TESIS – KI 092361

TEMPAT PENYIMPANAN ALUR KERJA UNTUK PENYEDIAAN ALUR KERJA YANG DAPAT DIKONFIGURASI

RIGGA WIDAR ATMAGI
NRP 5111201901

PEMBIMBING
Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc. Ph.D.

PROGRAM MAGISTER
BIDANG KEAHLIAN REKAYASA PERANGKAT LUNAK
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2014

THESIS – KI 092361

WORKFLOW REPOSITORY FOR PROVIDING CONFIGURABLE WORKFLOW

RIGGA WIDAR ATMAGI
NRP 5111201901

SUPERVISOR
Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc. Ph.D.

MASTER PROGRAM
FIELD OF SOFTWARE ENGINEERING
DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2014

TESIS – KI 092361

WORKFLOW REPOSITORY FOR PROVIDING CONFIGURABLE WORKFLOW

RIGGA WIDAR ATMAGI
NRP 5111201901

SUPERVISOR
Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc. Ph.D.

MASTER PROGRAM
FIELD OF SOFTWARE ENGINEERING
DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2014

WORKFLOW REPOSITORY FOR PROVIDING CONFIGURABLE WORKFLOW IN ERP

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom)
di
Institut Teknologi Sepuluh Nopember Surabaya

oleh:

<u>RIGGA WIDAR ATMAGI</u>
NRP. 5111201912

Tanggal Ujian: 11 Juli 2014
Periode Wisuda: September 2014

Disetujui oleh:

1. Prof.Ir.Drs.Ec.Riyanarto Sarno, M.Sc, Ph.D. (Pembimbing)
NIP. 19590803 198601 1 001

2. Dr. Ir. Siti Rochimah, MT. (Penguji)
NIP. 19681002 199403 2 00

3. Sarwosri, S.Kom., M.T. (Penguji)
NIP. 19760809 200112 2 001

4. Abdul Munif, S.Kom, M.Sc. (Penguji)
NIP. 051100114

Direktur Program Pascasarjana

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN INSTITUT TEKNOLOGI SEPULUH NOPEMBER PROGRAM PASCASARJANA

<u>Prof. Dr. Ir. Adi Soeprijanto, M.T.</u>
NIP. 19640405 199002 1 001

PENYEDIAAN ALUR KERJA ERP YANG DAPAT DIKONFIGURASI MELALUI REPOSITORI

Nama : Rigga Widar Atmagi
NRP : 5111201912
Pembimbing : Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc. Ph.D.

ABSTRAK

Workflow pada ERP dengan domain fungsi yang besar rentan dengan adanya duplikasi. Membuat *workflow repository* yang menyimpan berbagai macam *workflow* dari proses bisnis ERP yang dapat digunakan untuk menyusun *workflow* baru sesuai kebutuhan *tenant* baru

Metode yang diusulkan: Metode yang diusulkan terdiri dari 2 tahapan, *preprocessing* dan *processing*. Tahap *preprocessing* bertujuan untuk mencari *common* dan *sub variant* dari *existing workflow variant*. *Workflow variant* yang disimpan oleh pengguna adalah *Procure to Pay workflow*. Variasi tersebut diseleksi berdasarkan kemiripannya dengan *similarity filtering*, kemudian di*merge* untuk mencari *common* dan *sub variant*nya. *Common* dan *sub variant* disimpan menggunakan *metadata* yang dipetakan pada basis data relasional. Deteksi *common* dan *sub variant workflow* mencapai tingkat akurasi sebesar 92%. C*common workflow* terdiri dari 3-*common* dari 8-*variant workflow*. *Common workflow* tersebut memiliki tingkat kompleksitas lebih rendah 10% dari model sebelumnya.

Tahapan *processing* adalah tahapan penyediaan *configurable workflow*. Pengguna memasukan *query model* untuk mencari *workflow* yang diinginkan. Dengan menggunakan metode *similarity filtering*, didapatkan *common* dan/atau *sub variant* yang memungkinkan. Pengguna dapat menggunakan *common workflow* melalui *workflow designer* untuk melakukan rekomposisi ulang. Penyediaan *configurable workflow* oleh ERP mencapai tingkat 100% dimana apapun yang diinginkan pengguna dapat disediakaan *workflow*nya oleh ERP, ataupun sebagai dasar membentuk *workflow* yang lain. Berdasarkan hasil percobaan, tempat penyimpanan *workflow* dapat dibangun dengan arsitektur yang diajukan dan mampu menyimpan dan menyediakan *workflow*. Tempat penyimpanan ERP mampu mendeteksi *workflow* yang bersifat *common* dan sub *variant*. Tempat penyimpanan ERP mampu menyediakan *configurable workflow*, dimana pengguna dapat memanfaatkan *common* dan *sub variant workflow* untuk menjadi dasar mengkomposisi *workflow* yang lain.

Keyword: Alur kerja , Penyimpanan alur kerja, Metode kemiripan, Metode penggabungan, ERP

WORKFLOW REPOSITORY FOR PROVIDING CONFIGURABLE WORKFLOW IN ERP

Student Name : Rigga Widar Atmagi
Student ID : 5111201912
Supervisor : Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc., Ph.D.

ABSTRACT

Workflow in ERP which covered big domain faced duplication issues. Scope of this research was developing workflow from business process ERP which could be used for required workflow as user needs.

Proposed approach consisted of 2 stages preprocessing and processing. Preprocessing stages aimed for finding common and variant of sub workflow based on existing workflow variant. The workflow variants that were stored by user were procured to pay workflow. The workflows was filtered by similarity filtering method then merged for identifying the common and variant of sub workflow. The common and sub variant workflow were stored using metadata that mapped into relational database. The common and variant of sub workflow detection achieved 92% accuracy. The common workflow consisted of 3- the common workflow from 8-variant workflow. The common workflow has 10% lesser complexity than its predecessor.

Processing was providing configurable workflow. User inputted query model to find required workflow. Utilizing similarity filtering, possible the common and variant of sub workflow was collected. User used the common workflow through workflow designer to recompose. Providing configurable workflow ERP achieved 100%, where any user need would be provided by ERP, as workflow or as based template for creating other.

Based on evaluation, repository was built based on proposed architecture and was able to store or provide workflow. Repository detected workflow whether common or variant of sub workflow. Repository ERP was able to provide configurable ERP, where user utilized common and variant of sub workflow as based for creating one of their need.

Keyword: workflow, workflow repository, similarity, merging, ERP

ACKNOWLEDGEMENT

The writer would like to acknowledge for countless thanks to the Most Gracious and the Most Merciful Allah SWT, who always gives him the best of his life. Peace and salutation were always dedicated to Prophet Muhammad "Peace be upon Him" (pbuh). This manuscript was presented to fulfil one of the requirements with intent to accomplish the Master Program in Department of Informatics, Institut Teknologi Sepuluh Nopember. The writer would like to take his opportunity to express his deep gratitude to the following persons:

1. Mother, father, and sister for all of their support. Thank you for such priceless cares. He hopes that his graduation would be the best present for them.
2. Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc., Ph.D., who always encourages him to finish his research flawlessly, as well as, shares a plenty of precious knowledge with him.
3. Members of Workflow Management Research Group (Dewa, Mb. Riska, Mb. Astria, P. Beje, P. Sidiq, etc.), who create such a competitive atsmosphere to conduct this research.
4. Institut Teknologi Sepuluh Nopember (ITS), which gives him a scholarship, i.e., Fast-Track Germany Program, for two years.
5. Software Engineering Laboratory Crew, who always accompany him in the struggle of finishing his thesis,
6. P. Syai'in and his team, who support and motivate the writer to continue his higher education until pursuing Doctorate in Germany in 2014.
7. Lecturers in Department of Informatics ITS, who give suggestion and critics to the manuscript.
8. All of grantees of Fast-Track Scholarship (the 1st batch) ITS, who become his friends in this master study.
9. Staffs in Department of Informatics ITS, who help him to complete the administrative requirements for his study and graduation.

10. Anyone that cannot be mentioned directly or indirectly who help him in cases of completing this thesis. The writer does appreciate for any opinion and suggestion for the improvement of this thesis.

About all, there was too much wrongness of this thesis. The lack of it could be used to achieve the perfection on the future works. Thank you very much.

Surabaya,
July2014

The Writer

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

This section explained research background, research question, research objective, research hypothesis, research contribution, and research constraint in this thesis.

1.1 Background

In cloud era, ERP which was SaaS architecture arises. ERP system providers provided their product and enforced user to adopt their standard flow of work (Feuerlicht & Lozina, 2007; Granell, Dı ́az, & Gould, 2010). Each organization which was used ERP product, automatically have similar supporting flow of work in similar way, see Figure 1. 1. Since organizations have different needs, cultures and rules, mostly ERP product could not be adopted fully (Marcello La Rosa M. D., 2011) (Zhiqiang Yan R. P., 2009; P. J. PetiaWohed, 2007; Klemens, 2000).

Minor, or sometimes even major, adaptations were required to tailor the process to the local environments like local law or company cultures. In this situation, workflow platform was required. Workflow technology enabled process adaptation with rule changes. ERP product which was supported by workflow (Zarine, 2012) platform opened possibilities for user configuration, see Figure 1. 2 (Sameer Malhotra, 2007) (M. Rosemann, 2007). Early adaptation of workflow system in ERP was distributed. Each organization used its own in own repository. Repository of workflow should be played smart role by providing configurable workflow for user.

Proposed approach consisted of 2-stage, preprocessing and processing. Preprocessing was conducting by analyzing workflow variant, analyzing workflow repository and developing workflow repository. Analyzing workflow variant was conducted by finding the common workflow and variant of sub workflow. Matching workflow was supported by similarity filtering. Similarity filtering consisted of grounding similarity, context similarity, profile similarity and structure similarity.

Processing was conducted by publishing the common workflow and variant of sub workflow based on proposed metadata. The common workflow opened possibilities for providing configurable workflow. User's query model was compared with stored workflow by comparing each metadata.

Proposed approach was evaluated by proving that architecture and repositories have been well developed and well-built. Dataset in repository could store workflow metadata. Finding the common workflow and variant of sub workflow was evaluated by measuring accuracy using procure to pay dataset. Proposed similarity has been proved suit well better than utilizing average weight similarity. It has been evaluated and the result presented in Section 4. Providing configurable workflow was proved by measuring accuracy. Measuring accuracy was conducting by measuring how accurate repository could provide user's query model. User's query model should be fully accommodated by repository accurately.

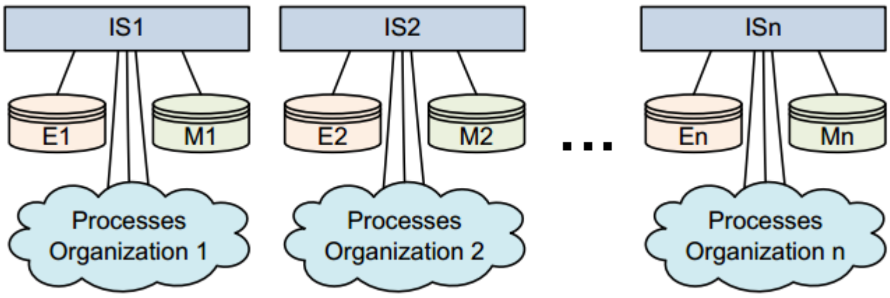


Figure 1. 1 Traditional situation with, per process, one model, and a dedicated system and database for each organization

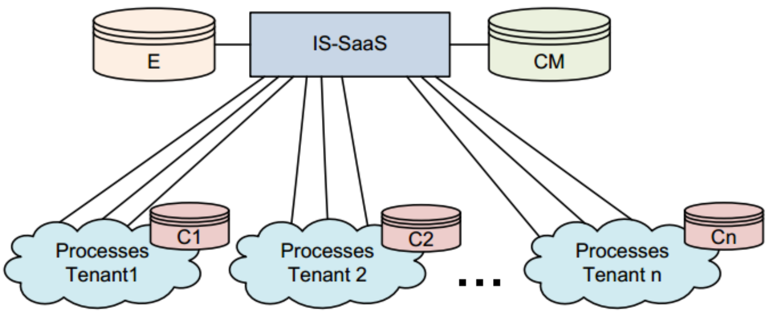


Figure 1. 2 Proposed situation using SaaS technology based on configurable services

The structure of this thesis was as follows. Chapter 1 explained the essential need of this research. The summary of several related researches was presented in Chapter 2. Chapter 3 provided an exclusive explanatory regarding the proposed method with a case study. A Chapter 4 presented the design and implementation of the proposed method. Chapter 5 explained the evaluation procedure including experimental design and result. In the end of this research, Chapter 6 reflects on the conclusion of this thesis.

Table 1. 1 Getting the common workflow approach

Process
A. Collecting tenant's workflow
B. Selecting workflow through similarity filtering
C. Merging closed-related workflow 1. Matching 2. Merging
D. Extracting workflow variant
E. Storing workflow into repository 1. List of variant 2. List common and variant of sub workflow

Table 1. 2 Providing common and variant of sub workflow approach

Process
A. Displaying common of sub workflow
B. Querying workflow based metadata into workflow variant data set
C. Matching common sub workflow by comparing workflow variant metadata

1.2 Question

The research questions of this research were as follows

1. How to develop workflow repository
2. How to provide workflows for configurable ERP

1.3 Objective

The objectives were as follows

1. Developing workflow repository for supporting efficient storing workflow variant.
2. Providing workflow and its description for supporting the composing a required workflow

1.4 Hypothesis

Hypothesis for this research was the proposed workflow repository containing common and variant of sub workflow which were selected by similarity filtering and merging method could compose a required workflow by selecting recommended workflow.

1.5 Constraint

The constraints of this research were the proposed method was applied to the specific case study, i.e., procure to pay application; workflow model and the variation of the case study were clearly defined. Also, the training dataset was presented in accordance with variation, which was commonly found in the case study. The testing dataset was generated depending on the behavior of the training dataset. Cyclic workflow was excluded from our dataset. If any cyclic workflow existed, then it has to be wrapped into Single Entry Single Exit (SESE) fragment.

CHAPTER 2
LITERATURE STUDY

2.1 Enterprise Resource Planning

ERP or Enterprise Resource Planning was an integrated system that rested on a wide range of enterprise business process modules in general. ERP system was referred to a system of interrelated due to the use of software as a means of integration between the desired resources to be integrated (Sarno & Herdiyanti, 2010).

Companies to manage data and examine crucial corporate data that spread in various /.areas of business, such as data from finance, marketing, production, human resources, managerial, and so on, often use ERP systems. ERP systems also facilitated the integration of data flow between departments were connected in the system.

2.1.1 Supplier Resource Management (SRM)

SRM was responsible for managing relationship between company and supplier. SRM was related to procure to pay process by providing each supplier with their information and performance.

2.1.2 Procurement Process

The procurement process involves the activities required to purchase, receive, and pay for the goods and services that an organization needs for its operations. The procurement process consists of six key steps: requirements determination, supplier selection, order processing, goods receipt, invoice verification, and payment processing. To function properly, procurement requires organizational data such as plant, storage locations, purchasing organization, and purchasing group.

2.1.3 Ordering Process

Ordering process involves the activities required to receive and respond to a customer inquiry, process a sales order, ship goods, and bill and receive payment from customers. Process consists of five key steps: (1) receipt of a customer inquiry or RFQ, (2) preparation of a quotation in response to the inquiry, (3) sales order processing, (4) shipping, and (5) billing and payment processing.

2.1.4 Production Process

The production process involves the various steps and activities necessary to manufacture or assemble finished goods and semi-finished goods. The two most common production strategies were make-to-stock and make to order. In make-to-stock, the materials were produced and stored in inventory for sale later. In make-to-order, production occurs only after the company receives a sales order.

If the company has adopted a make-to-order strategy, then the receipt of a customer order (fulfilment process) would trigger the need to produce the materials. If the company has adopted a make-to the material planning process, which was concerned with ensuring that sufficient quantities of materials were always available, triggers stock strategy, then production.

The production process consists of eight key steps: request production, authorize production, release production order, raw materials and semi-finished goods issue, production, production confirmation, finished goods receipt, and production order completion

2.1.5 Inventory Management

Inventory and warehouse management (IWM) processes were concerned with the storage and movement of materials in an organization. IM consists of four goods movements: goods receipt, goods issue, stock transfer, and transfer posting. Goods receipt was a movement of materials into inventory; it therefore results in an increase in inventory. Goods issue was a movement of materials out of inventory; it therefore results in a decrease in inventory. Stock transfers were used to move materials within the enterprise from one organization level or location to another in a simple way. Transfer postings were a straightforward way to change the status or type of stock, such as unrestricted use, in quality inspection, blocked, or in transit.

2.2 Service Oriented Architecture

Service-oriented Architecture (SOA) was a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. By providing functionality as coarse grained, loosely coupled and especially context free Web services, it makes this functionality interchangeable, reusable and extensible (Jae-Yoon Jung, 2006).

Workflow management system and ERP systems manage workflow. When managed correctly, a coherent picture of what was happening to an organization could be captured. The definition and management of processes allow for the integration of functional departmental data and applications allowing for better decision-making and planning (Reichert, 2012).

Services were software components, which were made accessible over the Internet. Services have to provide the following attributes or characteristics such as Self-Contained/Autonomous, Coarse-Grained, Visible/Discoverable, Stateless, Reusable, Compose able, loose coupling, and Self-describing. The arrangement of services to build service-oriented applications could be grouped into two categories: orchestration, see Figure 2. 1, and choreography, see Figure 2. 2.

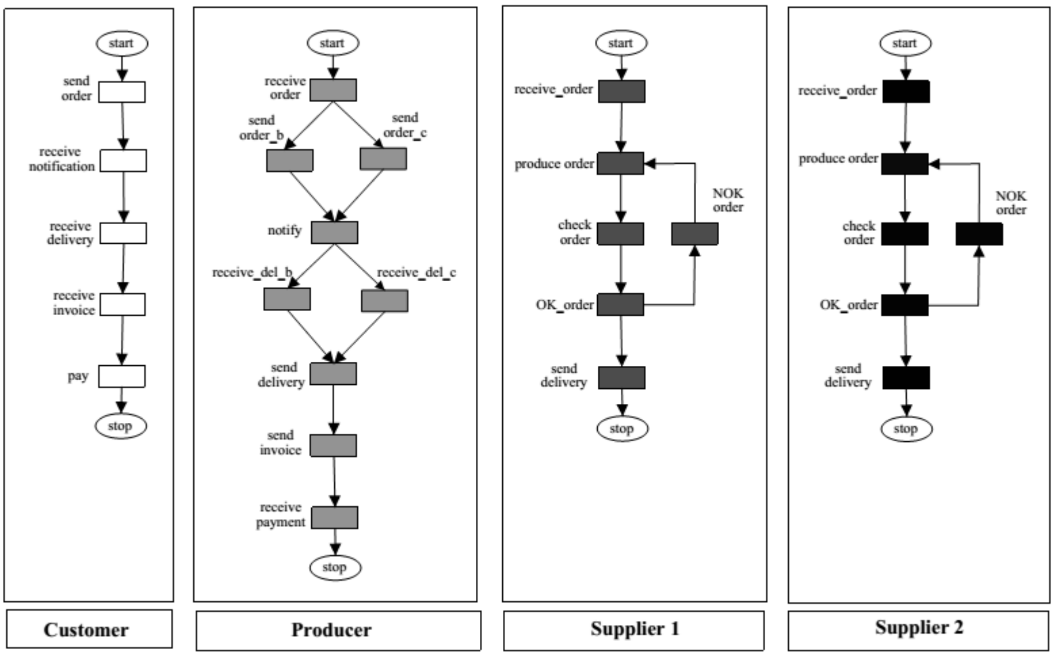


Figure 2. 1 Orchestration

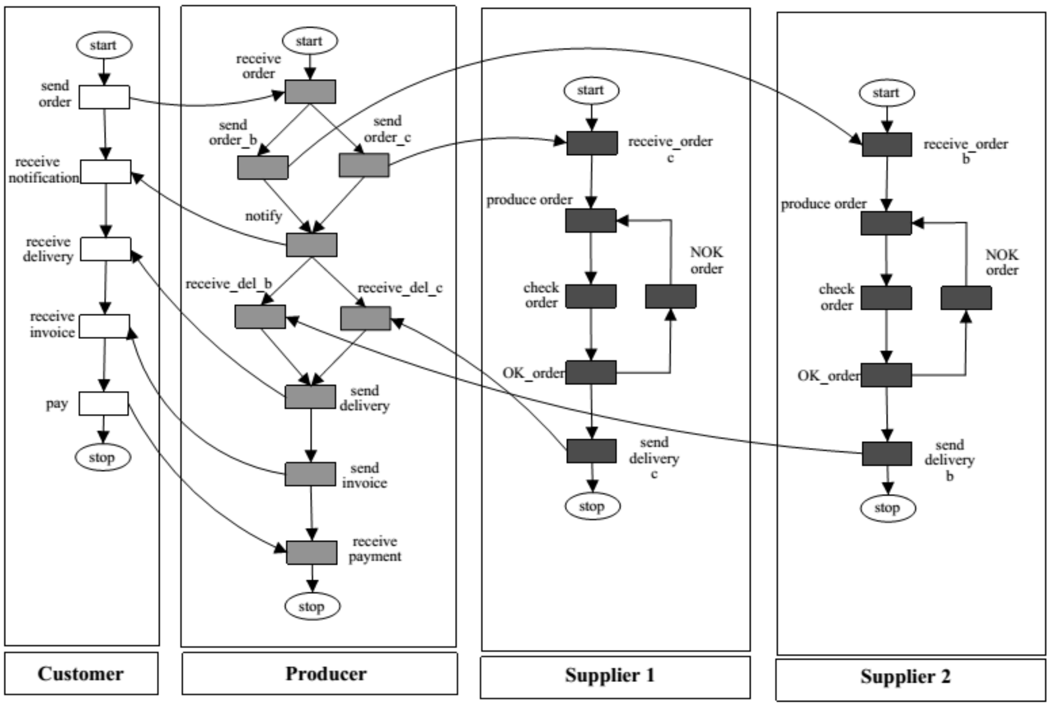


Figure 2. 2 Choreography

2.3 Business Process Model Repositories

Managing such large collections of business processes was fulfilled by providing common repository functions such as storage, search and version management. Repositories provide functionality that was specific for repositories that contain business process models. The repository for integrated process management was a repository for managing business process models throughout their lifecycle. As such it does not only have basic functions for storing and retrieving business process models, but also advanced functions for version and configuration management. Several example business process model repositories were as follows.

