes that has been overlooked is their complexity. High complexity in processes may result in bad understandability, errors, defects, and exceptions leading processes to need more time to develop, test, and maintain. Therefore, excessive complexity should be avoided. This paper describes an experiment designed to validate the Control-Flow Complexity (CFC) metric that we have proposed in our previous work. In order to demonstrate that our CFC metric serves the purpose it was defined for, we have carried out an empirical validation by means of a controlled experiment. The explanation of the steps followed to do the experiment, the results, and the conclusions obtained are the main objectives of this paper.

1. Introduction

Designing and improving processes is a key aspect for businesses to stay competitive in today's marketplace. Organizations have been forced to improve their business processes because customers are demanding better products and services. When an organization adopts a process management philosophy, process improvement can take place. Independently of the approach taken, which can be a continuous process improvement (CPI) [1], a business process redesign (BPR) [2], or a business process reengineering (BPR) [3] approach, methods need to be available to analyze the processes undergoing improvements. To achieve an effective process management, one fundamental area of research that needs to be explored is the complexity analysis of processes.

We define process complexity as the degree to which processes are difficult to analyze, understand, or explain. High complexity in a process has several undesirable drawbacks, it may result in bad understandability, errors, defects, and exceptions leading processes to need more time to develop, test, and maintain. For example, in software engineering it has been found that program modules with high complexity indices have a higher frequency of failures.

Surprisingly, in spite of the fact that there is a vast literature on software measurement of complexity [4], no research on process measurement of complexity has yet been carried out. Analyzing the complexity at all the stages of process design and development helps avoid the drawbacks associated with high complexity processes. Currently, organizations have not adopted complexity metrics as part of their process management projects. As a result, it may happen that simple processes to be designed in a complex way.

In [5] we have presented a Control-Flow Complexity (CFC) metric to measure the degree of complexity of business processes from a control-flow perspective. As Lord William Thomson Kelvin (1824-1907) said, "if you cannot measure it, you cannot improve it." The use of the CFC metric allow designers to improve processes, thus reducing the time spent reading and understanding processes in order to remove faults or adapt them to changed requirements. The CFC metric can be used to analyze the complexity of business processes, as well as workflows and Web processes. In this work we describe the empirically experiment with human subjects that we have carried out for validating the metric proposed in [5].

In the area of software measurement, a significant number of the metrics developed have had a reduced industrial acceptance. According to some research, one reason is that there is a lack of serious validation; and thus, a lack of confidence in the measurements [6]. To avoid that this type of problems also impact the area of business process management we demonstrate that our CFC metric serves the purpose it was defined for by carrying out an empirical validation by means of a controlled experiment.

This paper is structured as follows. Section 2 presents an introduction to process complexity. In section 3, we give the main elements behind the CFC metric that will be object of experimentation. Section 4 constitutes the core of this paper. We describe the experiment that we have carried out for empirically validating the proposed metric. Such an experiment plays a fundamental role in our work, since the experimentation is as a crucial part of the evaluation of new metrics and is critical for the success of any measurement activity [7]. Through empirical validation we can demonstrate with real evidence that the measure we proposed serve the purpose it was defined for. Finally, section 5 presents our conclusions.

2. Process Complexity

The Merriam-Webster dictionary definition of "complexity" includes references to 'the quality or state of being complex' and 'something complex'. The adjective "complex" is referred to as something 'hard to separate, analyze, or solve', being a synonym of complicated, intricate, and involved.

In software engineering, several definitions have been given to describe the meaning of software complexity. For example, Curtis [8] states that complexity is a characteristic of the software interface which influences the resources another system will expend or commit while interacting with the software. Card and Agresti [9] define system complexity as the sum of structural complexity and data complexity divided by the number of modules changed. Fenton [10] defines complexity as the amount of resources required for a problem's solution.

After analyzing the characteristics and specific aspects of processes, we believe that the definition that is better suited to describe processes complexity can be derived from [11]. Therefore, we define process complexity as, "The degree to which a process is difficult to analyze, understand or explain. It may be characterized by the number and intricacy of activity interfaces, transitions, conditional and parallel branches, the existence of loops, data-flow, control-flow, roles, activity categories, the types of data structures, and other process characteristics."

