

A NEW COMPLEXITY METRIC FOR BUSINESS PROCESS MODELS

**A MASTER'S THESIS
IN
INFORMATION TECHNOLOGY SERVICE MANAGEMENT
ATILIM UNIVERSITY**

**BY
EMRİYE OŐKUN**

JULY 2014

**A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES**

**OF
ATILIM UNIVERSITY**

**BY
EMRİYE OŐKUN**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF**

MASTER OF SCIENCE OF PHILOSOPHY

**IN
THE DEPARTMENT OF INFORMATION TECHNOLOGY SERVICE
MANAGEMENT**

JULY 2014

Approval of the Graduate School of Natural and Applied Sciences, Atılım University.

Prof. Dr. K. İbrahim AKMAN

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Murat KOYUNCU

Head of Department

This is to certify that we have read the thesis “A New Complexity Metric for Business Process Models” submitted by “Emriye ÇOŞKUN” and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Deepti MISHRA

Supervisor

Examining Committee Members

Assoc. Prof. Dr. Mohammad REHAN

Asst. Prof. Dr. Deepti MISHRA

Instructor Güler KALEM

Date: 10.07.2014

I declare and guarantee that all data, knowledge and information in this document has been obtained, processed and presented in accordance with academic rules and ethical conduct. Based on these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Emriye OŐKUN

Signature:

ABSTRACT

A NEW COMPLEXITY METRIC FOR BUSINESS PROCESS MODELS

Çoşkun, Emriye

M.S., Information Systems Engineering Department

Supervisor: Asst.Prof.Dr. Deepti Mishra

July 2014, 65 pages

In this thesis, CADAC (Cognitive Activity Depth Arc Control Flow) complexity metric that can accurately predict the perceived understandability of business process models is proposed.

CADAC is designed considering existing complexity metrics for business process models and cognitive weights of elements in business process models. CADAC metric is validated theoretically by using Weyuker's Properties and Briand's Framework and practically with examples. Also, practical implication is done for CADAC metric.

CADAC metric is designed considering only BPMN 2.0 language, not the other business process models techniques. For CADAC evaluations, all types of activities, only exclusive, parallel and inclusive gateways, and all types of sequence flows are considered.

Keywords: Complexity metrics, Business process models, Business process modelling notation, Cognitive weight.

ÖZ

İŞ AKIŞ MODELLERİ İÇİN YENİ BİR KARMAŞIKLIK METRİĞİ

Çoşkun, Emriye

Yüksek Lisans, Bilişim Sistemleri Mühendisliği Bölümü

Tez Yöneticisi: Yrd.Doç.Dr. Deepti Mishra

Temmuz 2014, 65 sayfa

Bu çalışmada iş akış modellerinin, algılanan anlaşılabilirliğini tam olarak tahmin edebilen CADAC (Cognitive Activity Depth Arc Control Flow) karmaşıklık metriği tasarlanır.

CADAC, iş akış modelleri için mevcut karmaşıklık metrikleri ve iş akış modellerindeki elementlerin kavramsal ağırlıkları dikkate alınarak tasarlanır. CADAC metriği, teorik olarak Weyuker' s Properties ve Briand' s Framework kullanılarak ve pratik olarak örneklerle doğrulanır. Ayrıca, CADAC metriği için pratik uygulama yapılır.

CADAC metriği sadece BPMN 2.0 dili dikkate alınarak tasarlanır, diğer iş akış modelleri teknikleri dikkate alınmaz. CADAC hesaplamaları için tüm tipteki faaliyetler, sadece harici, paralel ve dahili ağ geçitleri, ve tüm tipteki dizi akışları hesaba katılır.

Anahtar Kelimeler: Karmaşıklık metrikleri, İş akış modelleri, İş akış modelleme notasyonu, Kavramsal ağırlık.

GCCRIIS

To My Parents

ACKNOWLEDGMENTS

I express sincere appreciation to my supervisor Asst. Prof. Dr. Deepti Mishra for her guidance and insight throughout the research. Thanks also, to my parents, I offer sincere thanks for their continuous support and patience during this period.

TABLE OF CONTENTS

ABSTRACT	iii
ÖZ	iv
DEDICATION.....	v
ACKNOWLEDGMENTS	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	ix
LIST OF FIGURES	ix
LIST OF ABBREVIATIONS.....	x
CHAPTER	
I. INTRODUCTION	1
1.1 Statement of the Problem	1
1.2 Scope and Outline of the Thesis	3
II. BACKGROUND INFORMATION AND LITERATURE SURVEY	5
2.1 Business Process Modelling	5
2.1.1 Business Process Modelling Techniques	5
2.1.2 Business Process Models Elements	6
2.1.3 Complexity Metrics for Business Process Models	8
2.2 Literature Survey	8
2.2.1 Existing Complexity Evaluation Approaches	8

2.2.1.1 Seven Process Modelling Guidelines (7PMG)	8
2.2.1.2 Similarities between Software and Business Processes	9
2.2.2 Existing Complexity Metrics for BPMs	10
III. PROPOSED COMPLEXITY METRIC	17
3.1 Example for the Proposed Complexity Metric Calculations.....	23
IV. VALIDATON OF THE PROPOSED COMPLEXITY METRIC	24
4.1 Theoretical Validation	24
4.1.1 Weyuker' s Properties	24
4.1.2 Briand's Framework	34
4.2 Practical Implication	37
4.2.1 Examples of BPMN Designs with corresponding to BPEL XML codes.....	38
V. RESULTS AND COMPARATIVE ANALYSIS	41
5.1 Results	41
5.2 Analysis	41
VI. CONCLUSIONS	48
6.1 Conclusion	48
6.2 Contributions and Limitations	49
6.3 Directions for Further Research	50
REFERENCES	50

APPENDICES	53
A. Example1 for business process model	53
B. Example2 for business process model	53
C. Example3 for business process model	53
D. Example4 for business process model.....	53
E. Example5 for business process model.....	53
F. Example6 for business process model.....	53
G. Example7 for business process model.....	54
H. Example8 for business process model.....	54
I. Example9 for business process model.....	54
J. Example10 for business process model.....	54
K. Example for the Proposed Complexity Metric Calculation.....	54
L. Example of Mapping Concurrent Processing	54
M. Example of Mapping Synchronization in Concurrent Processing	54
N. Example11 for business process model.....	54
O. Example12 for business process model.....	55
P. Example13 for business process model.....	55

LIST OF TABLES

TABLE

2.1 Overview 7PMG	9
2.2 Similarities between Software and Business Process.....	10
2.3 Cognitive Weights W for BPM elements	15
3.1 Complexity Metrics with Elements for Business Process Models.....	18
3.2 Cognitive Weights adapted to BPMN	20
5.1 Existing Complexity Metrics and CADAC Metric Calculations.....	42

LIST OF FIGURES

FIGURE

1.1: Flow Chart of the Outline of the thesis.....	4
2.1: Start Event, Message (Intermediate) Event, End Event	6
2.2: Task, Sub-Process, Call Activity.....	7
2.3: Exclusive (XOR), Parallel (AND), Inclusive (OR) Gateways.....	7
2.4: Sequence Flow	8
3.1: Example of the Proposed Metric Calculations	23
4.1: Sequential Concatenation	28
4.2: AND Concatenation	29
4.3: Example of Mapping Concurrent Processing	38
4.4: Example of Mapping Synchronization in Concurrent Processing	39

LIST OF ABBREVIATIONS

BPEL	-	Business Process Execution Language
BPM	-	Business Process Modelling
BPMs	-	Business Process Models
BPMN	-	Business Process Modelling Notation
CADAC	-	Cognitive Activity Depth Arc Control Flow
CC	-	Cognitive Complexity
CFC	-	Control-Flow Complexity
CNC	-	Coefficient of Network Complexity
EPC	-	Event-Driven Process Chains
IC	-	Interface Complexity
IEEE	-	The Institute of Electrical and Electronics Engineers
LOC	-	Lines of Code
NAOC	-	Number of Activities and Control flow elements
ND	-	Nesting Depth
NOA	-	Number of Activities
NOAJS	-	Number of Activities and Joins and Splits
OMG	-	Object Management Group
7PMG	-	Seven Process Modelling Guidelines
SW	-	Software

- UML - Unified Modelling Language
- XML - Extensible Markup Language
- YAWL - Yet Another Workflow Language

GCPRIS

CHAPTER I

INTRODUCTION

1.1 Statement of the Problem

Business process modelling is very essential in the discipline for business process management in organizations. A business process means “the series of steps that a business executes to produce a product or service” according to Rummler and Brache (Rummler-Brache, 1995). The aim of process models is to openly document the processes of organizations and provide a better understanding of the organizations. Process models often become more complicated and include various defects that prevent understanding and they can cause inefficient processes. Human understandability of the model may change person to person depending on having different training and skills. Furthermore, humans who are working for development in organization may have difficulties while interpreting certain parts of a business model although it does not seem to be enormously complicated in terms of mathematical approaches on complexity.

In general terms, metric is a measurement standard, and it is the basic measure for quality. It is measured every aspect of goods, services, and processes that influence quality with the help of metrics. By using metrics, you remove opinions and feelings, and replace them with facts, and reflects that validate or reject claim of commitment to quality. There are many types of metrics such as business, business process models, software, product, and so on.

Business metric means a unit of measurement that ensures a way to quantify a process objectively. A business metric can be any measurement that assists management to understand its operations. Examples of business metrics can be number of products made per hour, percent of failures from a process, hours required

to deliver a specific number of outputs or provide a service, etc. For this reason, metrics should implement in order to quantify the influences of these factors. Processes can not be understood if they are not measured. If processes can be understood, then they can be corrected, controlled and improved, and thus costs reduce while improving the quality of business outputs. It is measured the extent to which goods and services are supporting customer expectations.

Software metrics measures the frequency software failures or bugs, and they are used in order to determine the current product or process, predict and improve quality of a product or process. Software metrics are classified into three categories that are product metrics, process metrics, and project metrics. Product metrics defined as the characteristics of the product such as complexity, size, performance, design features and quality level. Process metrics is used to enhance software development and maintenance. Examples of process metrics cover the effectiveness of failure removal during development, the pattern of testing removal arrival, and the response time of the fix process. Project metrics identify the project characteristics and execution. Examples of project metrics involve the number of software developers, cost, schedule, and productivity. Software metrics cover software quality metrics that focus on the quality sights of the product, process, and project.

Software processes and business processes have certain extensive characteristics. There are numbers of researchers about similarities between object oriented software programming and business process design (Vanderfeesten, 2004; Cardoso, 2006b and Latva-Koivisto, 2001). Each software complexity metric has its corresponding metric for business process models (Gruhn- Laue, 2006a).

Complexity measures the simpleness and understandability of a design in business processes. Complexity metrics can be software complexity metrics and complexity metrics for business process models.

Business process models form a basis for communication between stakeholders that are business process analysts, domain experts, software developers, etc. For this reason, models in business process should be easy to understand and also maintain. In order to create models that are easy to understand, firstly it is defined what "easy

to understand”. This means, it is defined that a model is easy or difficult to understand by using measurements. Measures can provide us sufficient information with regard to understandability and maintainability of a BPM. The measurements of BPM are called Complexity Metrics. There are many complexity metrics for business process models (Cardoso, 2006a; Cardoso, 2006b; Latva-Koivisto, 2001; Mendling, 2006; Gruhn- Laue, 2006a), but each of the metric has a different aspect. In this thesis, the aim is to propose a new metric by considering different aspects of existing complexity metrics for business process models.

1.2 Scope and Outline of the Thesis

The purpose of this thesis is to propose a new complexity metric for business process models. For this reason, the first step is to investigate existing complexity metrics for business models. Then, a new complexity metric for business process models is created with respect to different aspects of existing complexity metrics and cognitive weights of elements in BPMs, and the new metric is verified by comparing the new metric that it is produced and existing complexity metrics. It is used only BPMN as BPM technique, since it is the most popular technique in the business process model techniques.

In Chapter 2, firstly background information for business process models is investigated, and secondly literature survey is investigated on complexity metrics for business models. In Chapter 3, impacts of elements in BPMs on complexity is investigated, and according to the these impacts, a new complexity metric for business process models is proposed. Furthermore, it is shown that how the proposed metric runs by using an example. Then, in Chapter 4 the proposed metric is verified theoretically. What is more, practical implication is done for the proposed metric. In chapter 5, results and comparative analysis for the proposed metric are specified with examples. Finally, in Chapter 6 conclusions as contributions, limitations and directions for further research are specified.

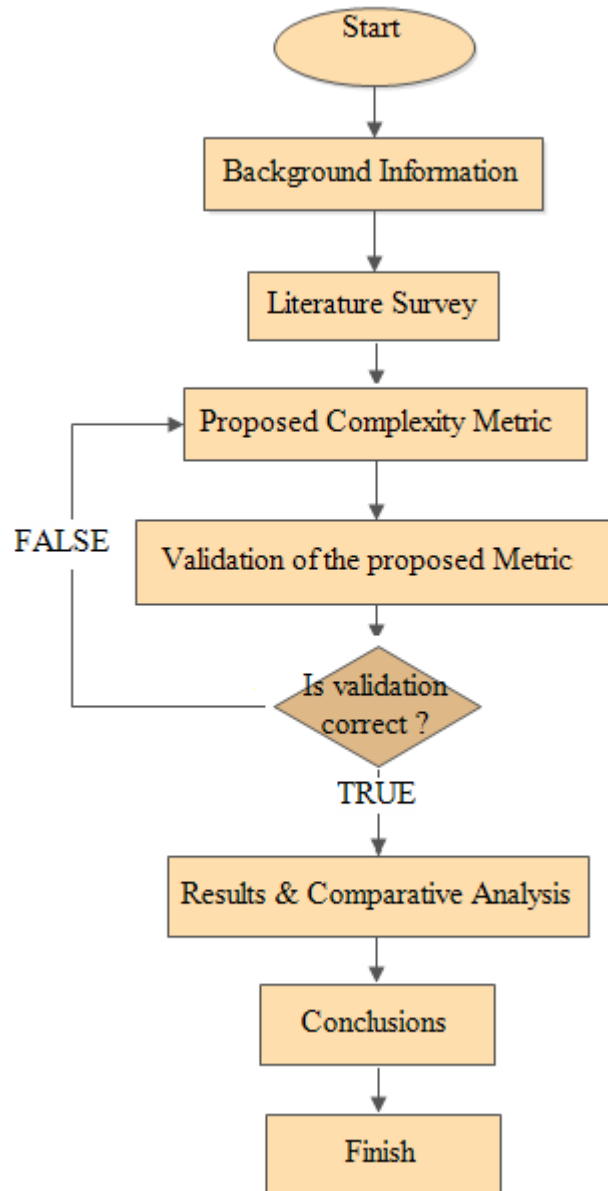


Figure 1.1: Flow Chart of the Outline of the thesis

According to the outline of the thesis, firstly background information has been searched, and then literature survey has been done about complexity metrics for BPMs. For this reason, approximately a hundred research papers have been found, but twenty five research papers that have the most important information about this thesis topic have been used. IEEE, ScienceDirect, Web of Science, Springer as e-resources databases have been searched, and the research has been done approximately fifteen hours per a week. In particular, preparing literature survey part has been hard to find advantages and disadvantages of existing complexity metrics for BPMs.

Then, based on background information and literature survey, it has been focused on proposing a new complexity metric. That is why, impacts of elements in BPMs on complexity has been considered, and according to the this impacts it has been proposed a new complexity metric for business process models.

After then, validation on the proposed metric has been done seven times because of existence of errors in the validation. When validation has not been correct, some changes on the proposed metric has been done. However, in the seventh validation, the proposed metric has been validated theoretically. What is more, practical implication on the proposed metric has been done. The validation part has been the hardest part of this work since it has been depend on the proposed metric, and it has been caused changing on the proposed metric. It has been studied on both the proposed metric and validation of the proposed metric nearly three months.