1. RepoX

   RepoX was an XML-based process model repository, which was a part of the METEOR workflow system. Although it was part of the METEOR project, it has been specifically developed with the intention to standardize the exchange of process models between a process definition tool and a workflow engine (known as interface 1 of the workflow reference model

2. Oryx

Oryx was a web-based process modeling tool that supports users browsing, creating, and storing and updating process models online. The tool uses a repository for storing the business process models that were created with it. Oryx could import processes from ERDF and JSON formats and processes could be transformed from BPEL. It could export process models in ERDF, JSON, RDF, PNML, XPDL, and XHTML formats or convert to Petri nets.

3. BP-Suite

   BP-Suite was a tool suite based for execution of processes specified in BPEL. It consists of BP-QL (a query language for business process definitions), BP-Mon (a tool for monitoring running business process instances) and BP-Ex (a tool for analyzing the logs of the executions). BP-Suite supports the activity, control-°ow, data, resource, organizational structure, monitoring and authorization aspects. It supports storing process models, process instances and historical information about process instances. Process models were stored in the BPEL XML format

4. ProcessGene

   The ProcessGene project provided a tool for querying business process models. It consists of four parts: a Scoping-Assistant (SA), a Query Specification Interface (QSI), a Query Interpreter (QI) and a Query Results Packager (QRP). Users of the ProcessGene provide querying scope and specifications by the SA and QSI; then the QI compiles specification to querying requirements and the QRP returns querying results. ProcessGene focuses on the activity, control flow and authorization aspects and supports reference and company specific process models.

5. Osiris

   OSIRIS (Open Service Infrastructure for Reliable and Integrated process Support) has been proposed for peer-to-peer process execution. The process repository support that it provided focuses on storing business process models, service specifications as they were provided or used by business processes and instances of executing business process models. In addition to that, OSIRIS provided supports typical peer-to-peer functions, such as concurrency control and load balancing. OSIRIS supports the activity, control-flow and data aspect and stores company specific process models (or services) and running process instances. Internally, process models and service specifications were stored in a database.

2.3.1 Workflow Metadata Modelling

The process data model prescribes how business process models and related data could be fed to the business process model repository and how they were stored internally. It consists of the meta-model, the storage model and the index model. The meta-model also prescribes how information that was stored in a business process model repository was presented to the end-user, by associating a notation with its concepts.

The storage model prescribes how the original information about the process must be technically provided to the BP Model Repository (external data model) and how it must be internally stored by the BP Model Repository (internal data model). Each BP Model Repository supports potentially support a large number of aspects, referred to Table 2. 1. Workflow metadata modelling comparison framework was presented in Table 2. 2

Table 2. 1 Workflow meta model aspect

Aspect	Definition
activity aspect (A)	contains concepts to describe the activities that were performed in the context of a process.
control-flow aspect (CF)	contains concepts to describe the control-fow relations between activities.
data aspect (D)	contains concepts to describe the information that was used and changed during the execution of a process.
resource aspect (R)	contains concepts to describe physical resources that were required to execute (activities in) a process, including human resources.
authorization aspect (Au)	contains concepts to describe who was authorized to perform which part of a process.
organization aspect (O)	contains concepts to describe the organizational structure, as it consists of people and organizational units, related to

Aspect	Definition
	a collection of processes
strategic goals aspect (G)	contains concepts to describe the hierarchy of strategic goals and to describe the relations of those goals to the processes that were meant to achieve them.

Table 2. 2 Workflow metadata modelling comparison framework

Repositories	Process meta model		Process storage model		Process index model
	Aspect	Notation	External	Internal	
RepoX	A, CF, D	Not specified	XML	Database (tables/ objects)	-
OryX	A, CF	BPMN, EPC, Petri nets	RDF	Database (tables/ objects)	-
BP-suite	A, CF, D, R, O, Au	BPEL	BPEL	Database (tables/ objects)	Categories
ProcessGene	A, CF, Au	Not specified	Not specified	Database (tables/ objects)	-
Osiris	A, CF, D	Not specified	Not specified	Database (tables/ objects)	-

2.3.2 Function Model Repository

A business process model repository should support a series of basic functions to effectively manipulate the processes that it stores. The storage functions were the functions to create, update and delete processes or parts of processes, by creating, updating or deleting instances of the concepts that were defined in the process meta model. In addition to that functions exist to import complete processes into the repository, using the interchange format from the external data model, and to export complete processes from the repository using that interchange format. The retrieval functions could be used to obtain the required process according to some criteria. There were three methods for retrieving processes: navigate, query and search. Workflow function model comparison framework was depicted in Table 2. 3.

Table 2. 3 Workflow function model comparison framework

Repositories	Process function model		Integration
	Storage	Retrieval	
RepoX	CRUD	Query, Navigation of Definition (SQL, XQuery)	-
OryX	CRUD	Query of Definition (BPMN-Q)	+Modelling
BP-suite	CRUD	Query of Definition , Instances, History	_Analysis
ProcessGene	CDU	Query of Definition (SQL)	-
Osiris	CDU	-	+Modelling

2.3.3 Process Management Repository

Advanced management functions could be subdivided into functions that were provided by general repositories and functions that were provided only by BP Model Repositories. The configuration management function makes it possible to maintain the relation between (a version of) a process and the (versions of) sub processes and activities that it consists of. Workflow management model framework was depicted in Table 2. 4.

Table 2. 4 workflow management model comparison framework

Repositories	Management model	
	General repository	Specific
RepoX	DBMS	Versioning, configuration
OryX	DBMS	-
BP-suite	DBMS	-
ProcessGene	DBMS	-
Osiris	DBMS	-

2.4 Process, Workflow and Services

2.4.1 Process

A process describes a sequence of single activities to accomplish a predefined assignment. The structure of a process features a clear definition of its activities and their dependencies. Because a process was designed once and executed several times, the term process instance was used for a single run (Karsten A. Schulz, 2004) (Mor Peleg, 2014).

2.4.2 Workflow

All parts of a process, which could be performed computerized, were called a workflow (Wolfgang Marquardt, 2003). An activity represented a simple task or an entire sub process. Each activity has a list of preconditions. Only if all preconditions were fulfilled the activity was ready for execution. Dependencies between two activities were defined by the control flow. The control flow was presented as unilateral control flow edges, which link activities, branches and loops.

2.4.3 Services

Workflow services were WCF-based services (Microsoft, 2014) that were implemented using workflows. Workflow services were workflows that use the messaging activities to send and receive Windows Communication Foundation (WCF) messages (Donglai Zhang, 2011).

2.5 Configurable Workflow

Process configuration (Babbie, 1990) (Mcoll, Jacoby, & Thomas, 2001) deals with the problem of managing families of workflow, i.e. workflows that were similar to one another in many ways, yet differ in some other ways from one organization, project, or industry to another. Configurable workflow was an integrated representation of multiple workflow reference of a same workflow in a given domain, such as multiple variants of an order-to-cash workflow operating in different markets, see Figure 2. 3.

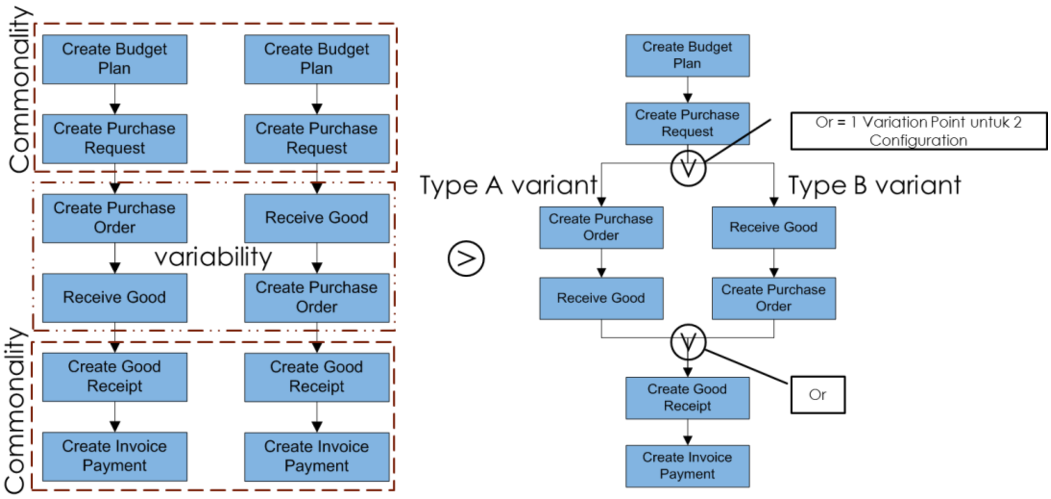


Figure 2. 3 Example configurable workflow

The right-hand side of this figure shows a configurable workflow for procure to pay application. This model was a merger between the two workflow with a variation point that represented by a configurable gateway. This inclusive split gateway has been marked with a thicker, red border: unlike a "normal" BPMN gateway, it does not represent a choice or a parallel split that would have an effect when the process was executed or simulated. Instead, a configurable gateway represented a design choice, i.e. a choice that would need to be made by an analyst to adapt the configurable workflow to a particular setting, such as a project or an organization. In our example the configurable gateway captures the fact that one needs to choose for a given screen project whether to select one path (receive good first) or the other (create purchase order before goods was received), or possibly both.

Therefore a core feature of configurable workflow was the explicit representation of variation points and their workflow. A variation point could be indicated in different ways, e.g. it could also be a special activity. A configurable workflow would typically feature many variation points, each capturing a decision that needs to be taken during process design. An analyst could configure this model by picking the most suitable workflow for each variation point. Once all these decisions have been taken, the configured process was individualized by removing that workflow that was no longer relevant, leading to an individualized process model. The latter could be used for further analysis, for simulation, or to produce an executable specification for a given set of requirements. Thus, a configurable workflow could foster the adoption of common or proven practices in a given domain, and reduce the modelling effort.

2.6 Discovery Technique for Workflow and Service (DTWS)

Workflow similarity search techniques have been developed from different angles (van der Aalst, 2006). These techniques mainly vary with respect to the information, incorporated in the workflow models that they use to determine similarity (Dijkman R. D.-B., 2009) and the underlying formalism that they use to determine similarity (Dumas, 2009).

All these metrics result in a similarity score between 0 and 1, where 0 indicates no similarity and 1 indicates identical elements. Hence, it was trivial to combine all metrics to obtain a weighted similarity score (Dijkman R. M., 2011) (Silva, et al., 2011) (van Dongen B. R., 2008).

2.6.1 Label Similarity

Label feature similarity could be measured in a number of different ways (van Dongen B. D., 2008). There was a syntactic similarity metric, which was based on string edit-distance (Zhiqiang Yan R. D., 2010). However, in realistic cases more advanced metrics should be used that take synonyms and stemming and if possible, domain ontologies into account (van Dongen B. D., 2008).

2.6.2 Structural Similarity

We could measure the structural similarity of two nodes, by determining the similarity of the (structural) roles that they have in their workflow graphs. If we consider an EPC as a graph, then its functions, events and connectors were nodes of the graph and the arcs were edges of the graph. We could then assign a similarity score to two EPCs by computing their graph-edit distance. The graph edit distance between two graphs was the minimal number of graph edit operations that was necessary to get from one graph to the other.

2.6.3 Context Similarity

To determine the contextual similarity between elements of a workflow model, we need a mapping between the elements in their input and output contexts. Such a mapping was in itself based on a similarity metric such as Syntactic Similarity or Semantic Similarity.

CHAPTER 3
METHODOLOGY

This section played an essential role to elaborate the explanatory regarding contribution given in this research. This section encompassed steps of methodology, general approach of whole methods, preprocessing phase, rule development, and method for building workflow repository approach.

The step of methodology for this research was widely defined in Figure 3. 1. The study began by studying literature related to technology services and workflows. Understanding of the literature would lead to the proper scope of the study.

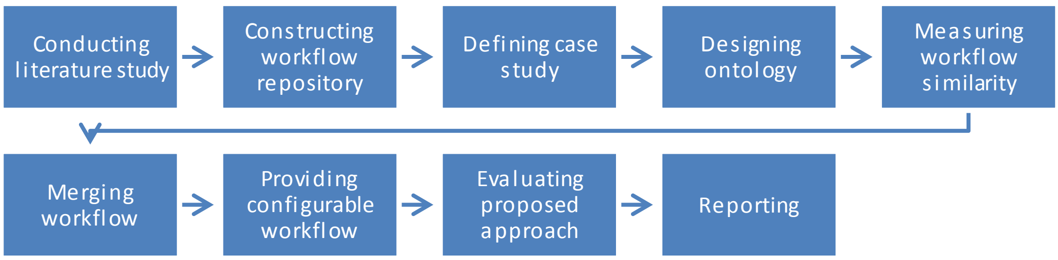


Figure 3. 1 Step of methodology

To obtain comprehensive understanding regarding proposed method, procure to pay process was selected as case study. The ideal procure to pay and its 8-variations were presented. Every particular step of the process was applied to case study. Repository stored and presented workflow as metadata. The properties were captured and mapped into metadata. This metadata was built based on ontology.

Repository collected tenant's workflows into database. Repository preprocessed common of sub workflow by selecting workflow variant through workflow similarity filtering. By using similarity filtering, repositories have closed-related workflow per domain knowledge. Each passed workflow was merged into common of sub workflow by matching and merging its node. Merging workflow consist of 2-approach, matching and merging. Matching compared node and edge workflow W_1 and W_2. If it has determined edge and node, then node and edge should be merged by adding, configuring, or deleting control flow in workflow. Workflow W_1 and W_2 was merged into common of sub workflow. This common of sub workflow was stored and has metadata from 2-workflow. Common of sub workflow have rich of variation. Variation was extracted into variation point and stored by referencing its node and workflow. See step by step process in Table 3. 1 .

Table 3. 1 Getting the common workflow approach

No	Process	Output
1.	Collecting tenant's workflow	Workflow variant
2.	Selecting workflow through similarity filtering	Closed-related workflow
3.	Merging closed-related workflow 3. Matching 4. Merging	Common and variant of sub workflow.
4.	Extracting workflow variant	Variation point
5.	Storing workflow into repository 3. List of variant 4. List common and variant of sub workflow	Stored workflow

Configuring approach was presented in Table 3. 2. Repository provided common of sub workflow by its description. User inputted their query by filing form based on workflow metadata. User query was compared with workflow variant. Result was recommendation of similar workflow variant. These workflow variant should be as query for retrieving in common sub workflow. Repository provided common of sub the common workflow which was containing recommendation of similar workflow variant. This result was taking into account as configurable workflow.

Table 3. 2 Configuring common and variant sub workflow approach

No	Process	Output
1.	Displaying common of sub workflow	List of sub workflow
2.	Querying workflow based metadata into	List of recommendation

No	Process	Output
	workflow variant data set	workflow variant
3.	Matching common sub workflow by comparing workflow variant metadata	Common sub workflow and workflow variant

At the end of this research, Whole documentation with respect to the proposed method in this research was written. The goal was obtaining essential recommendation to improve the impact. The result of this research was written in scientific journal for international publication.

3.1 Defining Case Study; Procure To Pay Process

In this research, procure to pay process, see Figure 3. 2, was investigated as a case study since it was frequently found in organization and committed with various manner. Based on 1-process model of procure to pay, 8-variation model was created. Such as variation model was possible. 8-possible variation of procure to pay was listed in APPENDIX A from Appendix 1 to Appendix 8.

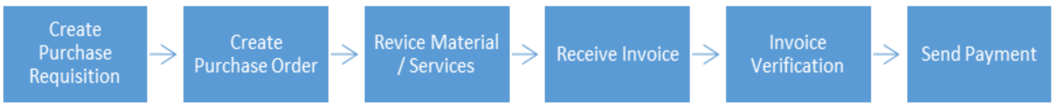


Figure 3. 2 Process model of procure to pay

3.1.1 Standard Operating Procedure

Figure 3. 2 was model depicting standard operational of procure to pay. The model was designed in Yet Another Workflow Language (YAWL), MS Visio and Petrinet model (PNML). These models should explained information about resources and rules.

Process started when user created purchase requisition after defining procurement plan and allocated fund. Purchase requisition creation was proposed to higher level to be approved. Higher level requested other information and wait until information was completed. After it was approved, user should perform auction/direct buying for selecting supplier. Request purchase order submitted when supplier has been selected. After purchase order was created, supplier could work as soon as possible.

Service acceptance/goods receiving was created in system when supplier has been finished their works. Supplier could submitted their invoice after deliverables was approved. Their invoice was evaluated and when information completed, then supplier could be paid.

3.1.2 Variants Needs and Issues

Some variation was found in application such as skipped activities or adding more activities. Skipped activities happened when one organization did not need some activities and other organization required more activities to ensure and guarantee procurement comply the organization rule.

Minor or mayor configuration need to be tailored in workflow. The possibilities were unlimited but there were some configuration which was impossible to be implemented and other was implemented easily.

Input and output of workflow should be concern for process designer. Activities worked when input and precondition were fulfilled. It was highly possible to do activities in parallel but it has data constraint to be considered.

3.1.3 Exceptional Variation

It was difficult to determine whether the variation was possible or not. There were some unpredictable needs. For instance, not every purchase order was created after purchase request was approved. This condition happened if there was a special privilege given by the board which was not defined in standard operational. Another example was receiving goods or service acceptance was not always recognized after project was delivered.

Such as inconsistence/un-common variant require some verification from expert. Configurable workflows help to do variation in single model. These cases possibility have been represented or reconfigured from configurable workflow.

3.2 Proposed Method

Proposed method consisted of 2 stages preprocessing, see Figure 3. 3 and processing, see Figure 3. 4. Preprocessing consisted of constructing workflow repository, designing ontology, measuring workflow similarity, clustering and merging workflow. Processing was providing workflow by publishing its metadata.

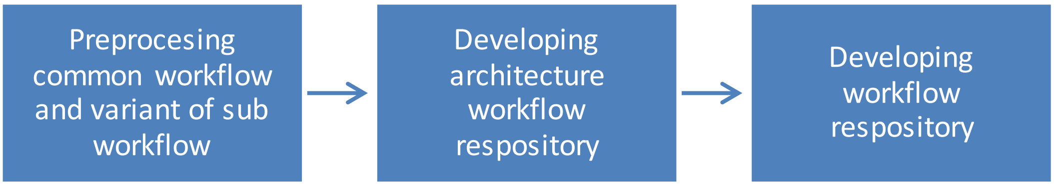


Figure 3. 3 Preprocessing

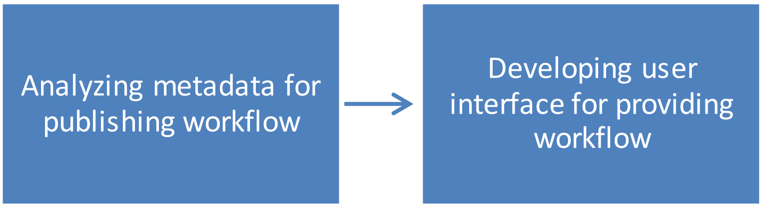


Figure 3. 4 Processing

3.2.1 Preprocessing

3.2.1.1 Developing Workflow Repository Architecture

This section described architecture and feature of workflow repository. The architecture, see Figure 3. 5 follow a three-tier model composed of an enterprise layer, an intermediary layer and a basic layer. The enterprise layer was the front-end of the repository. It hosts the repository manager which exposes the typical amenities of a repository such as simple querying. It could be accessed or cross-enterprise integration, or via a Web portal by the users of an organization (Paul Grefen, 1998) (Kim, 2012).

The business and privacy layer encapsulates the business logic and data of traditional software architecture. The business logic consists of the algorithms to operate over process model collections, e.g. matching algorithms, merging algorithms, individualization algorithms.

The repository manager accesses these logic-centric in the intermediary layer. It allowed users to batch operations via simple scripts that could be submitted through the repository manager. The bottom layer also hosts a set of data-centric which serves as an interface to access the underlying persistent data – the core of the repository. Each data-centric wraps one or more specific data entities and exposes the conventional features of the related RDBMS.

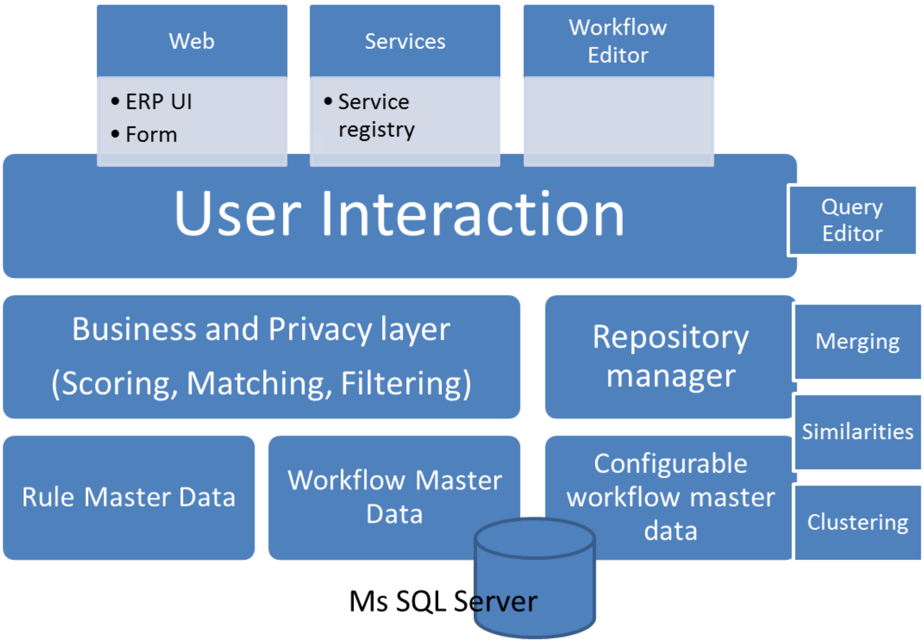


Figure 3. 5 Architecture

The repository manager accessed both process models and configurable workflow. It generated configurable workflow based on its original workflow model. Other features in repository manager were filtering, designing and presentation. Filtering offers capabilities to rank the process models in the repository based on their equality or degree of similarity to a query model, or to identify relevant workflow. Designing refers to ways to create, modify and complete process models based on existing content. Presentation provided support for improving the understanding of large process models and collections thereof.

The repository and database management layers provide the functions that were generally provided by repository and database management systems, respectively. The storage layer stores the process models, the related data and indices or classifications to enable fast querying, searching and navigation of the BP Model Repository. Process models could be stored both in an internal format Relational, object-oriented and XML databases have all been observed in this research.