3. The Control-flow Complexity Metric

Process measurement is the activity of assigning a number or a symbol to a process in order to characterize an attribute of the process according to given rules. Measures of process complexity can be seen as a good indicator of error-proneness or the likelihood of a process to have an execution fault. The rational is that as the structure of a process becomes more complex, business analysts lose track of how one activity is affected by another and so changes to the process can produce unexpected results.

Losing track of the structure of a process has been given as one reason why complex applications can never be deemed to be entirely "safe" [12]. Applying a CFC metric to processes allows business analysts to determine when a process has become too complex and needs corrective actions to be taken.

The CFC metric is calculated based on the split structures present in a process. Split structures are a very good candidate to develop a control-flow complexity metric since they are the elements that determine the control-flow of a process during its execution. Our work borrows some techniques from the branch of software engineering known as software metrics, namely McCabe's cyclomatic complexity [13]. A detailed explanation of the CFC metric can be found in [5]. The control-flow complexity for a process P is calculated as follows:

$$
\begin{aligned}
&CFC(P) = \\
&\quad \sum_{i \in \{\text{XOR-splits of P}\}} CFC_{XOR-split}(i) + \\
&\quad \sum_{j \in \{\text{OR-splits of P}\}} CFC_{OR-split}(j) + \\
&\quad \sum_{k \in \{\text{AND-splits of P}\}} CFC_{AND-split}(k)
\end{aligned}
$$

The greater the value of the CFC(P) the greater the overall architectural complexity of a process. CFC(P) analysis seeks to evaluate complexity without direct execution of processes. The function CFC(P) is computed based on the individual control-flow complexity of XOR, OR, and AND –splits.

Each individual complexity is calculated based on the notion of control-flow induced mental state. A mental state is a state that has to be considered when a designer is developing a process. Splits introduce the notion of mental states in processes. When a split (XOR, OR, or AND) is introduced in a process, the business process designer has to mentally create a map or structure that accounts for the number of states that can be reached from the split. The notion of mental state is important since there are certain theories [14] prove that complexity beyond a certain point defeats the human mind's ability to perform accurate symbolic manipulations, and hence result in error.

The XOR-split control-flow complexity is determined by the number of mental states that are introduced with this type of split. The function $\mathrm{CFC}_{\text{XOR-split}}(a)$, where 'a' is a XOR-split activity, computes the control-flow complexity of the split. For XOR-splits, the control-flow complexity is the fan-out of the split.

$$\mathrm{CFC}_{\text{XOR-split}}(a) = \text{fan-out}(a)$$

In this particular case, the complexity is directly proportional to the number of activities that follow a XOR-split and that a process designer needs to consider, analyze, and assimilate. The idea is to associate the complexity of an XOR-split with the number of states (Web services or workflow tasks) that follow the split. This rationale is illustrated in Figure 1.

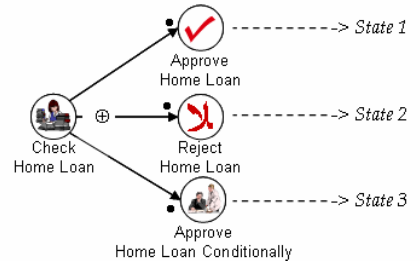

Figure 1. XOR-split control-flow complexity

The OR-split control-flow complexity is also determined by the number of mental states that are introduced with the split. For OR-splits, the control-flow complexity is 2^n-1, where n is the fan-out of the split.

$$\mathrm{CFC}_{\text{OR-split}}(a) = 2^{\text{fan-out}(a)}-1$$

This means that when a designer is constructing a process he needs to consider and analyze $2^n -1$ states that may arise from the execution of an OR-split construct.