Finally, comparisons of the proposed metric with existing complexity metrics in result and comparative analysis part have been done. This has been the easiest part with conclusion part. Last two parts has been done in three weeks.

CHAPTER II

BACKGROUND INFORMATION AND LITERATURE SURVEY

2.1 Business Process Modelling

BPM (Business Process Modelling) is the most essential task for Business Process Management. The aim of Business Process Models is to document the processes of organizations clearly and provide a better understanding.

2.1.1 Business Process Modelling Techniques

In order to design process models there are many BPM techniques. The most common languages are Business Process Model and Notation (BPMN), Event-driven process chain (EPC), Flow Charts, Unified Modelling Language (UML) Activity Diagrams, and Yet Another Workflow Language (YAWL) and Petri nets. In the thesis, it has been used only BPMN for designing all models since it is the most popular technique in the business process model techniques.

2.1.2 Business Process Models Elements

It has been used the OMG standard Business Process Model and Notation (BPMN) 2.0 for designing all models in this thesis. BPMN 2.0 includes the following five basic categories:

- **Flow objects:** event, activity, gateway
- **Data:** data object, data input, data output, data store
- **Connecting objects:** arrows, these indicate sequence flow, message flow, association, data association
- **Swimlane:** pool, lane
- **Artifact:** group, text annotation

In this thesis, the following elements of BPMN have been used:

Events are something that happens during a process and may have impacts on the business process. They are parts of Control Flow Elements. An event can be at the beginning (**start event**), at the end (**end event**) or during (**intermediate event**) the process. In this thesis, events have not been considered.



Figure 2.1: Start Event, Message (Intermediate) Event, End Event

Activity: An activity can be either task, sub-processes or call activity.

- **Task** is an atomic activity in a process flow.
- **Sub-Process** is a separable activity.
- **Call Activity** includes re-usable tasks and processes in the diagram

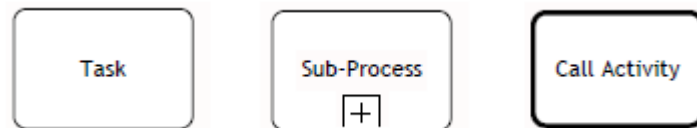


Figure 2.2: Task, Sub-Process, Call Activity

Task, sub-process and call activity as Activity have been used for the proposed metric.

Gateway: A gateway is used in order to control how sequence flows interact as they converge and diverge within a process. This means a gateway can act as split (converge) or join (diverge). Splits divide one flow arc into many (more than one), joins join multiple ones (many) into one. In this thesis, it has been taken account into only Exclusive, Parallel and Inclusive gateways, not the other gateways in order to calculate complexity for business process models. Complex, Event-Based and Parallel Event-Based gateways will be ignored for the proposed metric.



Figure 2.3: Exclusive (XOR), Parallel (AND), Inclusive (OR) Gateways

Split Gateway is divided into two types:

XOR (Exclusive) Split means it continues as soon as one of its incoming branches has been completed.

AND (Parallel) Split means it will wait until all incoming branches are completed before continuing.

Join Gateway is divided into three types:

XOR (Exclusive) Join means exactly one of the outgoing connections will be chosen.

AND (Parallel) Join means the control flow will continue for all outgoing connections simultaneously.

OR (Inclusive) Join means all outgoing connections whose condition evaluates to true those are selected.

Sequence Flow: A sequence flow is used to define the execution order of activities in a process. For each flow, there is only one source and only one target. In this thesis, it has been taken all Normal, Uncontrolled, Conditional, Default flows as Sequence Flow.

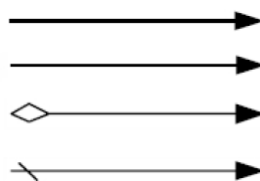


Figure 2.4: Sequence Flows

2.1.3 Complexity Metrics for Business Process Models

Business Process Models serve as a base for communication between the stakeholders in the software development process that are software designers, experts, business process analysts, managers, etc. In order to perform this, a business process model must be easy to understand and maintain. Hence, measures are useful since they can provide us suitable information about understandability and maintainability of the business process models. Complexity metrics for business process models are used to define whether a model is easy or difficult to understand.

2.2 Literature Survey

2.2.1 Existing Complexity Evaluation Approaches

2.2.1.1 Seven Process Modelling Guidelines (7PMG)

There are relationships between the model structure on the one hand error probability and understanding on the other hand according to seven process modelling guidelines (Mendling, 2010) and it is defined that how the guidelines are prioritized by industry experts.

Table 2.1: Overview 7PMG (Mendling, 2010)

G1	Use as few elements in the model as possible
G2	Minimize the routing paths per element
G3	Use one start and one end event
G4	Model as structured as possible
G5	Avoid OR routing elements
G6	Use verb-object activity labels
G7	Decompose a model with more than 50 elements

However, there is validity problem (Mendling, 2010) about 7PMG that does not relate to the content of a process model, but only to the way this content is organized and represented. It suggests ways of organizing different structures of the process model while the content is kept intact but the pragmatic issue of what must be

included in the model is still left out. The second limitation relates to the prioritizing guideline (Mendling, 2010) the derived ranking has a small empirical basis as it relies on the involvement of 21 process modellers only.

2.2.1.2 Similarities between Software and Business Processes

Key in many instances of innovation is the transfer of information and understanding developed in one discipline to the other (Kostoff, 1999). A business process model, regardless whether it is modelled in e.g. BPEL, EPC, BPMN or Petri Nets, exhibits many similarities with traditional software programs (Vanderfeesten, 2004). A business process model is made up of activities that contains operations on elementary data elements. Similarly, a software program consists of module or functions that cover inputs and provide outputs. The similarities are shown in Table 2.2.

Table 2.2: Similarities between Software and Business Process (Vanderfeesten, 2004)

Software	Business Process
Module/Class	Activity
Method/Function	Operation
Variable/Constant	Data Element

Furthermore, likewise interactions between modules and functions in a SW program are exactly defined, in a process model the order of activity is predefined using logic operators such as sequence, XOR-splits, OR-splits, and AND-splits. There is a mapping that can be constructed between software programs and business processes. Functions, procedures, or modules are mapped to activities. Two sequential software statements (i.e. instructions or functions) can be mapped to two sequential process activities. A 'switch' statement can be mapped to a XOR-split. In programs, threads can be used to model concurrency and can be mapped to AND-splits. Finally, the conditional creation of threads using a sequence of 'if-then' statements can be mapped to an OR-split (Cardoso, 2006a).

2.2.2 Existing Complexity Metrics for BPMs

One of the earliest and fundamental measures based on the analysis of software code is based on the basic count of the number of Lines of Code (LOC) of a program (Kalb, 1990). The simplest measure for the software metrics domain, the line of code metric is adapted as a number of activities (NOA) in the study of Cardoso (Cardoso, 2006a). The NOA metric characterizes only one specific view of size, namely length, it takes no account of functionality or complexity and it is not language-dependent and it is easier for users to understand. Another adaptation of the LOC metric for well-structured processes is to take into account as activities and process control-flow elements (i.e. control structures) (NOAC) (Cardoso, 2006a). On the other hand, for not well-structured processes like EPC and Workflow nets a third metric that counts the number of activities and the number of splits and joins (NOAJS) of a process is adapted (Cardoso, 2006a).

NOA = Number of activities (tasks and sub-processes) in a process

NOAC = Number of activities (tasks and sub-processes) and control-flow elements in a process

NOAJS = Number of activities (tasks and sub-processes), joins, and splits in a process.

The advantage of these metrics are, the "number of activities" is a simple, easy to understand measure for the size of a BPM.

Nevertheless, the "number of activities" metrics does not consider the structure of the model (Gruhn- Laue, 2006a). A BPM with 50 activities may be written using a well-structured control flow that is easy to understand, but in an unstructured control flow that makes understanding very hard.

There are four main types of complexity metrics for BPMs: **activity complexity**, **control flow complexity**, **data flow complexity**, **resource complexity** according to Cardoso (Cardoso, 2006b) and Latva-Koivisto (Latva-Koivisto, 2001).

The **Control-Flow Complexity (CFC) metric**, which borrows some ideas from McCabe's cyclomatic complexity (McCabe & Butler, 1989) is the number of mental states that have to be considered when a designer develops a process, and the CFC metric for BPMs counts the number of decisions in the flow of control (Cardoso, 2005). The CFCXOR – split, CFCOR – split, and CFCAND – split functions is calculated (Cardoso, 2006a) as follows :

The higher the value of CFCXOR – split, CFCOR – split, and CFCAND –split, the more complex is a process design, since developer has to handle all the states between control-flow constructs (splits) and their associated outgoing transitions and activities.

CFCXOR – split(a) = fan – out(a). The control-flow complexity of XOR-splits is determined by the number of branches that can be taken.

CFCOR– split(a) = 2fan – out(a) – 1. The control-flow complexity of OR-splits is determined by the number of states that may arise from the execution of an OR-split construct.

CFCAND – split(a) = 1. For an AND-split, the complexity is simply 1.

The higher the value of CFCXOR – split, CFCOR – split, and CFCAND –split, the more complex is a process design, since developer has to handle all the states between control-flow constructs (splits) and their associated outgoing transitions and activities.

The advantages of the CFC metric are that it can be used as a maintenance and quality metric, it gives the relative complexity of process designs, and it is easy to apply (Cardoso, 2006a).

The disadvantages of the CFC metric include the inability to measure data complexity, only control-flow complexity is measured (Cardoso, 2006a). Additionally, the same weight is placed on nested and non-nested loops. However,

deeply nested conditional structures are hard to understand than non-nested structures (Cardoso, 2006a).

Coefficient of Network Complexity (CNC) is a widely used metric in network analysis and was proposed to measure the degree of complexity of a critical path network (Latva-Koivisto, 2001). It is adapted as (**Coefficient of Network Complexity**) $CNC = \text{number of arcs} / (\text{no. of activities, joins, and splits})$ for using in business process (Cardoso, 2006a). CNC measure the degree of complexity of a critical path network.

The advantage of CNC is that the metric takes into account model structures.

Nevertheless, CNC seems too simple to capture the core of complexity, which is not balanced by its ease of computation or intuition.

Henry-Kafura (Henry-Kafura, 1981) proposed a metric based on the impact of the information flow in the software program's structure. The technique suggests identifying the number of calls to module (i.e. the flows of local information entering) as fan-in and the number of calls from a module (i.e. the flows of local information leaving) as fan-out. The complexity of a procedure (PC) is defined as: $PC = \text{Length} * (\text{fan-in} * \text{fan-out})^2$ Henry-Kafura (Henry-Kafura, 1981). This metric can be used in the same way for analyzing BPMs. The adapting of the Information Flow Metric for BPMs is **Interface Complexity (IC) metric** (Cardoso, 2006a) that is defined as:

$$IC = \text{Length} * (\text{fan-in} * \text{fan-out})^2$$

Fan-in: inputs of activities

Fan-out: outputs of activities

The length can be calculated using NOA.

The advantages of the IC metric are that it takes into account data-driven processes and it can be calculated prior to coding, during the design stage (Cardoso, 2006a).

The drawbacks of IC metric are that it can give complexity values of zero if an activity has no external interactions (Cardoso, 2006a). This typically only happens with the end activities of a process. This means, for example, EPC processes with a large percentage of end activities will have a low complexity.

For BPMs, the term well-structuredness is defined as: A model is well-structured if the split/join constructions are properly nested (Van der Aalst, 1998). Formally, well-structuredness is defined in terms of Petri net terminology (Van der Aalst, 1998), and it is modelled that BPM as workflow net (a special case of a Petri net). According to the research (Gruhn- Laue, 2006a), not being well structured in BPM informally means that there is a misfit between the split and join connectors. Using **Split-join ratio metric** (Mendling, 2006), the misfit between the split and join connectors in BPM can be calculated. A high value for split-join-ratio can be corrected by simply adding another unstructured element into the model which has too small split-join-ratio.

The disadvantage of split-join ratio metric is that the metric is too simple to measure unstructured model. Hence, split-join ratio metric is still under research.

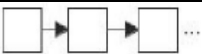
From the research about software complexity, it has been known that the metrics "maximum nesting depth" and "mean nesting depth" are suitable for measuring this factor which has influence on the overall complexity of the model: A greater nesting depth implies greater complexity (Gruhn- Laue, 2006a). The definition of the metrics "maximum nesting depth" and "mean nesting depth" for BPMs is straightforward: The **nesting depth** of an action is the number of decisions in the control flow that are necessary to perform this action (Gruhn- Laue, 2006a).


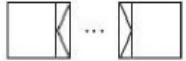



The advantage of the nesting depth model is that other than modern structured programming languages, common graph-oriented business modelling languages (for example UML activity diagrams or YAWL) do not require proper nesting, i.e. splits and joins does not have to occur pairwise (Gruhn- Laue, 2006a).



The disadvantage of the nesting depth model is that in the not well structured model may have a jump out of the control block. Thus, the splits and joins (control structure) are not properly nested.

Vanderfeesten (Vanderfeesten, 2008) introduced the Cross-Connectivity metric that measures the strength of the links between process model elements. Also, Cardoso (Cardoso, 2005) proposed the Control Flow Complexity metric and according to the metric mental states may be generated by a process model and the different types of routing elements. Both of two metrics are related to cognitive basis. Moreover, complexity metrics identifies the cognitive motion as a potential backbone according to Cardoso (Cardoso, 2006a). However, according to Gruhn- Laue (Gruhn- Laue, 2006b) it is argued that CFC is less useful if we define complexity as "difficulty to understand a model". In order to overcome the disadvantage of Control-Flow Complexity metric, **Cognitive Complexity Metrics (CC)** which is based on research about a piece of software by Shao- Wang (Shao- Wang, 2003) is introduced by Gruhn- Laue (Gruhn- Laue, 2006b). The cognitive weight of a BPM is the sum of the cognitive weights of a its elements (Gruhn- Laue, 2006b). The difficulty is measured by cognitive weight of a control structures in order to understand this control structure. Gruhn- Laue (Gruhn- Laue, 2006b) are introduced cognitive complexity metrics for bpm elements with YAWL symbols in Table 2.3.

Table 2.3: Cognitive Weights W for BPM elements (Gruhn- Laue, 2006b)

Workflow Pattern (Shao- Wang, 2003)	BPM control structure	corresponding software control structure	YAWL-Symbol	W
Sequence	consecutive steps in a work-flow	sequence		1

Exclusive Choice	XOR-split (exactly one of two branches is chosen) with corresponding XOR-join	branching with <i>if-then</i>		2
	XOR-split (exactly one of ≥ 3 branches is chosen) with corresponding XOR-join	branching with (with an arbitrary number of selectable cases)		3
Parallel Split and Synchronization	An AND-split activates all outgoing links in parallel, a corresponding AND-join synchronizes the flows of control	Execution of control flows in parallel		4
Multiple Choice and Synchronizing Merge	OR-split (a number of branches is chosen from 2 or more possible branches) with corresponding OR-join	branching with <i>case</i> , followed by parallel execution		7
(none)	Composite task (subtask, can be used for decomposing a BPM into modules)	Call of a user-defined function		2
Multiple Instances Patterns	Multiple Instance Activity (allows multiple instances of an activity to run concurrently).	branching, followed by parallel execution		6

Cancel Activity	Cancellation (by activating an activity one deactivates another one)	discussion (Gruhn- Laue, 2006b)		1
Cancel Case	Cancellation (by activating an activity one deactivates all elements within another part of the model)	comparable to a function		2 or 3

The disadvantage of Cognitive Complexity Metrics is that it is necessary to consider other concepts like cancellation or the multi-choice-pattern (Van der Aalst, 2003). This will be subject of further research, which should include validating experiments as well (Gruhn- Laue, 2006a).