3.2.1.2 Designing Ontology

Ontology was used as basic model of metadata. The concept could be converted into relational database or semantic repository (Shaokun Fan, 2012). Concept of our ERP was a SaaS ERP which consist several functional domain. Based on business requirement, the functional domain was repackaged into package. Each package has several functional domains. Each package has different purpose since user commonly need partial of ERP implementation. Illustration of domain ontology was presented in Figure 3. 6.

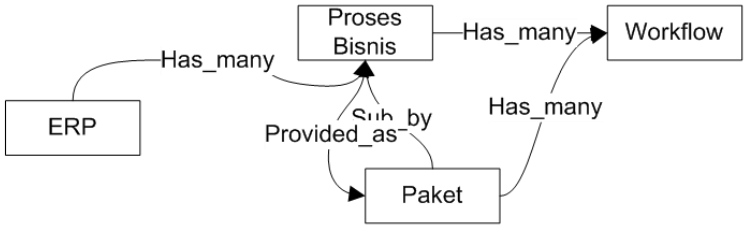


Figure 3. 6 ERP domain ontology

Concept of workflow metadata, see Figure 3. 7, was derived from known-model namely OWL-WS/S. this concept was adopted widely from represent workflow or service. In this research, OWL-WS/S concept was not enough. Additional annotation was proposed to enrich description of workflow. Addition was signed in black bold border (Chunhyeok Lim, 2011).

Workflow consists of 3-partial component, profile, grounding and model. Profile described what workflow is. Workflow profile based on OWL-WS/S was described by $value_{desc}$. Comparing workflow in multi stage similarities need more information about workflow, such as $value_{precond}$, $value_{postcond}$, $value_{inputdesc}$ and others (Ekanayake, 2011).

Grounding property was filled by mapping from xml definition of workflow. Mapping presented in Figure 3. 8. Input and its definition was default described in xml definition.

Model property was completed by defining sequence of workflow from xml. In xml workflow definition, arrangement of activity was iterated and mapped into metadata as presented in Figure 3. 9.

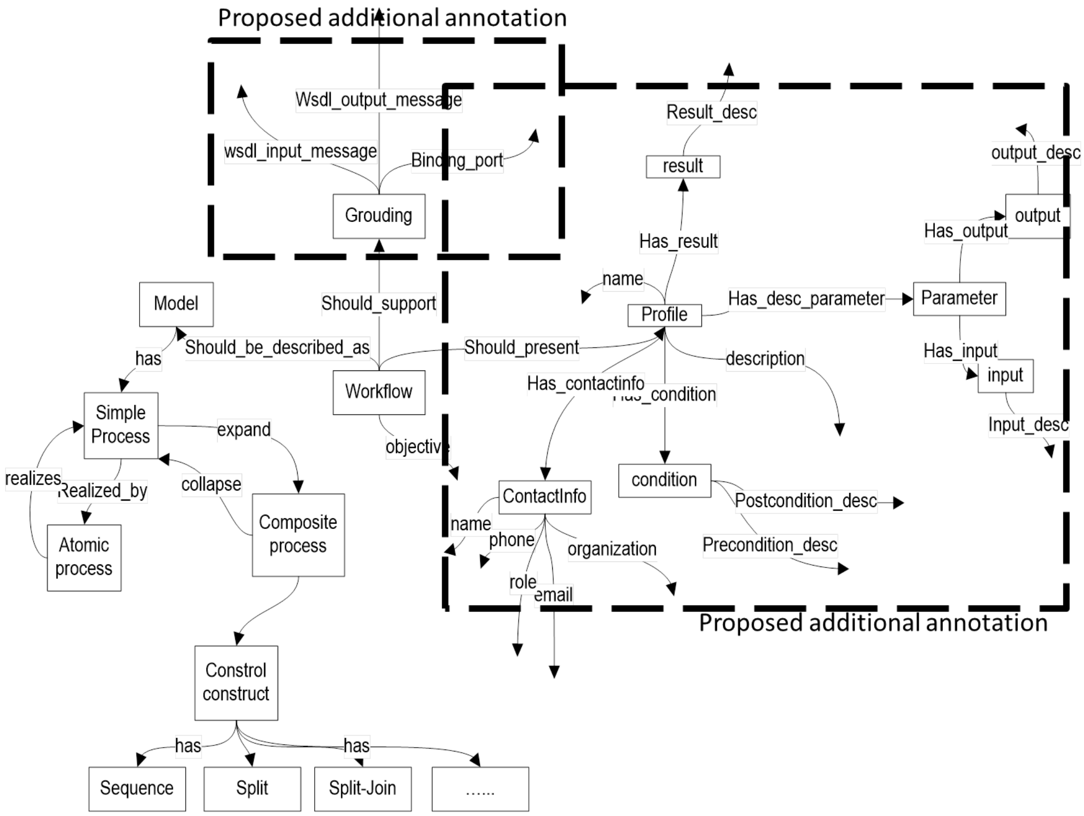


Figure 3. 7 Workflow metadata

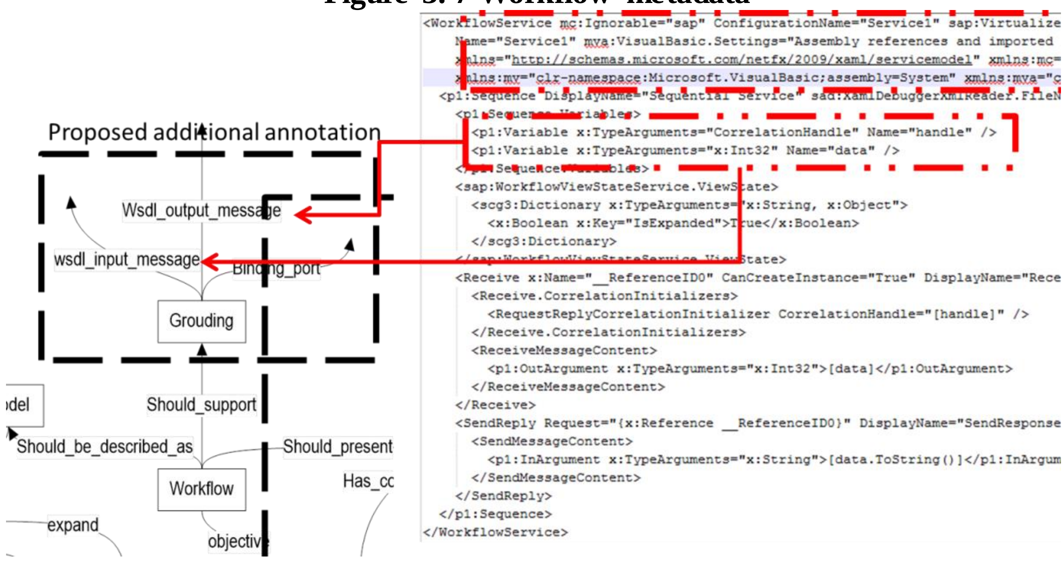


Figure 3. 8 Mapping grounding properties in metadata – XML

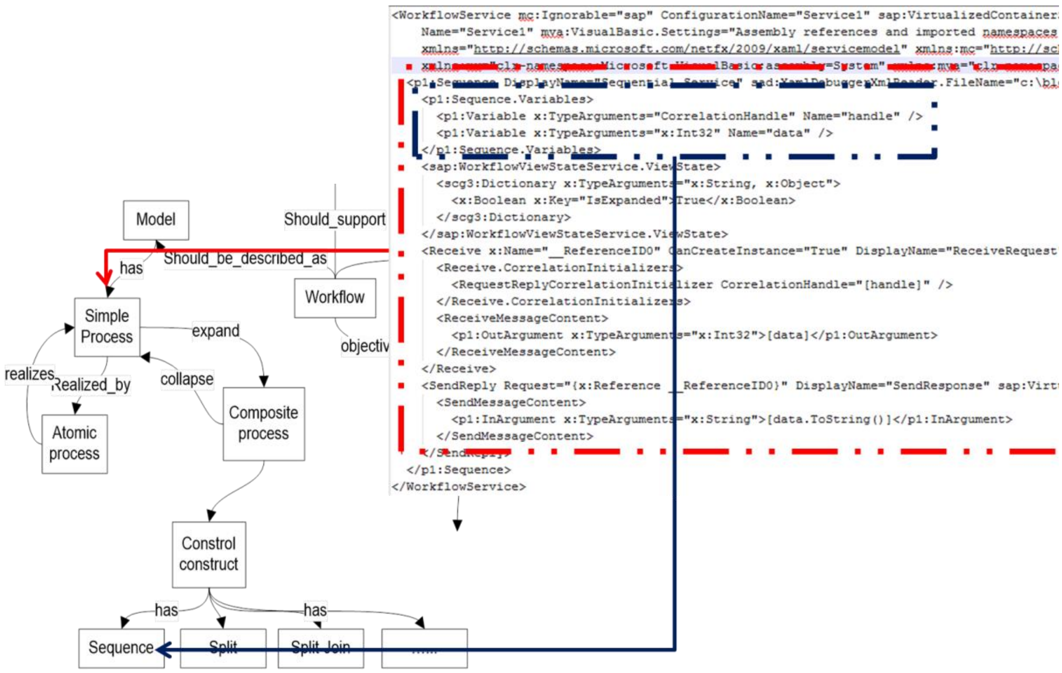


Figure 3. 9 Mapping model properties in metadata – XML

Importing workflow into repository provide way to store important structural information in workflow metadata. Repository allow user to specify concrete workflows using specific formats based on XML. Those XML files were then imported and analyzed by system, which was capable of mapping the information represented in the XML file to the relational database schema of workflow.

Some information was default provided by xml definition while other information was not. Supporting this situation, additional annotation was required. Additional annotation could be written in xml definition which was presented in Figure 3. 10.

Figure 3. 10 also demonstrated that primary table in database was filled by mapping workflow metadata. File was included in repository while completing workflow metadata. Default-required file was pnml file. Repository utilized prom plug in by converting pnml file into another files if there was no other file submitted.

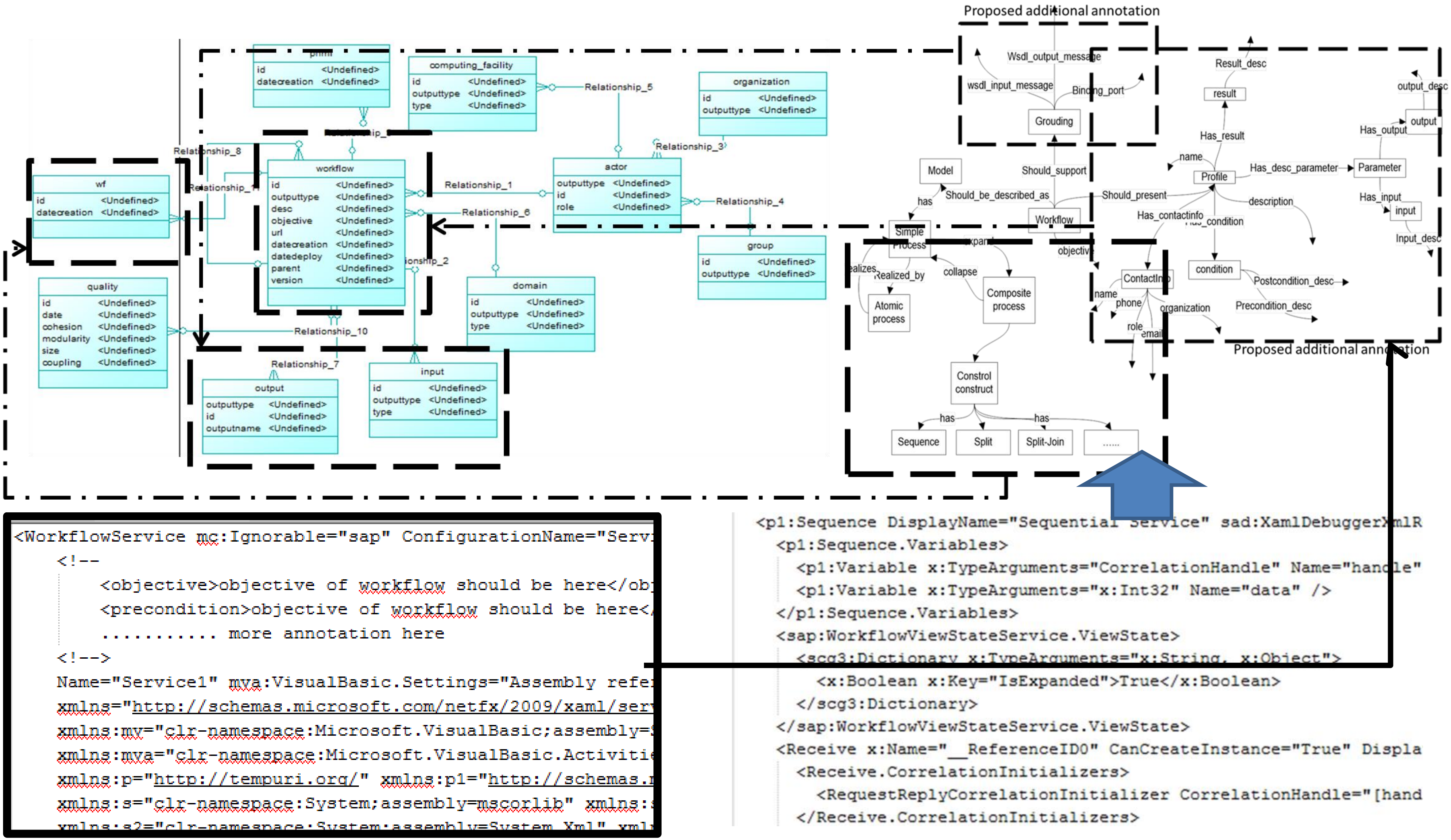


Figure 3. 10 Mapping database

3.2.1.3 Measuring Workflow Similarity

Measuring similarity used multi stage similarity. Multi stage similarity was considered as similarity filtering. Measuring workflow considered several elements which were contained in workflow model. Structural dimension contains the process model based on which the process instance was executed. For workflow, the structural dimension was represented by the process model that was adapted from the design time model for the particular workflow during instance adaptation (Lu, 2009).

Behavioral dimension contains executional information such as the set of tasks involved in the process execution (may differ from structural dimension due to choice constructs), the exact sequence of task execution, the performers and their roles in executing these tasks, the process-relevant data, and execution duration of the process instance. Contextual dimension contains descriptive information (annotations) from the process modeler about the reasoning behind the design of a particular workflow. Detail of similarity filtering was presented in Figure 3. 11.

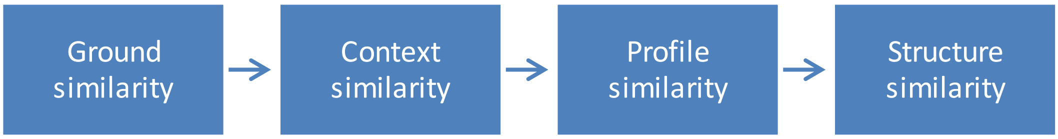


Figure 3. 11 Step of multi stage similarity

1. Grounding Similarity

To obtain executable and usable workflow, similarity was started by grounding similarity. Ground similarity performed by matching its input to other workflow input. When it was exact matching then workflow could passed. Unfortunately, this kind similarity only performed well in conditioned environment and primitive input. Detail illustration was presented in Figure 3. 12. If condition was fulfilled then Equation 1 was used.

In some others cases, proportional approach or jaccard method similarities was proposed. Proportional approach calculated how much similar input based on $value_{inputtype}$, $value_{inputdesc}$ and $value_{precondition}$. Order of input was important. When proportional approach was used, system needs to determine threshold value representing how proportional should be tolerated. Jaccard method similarities were based on previous research. Detail was presented in Figure 3. 13 and Equation 2 could be used.

Workflow considered three types of ports input ports (IP), output ports (OP), and multiport (MP) that could act either as input and output ports. Figure presented a comparison between activities A and A'. A has 3 input ports named X, Y, and W while A' has 2 input ports named X and Z. A has also 2 output ports and A' has2 output ports as well. In practical way for this research, grounding similarity could be treated as method similarity. One of the best method for this approach was stated in Equation 3 Jaccard similarity.

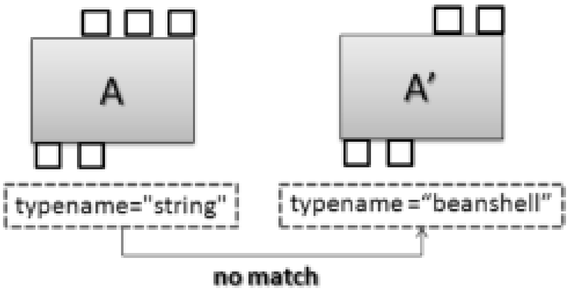


Figure 3. 12 Grounding similarity

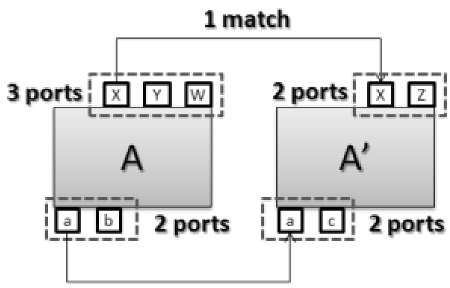


Figure 3. 13 Grounding similarity

Equation 1 Exact matching grounding similarity

$$G_{sim} = Count(IoW1_N == IoW2_M)$$

Equation 2 Proportional matching grounding similarity

$$G_{sim} = \frac{IoW1_N + IoW2_M - (2 \times matches)}{IoW1_N + IoW2_M}$$

Equation 3 Jaccard similarity

$$\boldsymbol{Jaccard_{sim} = \frac{a}{a+b+c}, where\ jaccard_{sim} is\ 0,,,1}$$

Where

$IoWx_N = input\ of\ workflow\ X\ with\ total\ input\ N$

$a = the\ number\ of\ dependencies\ on\ both\ entities$

$b = the\ number\ of\ dependencies\ on\ entity\ a\ only$

$c = the\ number\ of\ dependencies\ on\ entity\ b\ only$

2. Context Similarity

In some cases, workflows that have different knowledge domains may have similar or the same input. An example of such cases was the calculation of sales discount and drug dose calculations. Moreover, the workflow in one domain of knowledge has the same probability as in the case above.

Similarities context calculate the changes in the structure of the workflow by considering the input and output. At this stage, the threshold value should be determined. In the implementation phase, the system also stored the file in the form of workflows PNML, YAWL, or XAMLX in the database. PNML workflow files would be entered using Beehivez tool to calculate the similarity.

3. Profile Similarity

Profile attributes was attribute of workflow that explain what was that workflow and what workflow do. Getting data was illustrated in Figure 3. 15. The results were ranked. Profile attributes has many sub properties. Total profile similarity value was average total similarity value of sub properties. Similarity was counted by utilizing tool of Word Net and Semilar (semanticsimilarity.org).

Calculating profile similarity, see Figure 3. 14, through semantic similarity was the practical, widely used approach to address the natural language understanding issue in many core NLP tasks such as paraphrase identification, Question Answering, Natural Language Generation, and Intelligent Tutoring Systems.

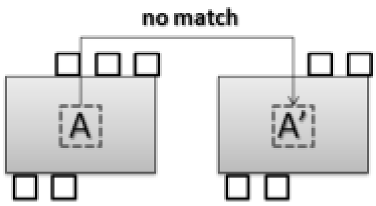


Figure 3. 14 Attribute profile similarities

Evaluating semantic similarity, SEMILAR, see Figure 3. 16, as semantic library toolkit was used. The SEMILAR corpus have word-level similarity qualitative judgments by human experts which could be used to further the understanding of the various word-to-word semantic similarity methods and their impact on the similarity of larger texts, e.g. sentences or paragraphs.

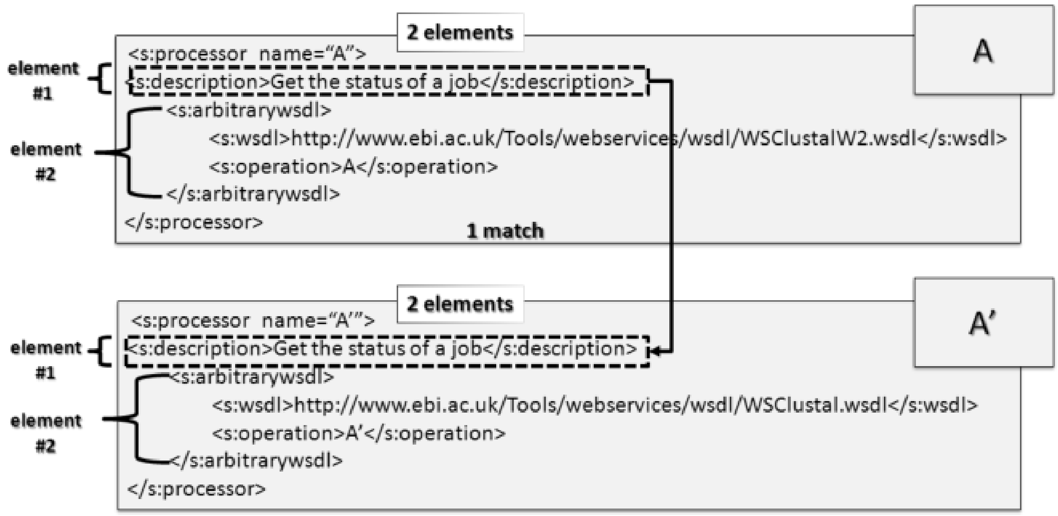


Figure 3. 15 Getting data for profile similarity

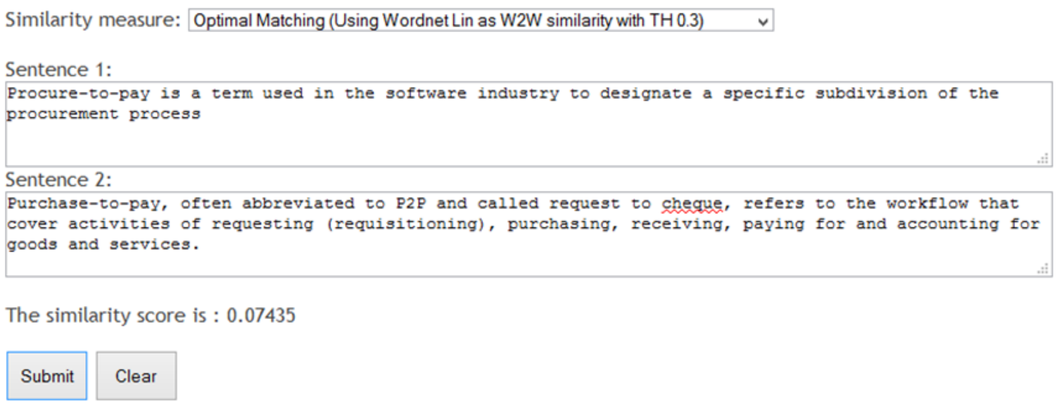


Figure 3. 16 Semilar toolkit

4. Structure Similarity

The structural feature of workflow was described by a complete or partial process model. Structural aspect was arguably the most important aspect of a workflow model. Defining the similarity based on the metric space was useful for quantifying the degree of match for structural feature.