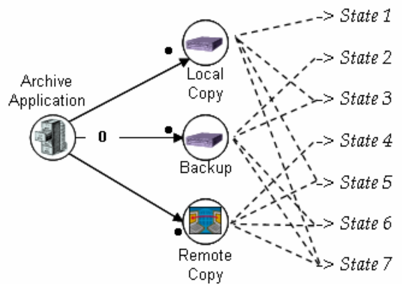

Figure 2. OR-split control-flow complexity

Mathematically, it would appear more obvious that 2^n states can be reached after the execution of an OR-split. But since a process that has started its execution has to finish, it cannot be the case where after the execution of an OR-split no transition is activated, i.e. no Web service or workflow task is executed. Therefore, this situation or state cannot happen.

For AND-splits, the control-flow complexity is simply 1.

$$\mathrm{CFC}_{\text{AND-split}}(a) = 1$$

A designer constructing a process needs only to consider and analyze one state that may arise from the execution of an AND-split construct since it is assumed that all the outgoing transitions are selected and followed.

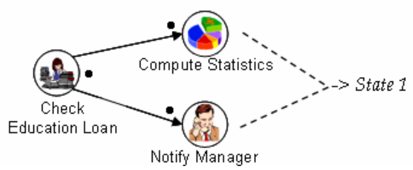

Figure 3. AND-split control-flow complexity

The higher the value of $\mathrm{CFC}_{\text{XOR-split}}(a)$, $\mathrm{CFC}_{\text{OR-split}}(a)$, and $\mathrm{CFC}_{\text{AND-split}}(a)$, the more complex is a process design, since developers have to handle all the states between control-flow constructs (splits) and their associated outgoing transitions and activities. Each formula to calculate the complexity of a split construct is based on the number of states that follow the construct.

4. Empirical Validation of the Control-flow Complexity Metric

In this section we describe the experiment we have carried out for empirically validating the CFC metric (see section 3). This empirical study is an experiment that compares what we believe to what we observe. Such an experiment plays a fundamental role in our work. Zelkowitz and Wallace [7] stress the importance of using experimental models for validating metrics. The authors suggest experimentation as a crucial part of the evaluation of new metrics.

For the experiment to be successful it needs to be wisely constructed and executed. Therefore, we have followed some suggestions, provided by Perry, Porter et al. [15], about the structure and the components of a suitable empirical study. To perform an experiment, several steps have to be taken in a certain order. An experiment can be divided into the following main activities [15]: research context, hypotheses, experimental design, threats to validity, data analysis and presentation, results and conclusions.

In the remainder of this section we explain how we have performed each of the activities described above.

4.1. Research Context

In this section the terminology is explained, the problem is defined, and a brief research review is undertaken to provide the historical context surrounding the problem.

Terminology and problem definition. Process complexity can be defined as the degree to which a business process is difficult to analyze, understand or explain. The control-flow complexity refers to the degree of complexity of a process from a control-flow perspective.

The CFC metric can be used to automatically measure the control-flow complexity of a process based on its structure. It allows designers to create less complex processes, thus reducing the time spent reading and understanding processes in order to remove faults or adapt the process to changed requirements.

Our goal is to analyze the CFC metric for the purpose of evaluating and validating the proposed metric. For a set of processes, we wish to determine the correlation between the output of the CFC metric and the perceived control-flow complexity from the point of view of process designers. In our experiments, process designers (subjects) were Master students from the Department of Mathematics and Engineering at the University of Madeira (Portugal).

Research Review. In [5] we have presented the CFC metric to analyze the degree of complexity of business processes. Nowadays, complexity analysis has an increased importance since the emergence of processes that span both between and within enterprises [16] have an inherent complexity. Therefore, methods should be used to support the design, improvement, and redesign of processes to reduce their complexity. The CFC can be used to analyze the complexity of business processes, as well as workflow and Web processes.

4.2. Hypotheses formulation

An important aspect of experiments is to know and to state in a clear and formal way what we intend to evaluate. Hypotheses are essential as they state the research questions we are asking. We present two hypotheses: an abstract and a concrete hypothesis.