Anti-patterns are recognized with discussing the use of "good" design patterns (Gustafsson, 2000). Anti-patterns commonly occurring solutions to a problem that are known to have negative consequences and in particular, finding anti-patterns in BPM should be very useful in order to uncover bad modelling style (Gruhn-Laue, 2006a). In a software complexity, a good design pattern helps to improve code quality, understandability and maintainability according to Gruhn-Laue (Gruhn-Laue, 2007).

Although, anti-patterns are quite useful, one will be able to fully leverage their potential when there is some kind of tool support and automatic detection, especially for large and complicated models (Baqae, 2009).

According to the literature survey on existing complexity metrics for business process models, there is no universally applicable solution as a complexity metric about understandability of business process models. However, cognitive weight

metric that proposed by Gruhn and Laue (Gruhn- Laue, 2006b) is the best at quantifiable measuring the understandability of existing BPMs.

GCRIIS

CHAPTER III

PROPOSED COMPLEXITY METRIC

Complexity metrics for BPMs have elements to measure complexity. Most of the complexity metrics are based on one or more elements of BPM. These elements are activity, control flow elements, depth, arc, handle, anti-pattern, cognitive weight.

Activity: task and sub-processes in a BPM

Control flow elements: events, AND-split, OR-split, XOR-split, AND-join, OR-join, XOR-join, fan-in, fan-out in a BPM

Depth: the number of decisions in the control flow that are necessary to perform the action for BPM (**Maximum Depth:** maximum nesting of structured blocks in a process model)

Arc: sequence flow in BPMs shows in which order the activities are performed

Handle: measuring undesirable jumps out of or into a structured control flow in BPMs

Cognitive weight: the sum of the cognitive weights of a business process model's elements in BPMs (Gruhn-Laue, 2006b)

Anti-pattern: counting the usage of anti-patterns in a BPM can help to detect poor modelling uncovering bad modelling style for BPMs

There are many complexity metrics for BPMs, that are pointed out in Chapter 2, with different types of elements as shown in Table 3.1. It has been considered how each of elements of BPM have an impact on complexity metrics for the thesis.

Table 3.1: Complexity Metrics with Elements for Business Process Models

Complexity Metrics	Complexity Metrics Elements						
	Activity	Control Flow	Depth (Max nesting depth)	Arc (sequence flow)	Handle	Anti-pattern	Cognitive Weight
1) NOA (Number of Activities in a Process Metric)	X						
2) NOAJS (Number of Activities, Joins and Splits in a Process Metric)	X	X					
3) NOAC (Number of Activities and Control Flow Elements in a Process Metric)	X	X					
4) CFC (Control Flow Complexity)		X					
5) CNC (Coefficient of Network Complexity)	X	X		X			
6) ND (Nesting Depth Metric)		X	X				

7) Split-Join Ratio		X			X		
8) IC (Interface Complexity)	X	X					
9) CC (Cognitive Complexity)							X
10) Anti-Pattern						X	

According to Latva and Koivisto (Latva-Koivisto, 2001), Gruhn and Laue (Gruhn-Laue, 2006a) no single metric can capture all aspects of a model's complexity. The authors suggest that different metrics should be used to include different aspects of a model's complexity. Thus, it has been considered all elements of BPM while proposing a new metric.

According to cognitive weight metric (Gruhn- Laue, 2006b), every element may not have equal impact on the complex of the BPMs, and different elements have different cognitive weights. For instance, sequence have less impact than any gateway with respect to cognitive weight for BPMs elements. Similarly, gateways have different weights among themselves, For example, XOR gateway has the smallest weight value, and according to number of splits it can have weight value as 2 or 3. AND gateway has weight value as 4, and OR gateway has the highest weight value as 7. Cognitive means comprehension effort for understanding.

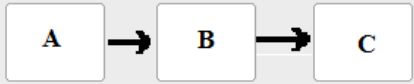
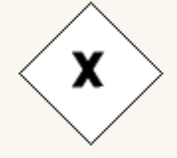
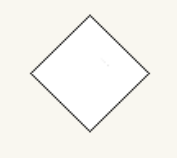
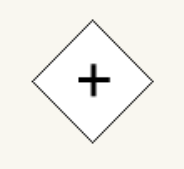
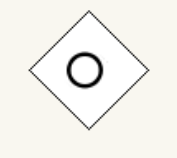
Considering the above details and literature survey, it is clear that cognitive weight metric proposed by Gruhn and Laue (Gruhn- Laue, 2006b) appears to be the best at quantifiably measuring the understandability of a BPM.


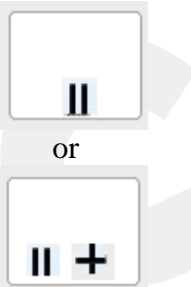
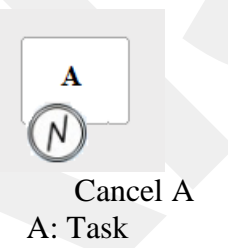
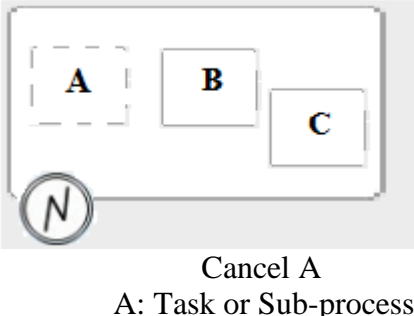
Hence, the proposed complexity metric has been based Cognitive Weight for BPM elements. Since software is the result of human creative activity, cognitive informatics plays an important role in understanding fundamental characteristics of

software according to Shao- Wang (Shao- Wang, 2003). Shao- Wang (Shao- Wang, 2003) proposed cognitive weights for the elements in the software.

Gruhn- Laue (2006b) adapted the similar approach to assign Cognitive Weight to BPMN elements as shown in Table 3.2.

Table 3.2: Cognitive Weights adapted to BPMN

Element	BPM Control Structure	BPMN	Weight (W)
Sequence Flow	Consecutive tasks or activities		1
Exclusive Gateway	XOR-split (exactly one of two branches is chosen) with corresponding XOR-join		2
	XOR-split (exactly one of ≥ 3 branches is chosen) with corresponding XOR-join		3
Parallel Gateway	An AND-split activates all outgoing links in parallel, a corresponding AND-join synchronizes the flows of control		4
Inclusive Gateway	OR-split (a number of branches is chosen from 2 or more possible branches) with		7

	corresponding OR-join		
(none)	Sub-processes		2
Multiple Instances Patterns	Multiple instances task		6
Cancel Activity	Cancellation (by activating an activity one deactivates another one)		1
Cancel Case	Cancellation (by activating an activity one deactivates all elements within another part of the model)		2 or 3

It has been proposed a new complexity metric for business process models also considering existing complexity metrics and elements of BPM. In the research, it has been seen that elements in BPMs have different influence on complexity. Control flow elements (control structures) have higher influence than number of activities according to Cardoso (Cardoso, 2006a). Moreover, nesting depth (Maximum Nesting Depth) (Gruhn- Laue, 2006a) has higher influence than control flow elements. It has been proposed a new complexity metric named as **CADAC**

(Cognitive Activity Depth Arc Control Flow) Metric considering cognitive weights (Gruhn- Laue, 2006b). According to cognitive weights (Gruhn- Laue, 2006b), weights to some elements for business process models have been given. It has been taken weights from the Cognitive Weights for BPM elements (Gruhn- Laue, 2006b) directly as: 1 for activities, 2 for Xor, 4 for And, 7 for Or. Also, It has been added number of Arc in the proposed metric since in the Cognitive Weights for BPM elements (Gruhn- Laue, 2006b) weight of Xor can be 2 or 3 depending on outgoing numbers. It has been taken weights of an Xor as 2 although it has greater than or equal to 3 branches, but it has been added weight 1 for each Arc and it has been added handled the problems about split numbers of XORs. Moreover, it has been added weight 4 for Maximum (fan-in x fan-out)² in CADAC calculation, (fan-in x fan-out)² is derived from Interface Complexity (IC) metric (Cardoso, 2006a). Lastly, it has been added weight 14 for Maximum Depth for CADAC Metric since depth has the highest influence on complexity in the elements. According to the proposed complexity metric CADAC, complexity is calculated as:

$$\text{CADAC Complexity} = (\text{NOA} \times 1) + (\text{Maximum Nesting Depth} \times 14) + (\text{Number of XORs} \times 2) + (\text{Number of ANDs} \times 4) + (\text{Number of ORs} \times 7) + (\text{Maximum (fan-in x fan-out)}^2 \times 4) + (\text{Number of ARCs} \times 1)$$

3.1 Example for the Proposed Complexity Metric Calculations

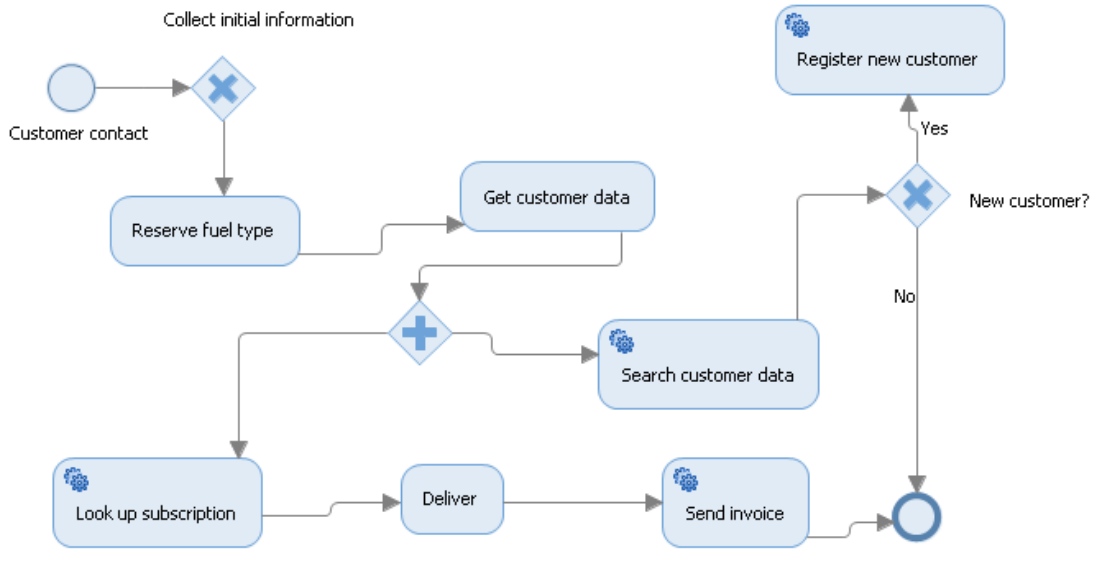


Figure 3.1 : Example for the Proposed Metric Calculations

It has been shown how the proposed complexity metric (CADAC) can be calculated using the example (Appendix K) above .

$$\text{CADAC Complexity} = (\text{NOA} \times 1) + (\text{Maximum Nesting Depth} \times 14) + (\text{Number of XORs} \times 2) + (\text{Number of ANDs} \times 4) + (\text{Number of ORs} \times 7) + (\text{Maximum (fan-in} \times \text{fan-out)}^2 \times 4) + (\text{Number of ARCs} \times 1)$$

According to the formula, firstly NOA, Maximum Nesting Depth, Number of XORs, Number of ANDs, Number of ORs, Maximum (fan-in x fan-out) and Number of ARCs should be calculated.

In the example, NOA = 7, Maximum Nesting Depth = 3 , Number of XORs = 2, Number of ANDs = 1, Number of ORs = 0, Maximum (fan-in x fan-out) = 1 X 1= 1, and Number of ARCs = 12.

Then, the complexity can be calculated according to CADAC metric.

$$\begin{aligned} \text{According to the example, the CADAC Complexity} &= (7 \times 1) + (3 \times 14) + (2 \times 2) \\ &+ (1 \times 4) + (0 \times 7) + ((1)^2 \times 4) + (12 \times 1) = 73 \end{aligned}$$

The complexity according to the CADAC metric is 73 for the example.

GCCRIS

CHAPTER IV

VALIDATON OF THE PROPOSED COMPLEXITY METRIC

4.1 Theoretical Validation

In the thesis for validating the proposed complexity metric CADAC it has been used Weyuker' s Properties (Weyuker, 1988) that are designed to evaluate software complexity measures. They should be satisfied by any good complexity measures. In the thesis it has been considered processes complexities, not software complexities. However, in Weyuker' s Properties there are concatenation operation of program blocks since they concern with software. Therefore, it has been used concatenation operations on processes (Cardoso, 2008).

Also, for validating CADAC it has been used Briand's Framework (Briand, 1996) that concerns software complexity measures. Concatenation operation in Briand's Framework are about software complexities. Therefore, it has been used concatenation operations on processes (Cardoso, 2008) for CADAC.

4.1.1 Weyuker' s Properties

Property 1

Non- Coarseness: $(\exists P)(\exists Q) (|P| \neq |Q|)$ where P and Q are two different systems.

It is required that a good metric should be able to discriminate between two different systems by returning different measurement results.

Let us take two processes P and Q . For P (example 1) in Appendix A, complexity metric measures are $NOA = 6$, Maximum Nesting Depth = 4, Number of XORs = 3,

Number of ANDs = 1, Number of ORs = 0, Maximum (fan-in x fan-out) = 3 X 1= 3, and Number of ARCs = 17. According to P, **CADAC** = (6 x 1) + (4 x 14) + (2 x 3) + (4 x 1) + (7 x 0) + ((3)² x 4) + (17 x 1) = 125. CADAC metric value for P is 125 for the example.

For Q (example 2) in Appendix B, complexity metric measures are NOA = 3, Maximum Nesting Depth = 1, Number of XORs = 1, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = 2 X 1= 2, and Number of ARCs = 6. According to Q, **CADAC** = (3 x 1) + (1 x 14) + (2 x 1) + (4 x 0) + (7 x 0) + ((2)² x 4) + (6 x 1) = 41. CADAC metric value for Q is 41for the example.

Hence, Property 1 is satisfied by the proposed metric CADAC.

Property 2

Granularity: *Let c be a non-negative number. Then there are only finitely many processes of complexity c.*

This property states that there will be a finite number of cases having the same metric value.

Let us take two processes A and B. For A (example 12) in Appendix O, complexity metric measures are NOA = 3, Maximum Nesting Depth = 2, Number of XORs = 0, Number of ANDs = 2, Number of ORs = 0, Maximum (fan-in x fan-out) = 1 X 1= 1, and Number of ARCs = 7. According to A, **CADAC** = (3 x 1) + (2 x 14) + (2 x 0) + (4 x 2) + (7 x 0) + ((1)² x 4) + (7 x 1) = 50. The CADAC metric value for A is 50.

For B (example 13) in Appendix P, complexity metric measures are NOA = 5, Maximum Nesting Depth = 2, Number of XORs = 2, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = 1 X 1= 1, and Number of ARCs = 9. According to B, **CADAC** = (5 x 1) + (2 x 14) + (2 x 2) + (4 x 0) + (7 x 0) + ((1)² x 4) + (9 x 1) = 50. The CADAC metric value for B is 50.

Since the universe of discourse deals with a finite set of applications, each of which has a finite number of process, this property is satisfied by CADAC.

Property 3

Non-uniqueness: *There are two distinct systems P and Q such that $|P| = |Q|$.*

This means that two different system designs may have the same metric value. A good metric should return same complexity for such processes.