3.2.1.4 Clustering Workflow

Workflow clustering was performed by adapting two traditional clustering algorithms (k-Means and AGNES) for workflow clustering. Clustering was guided by a semantic similarity measure for workflows, originally developed in the context of case-based reasoning. Workflow would be clustered automatically by system based on its $value_{desc}$ and $value_{objective}$. Total similarity which was used for clustering was average value from total similarity $value_{desc}$ and $value_{objective}$.

The threshold of this clustering process was an input parameter of the algorithm. It determined how much similarity was necessary in order to consider two workflows as part of the same cluster. Choosing the value of the threshold was an ad-hoc decision, based on assumptions and prior knowledge and experience.

Presenting example was in Figure 3. 17. Based on the similarity values, there were two groups considering a threshold value of 0.55, since all edges with values lower than 0.55 would be pruned. In this way, workflows #1 and #2 were part of one cluster and workflows #3 and #4 were part of another cluster.

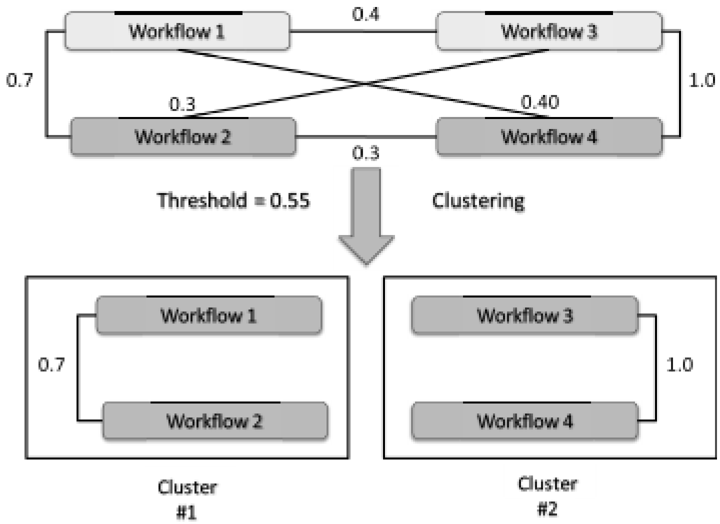


Figure 3. 17 Example of workflow cluster

3.2.1.5 Merging Workflow

Matching need to find which nodes was the first process graph match which nodes in the second process graph. Since there could be different candidate nodes in the second graph that may be matched to a given node in the first graph, and vice versa, the aim of matching two process graphs was to find the best mapping between their nodes. Example was illustrated in Figure 3. 18.

Referring to example in Figure 3. 19, given two graphs and a mapping between their nodes, matching score was computed in three steps. First, systems computing the matching score between each pair of nodes by computing their similarity. The similarity, and thus the matching score, of nodes of different types or between a split and a join were 0.

Workflow models, see example Figure 3. 20, were merged into a single process model. It was representing the behavior of all the individual models, i.e. a basic process model for all these process variants. The algorithm could be split up into three phases: the conversion of the process, the merge of the function graphs, and the conversion of the resulting function graph back into the used process modeling notation. EPC modelling notation was used giving give precise definitions for the algorithm's three phases.

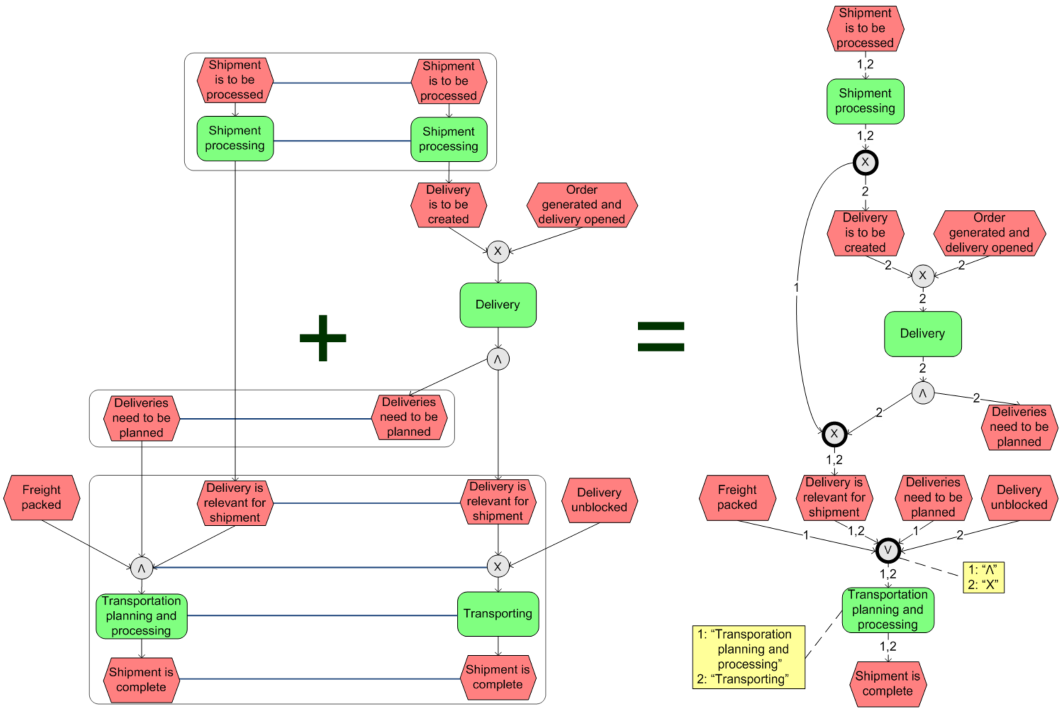


Figure 3. 18 Merging workflow

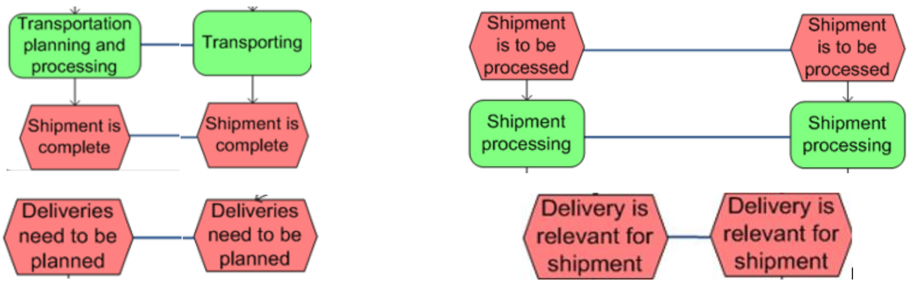


Figure 3. 19 Matched node

Merging was conducted by utilizing merge algorithm as a plug-in of ProM, see Figure 3. 21, which provided the necessary functionality to transform Petri nets to EPCs and vice versa, to import EPCs created with various software tools, to illustrate both Petri nets and EPCs, and to re-use EPCs with other data mining techniques (Xiping Liu, 2007) (Marcello La Rosa H. A.-B., 2010).

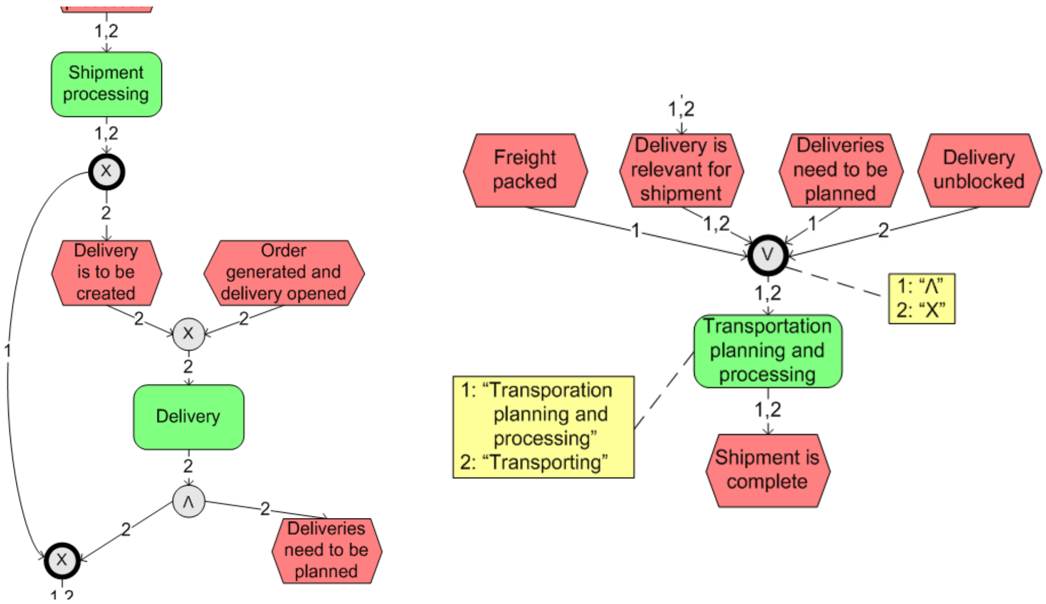


Figure 3. 20 Merging node and edge

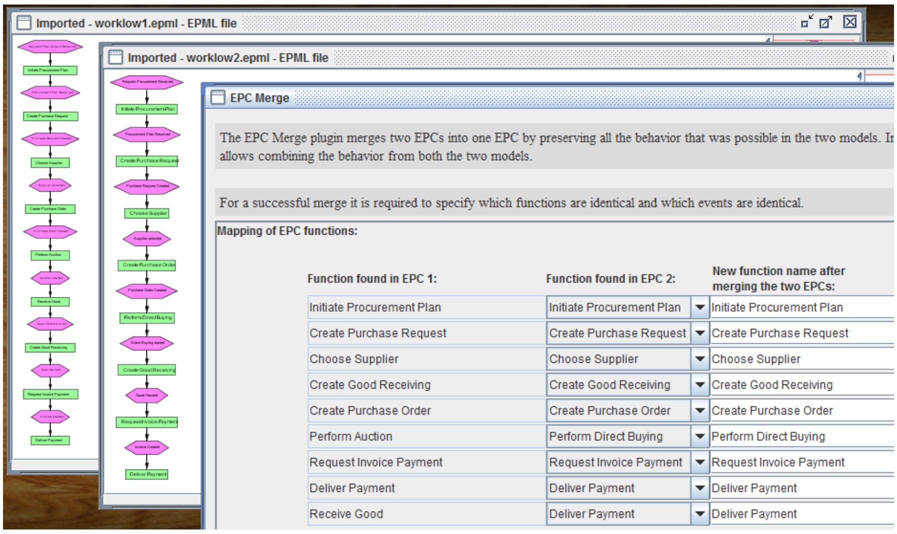


Figure 3. 21 Workflow was mapped onto a commonly agreed upon set of names

3.2.2 Processing

3.2.2.1 Storing the common workflow

Storing the common workflow was conducted by utilizing data structure tree/graph with one root. As mentioned in research constraint, cyclic workflow was excluded. If cyclic workflow was needed, then cyclic fragment has to convert into SESE.

Data structure tree for workflow explain that workflow has many activities. Workflows started by defining start activities. The next activities were derived from its root. Database schema for supporting workflow metadata and how to fill the database were depicted in Figure 3. 22

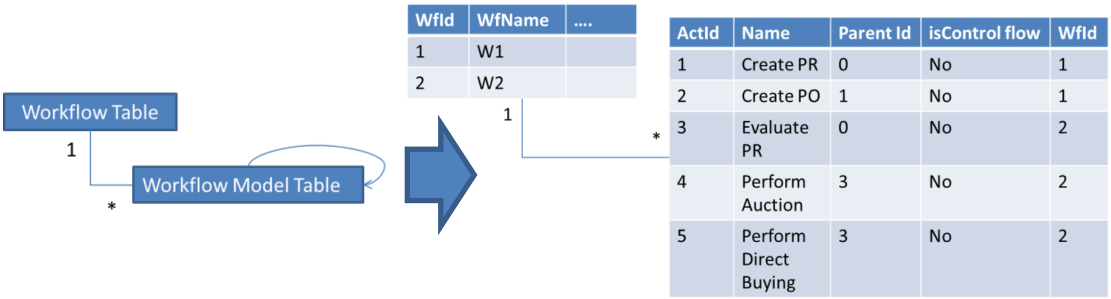


Figure 3. 22 Database schema for storing the common workflow

3.2.2.2 Providing Configurable Workflow

Configurable workflow covers all possible combinations of process parts, which were usually impossible to achieve by providing a set of individual process variants, see illustration in Figure 3. 23. But naturally, the model users do not need all these variations. Instead, they like to have a specific model covering exactly the required process behavior. Hence, the user of a configurable process model needs to configure the integrated model to that subset which depicts this desired behavior (W.M.P. van der Aalst A. K., 2001) (Juha Tiihonen, 2014).

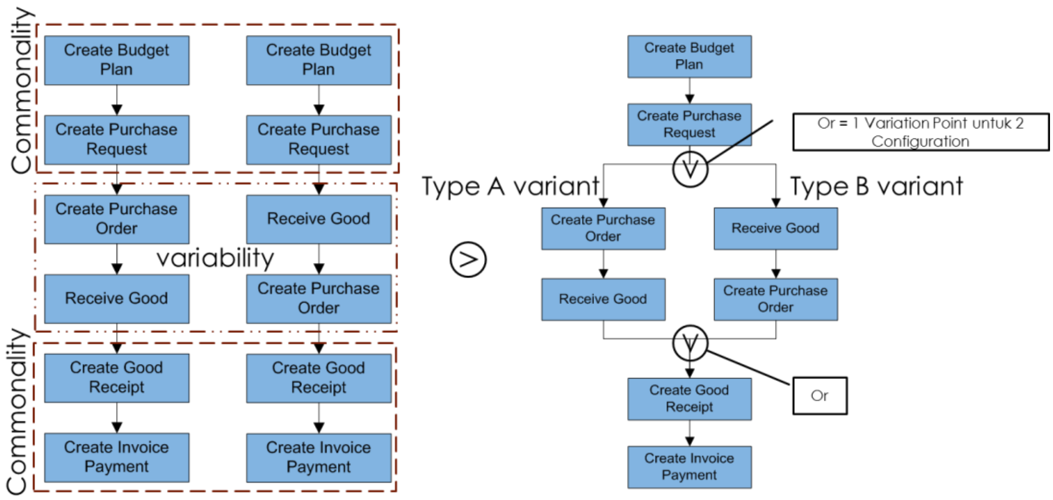


Figure 3. 23 Configurable workflow

1. Generating Question Model

Common of sub workflow consisted of merged node, merged edge and variation point. Variation point was correlating with variability. Capturing it, Variability was captured by means of configuration models composed of questions.

The space of possible answers to a question was represented as a set of facts, each of which could be set to true or false. These facts encode the variability of the system, e.g. optional features, values of configuration parameters, etc. (W.M.P. van der Aalst B. v., 2003) (Chen Xin, 2012).

Question model of workflow was generated manually. It was generated by querying workflow schema in database. Workflow table schema was depicted in Figure 3. 24. Workflow table schema adopted tree data structure, which activity was derived from root activity. By querying workflow model table and referencing workflow id, activity which has more than 1 edge could be counted. This activity has higher probably as variant point. By listing its query result question model could be mapped.

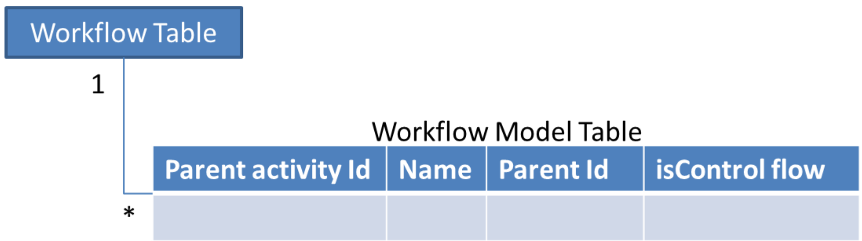


Figure 3. 24 Workflow table schema

2. Generating Individualize Model

Common of sub workflow covered all possible combinations of process parts, which was usually impossible to achieve by providing a set of individual process variants. But naturally, the model users did not need all these variations. Instead, they like to have a specific model covering exactly the required process behavior. Illustration was depicted in Figure 3. 25. Question was knowledge model for extending information. This model was generated by determining where variation point was. Start node acted as question and possibilities action was put as answer. User decided which answer that represented configurability of workflow and matched with their needs.

The individualization of the model or system was captured as configuration actions. As the question was answered, values were assigned to facts, and the resulting facts valuation determines which configuration actions should be performed to derive the individualized model or system, see Figure 3. 26.

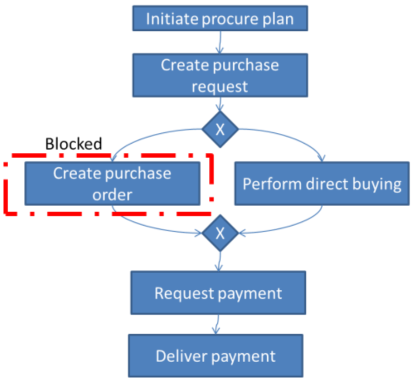


Figure 3. 25 Common of sub workflow and it variation point

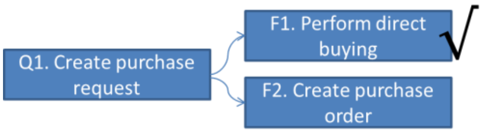


Figure 3. 26 Extracted variation point.

These common of sub workflow could thus provide a better starting point for the users of the configurable process model than the complete integrated model of all process variants could be. If configurations of the basic process model were known leading to the selected, established process variants, then these variants might probably be the closest to our requirements. In this way, the risky and time-consuming task of configuring the process from scratch could be avoided.

3. User Interface

User interface that was presented in this section consisted of user interface for providing common of sub workflow, user interface for query editor, and user interface for storing workflow variant.

Figure 3. 27 was the user interface for providing common of sub workflow. The list contained them by listing description of workflow variant. List of description was ordered by merging time activity. Figure 3. 28 was interface that used for the custom application. This input would be used for retrieval process. Querying could be proceed by defining keyword or determining detail information in metadata way. Figure 3. 29 was interface for helping user by adding annotation for their workflow.

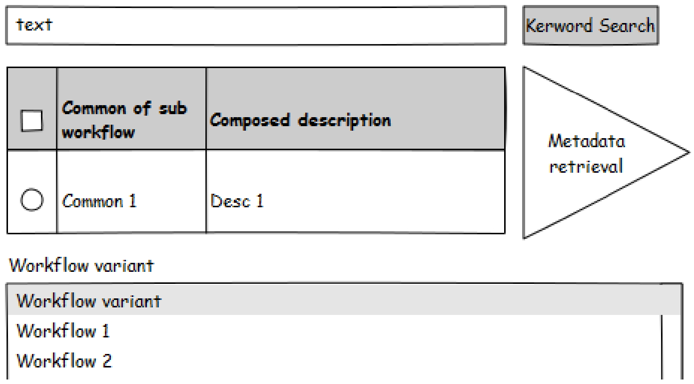


Figure 3. 27 User interface list form

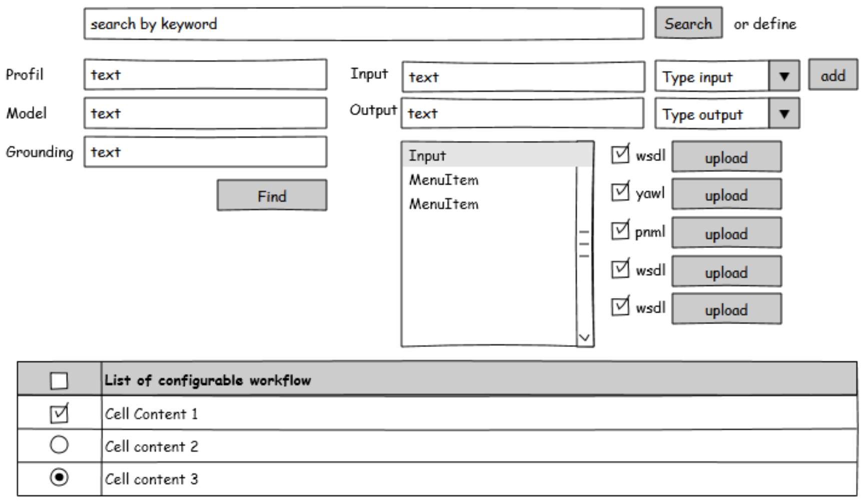


Figure 3. 28 User interface retrieval form

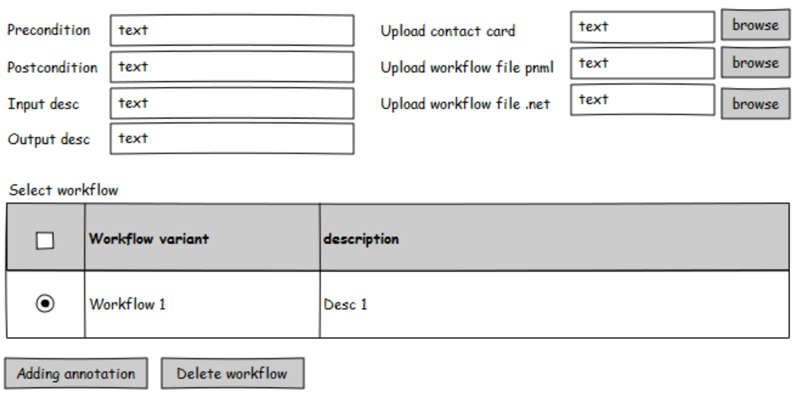


Figure 3. 29 User interface adding notation form

4. Tool Support

Synergia, see Figure 3. 30, was a set of tools supporting process configuration. Synergia consisted of stand-alone application responsible for a specific task in the configuration. Synergia provided an interface to map process facts from a configurable process model to domain facts from the domain configuration model and create an XML serialization of it.