Abstract Hypothesis: "The CFC metric is a good and accurate metric to evaluate and establish the control-flow complexity of processes."

Concrete Hypothesis: "There is a significant correlation between the CFC metric and the subject's rating of the control-flow complexity of processes."

4.3. Study Design

After the research context and the hypotheses formulation, the design of the study took place. A study's design is a detailed plan for collecting the data that will be used to test the hypotheses. This phase also explains how the experiment was conducted and has several components:

Variable selection. One component is a set of variables that link causes and effects. Typically, there are two kinds of variables: dependent and independent.

a) The independent variable is the control-flow structure of processes.
b) The dependent variable is the control-flow complexity of processes which varies when the control-flow structure of processes changes.

Subjects selection. Our subjects were students of the Department of Mathematics and Engineering enrolled in the first-year of a Master program in Computer Science at the University of Madeira, Portugal. Nineteen subjects were selected. Most of the students had industrial experience in several areas, but none had experience with business process management systems and methodologies. By the time the experiment was done, all the students had taken a 50 hours course on Business Process Management (BPM) and, therefore, gained experience in the design and development of business processes.

Experiment design. The objects to be rated were business processes graphically designed with the process language used by METEOR workflow management system [17]. An example of the processes analyzed and rated by the subjects is illustrated in Figure 4. The independent variable was measured using the CFC metric presented in section 3. The dependent variable was measured according to subject's ratings. All the tests were solved by the same group of subjects.

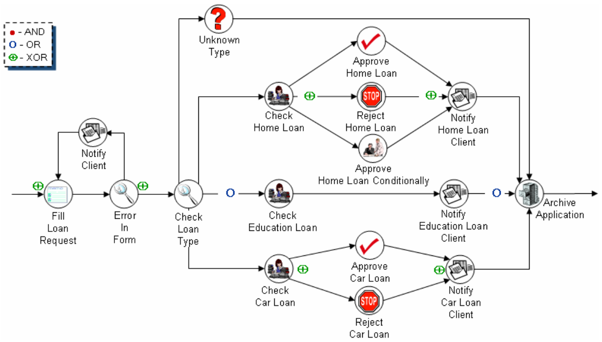

Figure 4. Example of an object rated by the subjects

We prepared the material we had to give to the subjects. The material consisted of 22 professionally-designed, error-free, processes (objects) of the same universe of discourse, related to bank loan applications. The subjects were told how to carry out the experiment. Each subject carried out the experiment alone, in class, and could use unlimited time to solve it. We collected all the data, including subjects' rating and the measurements automatically calculated by means of the CFC metric. All tests were considered valid because all of the subjects had at least medium experience in designing and analyzing business processes.

4.4. Threats to Validity

Threats to validity are influences that may limit our ability to interpret or draw conclusions from the study's data. We will discuss the empirical study's various threats to validity (construct, internal, and external validity) and the way we attempted to alleviate them.

Construct validity. All the measurements of the dependent variable were subjective and based on the perception of the subjects. As the subjects involved in this experiment had medium experience in BPM design we think their ratings can be considered significant. The independent variable that measures the control-flow complexity of processes can also be considered constructively valid because from a complexity theory point of view, a system is called complex if it is composed of many different types of elements.

Internal validity. We have considered the different aspects that could threaten the internal validity of the study, such as differences among subjects, precision of subject' ratings, learning effects, fatigue effects, and subject incentive. Subjects were knowledgeable concerning the evaluation issues. Analyzing the results of the experiment we can empirically observe the existence of a correlation between the independent and the dependent variable.

External validity. One threat to external validity has been identified: subject selection. This threat can limit the ability to generalize the results to settings outside the study. The subjects were Master students that had recently taken a 50 hours course on BPM gaining an in-depth experience in the design and development of business processes. In order to extract a final conclusion that can be generalized, it is necessary to replicate this experiment with a more diversified number of subjects, including practitioners and designers with less experience.