Let us take two processes P (example 2) in Appendix B and Q (example 11) in Appendix N. Maximum (fan-in x fan-out) value for Process P is 2 with two fan-in and one fan-out. Also, maximum (fan-in x fan-out) value for Process Q is 2 with two fan-in and one fan-out. The number of the other elements of P and Q are the same. However, the design of P is different from the design of Q.

For P, complexity metric measures are NOA = 3, Maximum Nesting Depth = 1, Number of XORs = 1, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = $2 \times 1 = 2$, and Number of ARCs = 6. According to P, **CADAC** = $(3 \times 1) + (1 \times 14) + (2 \times 1) + (4 \times 0) + (7 \times 0) + ((2)^2 \times 4) + (6 \times 1) = 41$. CADAC metric value for P is 41.

For Q, complexity metric measures are NOA = 3, Maximum Nesting Depth = 1, Number of XORs = 1, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = $2 \times 1 = 2$, and Number of ARCs = 6. According to Q, **CADAC** = $(3 \times 1) + (1 \times 14) + (2 \times 1) + (4 \times 0) + (7 \times 0) + ((2)^2 \times 4) + (6 \times 1) = 41$. CADAC metric value for Q is also 41.

Hence, Property 3 is satisfied by the proposed metric CADAC.

Property 4

Design Implication: $(\exists P)(\exists Q) (P \equiv Q \text{ and } |P| \neq |Q|)$.

According to this property, although two processes that are P and Q may have the same functionality, the process's complexity is determined by the details of the design.

This property points out that implementation is important. There can be different activity flow interdependency for the same functionality. If there exist processes, P and Q with the same functionality such that they have same set of activities but they differ in their activity flow interdependencies making their implementation different and hence their complexities are different.

Let us take two designs P (example 2) in Appendix B and Q (example 12) in Appendix O that have the same activities and the numbers of activities are 3 for both of them. For P, the other complexity metric measures are Maximum Nesting Depth = 1, Number of XORs = 1, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = 2 X 1 = 2, and Number of ARCs = 6. According to P, **CADAC** = $(3 \times 1) + (1 \times 14) + (2 \times 1) + (4 \times 0) + (7 \times 0) + ((2)^2 \times 4) + (6 \times 1) = 41$. CADAC metric value for P is 41.

For Q, the other complexity metric measures are Maximum Nesting Depth = 2, Number of XORs = 0, Number of ANDs = 2, Number of ORs = 0, Maximum (fan-in x fan-out) = 1 X 1 = 1, and Number of ARCs = 7. According to Q, **CADAC** = $(3 \times 1) + (2 \times 14) + (2 \times 2) + (4 \times 0) + (7 \times 0) + ((1)^2 \times 4) + (9 \times 1) = 50$. CADAC metric value for Q is also 50.

Hence, Property 4 is satisfied by proposed metric CADAC.

Property 5

Monotonicity: $(\exists P) (\exists Q) (|P| \leq |P;Q| \ \& \ |Q| \leq |P;Q|)$

This property states that two interacting processes may have zero or more (but never negative) complexity to that which is present in the two initial processes themselves. This complexity is introduced whenever processes concatenate, and a good metric should be able to find out it.

In order to apply property 5 it should be considered concatenation operations for processes that are Sequence, XOR, AND, OR operations.

According to this property for any processes P and Q, the complexity of $P * Q, * \in \{-, \bullet, \oplus, 0\}$, is greater than or equal to the original complexity of P.

Case 1

Sequential Concatenation (-) : $(\forall P)(\forall Q) (P - Q \geq P)$

In the sequence concatenation when a process P is concatenated sequentially with a process Q, then the resulting process become P-Q (Cardoso, 2008) .

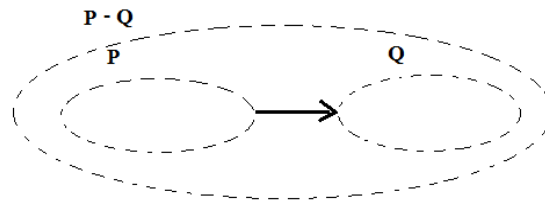


Figure 4.1 : Sequential Concatenation (Cardoso, 2008)

For two processes P (example 1) in Appendix A and Q (example 13) in Appendix P, CADAC for P is 125, and CADAC for Q is 50.

$|P - Q| = |P| + |Q| + (1 \times 1) = 125 + 50 + 1 = 176$, thus it is satisfied that $|P - Q| \geq |P|$ ($176 \geq 125$). The sequential concatenation operation the positivity also holds since $|P - Q| > |P|$.

Case 2

AND Concatenation (•) : $(\forall P)(\forall Q) (P \bullet Q > P)$

In the AND concatenation when a process P is concatenated with a process Q by using an AND-split and an AND-join, then the resulting process become $P \bullet Q$ (Cardoso, 2008).

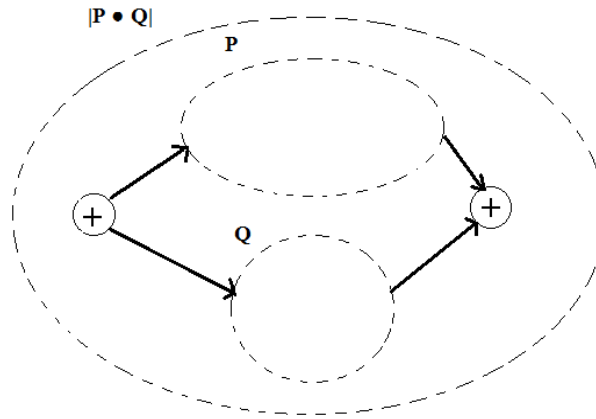


Figure 4.2 : AND Concatenation (Cardoso, 2008)

For the AND concatenation operation ' \bullet ', the weak positivity holds. For two processes P (example 1) in Appendix A and Q (example 13) in Appendix P, CADAC for P is 125, and CADAC for Q is 50.

$|P \bullet Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 4) + (1 \times 4) = 125 + 50 + 4 + 14 + 8 + 4 = 205$, thus it is satisfied that $|P \bullet Q| \geq |P|$ ($205 \geq 125$). The AND concatenation operation the positivity also holds since $|P \bullet Q| > |P|$.

Case 3

XOR Concatenation (\oplus): ($\forall P$)($\forall Q$) ($P \oplus Q > P$)

In the XOR concatenation when a process P is concatenated with a process Q by using an XOR-split and an XOR-join, then the resulting process become $P \oplus Q$ (Cardoso, 2008).

Likewise the AND concatenation above, XOR concatenation is calculated for two processes P (example 1) in Appendix A and Q (example 13) in Appendix P. CADAC for P is 125, and CADAC for Q is 50.

$|P \oplus Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 2) + (1 \times 4) = 125 + 50 + 4 + 14 + 4 + 4 = 201$, thus it is satisfied that $|P \oplus Q| \geq |P|$ ($201 \geq 125$). The XOR concatenation operation the positivity also holds since $|P \oplus Q| > |P|$.

Case 4

OR Concatenation (o): $(\forall P)(\forall Q)(P \circ Q > P)$

In the OR concatenation when a process P is concatenated with a process Q by using an OR- split and an OR-join, then the resulting process become P o Q (Cardoso, 2008).

Likewise the AND concatenation above, OR concatenation is calculated for two processes P (example 1) in Appendix A and Q (example 13) in Appendix P. CADAC for P is 125, and CADAC for Q is 50.

$|P \circ Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 7) + (1 \times 4) = 125 + 50 + 4 + 14 + 14 + 4 = 211$, thus it is satisfied that $|P \circ Q| \geq |P|$ ($211 \geq 125$). The XOR concatenation operation the positivity also holds since $|P \circ Q| > |P|$.

Hence, Property 5 is satisfied by the proposed metric CADAC.

Property 6

Non-equivalence of Interaction:

- a) $(\exists P)(\exists Q)(\exists R)(|P| = |Q| \ \& \ |P ; R| \neq |Q ; R| \ \& \ * \in \{-, \oplus, \bullet, o\})$
- b) $(\exists P)(\exists Q)(\exists R)(|P| = |Q| \ \& \ |R ; P| \neq |R ; Q| \ \& \ * \in \{-, \oplus, \bullet, o\})$

This property states that it is possible to have two identical processes, but when concatenated to a third same process, their complexity results are not equal. A good metric can discriminate between two identical processes.

In order to apply property 6 it has been considered concatenation operations for processes that are Sequence, XOR, AND, OR operations as property 5.

According to this property there exist processes P, Q, and R, such that $|P| = |Q|$ and $|P * R| \neq |Q * R|$, where $* \in \{-, o, \bullet, \oplus\}$.

Likewise property 5, property 6 has four distinct cases.

For P (example 12) in Appendix O CADAC is 50, Q (example 13) in Appendix P CADAC is 50, and R (example 2) in Appendix B CADAC is 41.

Case 1

Sequential Concatenation (-) : $|P - R| = |P| + |R| + (1 \times 1) = 50 + 41 + 1 = 92$ and $|Q - R| = |Q| + |R| + (1 \times 1) = 50 + 41 + 1 = 92$, since $|P| = |Q|$, it holds that $|P - R| = |Q| + |R| + 1 = 92$, thus $|P - R| = |Q - R|$.

Case 2

XOR Concatenation (\oplus): $|P \oplus R| = |P| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 2) + (1 \times 4) = 50 + 41 + 4 + 14 + 4 + 4 = 117$, and $|Q \oplus R| = |Q| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 2) + (1 \times 4) = 50 + 41 + 4 + 14 + 4 + 4 = 117$, since $|P| = |Q|$, it holds that $|P \oplus R| = |Q| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 2) + (1 \times 4) = 50 + 41 + 4 + 14 + 4 + 4 = 117$, thus $|P \oplus R| = |Q \oplus R|$.

Case 3

AND Concatenation (\bullet): $|P \bullet R| = |P| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 4) + (1 \times 4) = 50 + 41 + 4 + 14 + 8 + 4 = 121$, and $|Q \bullet R| = |Q| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 4) + (1 \times 4) = 50 + 41 + 4 + 14 + 8 + 4 = 121$, since $|P| = |Q|$, it holds that $|P \bullet R| = |Q| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 4) + (1 \times 4) = 50 + 41 + 4 + 14 + 8 + 4 = 121$, thus $|P \bullet R| = |Q \bullet R|$.

Case 4

OR Concatenation (\circ): $|P \circ R| = |P| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 7) + (1 \times 4) = 50 + 41 + 4 + 14 + 14 + 4 = 127$, and $|Q \circ R| = |Q| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 7) + (1 \times 4) = 50 + 41 + 4 + 14 + 14 + 4 = 127$, since $|P| = |Q|$, it holds that $|P \circ R| = |Q| + |R| + (4 \times 1) + (1 \times 14) + (2 \times 7) + (1 \times 4) = 50 + 41 + 4 + 14 + 14 + 4 = 127$, thus $|P \circ R| = |Q \circ R|$.

Hence, Property 6 is not satisfied by the proposed metric CADAC.

Property 7

Significance of Permutation: $(\exists P)(\exists Q)$ If Q is formed by permuting order of the activities of P , then $|P| \neq |Q|$.

According to this property there are two processes P and Q such that Q is formed by permuting order of the activities of P , and $|P|$ is not equal to $|Q|$.

This property specifies that permutation is significant. It means that permutation of elements within a process change metric value. The purpose is to provide that metric values change due to the permutation of process activities.

Let us take a process P that contains an AND-split and an OR-split. The other complexity metric measures are NOA= 6, Maximum Nesting Depth = 2 , Maximum (fan-in and fan-out) = 3, Number of XORs = 0, and Number of ARCs = 12. According to P , $CADAC = (6 \times 1) + (2 \times 14) + (2 \times 0) + (4 \times 1) + (7 \times 1) + ((3)^2 \times 4) + (12 \times 1) = 93$. CADAC metric value for P is 93.

Let us assume that process Q is a permutation of the activities of process P . This means, the AND- split and OR-split are exchanged. For Q , the other complexity metric measures are the same as P that are NOA= 6, Maximum Nesting Depth = 2, Maximum (fan-in and fan-out) = 3, Number of XORs = 0, and Number of ARCs = 12. According to P , $CADAC = (6 \times 1) + (2 \times 14) + (2 \times 0) + (4 \times 1) + (7 \times 1) + ((3)^2 \times 4) + (12 \times 1) = 93$. CADAC metric value for Q is also 93.

Hence, Property 7 is not satisfied by the proposed metric CADAC.

Property 8

No change on renaming: If P is a renaming of Q , then $|P| = |Q|$.

This property specifies that uniformly changing activity names should not affect complexity of a process. If P and Q are two processes then renaming of P to Q does not change their system complexity, i.e. $|P| = |Q|$.

This property requires that when the name of processes or activities change, metric should remain unchanged. As, the proposed metric does not depend on the name of processes or activities, this property is satisfied by CADAC.

Property 9

Interaction Complexity: P and Q for which $|P| + |Q|$ is less than $|P \circledast Q|$.

This property states that interaction between parts of a process cause additional positive complexity i.e., it makes additional complexity a requirement when two processes keep on interacting for some time, or as the process grows with age. Since, growth in process complexity occurs when new nodes are added and none of the nodes has negative values, then it is clear that the complexity of the new process is always equal to or greater than the sum of the two original processes. A good metric should be able to find out this change in behaviour.

$$(\exists P)(\exists Q) (|P \circledast Q| > |P| + |Q| \ \& \ \circledast \in \{-, \oplus, \bullet, \circ\})$$

This property states that the complexity of a process formed by concatenating two processes is greater than the sum of their complexities for at least in some cases. This means there may be interactions between the concatenated processes.

In order to apply property 9 it should be considered concatenation operations for processes that are Sequence, XOR, AND, OR operations as property 5 and 6.

For two processes P (example 1) in Appendix A and Q (example 13) in Appendix P, CADAC for P is 125, and CADAC for Q is 50.

Case 1

Sequential Concatenation (-) : $|P - Q| = |P| + |Q| + (1 \times 1) = 125 + 50 + 1 = 176$, thus $|P - Q| \geq |P| + |Q|$ ($176 \geq 175$).

Case 2

XOR Concatenation (\oplus): $|P \oplus Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 2) + (1 \times 4) = 125 + 50 + 4 + 14 + 4 + 4 = 201$, thus $|P \oplus Q| \geq |P| + |Q|$ ($201 \geq 175$).

Case 3

AND Concatenation (●): $|P \bullet Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 4) + (1 \times 4) = 125 + 50 + 4 + 14 + 8 + 4 = 205$, thus $|P \bullet Q| \geq |P| + |Q|$ ($205 \geq 175$).

Case 4

OR Concatenation (○): $|P \circ Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 7) + (1 \times 4) = 125 + 50 + 4 + 14 + 14 + 4 = 211$, thus $|P \circ Q| \geq |P| + |Q|$ ($211 \geq 175$).

Hence, Property 9 is satisfied by the proposed metric CADAC.

Interoperability Property (Cardoso, 2008)

Measures must be interoperable (i.e., independent of the process specification language). If $P = Q$, possibly specified with different languages (i.e., $P \in L_p$ and $Q \in L_q$, where L_p and L_q are process modelling languages), then $|P| = |Q|$ (Cardoso, 2008). The proposed metric CADAC can be applied not only BPMN language but also the other BPM languages by providing transformation of any BPM language to BPMN. Hence, Interoperability Property (Cardoso, 2008) is satisfied by the proposed metric CADAC since not only BPMN language that it has been taken account into for the thesis but also the other BPM languages include activity, control-flow, arc, depth, AND, OR, XOR split and join, fan-in and fan-out elements.

4.1.2 Briand's Framework

Briand's generic measurement framework (Briand, 1996) categorizes software metrics into five as size, length, complexity, cohesion and coupling. In this thesis, it has been used only complexity for validating the proposed metric CADAC since it has been considered only complexity.