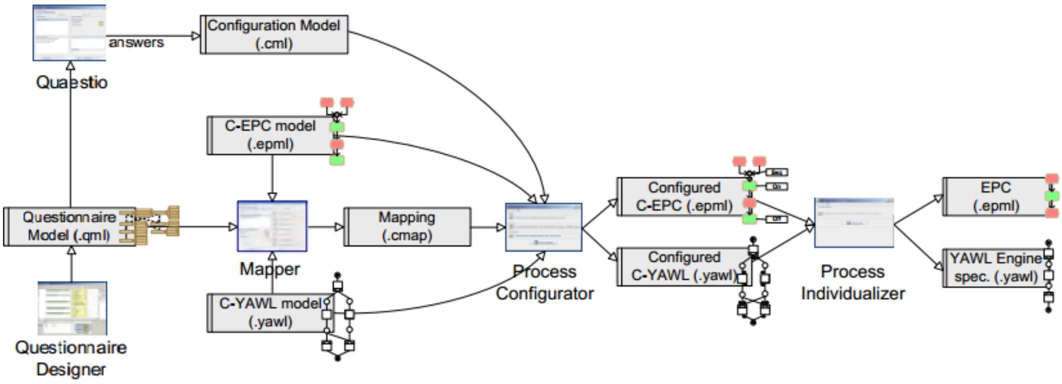


Figure 3. 30 Synergia toolkit

3.3 Evaluation Method

Our evaluation method was presented as Figure 3. 31. Detail evaluation was presented in following section.

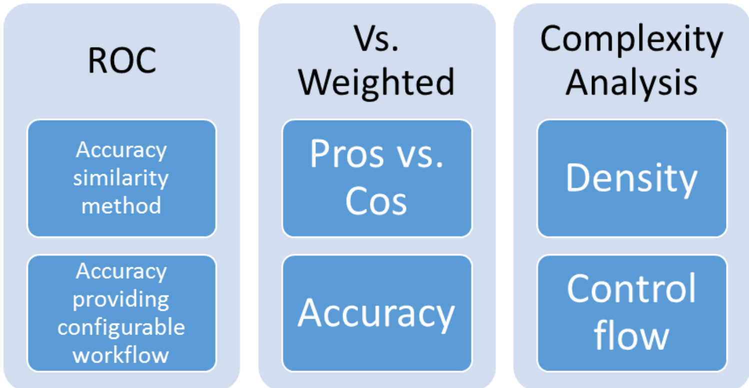


Figure 3. 31 Proposed evaluation method

3.3.1 Measuring Accuracy

Receiver Operating Characteristic (ROC) analysis as a framework to measure the accuracy of the proposed method was used. Measuring accuracy was conducting in 2 different stages. Measuring first similarity was conducted when determining common of sub workflow. The second stages were used for calculating how accurate configurable workflow was provided.

3.3.2 Vs. Averaged Weighting Based Similarity.

In this research, similarity filtering was proposed. As comparison, it could be compared with well-known similarity such as average weighting based similarity. Comparison result lead to conclusion which similarity would fit into repository

3.3.3 Calculating Complexity

The complexity of the algorithm for merging connectors was linear on the maximum number of connectors, which was bounded by the number of edges e of the largest graph to be merged. Complexity analysis should be lower in merging. Complexity analysis was corresponding to understandability, usability and configurability.

3.4 Final Report

In the last step of this research, all of material which was required and conducted during this research was written. Such material embraces research background and motivation, related literature and basic concept, research methodology, research evaluation and its result, and conclusion of the whole parts in this thesis. Complete parts of this research on the book format were written, as well as, publish essential parts of this research in an international journal.

[This page was deliberately left blank]

CHAPTER 4

RESULT OF RESERCH AND EVALUATION

This section presented the scenario and the result of evaluation. It played important roles to prove the hypothesis given in the background in this thesis.

4.1 Scenario

Evaluation that was conducted in this research has goal to answer research question above, to prove, and measure proposed method before it was implemented into system. Scenario was conducted in environment which was depicted in Table 4. 1 and Table 4. 2.

Table 4. 1 Hardware

No	Name	Specification
1.	Processor	Intel Core i3 2.4 GHz
2.	Memory	2048 RAM
3.	Hard disk	300 GB

Table 4. 2 Software

No	Name	Specification
1.	Operating System	Windows 7 Ultimate
2.	Programming	Java, .Net
3.	Database	MySQL
4.	Web Server	Apache Tomcat 6

4.2 Preprocessing

4.2.1 Evaluation Using ROC

Receiver Operating Characteristic (ROC) analysis as a framework to measure the accuracy of the proposed method was used. True-Positive (TP) was obtained from the number of cases, which were actually matches and detected by the system. True-Negative (TN) represented the number of cases, which were not matches and not detected by the system. False-Positive (FP) came from the number of legal cases, which were detected by the system. False Negative (FN) was the number of mismatch cases, which were not detected by the system.

Evaluation was conducted by utilizing procure to pay process. Procure to pay process was generated into 8-variant. The variants were depicted in Appendix 1 to Appendix 8. This variant was assumed as tenant's variant workflow and stored in repository.

Starting storing approach, repository inspected variant by calculating similarity filtering. As assumption, all 8 variant models have same grounding properties. It means variants have similar input. It was reflected based from similar starting node in their workflow graph.

First step was calculating context similarity, see Table 4. 3. Variant was compared with other variant in matrix 2x2. Order of comparing did not take into account since the result was same. System determines threshold value each stage similarity filtering by averaging total similarity value that excluding exacts matching. Workflow which passed first stage, then processed in second stage.

Second stage was profile similarity, see Table 4. 4. System determines threshold value in same way with context similarity in first stage. Total value of profile similarity was average of similarity value of properties in profile attributes. Workflow properties for profile attributes were explained in previous section about designing metadata.

The last stage was structural similarity, see Table 4. 5. Structural similarity has similar approach for determining its threshold. Structural similarity was calculated by utilizing Beehivez using jaccard structured similarity.

Confirming proposed approach about similarity method, then further evaluation about similarity was conducted. Similarity approach was evaluated in larger dataset which consist of 17 domain workflow, see Table 4. 6 and corresponding with profile attribute in Appendix 9, including procure to pay. In default case, procure to pay as data training, it was getting 92% accuracy, see trend diagram in Figure 4. 1. Figure 4. 1 explained that trend accuracy was increasing in each step of similarity filtering.

Table 4. 3 Context similarity for data training

No	Model/Model	W1	W2	W3	W4	W5	W6	W7	W8
1	W1	1	0.5	0.3	0.08	0.09	0.1	0.12	0.11
2	W2		1	0.33	0.09	0.09	0.1	0.13	0.12
3	W3			1	0.13	0.14	0.11	0.14	0.13
4	W4				1	0.24	0.07	0.07	0.15
5	W5					1	0.07	0.13	0.24
6	W6						1	0.36	0.1
7	W7							1	0.15
8	W8								1

Legend
Passed workflow
Excluded pair
Un passed workflow

Table 4. 4 Profile similarity for data training

No	Model/Model	W1	W2	W3	W4	W5	W6	W7	W8
1	W1	1	0.28	0.47	0.18	0.2	0.36	0.48	0.39
2	W2		1	0.54	0.25	0.56	0.5	0.49	0.54
3	W3			1	0.33	0.51	0.55	0.66	0.71
4	W4				1	0.25	0.35	0.31	0.26
5	W5					1	0.65	0.46	0.64
6	W6						1	0.48	0.73
7	W7							1	0.59
8	W8								1

Legend
Passed workflow
Excluded pair
Un passed workflow

Table 4. 5 Structure similarity for data training

No	Model/Model	W1	W2	W3	W4	W5	W6	W7	W8
1	W1	1	0.27	0.22	0.35	0.2	0.11	0.11	0.31
2	W2		1	0.26	0.22	0.36	0.07	0.07	0.2
3	W3			1	0.21	0.26	0.11	0.11	0.17
4	W4				1	0.23	0.14	0.14	0.45
5	W5					1	0.1	0.1	0.2
6	W6						1	0.76	0.16
7	W7							1	0.15
8	W8								1

Legend
Passed workflow
Excluded pair
Un passed workflow

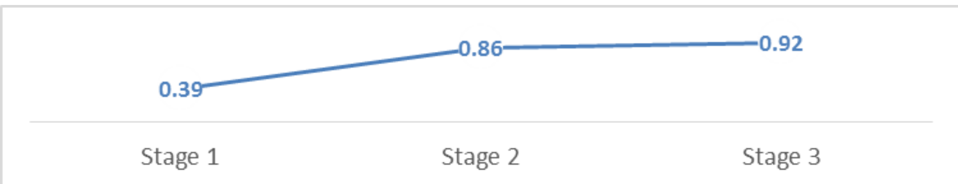


Figure 4. 1 Accuracy trend for data training

Table 4. 6 ERP domain knowledge

No	Domain	Variation
1.	Procure to Pay	8
2.	Pembuatan Jurnal Make To Order	5

No	Domain	Variation
3.	Proses Analysis Revenue	5
4.	Proses Evaluasi Supplier	9
5.	Proses MRL	20
6.	Proses Pembuatan Kontrak	21
7.	Proses Pembuatan Order Card HPP SPH	6
8.	Proses Pembuatan Purchase Request	5
9.	Proses Pembukuan	5
10.	Proses Pencatatan Transaksi Pembelian Untuk Produksi	5
11.	Proses Penerimaan Barang	5
12.	Proses Pengeluaran Barang	10
13.	Proses Penyesuaian dan Kalkulasi Harga	5
14.	Proses Production Order	5
15.	Proses Produksi	6
16.	Proses Sourcing	11
17.	Proses Transaksi Perusahaan Manufaktur	5

Similarity filtering was conducted by performing similarity calculating over 8-variant of workflow to each variant in 17 other ERP process domain. The result has shown that system which performed this approach also getting false positive attribute. The distribution of ROC attributes was depicted in Figure 4. 3. From this kind of evaluation, could be demonstrated in Figure 4. 2, that trend was still increasing and reach 51%.

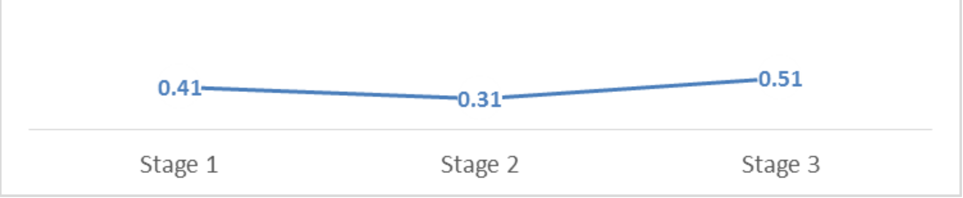


Figure 4. 2 Accuracy trend for data testing

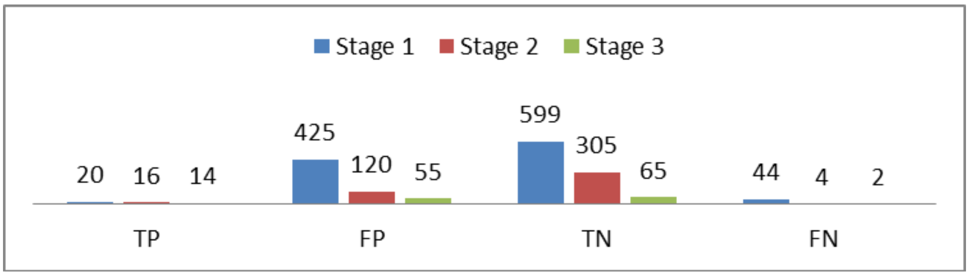


Figure 4. 3 ROC attributes for data testing

4.2.2 Vs. Average Similarity Method

Proposed similarity filtering was based on multi stage filtering including 4 kind of similarity approach. Well known method for similarity was average weighting similarity. This kind of similarity could be including several other similarity approaches. Average similarity worked by multiplying similarity value by weight. The weight was corresponding importance and impact of similarity approach.

Comparing with proposed approach, average similarity was conducted. Average similarity includes context similarity, profile similarity and structural similarly. In two scenarios, weight for each scenario was 30:40:30 and 20:60:20. Determining threshold value was average of similarity value excluding exact matching. First and second scenario each similarity value was multiplying with its weight. Both of them reach accuracy 60%. Proposed approach has higher accuracy than average weighting similarity,

Table 4. 7 Average similarity 30:40:30, threshold 0.29

No	Model/Model	W1	W2	W3	W4	W5	W6	W7	W8
1	W1	1	0.34	0.34	0.2	0.17	0.21	0.26	0.28
2	W2		1	0.39	0.19	0.36	0.25	0.26	0.31
3	W3			1	0.23	0.32	0.29	0.34	0.37
4	W4				1	0.24	0.2	0.19	0.28
5	W5					1	0.31	0.25	0.39
6	W6						1	0.53	0.37
7	W7							1	0.33
8	W8								1

Legend
Passed workflow
Excluded pair
Un passed workflow

Table 4. 8 Average similarity 20:60:20, threshold 0.35

No	Model/Model	W1	W2	W3	W4	W5	W6	W7	W8
1	W1	1	0.32	0.39	0.19	0.18	0.26	0.33	0.32
2	W2		1	0.44	0.21	0.43	0.33	0.33	0.39
3	W3			1	0.27	0.39	0.37	0.45	0.49
4	W4				1	0.24	0.25	0.23	0.28
5	W5					1	0.42	0.32	0.47
6	W6						1	0.51	0.49
7	W7							1	0.41
8	W8								1

Legend
Passed workflow
Excluded pair
Un passed workflow

4.2.3 Complexity Analysis

Performance of merging method was evaluated by analyzing its complexity. Workflows were not static complex systems. They were constantly undergoing revisions, adaptations, changes, and modifications to meet end user's needs. The complexity of workflows tends to increase as they were maintained over a period of time. The typical evolution complexity of workflows was illustrated in Figure 4. 4.

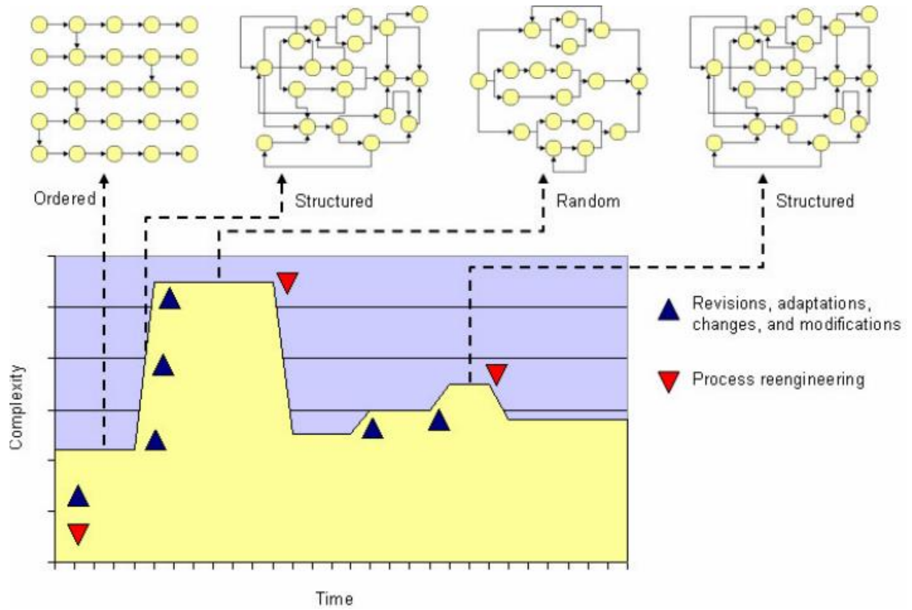


Figure 4. 4 Workflow complexity evolve over time

Complexity analysis was conducted in different scenario. Inspecting other possibilities, merging Workflow1 to workflow 8 was inspected orderly as depicted in Table 4. 9 to Table 4. 15. For certain reason, other workflow namely MTO24 was evaluated in Table 4. 16.

The Control Flow Complexity (CFC) metric evaluates the complexity introduced in a process by the presence of XOR split, OR-split, and AND-split constructs. CFC of all split constructs needs to be added together. The value of CFC should correspond to the values of McCabe's Cyclomatic Complexity for which in practice, the industry interpretation was the following: from 1 to 10, the model was simple; from11 to 20, it was slightly complex; from 21 to 50 it was complex; and above 50 it was untestable.

Size was a common metric that has been empirically validated as an indicator of model complexity. Size could be measured by simply counting all elements within the model like arcs, places and transition.

Density was the ones that most convincingly related to model understandability. Density metric relates the number of available connections to the number of maximum connections for the given number of nodes. The simplest model would be a perfectly sequential model that would have 0 as its density. The most complex model would have an arc between every node in that model and have density as 1.

Complexity trend was depicted in Figure 4. 5 which was shown density became lower and lower. The definition of the density metric builds on the assumption that a lower value was associated with an hardly understanding of the model, which implies as a consequence a higher error-probability. All possible CFC index was counted as simple. lower values would mean make the model more easily to understand. All possible workflow size was bigger and bigger. Size was aligning with complexity. Bigger size indicated workflow was complex. If the workflow was complex, then it has higher error probability.

Table 4. 9 Complexity analysis W1-W2

Name	Type	Value
Control-Flow	Control-Flow	4
Density	Coupling	0.1
AND-Joins	Size	0
AND-Splits	Size	0
XOR-Joins	Size	1
XOR-Splits	Size	1
Arcs	Size	20
Places	Size	10
Transitions	Size	10

Table 4. 10 Complexity analysis W1-W3

Name	Type	Value
Control-Flow	Control-Flow	11
Density	Coupling	0.09090909
AND-Joins	Size	0
AND-Splits	Size	0
XOR-Joins	Size	2
XOR-Splits	Size	3
Arcs	Size	28
Places	Size	11
Transitions	Size	14

Table 4. 11 Complexity analysis W1-W4

Name	Type	Value
Control-Flow	Control-Flow	84
Density	Coupling	0.051226553
AND-Joins	Size	4
AND-Splits	Size	0
XOR-Joins	Size	7
XOR-Splits	Size	11
Arcs	Size	71
Places	Size	21
Transitions	Size	33

Table 4. 12 Complexity analysis W1-W5

Name	Type	Value
Control-Flow	Control-Flow	99
Density	Coupling	0.043
AND-Joins	Size	5
AND-Splits	Size	0
XOR-Joins	Size	9
XOR-Splits	Size	14
Arcs	Size	86
Places	Size	25
Transitions	Size	40

Table 4. 13 Complexity analysis W1-W6

Name	Type	Value
Control-Flow	Control-Flow	154
Density	Coupling	0.030295566
AND-Joins	Size	6
AND-Splits	Size	0
XOR-Joins	Size	13
XOR-Splits	Size	18
Arcs	Size	123
Places	Size	35
Transitions	Size	58

Table 4. 14 Complexity analysis W1-W7

Name	Type	Value
Control-Flow	Control-Flow	154
Density	Coupling	0.025664361
AND-Joins	Size	6
AND-Splits	Size	0
XOR-Joins	Size	16
XOR-Splits	Size	22
Arcs	Size	141
Places	Size	41
Transitions	Size	67

Table 4. 15 Complexity analysis W1-W8

Name	Type	Value
Control-Flow	Control-Flow	222
Density	Coupling	0.024475524
AND-Joins	Size	10
AND-Splits	Size	0
XOR-Joins	Size	18
XOR-Splits	Size	26
Arcs	Size	168
Places	Size	44
Transitions	Size	78

Table 4. 16 Complexity analysis MTO24

Name	Type	Value
Control-Flow	Control-Flow	1
Density	Coupling	0.009433962
Cross-Connectiv...	Cohesion	0.49963369963...
Num of Functions	Size	51
Num of Events	Size	52
Num of ORs	Size	0
Num of XORs	Size	0
Num of ANDS	Size	2

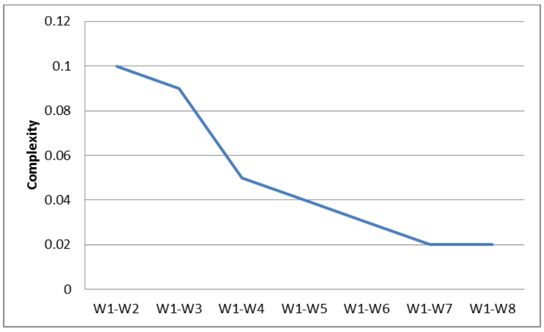


Figure 4. 5 Workflow trend

Table 4. 17 Complexity analysis W1, W2 and W3

Table 4. 18 Complexity analysis W1 and MTO24

Name	Type	Value
Control-Flow	Control-Flow	2
Density	Coupling	0.04347826
Cross-Connectiv...	Cohesion	0.13614985837...
Num of Functions	Size	10
Num of Events	Size	10
Num of ORs	Size	0
Num of XORs	Size	2
Num of ANDS	Size	0

Name	Type	Value
Control-Flow	Control-Flow	3
Density	Coupling	0.007936508
Cross-Connectiv...	Cohesion	0.30593923322...
Num of Functions	Size	59
Num of Events	Size	61
Num of ORs	Size	0
Num of XORs	Size	2
Num of ANDS	Size	2

Table 4. 19 Complexity analysis for W6, W7

Table 4. 20 Complexity analysis for W4, W5, W8

Name	Type	Value
Control-Flow	Control-Flow	8
Density	Coupling	0.0952381
Cross-Connectiv...	Cohesion	0.06977146936...
Num of Functions	Size	12
Num of Events	Size	12
Num of ORs	Size	2
Num of XORs	Size	6
Num of ANDS	Size	0

Name	Type	Value
Control-Flow	Control-Flow	14
Density	Coupling	0.1
Cross-Connectiv...	Cohesion	0.03338508003...
Num of Functions	Size	13
Num of Events	Size	13
Num of ORs	Size	2
Num of XORs	Size	10
Num of ANDS	Size	1

4.3 Processing

4.3.1 Providing Configurable Workflow

Evaluation querying common of sub workflow was conducted to measure how accurate common of sub workflow was provided. Querying configurable workflow achieve accuracy 92% for getting recommendation of workflow variant and 1 for getting configurable workflow. Getting configurable workflow achieve 1 since all workflow composed configurable workflow was recommendation of workflow variant. Querying configurable workflow was also conducted in data testing. Testing achieve accuracy 51% for getting recommendation of workflow variant and 1 or getting configurable workflow. This result proved that similarity filtering was success for collecting majority of similar workflow into configurable workflow. User need recompose them using workflow designer for creating new

Similarity filtering recommended 3 common of sub workflow for providing configurable workflow. The recommendation was merged model from W1, W2 and W3 and merged model from W6 and W7. These merged models were depicted in Appendix 10 and Appendix 11 (Appendix C). Similarity between them was 0.06 in context similarity and 0.13 in structure similarity.

Merged model from W1, W2 and W3 should be mapped into question model. the model has variation in purchase order. From purchase order user could choose his configuration into Perform auction, perform direct buying or nothing.