4.5. Data Analysis and Presentation

Two main approaches to presenting and analyzing data can be chosen: quantitative and qualitative analysis. Since our subjects rated processes using a numerical scale from 0 to 100, we have selected quantitative analysis to draw conclusions from our data. The qualitative analysis was done in conjunction with a statistical analysis.

As we have said previously, our goal is to determine if any correlation exists between subjects' ratings and the CFC metric proposed in [5] and briefly described in section 3. Since the data collected in the experiment is distribution free, the Spearman Rank-Difference Correlation Coefficient [18], r_S, was used to determine the correlation of the data collected in the experiment. The Spearman r_S is a non-parametric statistic used to show the relationship between two variables which are expressed as ranks (the ordinal level of measurement). The correlation coefficient is a measure of the ability of one variable to predict the value of another variable. Using Spearman's correlation coefficient, the CFC metric was correlated separately to the different subject's rates of control-flow complexity. In our experiment the null hypothesis was:

H_0: "there is no correlation between the CFC metric and the subject's rating of control-flow complexity".

The probability that the null hypothesis would be erroneously rejected was controlled with two confidence levels: $\alpha_1=0.005$ and $\alpha_2=0.05$. The decision rules for rejecting the null hypothesis were:

For α_1: reject H_0 if $r_S >= 0.586$; For α_2: reject H_0 if $r_S >= 0.425$

5. Results

The analysis performed on the collected data led to some interesting results. Table 1 shows summary statistics describing the Spearman rank-difference correlation coefficient between subjects' ratings and the values given by the CFC metric. For each subject, the correlation coefficient, r_S, is given.

Table 1. Correlation coefficients

		rs	α_1	α_2
Subject	1	0,741	Reject H_0	Reject H_0
	2	0,576	Accept H_0	Reject H_0
	3	0,487	Accept H_0	Reject H_0
	4	0,974	Reject H_0	Reject H_0
	5	0,732	Reject H_0	Reject H_0
	6	0,693	Reject H_0	Reject H_0
	7	0,733	Reject H_0	Reject H_0
	8	0,848	Reject H_0	Reject H_0
	9	0,620	Reject H_0	Reject H_0
	10	0,638	Reject H_0	Reject H_0
	11	0,720	Reject H_0	Reject H_0
	12	0,677	Reject H_0	Reject H_0
	13	0,833	Reject H_0	Reject H_0
	14	0,487	Accept H_0	Reject H_0
	15	0,767	Reject H_0	Reject H_0
	16	0,704	Reject H_0	Reject H_0
	17	0,835	Reject H_0	Reject H_0
	18	0,899	Reject H_0	Reject H_0
	19	0,664	Reject H_0	Reject H_0

Based on the data from Table 1 and taking in consideration α_1, the values of r_S are greater than 0.586 for 84% of the subjects; therefore we reject the null hypothesis. Taking in consideration α_2, all the values of r_S are greater than 0.425, therefore we also reject the null hypothesis. For α_1, our confidence level is 95%, and for α_2 our confidence level is 99.5%.

After analyzing the data we gathered, we concluded that the obtained results reveal that there exists a high correlation between the CFC metric and the subject's rating of control-flow complexity. This leads us back to our original goal which was to demonstrate that the CFC metric serves the purpose it was defined for, measure the control-flow complexity of processes. The results obtained are believable and there are no ambiguities in our interpretation. We also believe that no external elements have influenced our results. The diffusion of the experimental results and the way they are presented are relevant so that they are really put into use. Therefore, we published our findings in this paper and we are also planning to develop a Web-based system to allow other researcher to replicate our experiment.

Our results recommend the use of the CFC metric to create less complex processes, thus reducing the time spent reading and understanding processes in order to remove faults or adapt the processes to changed requirements. The complexity measurement enables process managers and administrators to calculate the complexity of processes generated by others. Process designers can analyze the complexity of a particular process in development. Process consultants can contribute with new process components, needing methods to analyze the complexity of the proposed solutions. End-users can inquire about the complexity of processes before starting process instances.

6. Conclusions

The complexity of processes is intuitively connected to effects such as readability, effort, testability, reliability, and maintainability. Therefore, it is important to develop metrics to identify complex processes. Afterwards, these processes can be reengineered, improved, or redesigned to reduce their complexity.