Property 1

Non-negativity: $Complexity(S) \geq 0$

The complexity of a system $S = \langle E, R \rangle$ is non-negative.

In a BPMN, number of activities is always greater than or equal to zero, but never less than zero. Also, number of Arc, XOR, AND, OR, Maximum (fan-in & fan-out)

and Maximum Depth values can be zero or greater than zero, but not negative. These means it could happen that $CADAC = 0$ or $CADAC > 0$, but never $CADAC < 0$.

Hence, Property 1 is satisfied by the proposed metric CADAC.

Property 2

Null Value: $R = \emptyset \Rightarrow Complexity(S) = 0$

The complexity of a system $S = \langle E, R \rangle$ is null if R is empty.

If there is no activity (i.e., $NOA = 0$), then the other elements that are number of Arc, XOR, AND, OR, Maximum (fan-in & fan-out) and Maximum Depth values are zero since all these elements exist depending on existence of activity. Thus, $CADAC = 0$.

Hence, Property 2 is satisfied by the proposed metric CADAC.

Property 3

Symmetry: $(S = \langle E, R \rangle \text{ and } S^{-1} = \langle E, R^{-1} \rangle) \Rightarrow Complexity(S) = Complexity(S^{-1})$

The complexity of a system $S = \langle E, R \rangle$ does not depend on the convention chosen to represent the relationships between its elements.

The number of elements in CADAC does not depend on the convention used to represent the elements.

Let us take two processes P and Q. Maximum (fan-in x fan-out) value for Process P is 2 with two fan-in and one fan-out. Although Process Q has one fan-in and two fan-out that is different from P design, Q has the same Maximum (fan-in x fan-out) value that is 2.

For P, complexity metric measures are $NOA = 6$, Maximum Nesting Depth = 2, Number of XORs = 2, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = $2 \times 1 = 2$, and Number of ARCs = 12. According to P, **CADAC** = $(6 \times 1) + (2 \times 14) + (2 \times 2) + (4 \times 0) + (7 \times 0) + ((2)^2 \times 4) + (12 \times 1) = 66$. CADAC metric value for P is 66.

For Q, complexity metric measures are NOA = 6, Maximum Nesting Depth = 2, Number of XORs = 2, Number of ANDs = 0, Number of ORs = 0, Maximum (fan-in x fan-out) = 2 X 1 = 2, and Number of ARCs = 12. According to Q, **CADAC** = (6 x 1) + (2 x 14) + (2 x 2) + (4 x 0) + (7 x 0) + ((2)² x 4) + (12 x 1) = 66. CADAC metric value for Q is also 66.

Hence, Property 3 is satisfied by the proposed metric CADAC.

Property 4

Module Monotonicity: ($S = \langle E, R \rangle$ and $m1 = \langle Em1, Rm1 \rangle$ and $m2 = \langle Em2, Rm2 \rangle$ and $m1 \cup m2 \subseteq S$ and $Rm1 \cap Rm2 = \emptyset$) \Rightarrow Complexity (S) \geq Complexity(m1) + Complexity(m2)

The complexity of a system S = $\langle E, R \rangle$ is no less than the sum of the complexities of any two of its modules with no relationships in common.

($\exists P$)($\exists Q$) ($|P * Q| > |P| + |Q|$ & $*$ $\in \{-, \oplus, \bullet, \circ\}$)

This property states that the complexity of a process formed by concatenating two processes is greater than the sum of their complexities for at least in some cases. This means there may be interactions between the concatenated processes.

In order to apply property 4 it should be considered concatenation operations for processes that are Sequence, XOR, AND, OR operations.

For two processes P (example 1) in Appendix A and Q (example 13) in Appendix P, CADAC for P is 125, and CADAC for Q is 50.

Case 1

Sequential Concatenation (-) : $|P - Q| = |P| + |Q| + (1 \times 1) = 125 + 50 + 1 = 176$, thus $|P - Q| \geq |P| + |Q|$ ($176 \geq 175$).

Case 2

XOR Concatenation (\oplus): $|P \oplus Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 2) + (1 \times 4) = 125 + 50 + 4 + 14 + 4 + 4 = 201$, thus $|P \oplus Q| \geq |P| + |Q|$ ($201 \geq 175$).

Case 3

AND Concatenation (•): $|P \bullet Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 4) + (1 \times 4) = 125 + 50 + 4 + 14 + 8 + 4 = 205$, thus $|P \bullet Q| \geq |P| + |Q|$ ($205 \geq 175$).

Case 4

OR Concatenation (o): $|P \circ Q| = |P| + |Q| + (4 \times 1) + (1 \times 14) + (2 \times 7) + (1 \times 4) = 125 + 50 + 4 + 14 + 14 + 4 = 211$, thus $|P \circ Q| \geq |P| + |Q|$ ($211 \geq 175$).

Hence, Property 4 is satisfied by the proposed metric CADAC.

Property 5

Disjoint Module Additivity: ($S = \langle E, R \rangle$ and $S = m1 \cup m2$ and $m1 \cap m2 = \emptyset$)
 $\Rightarrow \text{Complexity}(S) = \text{Complexity}(m1) + \text{Complexity}(m2)$

The complexity of a system $S = \langle E, R \rangle$ composed of two disjoint modules $m1, m2$ is equal to the sum of the complexities of the two modules.

Let $m1$ and $m2$ be any two disjoint pools in S such that $S = m1 \cup m2$. Obviously, CADAC values are $S = m1 + m2$.

Hence, Property 5 is satisfied by the proposed metric CADAC.

4.2 Practical Implication

Business Process Modelling Notation (BPMN) has been developed in order to provide a standard notation readily understandable for business analysts, business managers and technical developers. BPMN is supported by various modelling tools and BPMN models can be directly executed by Business Process Execution Language for Web Services (BPEL) since BPEL is a mainly block-structured language supported by several execution platforms. In this way, a standardized bridge for the gap between the business process design and process implementation is created by BPMN.

4.2.1 Examples of BPMN Designs with corresponding to BPEL XML codes

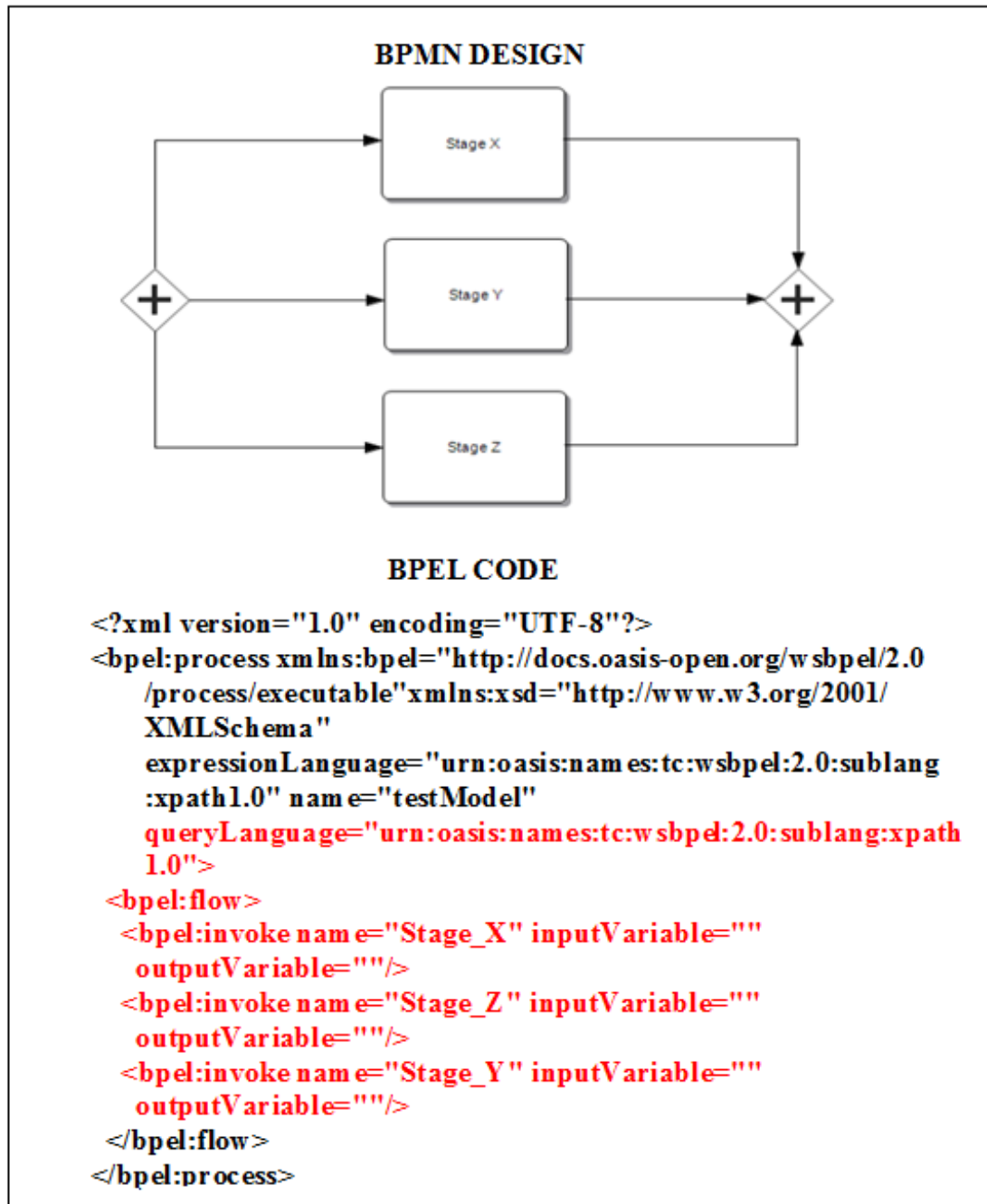


Figure 4.3 : Example of Mapping Concurrent Processing (Appendix L)

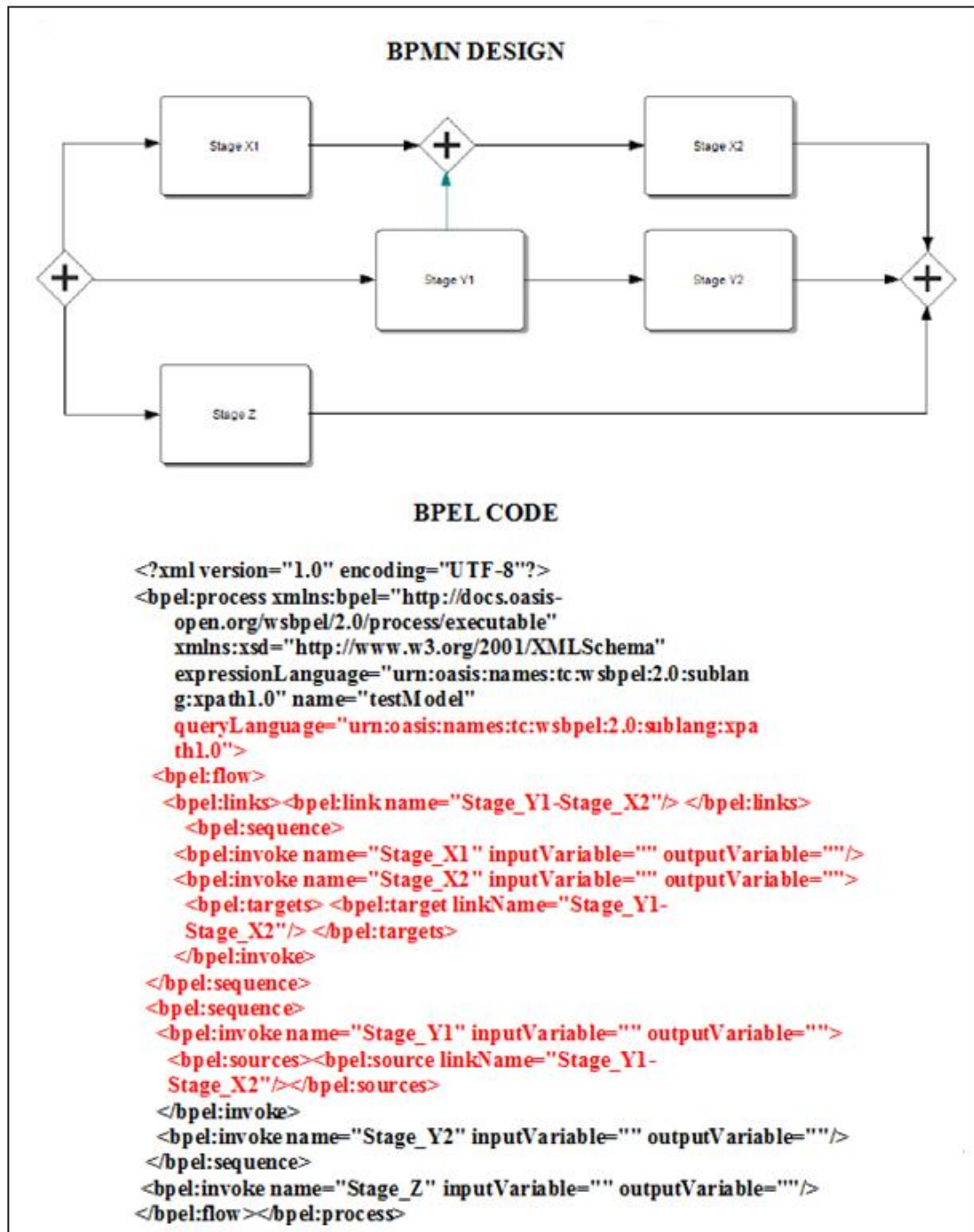


Figure 4.4 : Example of Mapping Synchronization in Concurrent Processing (Appendix M)

The proposed complexity metric CADAC is designed calculation for business process models. Based on transforming BPMN to BPEL XML source code, CADAC metric has understandability as quality attribute.

CADAC metric is directly linked to a programmer's comprehension effort when a source code is reused in XML design. CADAC metric takes into account the complexity within the XML format. Therefore, process models with higher CADAC values will take more effort in understanding. Therefore, these models are more difficult to reuse than models with lower CADAC.

Kaner (Kaner, 2004) proposed a framework for the evaluation of a SW metric, according the framework the proposed metric as follows:

The purpose of the metric: The purpose of Cognitive Activity Depth Arc Control Flow (CADAC) metric is to predict the complexity of a process model in terms of understandability as quality attribute.

Scope of the metric: The scope of CADAC metric is a business process model in BPMN language.

Attributes measured by the metric: CADAC metric measures the complexity of a process model in terms of understandability as quality attribute.

Natural scale of the attribute: Quality attribute understandability can be measured according to ordinal scale.

Natural variability of the attribute: Quality attributes are subjective, so they can be comment different from person to person. Understandability of the same process model may change according to the perspective of humans.

Metrics definition: CADAC complexity metric definition is specified in Chapter 3.

Measuring instrument to perform the measurement: The measuring instrument for CADAC can be counted by a human or a machine. CADAC metric can be calculated either manually by humans or a tool.

Natural scale of the metric: The natural scale of CADAC metric is ratio scale.

The natural variability of the readings from the measuring instrument: Reading from the measuring instrument (either manually or a tool) is based on mathematical formulas and in objective nature. Hence, no variability of readings (measurement error) from the instrument is expected.

Relationship of the attribute to the metric value: CADAC metric is related to the quality of process models. If the value of CADAC metric increases for a process, the quality (understandability) of that process will decrease.

The natural and foreseeable side effects of using the measuring instrument: CADAC metric calculations manually may provide an incorrect result so, using a tool in order to calculate this metric values will be a more reliable. What is more, it will take less efforts in the measurement.