User could chose on or many options to be used in their system. Dot dash in Figure 4. 6 shown that after chose purchase order variation, user should defined next variation about Create Good Receive. The option was used it or not. These models help out user and modeler to identify workflow configuration. Figure 4. 7 showed that question model for merged model W6 and W7. There were 3 variation points.

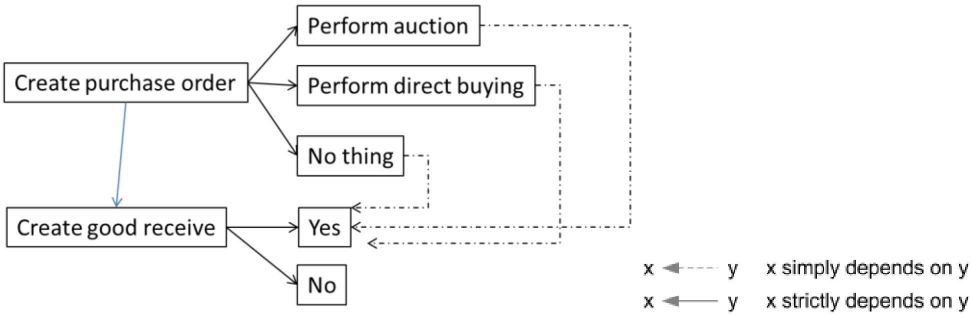


Figure 4. 6 Question model W1,W2, W3

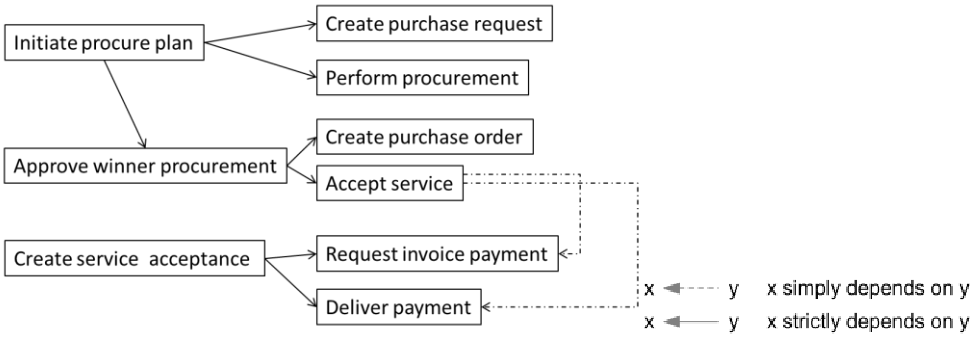


Figure 4. 7 Question model W6 and W7

Adding configuration could be tested in question detail by adding answer (adding answer/option then possibilities would be richer). As an example, based on Figure 4. 6, user could add Perform Outline Agreement from Purchase Order. When there were options adding in question model, it meant model should be configured by adding node after control flow Purchase Order.

Adding component into configurable workflow was simple task. Selected configurable workflow was loaded into workflow designer in repository, see Figure 4. 8. Workflow designer in repository allows to design and execute multiple xaml based workflows in a windowed environment akin with Visual Studio.. Workflow designer repository consisted feature such as:

1. Develop xaml based workflows and workflow services

2. Load 2 workflow or more
3. Toolbox support for all standard wf activities which has been filled by all standard wf 4.0 activities
4. Add custom activity into workflow model. Execute Workflows Concurrently
5. Execute Workflows Concurrently
6. Show Workflow Validation Errors
7. Debug Workflows
8. Docking

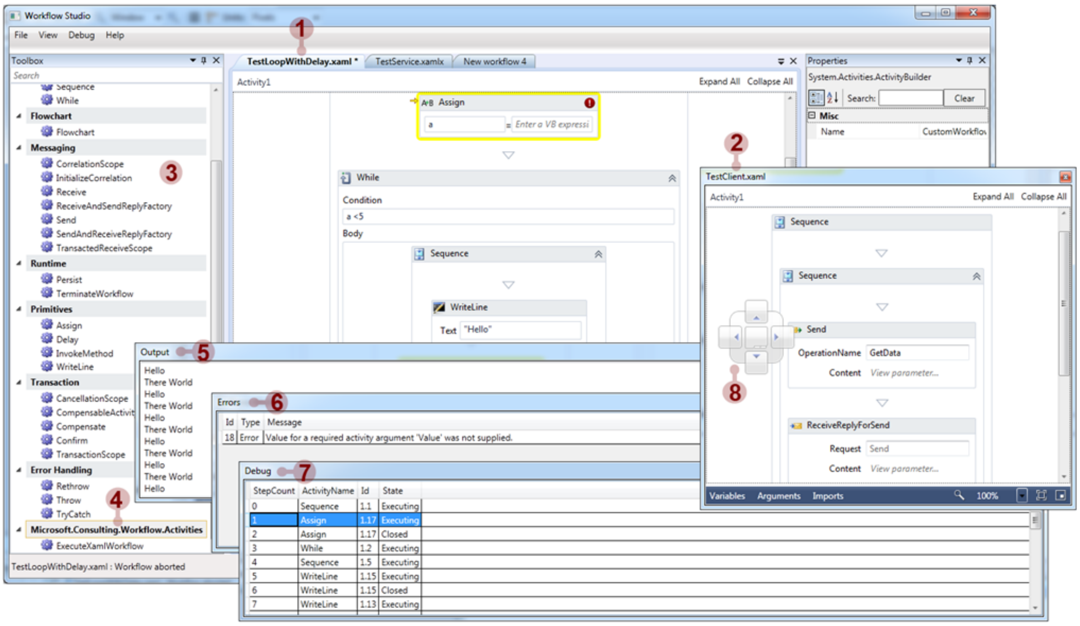


Figure 4. 8 Workflow repository designer studio

4.3.2 Prototype Implementation

As a proof of concept, we implemented a prototype of repository according to the architecture described in Section 3. The prototype supports the following features

1. Basic features: model import/export, model search, model classification
2. Filtering functionality: similarity search
3. Design functionality: merge-driven creation

These features could be accessed via a Web portal, or directly by using the available Web services, see Figure 4. 9 Workflow repository. The portal exposes the above features through a graphical interface to provide process models visualization and editing capabilities, see Figure 4. 8. Specifically, the portal was implemented using the so-called “model-view-controller” pattern, where the portal itself was merely a view on the models stored in the underlying database.

Internally, both the models archive and the canonical models archive were implemented using a single MS SQL service database, although these were exposed as two separate logical entities through data-centric Web services. Currently, the process modeling languages that were supported were EPCs.

In the following we describe two example scenarios that were supported by the prototype repository. In the first scenario an organization could use the workflow repository for advanced search functionality. The collection of process models to be analyzed could originate from windows workflow designer and EPC Editor Eclipse and then imported into the repository. User could search for a particular model based on keywords or by using an input model that was similar to the model to be searched for.

In the second scenario, recomposing workflow creation provided by the repository could be used to assist an organization integrating their process models with those of another organization. In case of merger between 2 organizations, both organizations could import their sets of business processes in the repository, see Figure 4. 10.

Subsequently, they could use the similarity search function to search for pairs of models that were similar. Matching similarity establish between elements from one process and elements from the other process and merge the two models into a configurable process model, using this match. The resulting model would capture both the commonalities between the two models, and their differences, in the form of variation points. This new model could then provide a roadmap for implementing changes to the current business services and IT infrastructure supporting the business process, in order to recomposing them, see Figure 4. 11 Workflow editor.

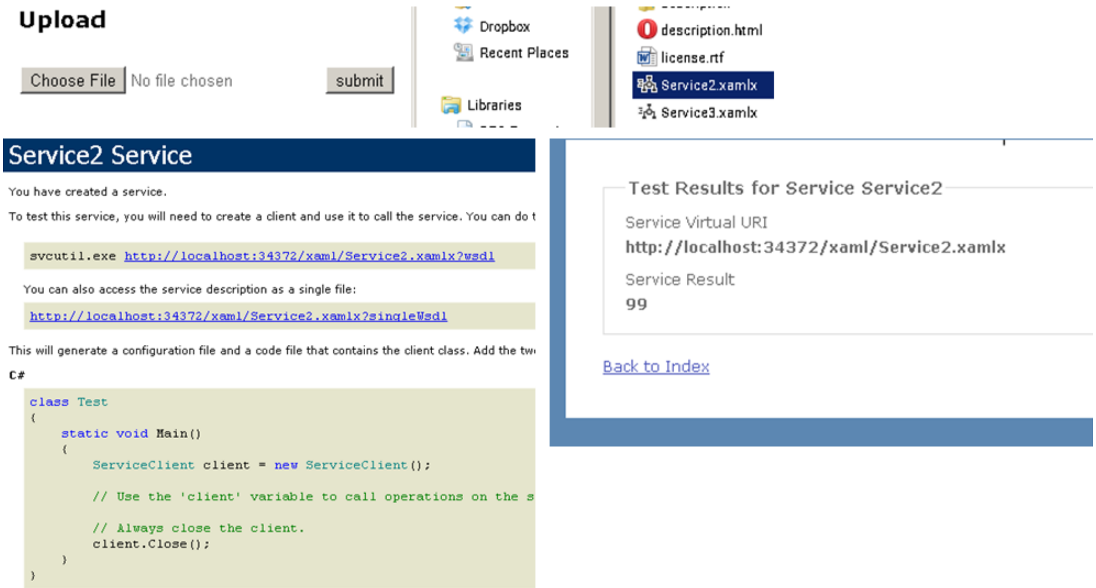


Figure 4. 9 Workflow repository

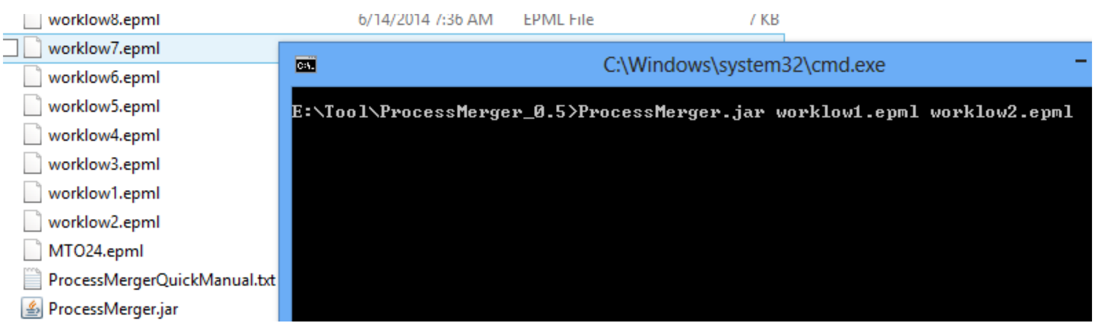


Figure 4. 10 Generating model for identifying common and variant of sub workflow

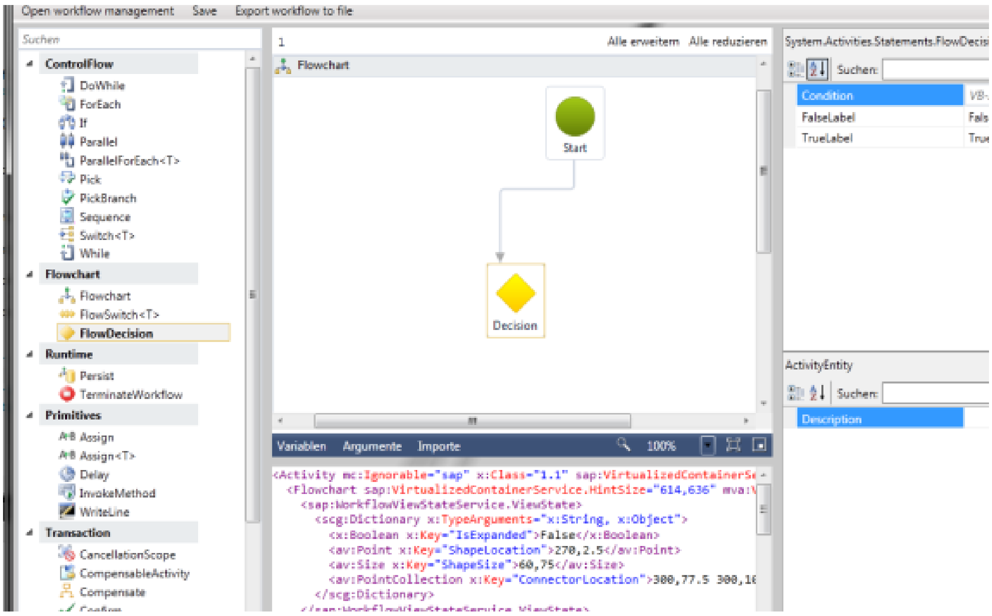


Figure 4. 11 Workflow editor

[This page was deliberately left blank]

CHAPTER 5
CONCLUSION AND FUTURE WORK

This section was the final section in this thesis. The goal was to summarize the whole content of this report and to provide some recommendation for related researches.

5.1 Threat of Validity

This research used data from ERP system. ERP that was used was developed for internal research. The controlled experiment was conducted in a small scale. Procure to pay workflow was modelled by referencing real world procure to pay in SAP ERP. Modelling workflow which was not based on real situation should be avoided. Another factor that may have influenced the final results of the experiment was that all of the subjects of the experiment were domain workflows in data testing and common of sub workflow. Domain workflow in data testing was generated by student in ERP class. These students only had very little theoretical background in BP modeling and they had had very little practice with actual modeling. it was shown that people who have had more actual practice in creating BP models understand their specifics in greater detail and therefore understand more complex models easier. It could be argued that if the subjects of the controlled experiment had had more real-life experience in BP modeling, the results could have been different. The next factor that could have affected the results was common of sub workflow. The subjects were not data aware. It was purely about process. When data constraint was taking into account, variation could have been different.

5.2 Conclusion

Conclusion for this research was.

1. Architecture of workflow repository has been implemented by utilizing several plugin. The architecture was supported by MS SQL server database and portal interface via web and desktop application.

2. Repository provided workflow metadata in xamlx format. This format has advantages for displaying workflow in graph via workflow editor. Comparing with other repository which was stated in Section 1, external storage model in xamlx format has many advantages than RDF or BPEL. RDF was not executable. BPEL faced interoperability issue.
3. Workflow repository has storage for common of sub workflow and storage of workflow variant. Common of sub workflow was made by workflow variant.
4. Constructing common of sub workflow was preceded by similarity filtering which consists of several similarity approach such profile similarity, grounding similarity and context similarity.
5. Comparing similarity filtering to average weighting similarity produced better result. Accuracy for similarity filtering has reached 92% than average only reached 80%.
6. Merging process produce common of sub workflow. Evaluating result product of merging process, it was density, control flow complexity and size. Common of sub workflow has bigger size and harder understandable.
7. Resolving complexity issue, question model was used. Variation point could be modelled into list of questions answer.
8. Repository could provide configurable ERP by providing common of sub workflow in portal. If any user query about their needs then system was looking for to the workflow variant data set first. All recommendation in workflow variant was matched to common of sub workflow data set. Common of sub workflow which contains majority recommendation then was provided.
9. Using proposed similarity filtering, system achieved 92% accuracy. Providing configurable workflow achieved 100% accuracy. System could achieve higher accuracy configurable workflow since, in preprocessing stage, system has been investigated workflow by its similarity and the common workflow was constructed based on recommendation of them.

5.3 Future Work

Future works were as follow

1. Implement the runtime query processing approach so that empirical evaluation could be performed and scalability and complexity analysis could be rigorously conducted.
2. Find a way to handle complex data type among platform.
3. Build workflow fragment knowledge repository, which focus on fragment. This kind of repository would be built in different system than workflow repository. It drive vision which workflow repository as workflow collaborator.

[This page was deliberately left blank]

REFERENCES

Babbie, E. R. (1990). *Survey research methods*. Wadsworth Pub. Co.

Chen Xin, C. B.-g.-h.-s. (2012). A Model of Workflow Composition for Emergency Management. *Physics Procedia, Volume 33, Pages 567-572*.

Chunhyeok Lim, S. L. (2011). Storing, reasoning, and querying OPM-compliant scientific workflow provenance using relational databases. *Future Generation Computer Systems, Volume 27, Issue 6, Pages 781-789*.

Dijkman, R. D.-B. (2009). Graph Matching Algorithms for Business Process Model Similarity Search. *Business Process Management. LNCS, vol. 5701 Springer, Heidelberg*.

Dijkman, R. M. (2011). Similarity of Business Process Models:Metrics and Evaluation. *Information Systems*.

Donglai Zhang, P. C. (2011). Web services workflow with result data forwarding as resources. *Future Generation Computer Systems, Volume 27, Issue 6, Pages 694-702*.

Dumas, M. G.-B. (2009). Similarity Search of Business Process Models. *Technical Committee on Data Engineering*.

Ekanayake, C. C.-C. (2011). Fragment-based version management for repositories of business process models. *On the Move to Meaningful Internet Systems*.

Feuerlicht, G., & Lozina, J. (2007). Understanding Service Reusability. *SYSTEMS INTEGRATION*.

Granell, C., Dı ´az, L., & Gould, M. (2010). Service-oriented applications for environmental models: Reusable geospatial services. *Environmental Modelling & Software*.

Jae-Yoon Jung, H. K.-H. (2006). Standards-based approaches to B2B workflow integration. *Computers & Industrial Engineering, Volume 51, Issue 2, Pages 321-334*.

Juha Tiihonen, W. M. (2014). Chapter 21 - Configuring Services and Processes. *Knowledge-Based Configuration, Pages 251-260*.

Karsten A. Schulz, M. E. (2004). Facilitating cross-organisational workflows with a workflow view approach. *Data & Knowledge Engineering, Volume 51, Issue 1, Pages 109-147*.

Kim, K. (2012). A model-driven workflow fragmentation framework for collaborative workflow architectures and systems. *Journal of Network and Computer Applications, Volume 35, Issue 1, Pages 97-110*.

Klemens, E. K. (2000). XML Schema Directory: A Data Structure for XML Data Processing. *IEEE*.

Lu, R. S. (2009). On managing business processes variants. *Data & Knowledge Engineering*.

M. Rosemann, W. v. (2007). A configurable reference modelling language. *Information Systems, Volume 32, Issue 1, Pages 1-23*.

Marcello La Rosa, H. A.-B. (2010). APROMORE: An Advanced Process Model Repository. *Expert Systems with Applications*.

Marcello La Rosa, M. D. (2011). Configurable multi-perspective business process models. *Information Systems, Volume 36, Issue 2, Pages 313-340*.

Mcoll, E., Jacoby, A., & Thomas, L. (2001). Design and use of questionnaires: A review of best practice applicable to surveys of health service staff and patients. *National Co-ordinating Centre for HTA*.

Microsoft. (2014, 1 19). *Workflow Services Overview*. Retrieved from Workflow Services Overview: http://msdn.microsoft.com/en-us/library/dd456797(v=vs.110).aspx

Mor Peleg, A. G.-F. (2014). Chapter 16 - Guidelines and Workflow Models. *nical Decision Support (Second Edition), Pages 435-464*.

P. J. PetiaWohed, J. (2007). A Universal Repository for Process Models. *The Department of Computer and Systems Sciences*.

Paul Grefen, R. R. (1998). A reference architecture for workflow management systems. *Data & Knowledge Engineering, Volume 27, Issue 1, Pages 31-57*.

Reichert, M. U. (2012). *Enabling flexibility in process-aware information systems: challenges, methods, technologies.* Springer.

Sameer Malhotra, D. J. (2007). Workflow modeling in critical care: Piecing together your own puzzle. *Journal of Biomedical Informatics, Volume 40, Issue 2, Pages 81-92*.

Sarno, R., & Herdiyanti, A. (2010). A Service Portfolio for an Enterprise Resource Planning. *International Journal of Computer Science and Network*, *10*(3).

semanticsimilarity.org. (n.d.). *Semilar*. Retrieved from Semilar: http://www.semanticsimilarity.org/

Shaokun Fan, J. L. (2012). A framework for transformation from conceptual to logical workflow models. *Decision Support Systems, Volume 54, Issue 1, Pages 781-794*.

Silva, V., Chirigati, F., Maia, K., Ogasawara, E., de Oliveira, D., Braganholo, V., et al. (2011). Similarity-based workflow clustering. *Journal of Computational Interdisciplinary Sciences*.

van der Aalst, W. d. (2006). Comparing Two Process Models based on Observed Behavior. *Springer, Heidelberg*.

van Dongen, B. D. (2008). Measuring Similarity between Business Process Models. *Bellahs`ene, Z., L´eonard, M. (eds.) CAiSE, Springer, Heidelberg* .

van Dongen, B. R. (2008). Measuring Similarity between Business Process Model. *Springer Berlin Heidelberg*.

W.M.P. van der Aalst, A. K. (2001). A reference model for team-enabled workflow management systems. *Data & Knowledge Engineering, Volume 38, Issue 3, Pages 335-363*.

W.M.P. van der Aalst, B. v. (2003). Workflow mining: A survey of issues and approaches. *Data & Knowledge Engineering, Volume 47, Issue 2, Pages 237-267*.

Wolfgang Marquardt, M. N. (2003). Workflow and information centered support of design processes. *Computer Aided Chemical Engineering, Volume 15, Pages 101-124*.

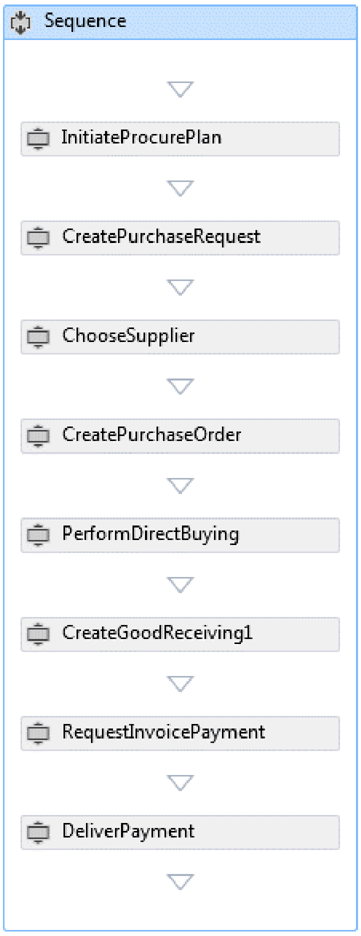
Xiping Liu, W. D. (2007). On design, verification, and dynamic modification of the problem-based scientific workflow model. *Simulation Modelling Practice and Theory, Volume 15, Issue 9, Pages 1068-1088*.

Zarine, S. J. (2012). Improving ERP Integration. *International Conference on Computer & Information Science (ICCIS), pp. 979–984*.