In our previous research we have proposed the Control-Flow Complexity (CFC) metric to be applied to processes. The CFC is a design-time metric that can be used to evaluate the difficulty of producing a business process, a Web process, or a workflow before an actual implementation exists. When process control-flow complexity analysis becomes part of the process development cycle, it has a considerable influence in the design phase, leading to less complex processes.

In order to demonstrate that our CFC metric serves the purpose it was defined for, we have carried out an empirical validation by means of a controlled experiment. Our experiment has involved 19 graduate students in Computer Science, as part of a research project, and tested if the control-flow complexity of a set of 22 business processes could be predicted using the CFC metric. Analyzing the collected data using statistical methods we have concluded that the CFC metric is highly correlated with the control-flow complexity of processes. This metric can, therefore, be used by business process analysts and process designers to analyze the complexity of processes and, if possible, develop simpler processes.

6. Acknowledgements

We would like to thank the Computer Science graduate students from the University of Madeira of class 2004-05 for their contribution in this study.

7. References

[1].Harrington, H., *Process Breakthrough: Business Process Improvement.* Journal of Cost Management (Fall), 1993: p. 30-43.

[2].Wastell, D., P. White, and P. Kawalek, *A methodology for business process re-design: experiences and issues.* Journal of Strategic Information Systems, 1994. **3**(1): p. 23-40.

[3].Ould, M.A., *Business Processes: Modelling and analysis for re-engineering and improvement*. 1995, Chichester, England: John Wiley & Sons. 224.

[4].Zuse, H., *A Framework of Software Measurement*. 1997, Berlin: Walter de Gruyter Inc.

[5].Cardoso, J., *Evaluating Workflows and Web Process Complexity*, in *Workflow Handbook 2005*, L. Fischer, Editor. 2005, Future Strategies Inc.: Lighthouse Point, FL, USA. p. 284-290.

[6].Fenton, N. and M. Neil, *Software metrics: successes, failures and new directions.* Journal Systems Software, 1999. **47**(2-3): p. 149-157.

[7].Zelkowitz, M.V. and D.R. Wallace, *Experimental Models for Validating Technology.* IEEE Computer, 1998. **31**(5): p. 23-31.

[8].Curtis, B., *Measurement and Experimentation in Software Engineering.* Proceedings of the IEEE, 1980. **68**(9): p. 1144-1157.

[9].Card, D. and W. Agresti, *Measuring Software Design Complexity.* Journal of Systems and Software, 1988. **8**: p. 185-197.

[10].Fenton, N., *Software Metrics: A Rigorous Approach*. 1991, London: Chapman & Hall.

[11].IEEE, *IEEE 610, Standard Glossary of Software Engineering Terminology*. 1992, New York: Institute of Electrical and Electronic Engineers.

[12].Borning, A., *Computer System Reliability and Nuclear War*, in *Communications of the ACM*. 1987. p. 112-131.

[13].McCabe, T., *A Complexity Measure.* IEEE Transactions of Software Engineering, 1976. **SE-2**(4): p. 308-320.

[14].Miller, G., *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information.* The Psychological Review, 1956.

[15].Perry, D.E., A.A. Porter, and L.G. Votta, *Empirical Studies of Software Engineering: A Roadmap*, in *The Future of Software Engineering*, A. Finkelstein, Editor. 2000, ACM Press. ISBN 1-58113-253-0.

[16].Sheth, A.P., W.v.d. Aalst, and I.B. Arpinar, *Processes Driving the Networked Economy.* IEEE Concurrency, 1999. **7**(3): p. 18-31.

[17].Kochut, K.J., *METEOR Model version 3*. 1999, Large Scale Distributed Information Systems Lab, Department of Computer Science, University of Georgia: Athens, GA.

[18].Siegel, S. and J. N. John Castellan, *Nonparametric Statistics for The Behavioral Sciences*. 1988: McGraw-Hill. 399.