CHAPTER V

RESULTS AND COMPARATIVE ANALYSIS

5.1 Results

In order to constitute a new metric that evaluate complexity of a BPMN model, complexity metrics elements for business process models and their cognitive weights according to Gruhn-Laue (Gruhn-Laue, 2006b) are used. Existing complexity metrics have advantages and disadvantages that are specified in chapter 2. Elements of BPMs; activity, control flow, depth, arc, cognitive weight have different impacts on complexity, so these different impacts have been considered while creating a new metric in chapter 3.

5.2 Analysis

In chapter 5, comparisons of existing complexity metrics and CADAC have been done.

Table 5.1: Existing Complexity Metrics and CADAC Metric Calculations

Examples	NOA	NOAJS	NOAC	IC	CNC	CFC	ND	CADAC
1	6	10	15	54	1.70	5	4	125
2	3	4	6	12	1.50	2	1	41
3	6	8	10	24	1.50	5	2	66
4	5	6	8	20	1.33	2	1	45
5	5	11	13	5	1.36	8	2	78
6a	3	4	7	3	1.50	2	1	29
6b	6	8	10	6	1.25	2	2	56
7	7	8	14	28	1.75	3	1	53
8a	3	5	8	3	1.60	3	2	47
8b	2	4	8	2	2.00	3	2	46
9	10	13	15	40	1.30	6	3	91
10	4	6	8	4	1.33	4	2	58

Table 5.1 shows complexity calculations of existing metrics and CADAC Metric for 10 BPMN examples from different domains that it has been selected randomly. According to the calculations in Table 5.1 there are different results about complexity for the different BPM examples.

It can be understood for a complexity metric, the bigger difference between two examples the better differentiated complexity. Thus, when the values of examples are the same, than complexity cannot be differentiated for the examples, and it can be said the examples have the same complexity.

For examples 1 and 5, according to the CADAC metric, example 1 (the value is equal to 125) is more complex than 5 (the value is equal to 78). But, according to the NOAJS metric the complexity is vice versa. The NOAJS metric only consider number of activities plus number of gateways (AND, OR, XOR, etc.). However, CADAC metric considers not only activities and gateways with their splits but also other elements that are arc (sequence flow), maximum depth and maximum (fan-in and fan-out). Although number of activities, splits of gateways number are more in example 5, number of maximum depth and maximum (fan-in & fan-out) are higher in example 1. Since, maximum nesting depth has higher influence on complexity than control flow elements according to Gruhn-Laue (Gruhn-Laue, 2006a), it is verified that example 1 is more complex than example 5. Also, for examples 2, 6a and 8b, according to the NOAJS metric all the examples have the same complexity with the value 4. However, according to the CADAC metric the complexity of example 2 (the value is 41) is higher than example 6a (the value is 29) since the value of maximum (fan-in and fan-out) for example 2 is 2 that is higher than the value of maximum (fan-in and fan-out) for example 6 is 1. Example 8b has the higher complexity (the value 46) among examples 2, 6a and 8b because example 8b have nesting depth value as 2, number of XOR gates as 2, number of arcs as 8, and those values are higher than example 6a that have nesting depth value as 1, number of XOR gates as 1, number of arcs as 6, and example 2 that have nesting depth value as 1, number of XOR gates as 1, number of arcs as 6.

In example 1 and 9, the NOAC metric has the same results, but CADAC metric for example 1 is 125 and for example 9 is 91. According to the CADAC, CNC, ND, IC

metrics example 1 is more complex than example 9. If it has been looked at the other metrics NOA, NOAJS and CFC, the complexity is vice versa. Nesting depth has higher influence than number of activities and control flow elements according to Gruhn- Laue (Gruhn-Laue, 2006a). According to CADAC formula, nesting depth has highest impact among elements in CADAC (nesting depth weight value = 14 in CADAC). Example 1 has nesting depth value as 4, and example 9 has nesting depth value as 3. Also, example 1 has maximum (fan-in and fan-out) value as 3 and example 9 has maximum (fan-in and fan-out) value as 2. Hence, example 1 is more complex than example 9 according to CADAC.

In example 2 and 4, the CFC and ND metrics have the same results, but for example 2 CADAC value is 41 and for example 4 CADAC value is 45, that shows example 4 is more complex than example 2. If it has been looked at the other metrics, all of them have the same results except for CNC values. As CNC, it is vice versa. Both of the examples 2 and 4 have the same maximum (fan-in * fan-out), but number of arc in example 2 is 6 and in example 4 is 8. Thus, example 2 has higher CNC than example 4. Also, the NOA results show example 4 is more complex than example 2. This means for IC results, example 4 is more complex than example 2. According to the CADAC, the impact of number of arc ($\text{arc} * 1$) is less than the impact of maximum (fan-in and fan-out) ($\text{maximum}(\text{fan-in} * \text{fan-out})^2 * 4$). Hence, CADAC metric act as IC metric but not as CNC metric for example 2 and example 4. For examples 2, 4 and 7, according to the ND metric all the examples have the same complexity with the nesting depth as 1. However, according to the CADAC metric the complexity of example 4 (the value is 45) is higher than example 2 (the value is 41), and example 7 has the highest complexity (the value 53) among examples 2, 4 and 7.

For examples 2 and 5, according to the CADAC metric, example 5 (the value is equal to 78) is more complex than 2 (the value is equal to 41). But, according to the CNC metric the complexity is vice versa. The CNC metric only considers number of arcs, activities, gateways. However, the CADAC metric does not only take the number of arcs, activities and gateways into account, but the structure of the model as well by considering maximum nesting depth value. Also, for examples 2, 3 and 6a, according to the CNC metric all the examples have the same complexity with the

value 1.50. However, according to the CADAC metric the complexity of example 2 (the value is 41) is higher than example 6a (the value is 29), and example 3 has the highest complexity (the value 66) among examples 2, 3, and 6a. According to example 2 and 5, the values of NOA, NOAJS, NOAC, CFC, ND, CADAC metrics for example 5 is more complex than example 2. For IC and CNC values it is vice versa. Since example 5 have 2 OR gates, 2 XOR gates, 2 AND gates, nesting depth value as 2 and number of arc as 15 those values are higher than example 2 that have 1 XOR gate, nesting depth value as 1 and number of arc as 6, complexity of example 5 is higher than example 2 according to CADAC metric.

In example 2 and 6a, the NOA, NOAJS, CNC, CFC, ND metrics have the same result, but in the CADAC metric the results for example 2 is 41 and for example 6a is 29. Thus, according to the CADAC metric, example 2 is more complex than example 6a. If it has been looked at the IC metric it is verified that example 2 is more complex than example 6a. However, the NOAC metric shows vice versa since NOAC takes into account activities, gateways and events. Number of activities, gateways have the same numbers for example 2 and 6a but, number of events for example 2 is 2 and example 6a is 3. Therefore according to NOAC metric example 6a is more complex than example 2.

According to the CADAC metric, example 3 (the value is equal to 66) is more complex than example 7 (the value is equal to 53). But, according to the NOAC metric, the complexity is vice versa. The NOAC metric only considers number of activities and number of control flow elements. However, the CADAC metric considers not only number of activities and control flow elements but also number of arc (sequence flow), and maximum depth. Although number of activities plus control flow elements is higher in example 7, the number of maximum depth is higher in example 3. Since, Maximum nesting depth have higher influence on complexity than control flow elements according to Gruhn-Laue (Gruhn-Laue, 2006a), it is verified that 3 is more complex than 7. Furthermore, for examples 4, 8a, 8b and 10, according to the NOAC metric all the examples have the same complexity with the value 8. However, according to the CADAC metric the complexity of example 8a (the value is 47) is bigger than example 8b (the value is 46), and example of 8b is higher than 4

(the value is 45). Example 10 has the highest complexity (the value 58) among examples 4, 8a, 8b and 10.

In examples 4 and 8b, the NOAC metric have the same result that is 8. According to the CADAC, CFC, ND, CNC, example 8b is more complex than example 4. But, according to the values of NOA, NOAJS and IC, it is vice versa. According to the NOA, example 4 is more complex than 8b as it only considers number of activities. Control flow elements have higher influence on complexity than number of activities according to Cardoso (Cardoso, 2006a). Although, the CADAC values have very close values (for example 4 is 45, example 8b is 46) gives the same result as CFC that shows example 8b is more complex than example 4.

For examples 5 and 9, according to the CADAC metric, example 9 (the value is equal to 91) is more complex than example 5 (the value is equal to 78). But, according to the CFC metric the complexity is vice versa. The CFC metric only considers counting the number of decisions (AND, OR, XOR splits) in the flow of control. However, the CADAC metric considers number activities, control flow elements, arc (sequence flow) and maximum depth. Since, Maximum nesting depth have higher influence on complexity than CFC metric according to Gruhn-Laue (Gruhn-Laue, 2006a), it is verified that example 9 is more complex than example 5. Furthermore, for examples 2, 4, 6a, and 6b, according to the CFC metric all the examples have the same complexity with the value 2. However, according to the CADAC metric the complexity of example 4 (the value is 45) is bigger than example 2 (the value is 41), and example of 2 is bigger than 6a (the value is 29). Example 6b has the biggest complexity (the value 56) among examples 2, 4, 6a, and 6b. The complexity of example 9 is higher than example 5 according to NOA, NOAJS, NOAC, IC, ND, CADAC. However, for CNC, CFC it is vice versa. Since nesting depth has higher influence on complexity than control flow elements according to CADAC, example 9 is more complex than example 5 with respect to CADAC. By contrast, according to the CFC, it is vice versa since CFC metric considers only AND OR XOR splits control flow elements.

For examples 6a and 8b in the Table 4.1, according to CADAC metric, example 8b (the value is equal to 46) is more complex than example 6a (the value is equal to 29).

But, according to the NOA metric the complexity is vice versa. The NOA metric only consider number of activities. However, the CADAC metric considers not only number of activities but also number of control flow elements, arc (sequence flow), maximum depth. Although number of activities is higher in example 6a, number of control flow elements and depth are higher in example 8b. Since, control flow elements have higher influence on complexity than number of activities according to Cardoso (Cardoso, 2006a), it is verified that 8b is more complex than 6a. Also, for examples 2, 6a and 8a, according to the NOA metric all the examples have the same complexity with the value 3. However, according to the CADAC metric the complexity of example 2 (the value is 41) is bigger than example 6a (the value is 29), and example 8a has the biggest complexity (the value 47) among examples 2, 6a and 8a since CADAC consider different elements that have different cognitive weights.

In examples 6b and 8a, the ND metric have the same result that is 2. The values of NOA, NOAJS, NOAC, IC, CADAC show that 6b is more complex than 8a. Whereas, the values of CNC, and CFC is vice versa. However, the CFC values of examples 6b (2) and 8a (3) are very close.

Results of NOA, NOAC, IC, CNC, CFC for example 7 is more complex than example 6b. However, for ND and CADAC values it is vice versa. In examples 6b and 7, the NOAJS metric have the same result that is 8. All the other values of example 6b and example 7 are very close except for IC, CNC, NOAC, but the ND for example 6b is more complex than 7b with very close values. Also, the CADAC for example 6b is more complex than 7b with very close values.

For examples 6b and 10, according to the CADAC metric, example 10 (the value is equal to 58) is more complex than 6b (the value is equal to 56). But, according to the IC metric the complexity is vice versa. The IC metric only considers number of activities and number of maximum (fan-in & fan-out). Control flow elements also include number of fan-in and fan-out. However, the CADAC metric considers not only number of activities and maximum (fan-in & fan-out) but also number of the other control flow elements, arc (sequence flow) and maximum depth. Since, maximum nesting depth has higher influence on complexity than control flow elements according to Gruhn- Laue (Gruhn- Laue, 2006a), it is verified that example

10 is more complex than example 6b. Besides, for examples 6a and 8a, according to the IC metric both of the examples have the same complexity with the value 3. However, according to the CADAC metric the complexity of example 8a (the value is 47) is bigger than example 6a (the value is 29).

In examples 7 and 8a, the CFC metric have the same result that is 3. According to the NOA, NOAC, NOAJS, IC, CNC metrics, example 7 is more complex than example 8a. For ND and CADAC values, it is vice versa. In CADAC formula, the ND has the highest impact ($ND * 14$), so the CADAC act as the ND.

In examples 7 and 8b, the CFC metric have the same result that is 3. According to the NOA, NOAC, NOAJS, IC and CADAC metrics example 7 is more complex than example 8a. For ND, CNC values, it is vice versa. Although example 8b have higher values from example 7 in terms of nesting depth and XOR gateway, example 7 have higher values from example 8b in terms of number of activity, arc, maximum (fan-in and fan-out). Hence, with respect to CADAC example 7 is more complex than example 8b.

Consequently, according to the comparisons of complexity metrics NOA, NOAJS, NOAC, IC, CNC, CFC, ND and CADAC 10 ten examples, better solution is gained by using CADAC metric since CADAC considers cognitive weights of elements.

CHAPTER VI

CONCLUSIONS

6.1 Conclusion

The Business Process Models document processes of organizations clearly, and they provide a better understanding of the organizations. Business models can be seen more difficult, and can include various faults that prevent understanding. This situation can bring about inefficient processes. Depending on having different skills, experience and training, human understandability of a business process model may vary. What is more, people who are working in development department may have problems in order to interpret some parts of the model even though it does not seem to be too complicated in terms of complexity.

Although there are several complexity metrics for business process models, each of these complexity metric has different aspect to predict whether or not a given model will be perceived as easy to understand.

In this thesis, the aim is to propose a new metric by considering different aspects of existing complexity metrics for business process models. There are several elements for BPMs. Each complexity metric has one or more these elements for BPMs, and these elements provide different aspects for complexity. Also, each element has different cognitive weight about complexity. In this work, the proposed metric considers all elements of BPMs, and cognitive weights of these elements.

According to the literature survey on complexity metrics for business process models, there are many complexity metrics for business process models (Cardoso,

2006a; Cardoso, 2006b; Latva-Koivisto, 2001; Mendling, 2006; Gruhn- Laue, 2006a), and they have different traits for finding complexity of a business process model. Each complexity metric for BPMs has one or more complexity metric elements. Complexity metrics for BPMs are **Activity, Control Flow, Depth (maximum nesting depth), Arc (sequence flow), Handle, Anti-pattern, Cognitive Weight**. It has been taken into account existing complexity metrics elements while proposing a new complexity metric.

Also, in the literature survey it has been seen that there is no universally applicable solution as a complexity metric about human understandability of business process models. However, **Cognitive Weight** metric that proposed by Gruhn and Laue (Gruhn-Laue, 2006b) is the best at quantifiable measuring the understandability of existing BPMs. Also, it has been taken into account cognitive weight for BPMs elements while proposing a new complexity metric.

Hence, it has been designed a new complexity metric **CADAC (Cognitive Activity Depth Arc Control Flow) Metric** that can accurately predict the perceived understandability of process models since elements for business process models and cognitive weights for business process models elements have been considered.

Then, it has been validated CADAC metric theoretically using Weyuker's Properties and Briand's Framework and practically by using examples from different domain. What is more, practical implication for the proposed metric has been done.

6.2 Contributions and Limitations

In this thesis, there are some contributions. First of all, the proposed metric CADAC for BPMs have been constituted taking into account existing complexity metrics elements. It has been seen that elements of BPMs have different impacts on complexity according to cognitive weights complexity (Gruhn-Laue, 2006b). Thus, while constituting the proposed metric, cognitive weights of elements in BPMs have been considered.

The proposed metric for BPMs has been done considering only BPMN language. Other contribution of the thesis is that the proposed metric CADAC can be applied not only BPMN language but also the other BPM languages by providing transformation of any BPM language to BPMN. It means if transformation of BPMN to any BPM language for a model is provided, then CADAC metric can be calculated for the model.