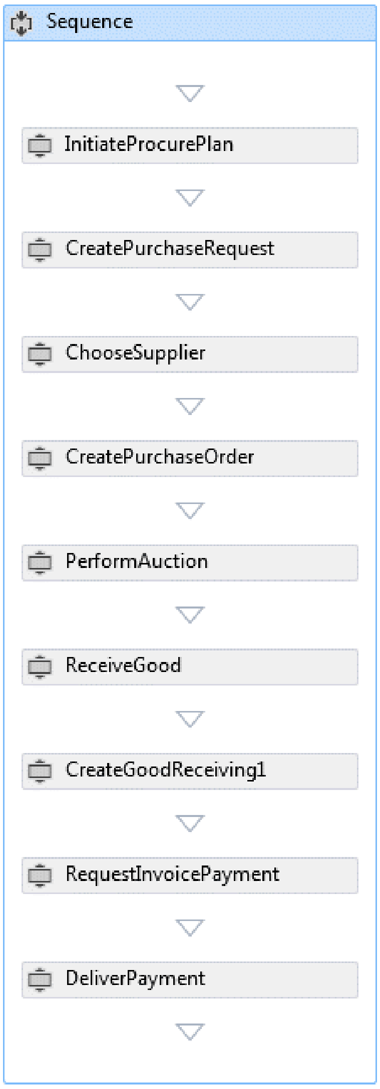
Zhiqiang Yan, R. D. (2010). Fast Business Process Similarity Search with Feature-Based Similarity Estimation. *Springer-Verlag Berlin Heidelberg*.

Zhiqiang Yan, R. P. (2009). Business Process Model Repositories - Framework and Survey. *Information Sciences*.

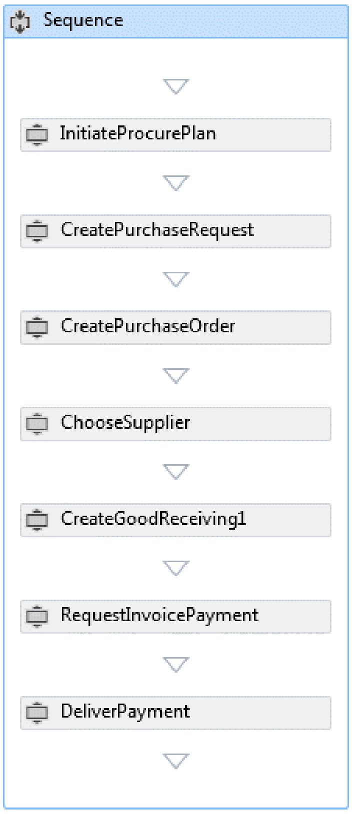
APPENDIX A WORKFLOW PROCURE TO PAY



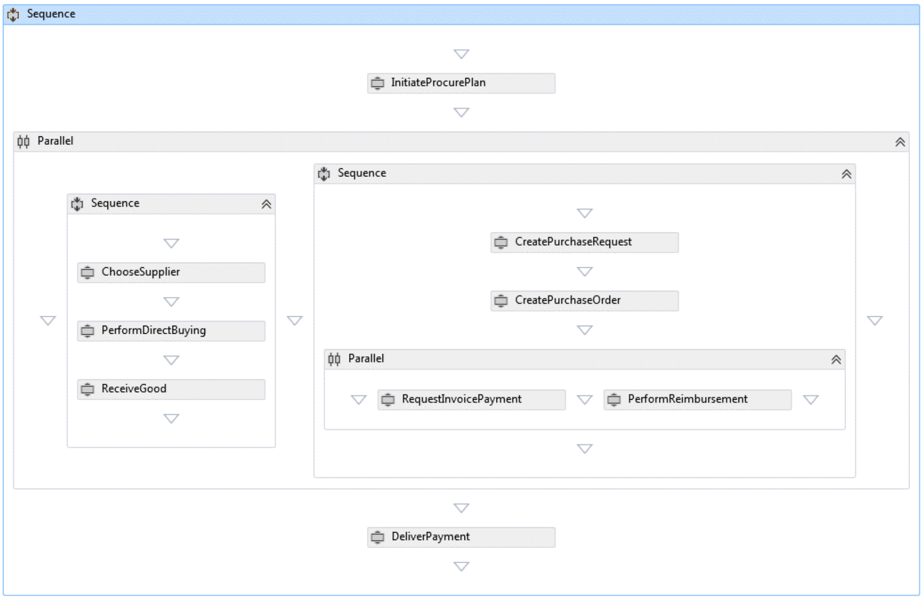
Appendix 1 Workflow 1



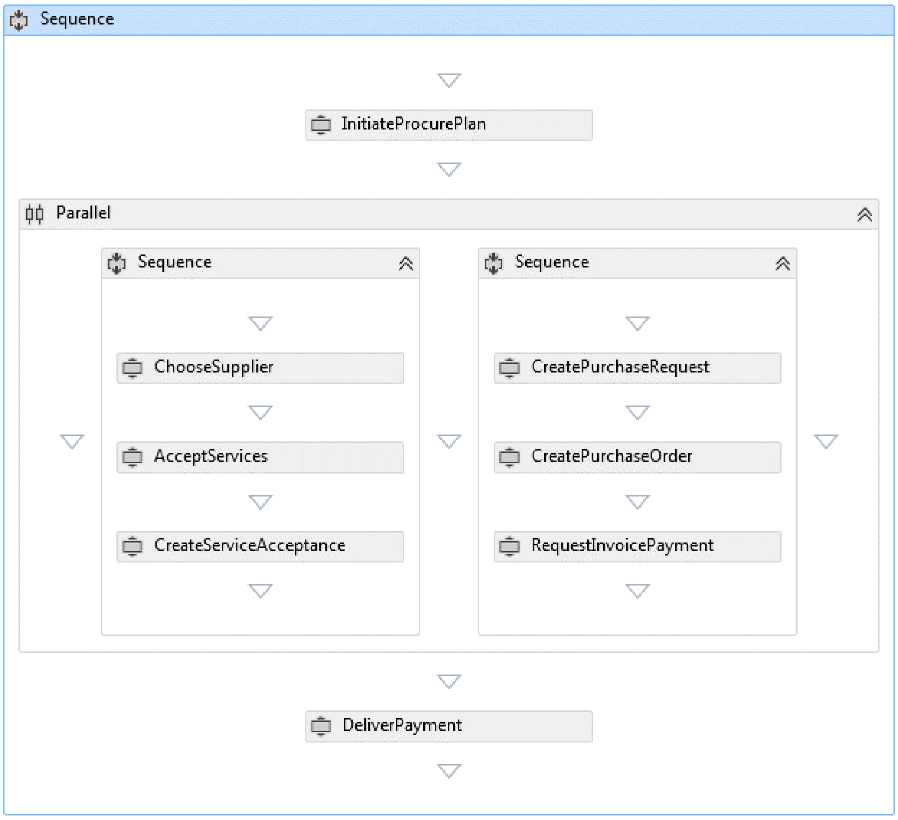
Appendix 2 Workflow 2



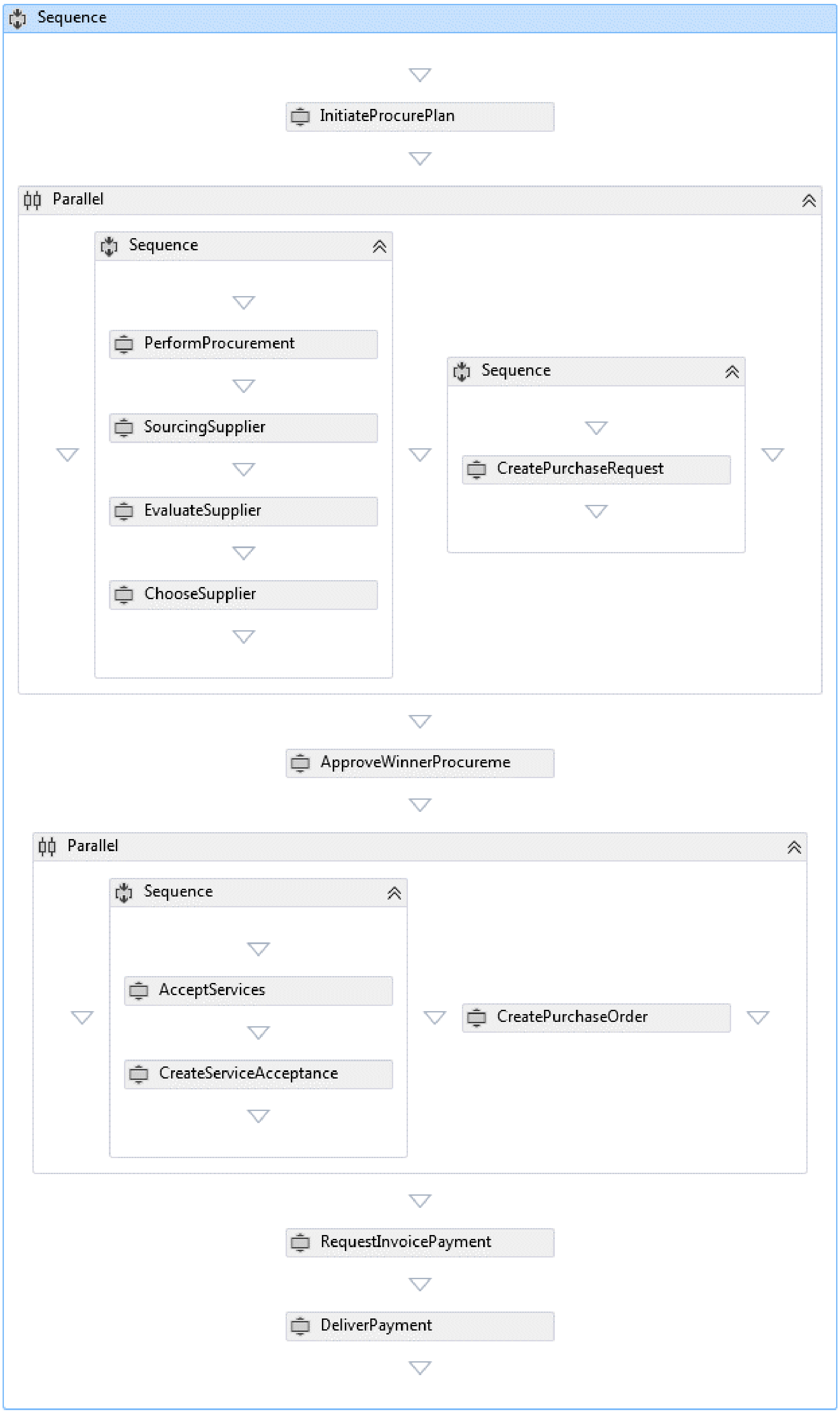
Appendix 3 Workflow 3



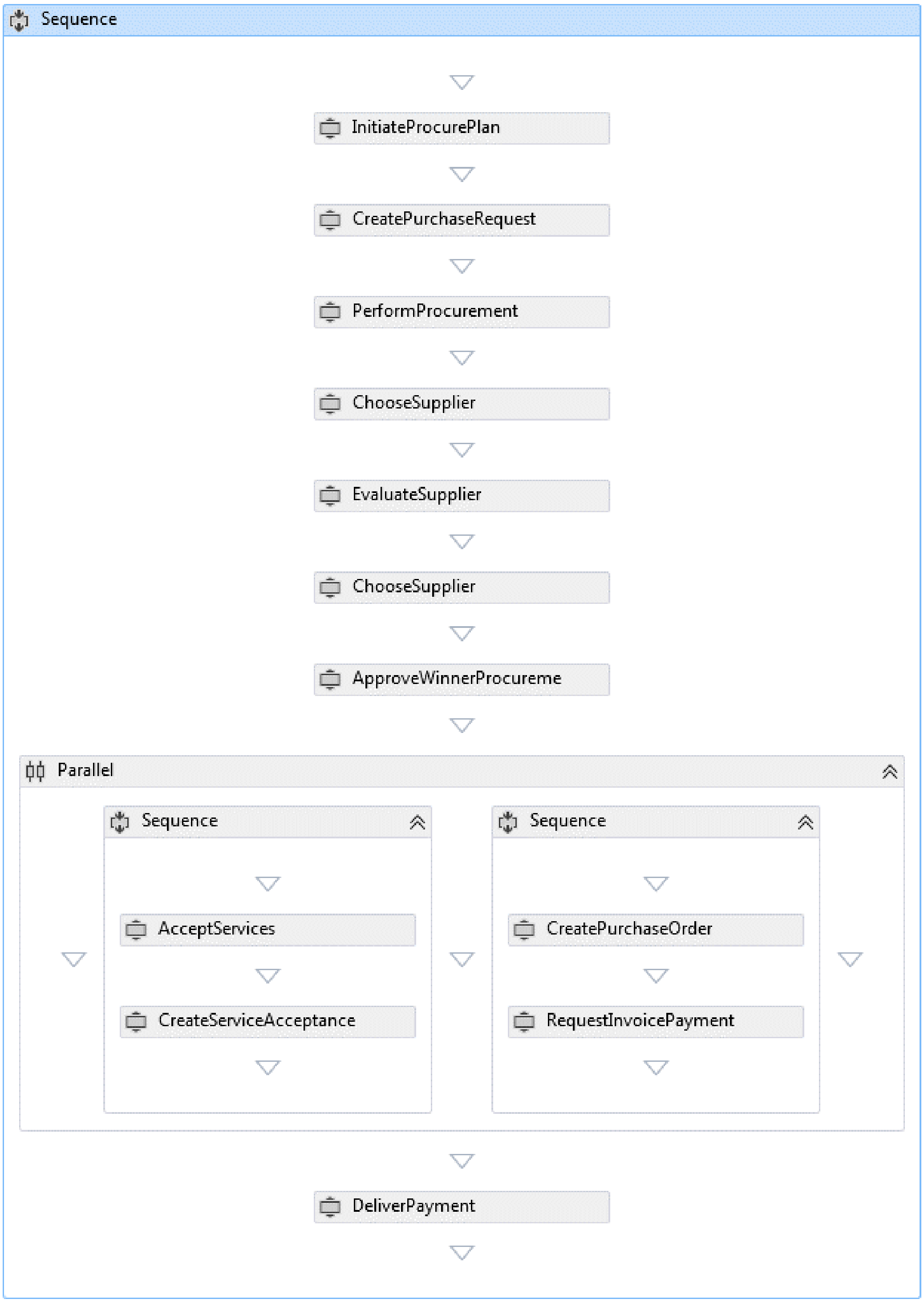
Appendix 4 Workflow 4



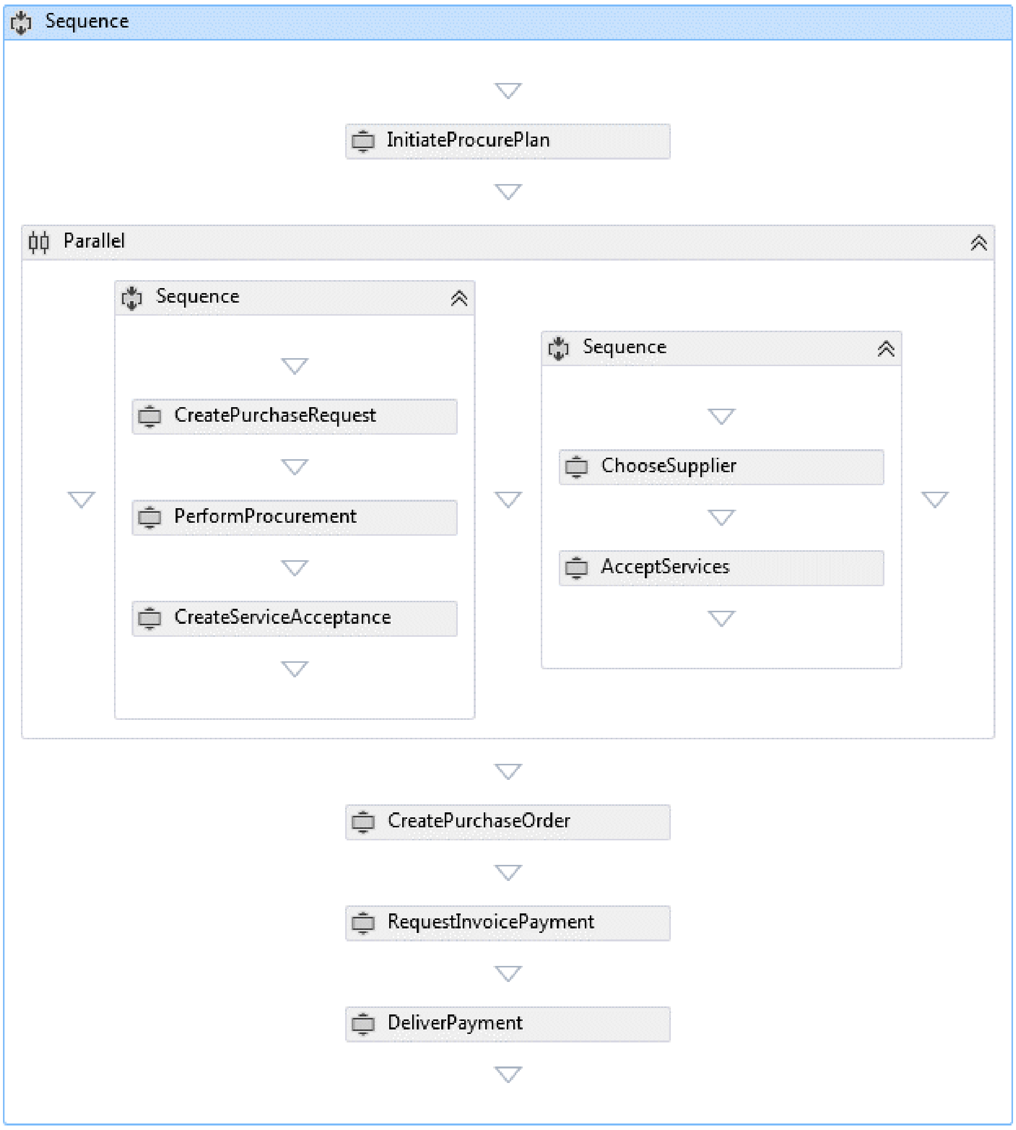
Appendix 5 Workflow 5



Appendix 6 Workflow 6



Appendix 7 Workflow 7



Appendix 8 Workflow 8

[This page was deliberately left blank]

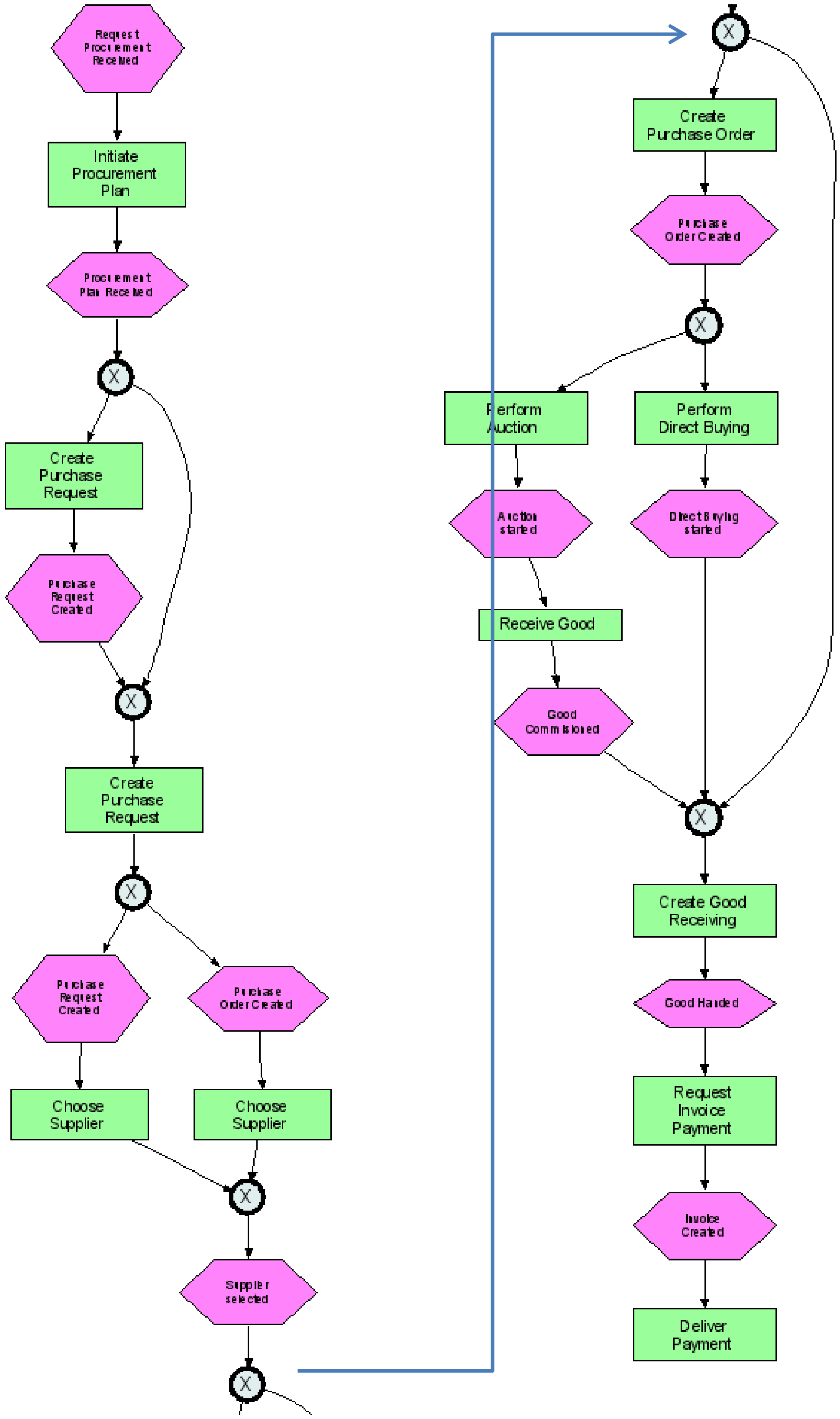
APPENDIX B PROFILE ATRIBUTE FOR DATA TRAINING