Also, the proposed metric has some limitations. Some of BPMN elements are not used in CADAC metric. For CADAC metric calculations, all types of activities, only exclusive, parallel and inclusive gateways, and all types of sequence flows in BPMN are considered.

The other limitation of CADAC is that it requires too much consideration. In order to calculate CADAC, activity, gateways (AND, OR, XOR), arc, maximum (fan-in and fan-out), and nesting depth are taken into account. Hence, complexity calculation for CADAC complexity metric is not easy as much as NOA, CFC or ND complexity metrics. CADAC requires more effort and time for complexity calculation.

6.3 Directions for Further Research

The proposed complexity metric CADAC is designed as calculation, and it has a formula. Therefore, designers can calculate complexity of business process models manually by using CADAC metric.

In further research, the proposed metric should be calculated from BPMN, so a tool that can calculate automatically complexity for business process models according to CADAC should be designed.

REFERENCES

- Baqaie, M. F.**, [2009], “*Ensuring Quality in Business-driven Development of IT Systems using Workflow Patterns*”, University of Paderborn, Germany, 2009.
- Briand, L.**, [1996], Morasca, S. & Basili, V., “*Property-based Software Engineering Measurement*”, IEEE Transactions on Software Engineering, pp 68-86, January, 1996.
- Cardoso, J.**, [2005], “*How to Measure the Control-flow Complexity of Web Processes and Workflows*”, Fischer, L., editor, Workflow Handbook 2005, Future Strategies, Lighthouse Point, 2005.
- Cardoso, J.**, [2006a], Mendling, J., Neuman, G. & Reijers, H.A., “*A Discourse on Complexity of Process Models*”, Lecture Notes in Computer Science 4103, Springer-Verlag, Berlin, pp. 115-126.2006.
- Cardoso, J.**, [2006b], “*Complexity analysis of BPEL web processes*”, Journal of Software Process: Improvement and Practice, 2006.
- Cardoso, J.**, [2008], “*Business Process Control-Flow Complexity: Metric, Evaluation, and Validation*”, Int. J. Service Res. 5(2): 49-76, 2008.
- Gruhn, V. & Laue, R.**, [2006a], “*Complexity Metrics for Business Process Models*”, Lecture Notes in Informatics, 85, pp. 1-12. 2006.
- Gruhn, V. & Laue, R.**, [2006b], “*Adopting the cognitive complexity measure for business process models*”, In Fifth IEEE International Conference on Cognitive Informatics, Beijing, China, pp. 236-241. IEEE Computer Society, 2006.
- Gruhn, V. & Laue, R.**, [2007], “*2 Approaches for Business Process Model Complexity Metrics*”. Technology for Business Information System, pp. 13-24@Springer, 2007.
- Gustafsson, J.**, [2000], “*Metrics calculation in MAISA*”, 2000.
- Henry, S. & Kafura, D.**, [1981], “*Software structure metrics based on information-flow*”. IEEE Transactions on Software Engineering, 7(5):510-518, 1981.

Kalb, G. E., [1990], “*Counting lines of code, confusions, conclusions, and recommendations*”, Briefing to the 3rd Annual REVIC User’s Group Conference, 1990.

Kaner, C., [2004], Bond, W. P., “*Software Engineering Metrics: What Do They Measure and How Do We Know?*”, In METRICS 2004. IEEE CS, Press, 2004.

Kostoff, R. N., [1999], “*Science and Technology Innovation*”, Office of Naval Research, Arlington, 1999.

Latva-Koivisto, A. M., [2001], “*Finding a complexity for business process models*,” Helsinki University of Technology, Tech. Rep., Feb 2001.

McCabe, T.J. & Butler, C.W., [1989] “*Design complexity measurement and testing*”, Communications of the ACM, 32:1415–1425, 1989.

Mendling, J., [2006], Moser, M., Neumann, G., Verbeek, H., Dongen, B. & van der Aalst, W.M.P., “*A Quantitative Analysis of Faulty EPCs in the SAP Reference Model*”, BPM Center Report BPM-06-08, BPMCenter.org, 2006.

Mendling, J., [2010], Reijers, H.A & van der Aalst, W.M.P., “*Seven process modelling guidelines (7PMG)*”, Information and Software Technology, Volume 52, Issue 2, February 2010, Pages 127-136

Reijers, H. A. & Vanderfeesten, I., [2004], “*Cohesion and Coupling Metrics for Workflow Process Design*”, In J. Desel, B. Pernici, and M. Weske, editors, Proceedings of the 2nd International Conference on Business process Management (BPM 2004), Lecture Notes in Computer Science volume 3080, pp. 290-305, Springer-Verlag, Berlin., 2004.

Rummler, G. A., Brache, A. P., [1995], “*Improving Performance: How to Manage the WhiteSpace in the Organization Chart*” (2nd ed.), Jossey-Bass, 1995.

Shao, J., Wang, Y., [2003], “*A new measure of software complexity based on cognitive weights*”. IEEE Canadian Journal of Electrical and Computer Engineering, 28(2): 69-74, 2003.

Van der Aalst, W.M.P., [1998], “*The Application of Petri Nets to Workflow Management*”, The Journal of Circuits, Systems and Computers 8, 21–66, 1998.

Van der Aalst, W.M.P., [2003], Hofstede, A., Kiepuszewski, B., Barros, A., “*Workflow patterns*”, Distributed and Parallel Databases 14, 2003.

Vanderfeesten, I., [2004], Reijers, H.A, Mendling, J., Van der Aalst, W.M.P. & Cardoso, J., “*Quality Metrics for Business Process Models*”, Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.74.6133&rep=rep1&type=pdf>

Vanderfeesten, I., [2008], Reijers, H.A, Mendling, J., Van der Aalst, W.M.P. & Cardoso, J., “*On a Quest for Good Process Models: The Cross-Connectivity Metric*”, Lecturer Notes in Computer Science, 5074, pp. 480-494.

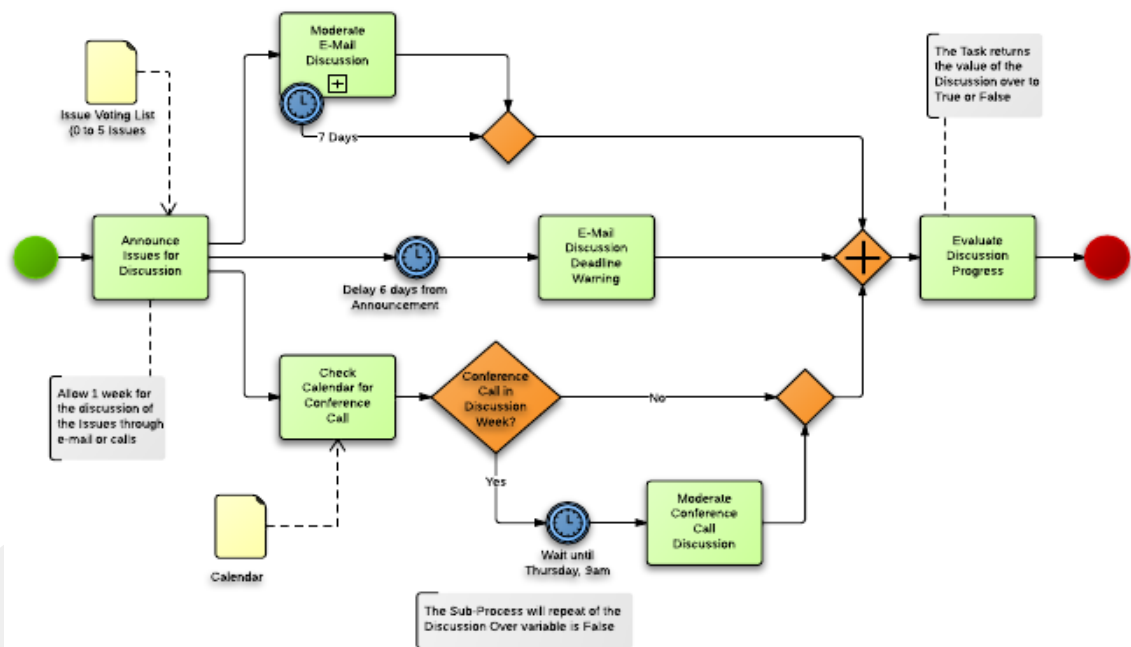
Weyuker, E.J., [1988], “ *Evaluating software complexity measures*”, IEEE Transactions on Software Eng., 14(9), 1357–1365, [1988].

GCRIIS

APPENDICES

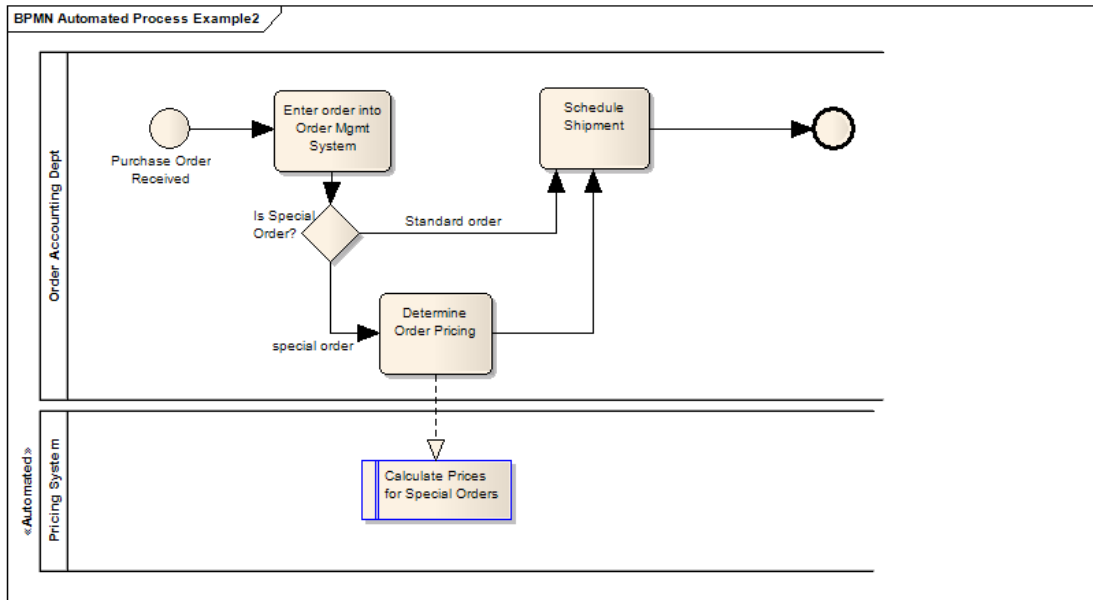
Appendix A – Example1 for business process model

URL: <https://www.lucidchart.com/community/examples/view/4d72-6c88-4f353354-976c-1f370a57027e>, [Accessed 10 September, 2013]



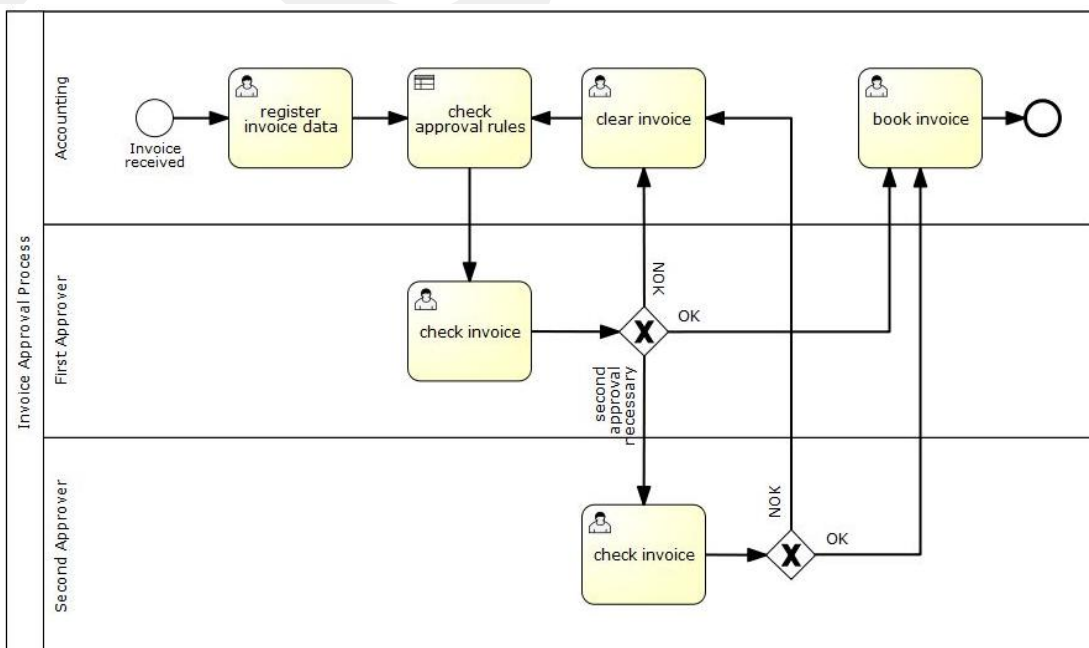
Appendix B – Example2 for business process model

URL: <http://blogs.msdn.com/b/nickmalik/archive/2008/06/26/blame-the-computer-a-business-process-modeling-anti-pattern.aspx>, [Accessed 10 September, 2013]



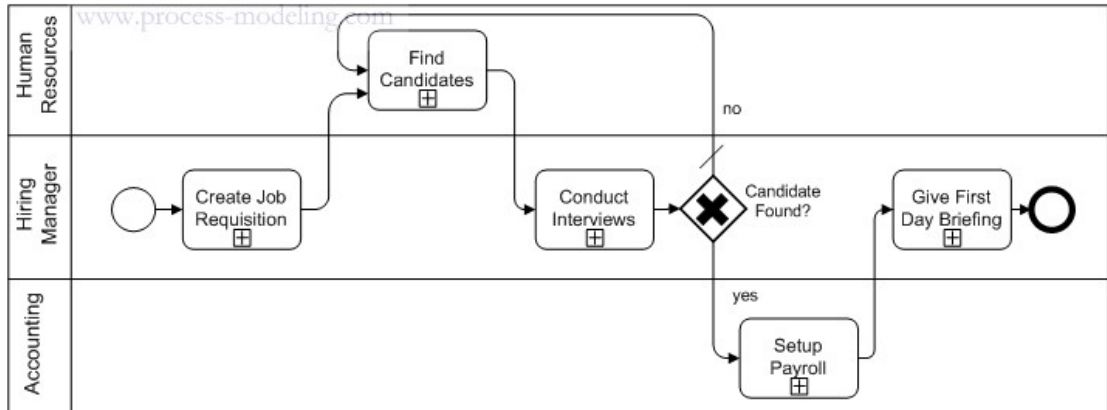
Appendix C – Example3 for business process model

URL: <http://www.saperionblog.com/lang/de/how-to-model-bpmn-for-dynamic-process-execution/2996/>, [Accessed 13 December, 2013]



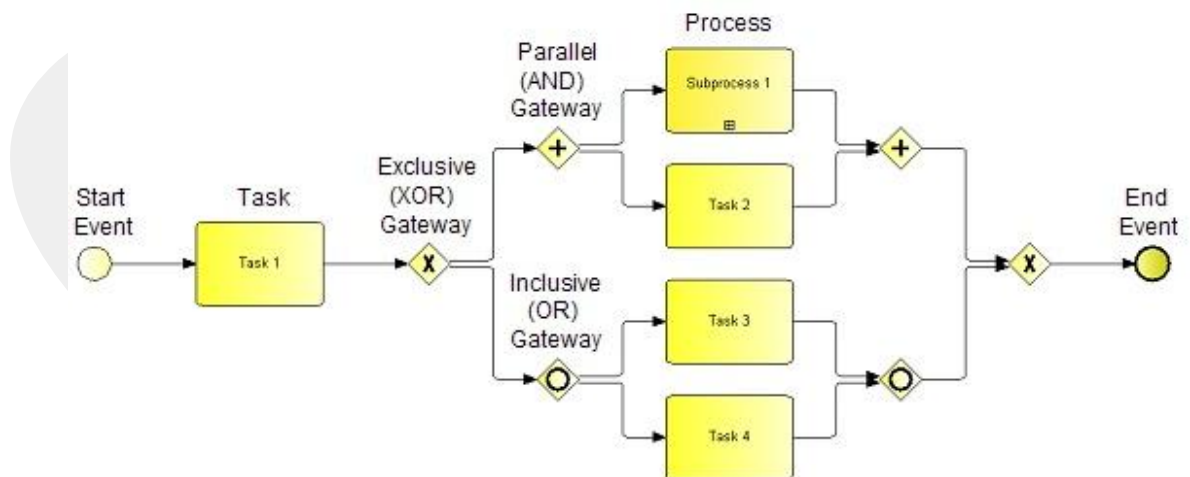
Appendix D – Example4 for business process model

URL: <http://www.rickgeneva.com/wp/posts/swimlane-lane-or-pool-learn-to-swim-first/>, [Accessed 13 December, 2013]



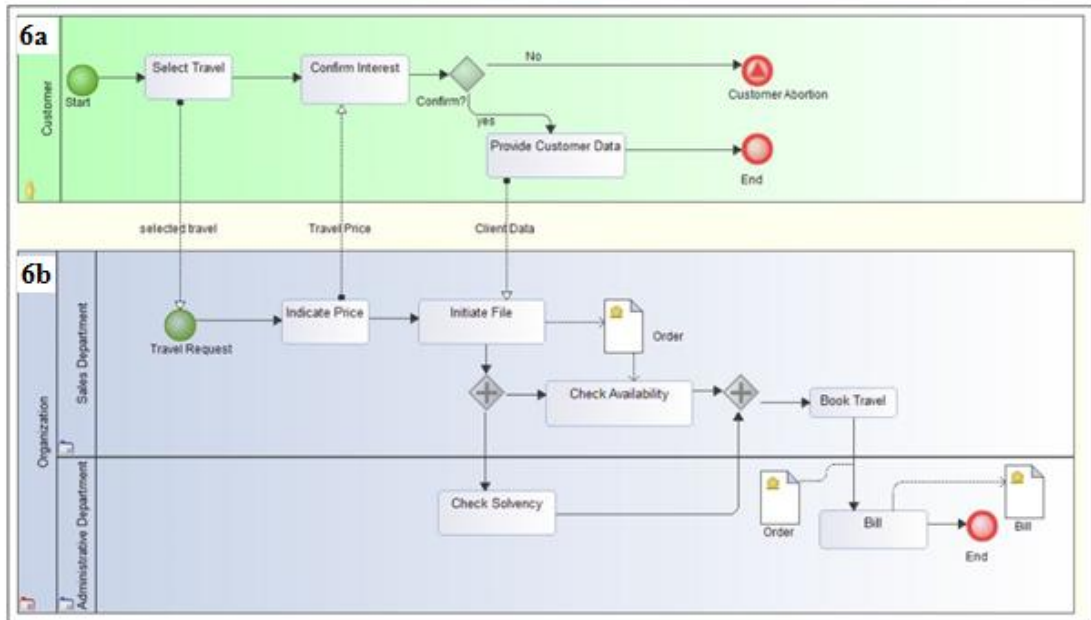
Appendix E – Example5 for business process model

URL: http://www.ibm.com/developerworks/websphere/techjournal/0702_koehler/0702_koehler.html, [Accessed 20 December, 2013]



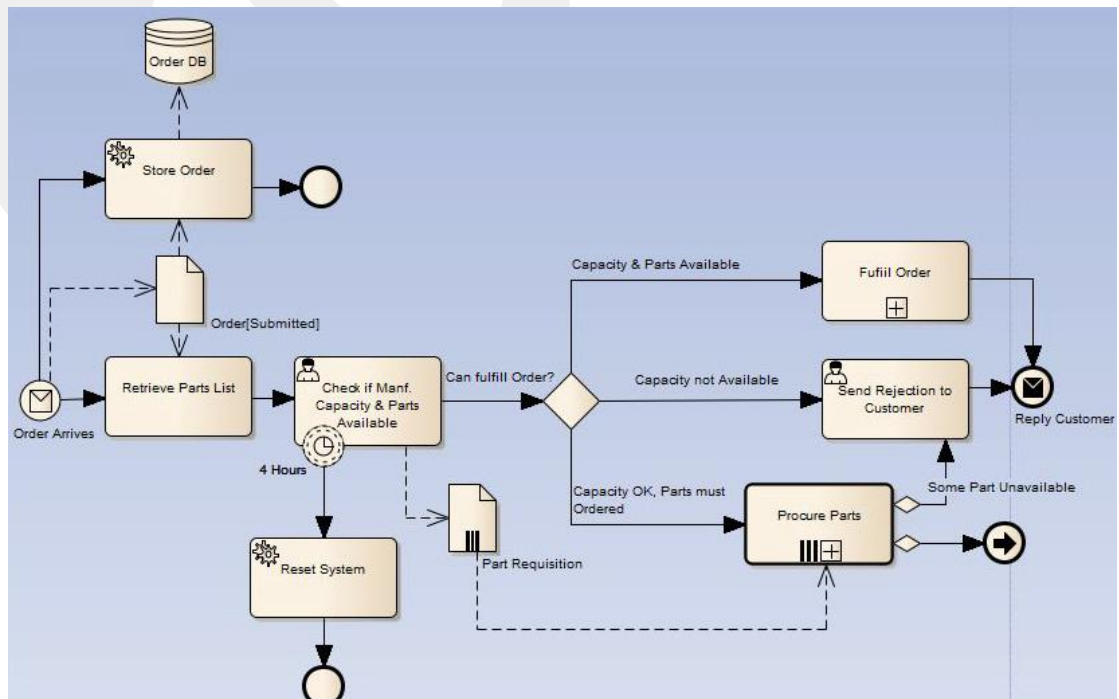
Appendix F – Example6 for business process model

URL: [http://www.modeliosoft.com/en/resources/diagram-examples/bpmn diagrams.html](http://www.modeliosoft.com/en/resources/diagram-examples/bpmn%20diagrams.html), [Accessed 10 January, 2014]



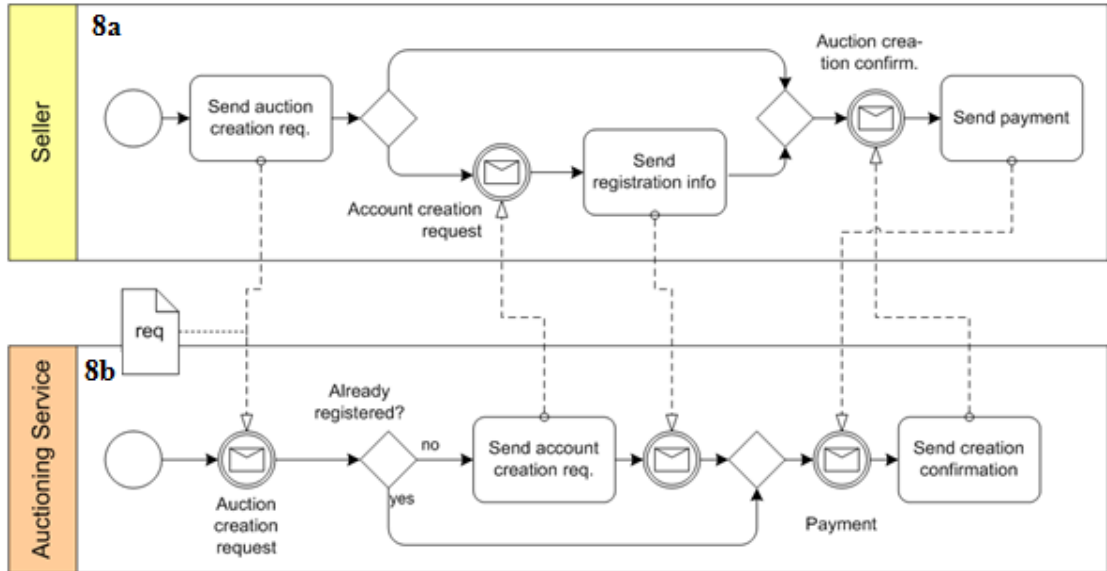
Appendix G – Example7 for business process model

URL: http://www.sparxsystems.com/enterprise_architect_user_guide/9.2/domain_based_mod_els/bpmn_2_0_business_process.html, [Accessed 10 January, 2014]



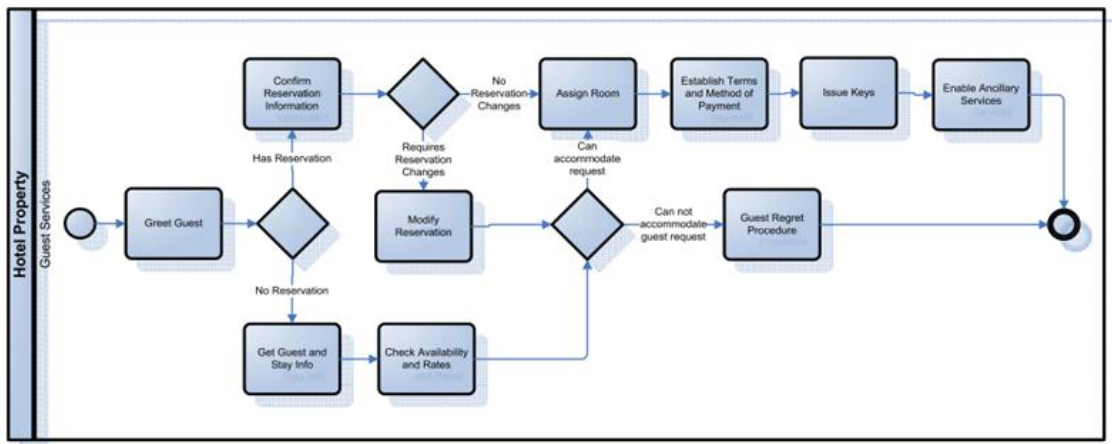
Appendix H – Example8 for business process model

URL: <http://blog.bpmn.info/wp-content/uploads/2008/10/chor1.png>, [Accessed 10 January, 2014]



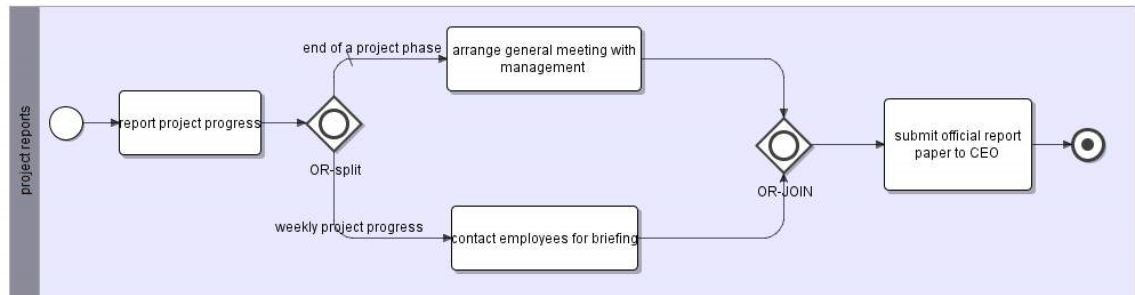
Appendix I – Example9 for business process model

URL: http://wiki.htng.org/mediawiki/index.php?title=Architecture:_Appendix_-_SOA_ _Realization, [Accessed 4 February, 2014]



Appendix J – Example10 for business process model

URL: <http://giannisgiataganas.blogspot.com/2011/04/pattern-bpmn-examples.html>, [Accessed 4 February, 2014]



Appendix K – Example of the Proposed Complexity Metric Calculation

URL: http://pic.dhe.ibm.com/infocenter/rsahelp/v8r5/index.jsp?topic=%2Fcom.ibm.ccl.soa.deploy.uml.doc%2Ftopics%2Fuml_topo_execution_bpmn_tsk.html, [Accessed 3 September, 2013]

Appendix L- Example of Mapping Concurrent Processing

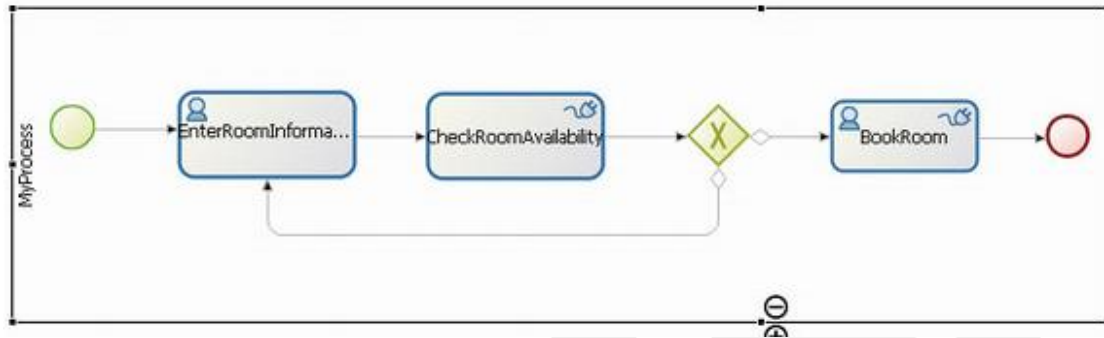
URL: <http://turing.crema.unimi.it/ScuolaDottorato/files/slides/BPE15.pdf>, [Accessed 22 March, 2014]

Appendix M- Example of Mapping Synchronization in Concurrent Processing

URL: <http://turing.crema.unimi.it/ScuolaDottorato/files/slides/BPE15.pdf>, [Accessed 22 March, 2014]

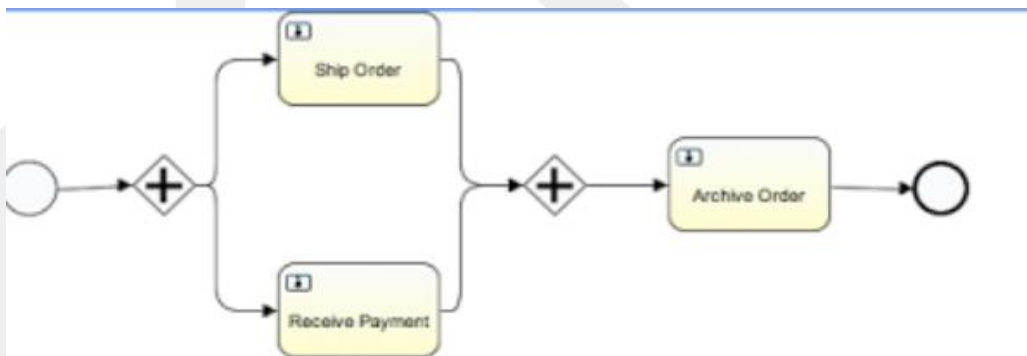
Appendix N – Example11 for business process model

URL: <http://www.ibm.com/developerworks/opensource/library/j-bpm1/figure1.jpg>,
[Accessed 2 April, 2014]



Appendix O – Example12 for business process model

URL: <http://bpmn20inaction.blogspot.com.tr/2013/03/activiti-designer-5120.html>,
[Accessed 2 April, 2014]



Appendix P – Example13 for business process model

URL: <http://oryx.bpmn-community.org/backend/poem/model/1201/png>, [Accessed 3 April, 2014]