Appendix 9 Default description for data training

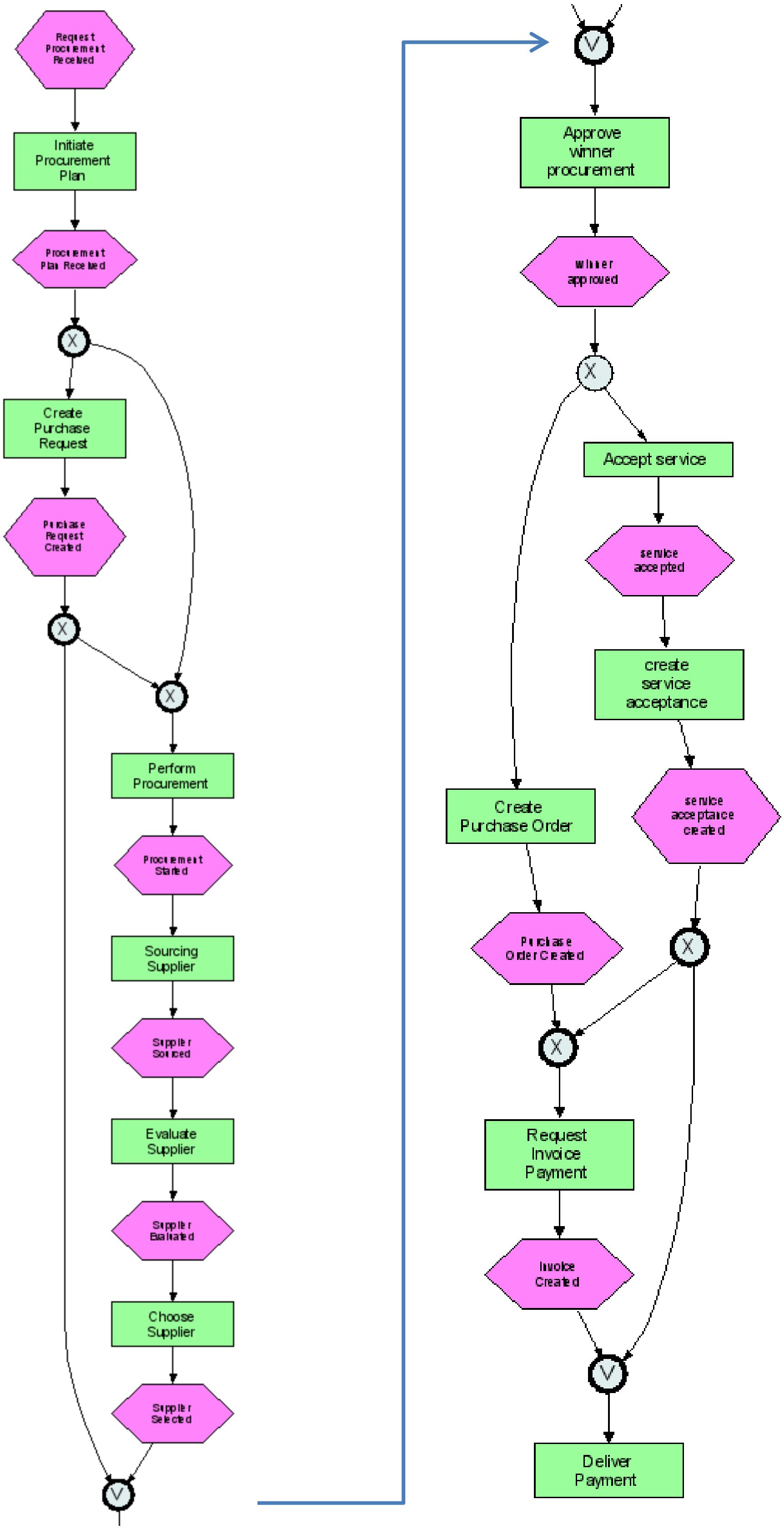
No	Name	Description
1.	Procurement Process Cycle	Procure-to-pay was a term used in the software industry to designate a specific subdivision of the procurement process.
2.	Procure to Pay P2P	Purchase-to-pay, often abbreviated to P2P and called request to cheque, refers to the workflow that cover activities of requesting (requisitioning), purchasing, receiving, paying for and accounting for goods and services.
3.	Purchase To Pay (P2P)	Procure to pay (purchase to pay or P2P) was the process of obtaining and managing the raw materials needed for manufacturing a product or providing a service.
4.	Procure-to-Pay Solution	Procure-to-Pay was an integrated solution that links purchasing and payables to maximize return on invested capital.
5.	Procure-to-Pay and Commercial	The process of buying goods, which includes the initial decision to make the purchase, the process of selecting the goods, and the transaction, made to pay for the goods purchased.
6.	Procure to Pay	The Procure-to-Pay (PTP) process contains all steps from purchasing goods from a supplier, to paying the supplier.
7.	Procure-to-Pay	Procurement Process: – Every organisation that operates a business has to purchase materials such as raw materials, services etc.
8.	Procure To Pay (P2P)	The P2P or Procure To Pay cycle was vital to an organization as in this process the organization buys and received goods or services from its vendors and makes necessary payments.

[This page was deliberately left blank]

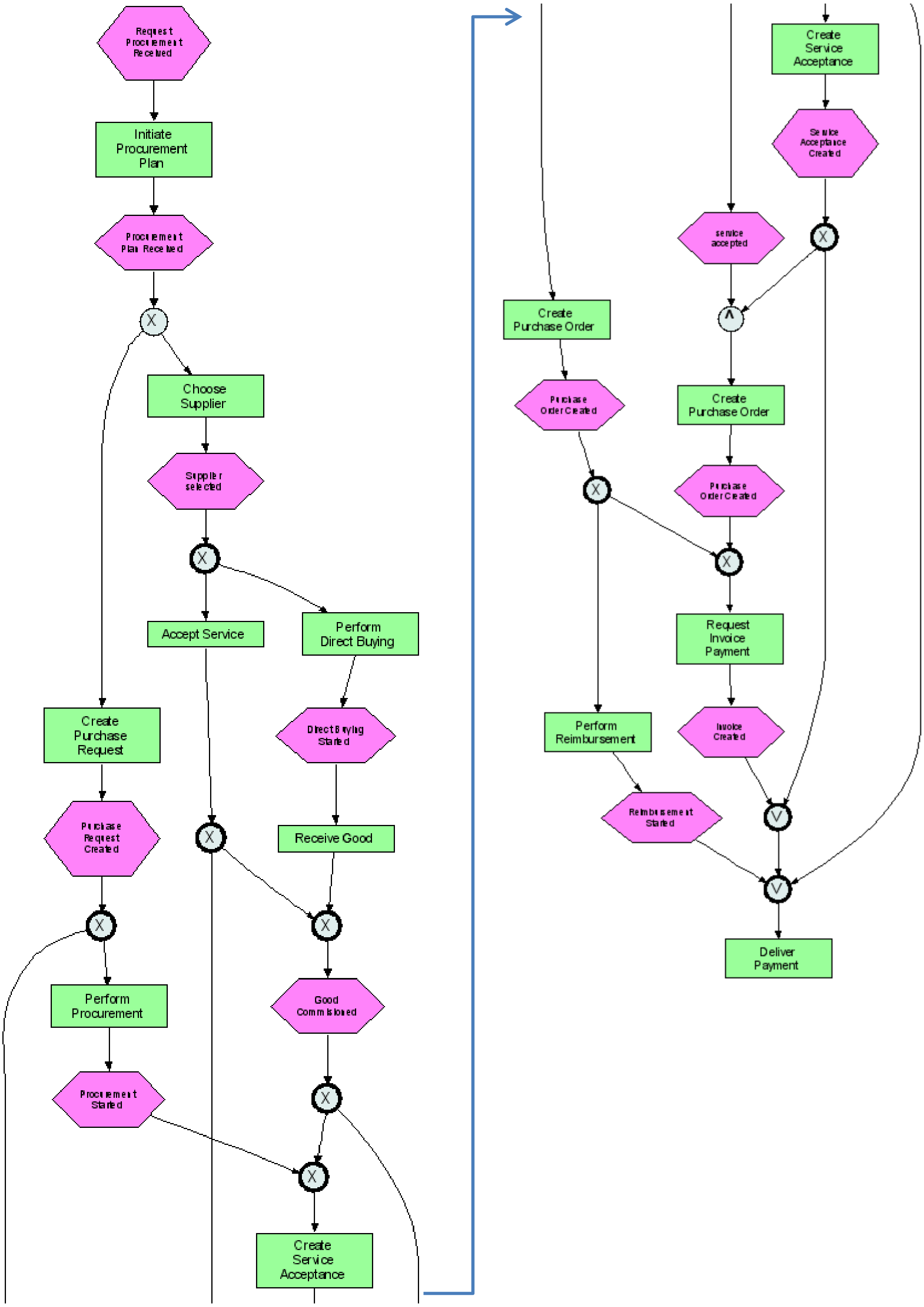
APPENDIX C CONFIGURABLE WORKFLOW



Appendix 10 Configurable workflow from W1,W2, and W3

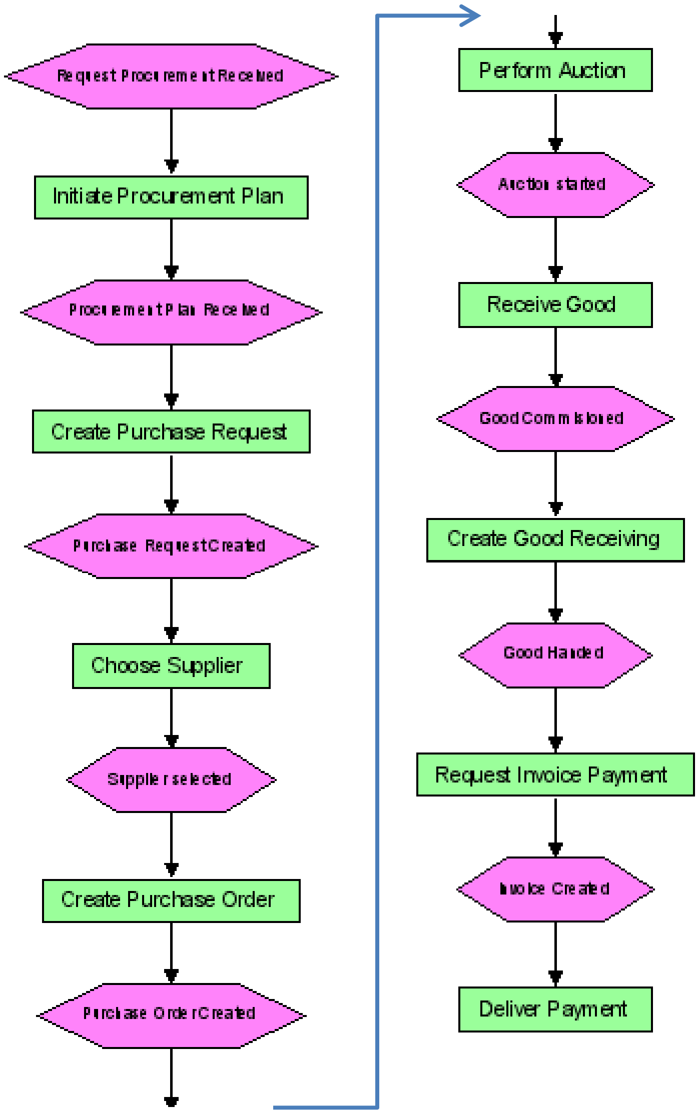


Appendix 11 Configurable workflow fromW6 and W7

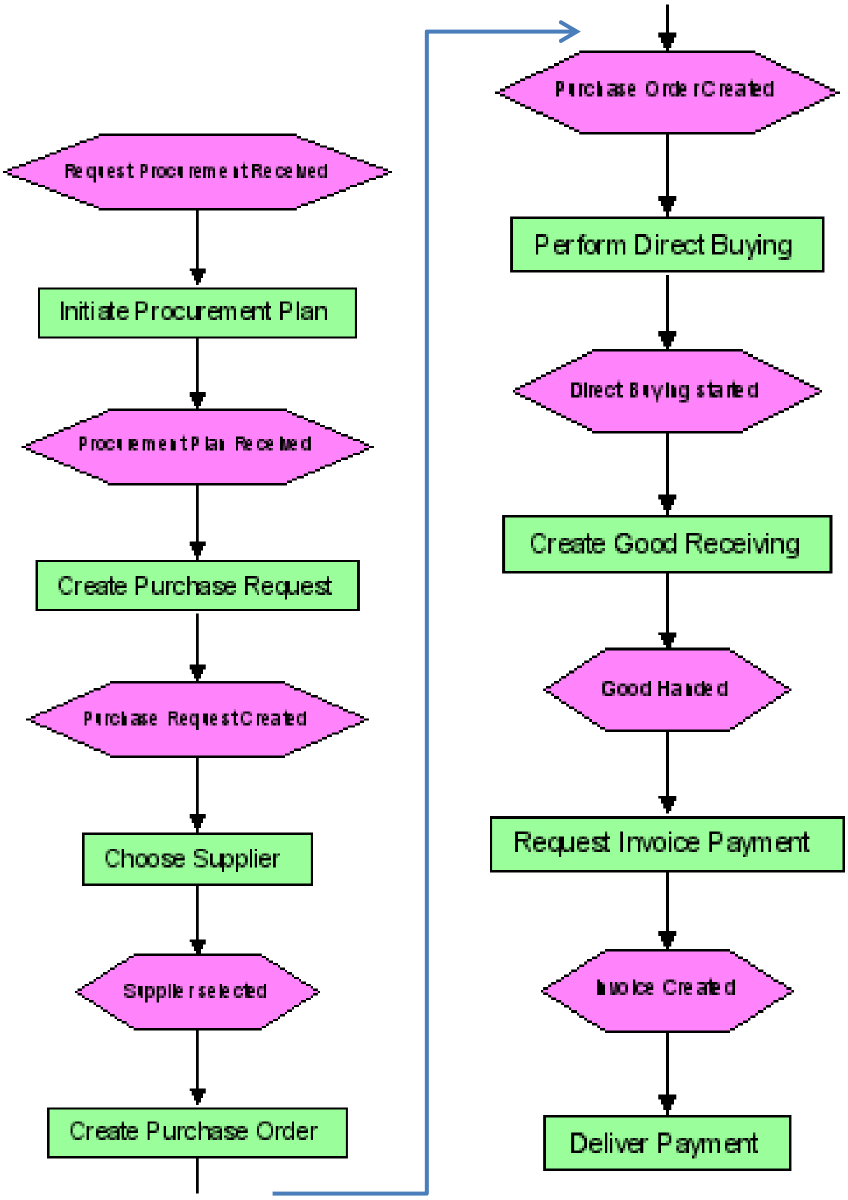


Appendix 12 Configurable workflow from W4,W5,W8

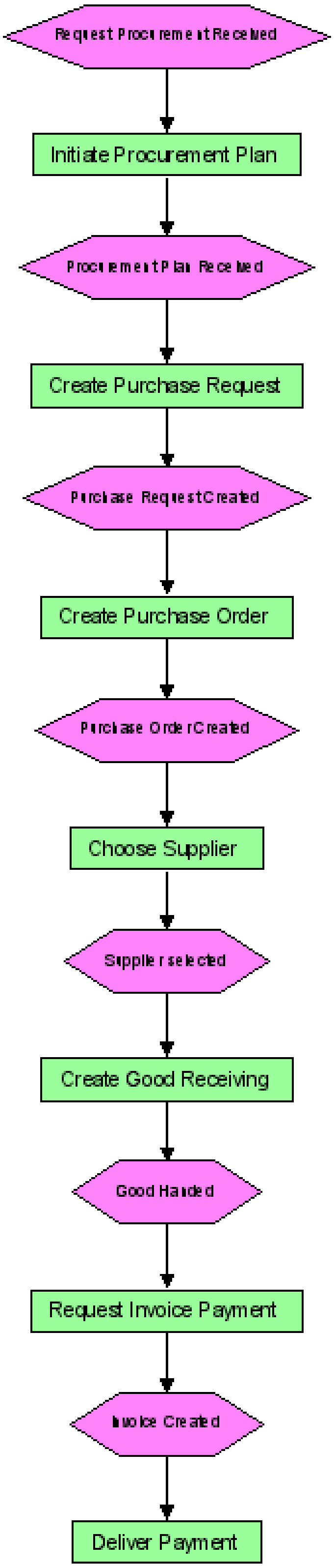
APPENDIX D WORKFLOW EPC MODEL



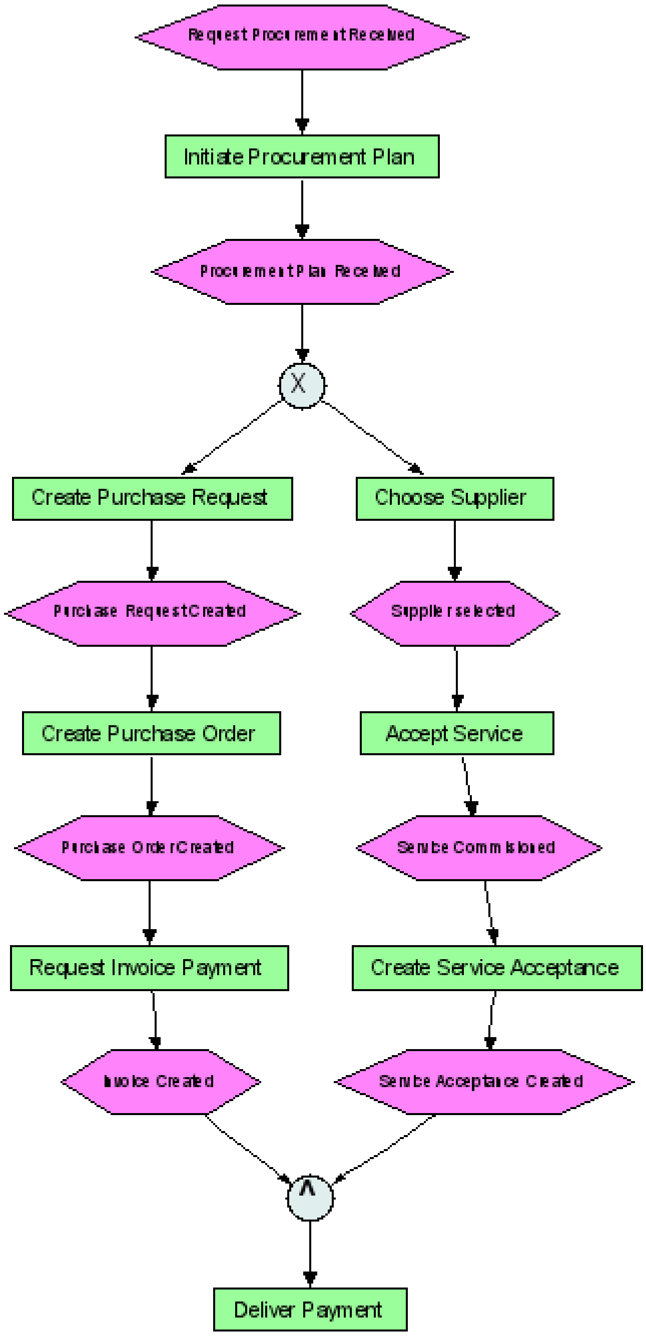
Appendix 13 Workflow 1 in EPC model



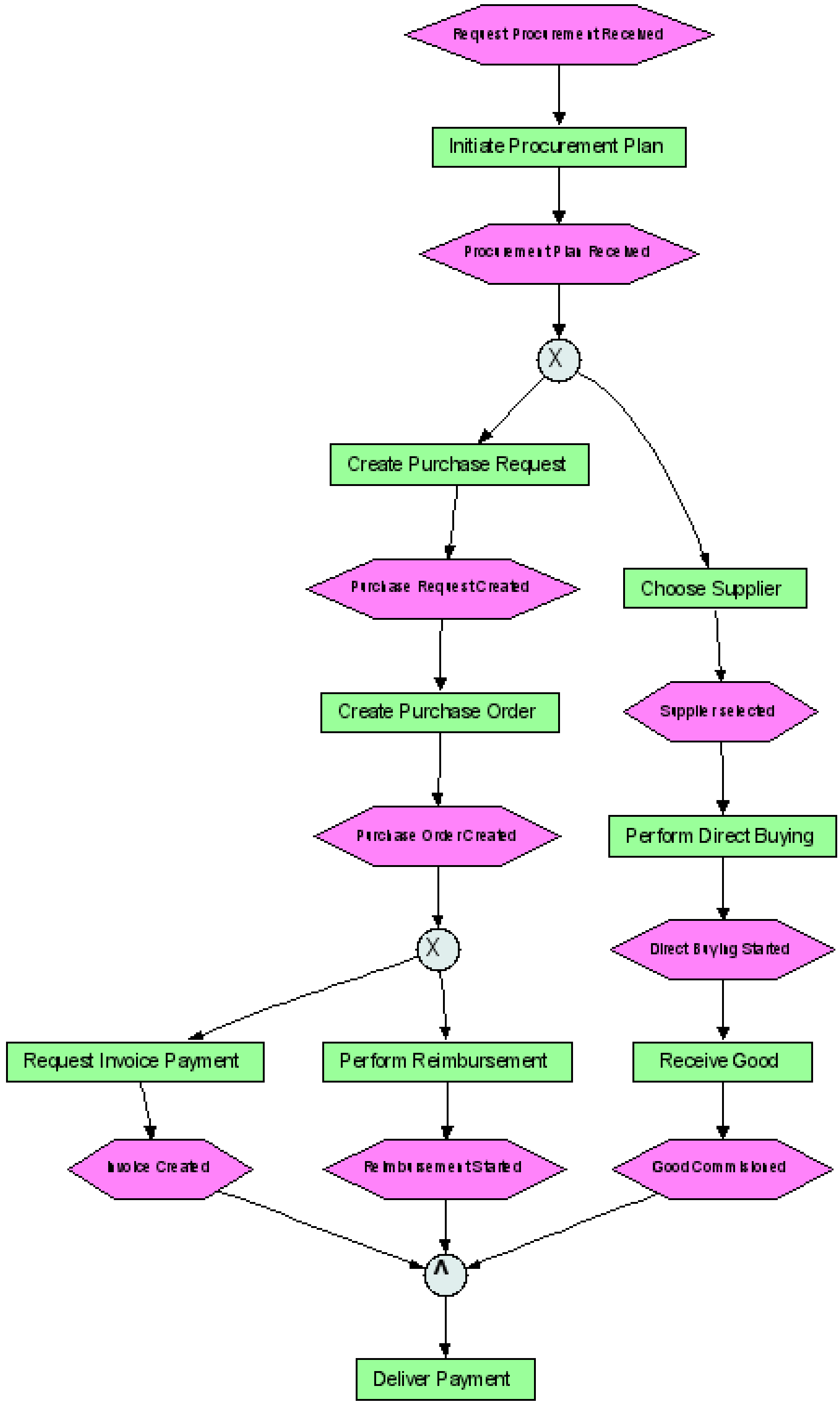
Appendix 14 Workflow 2 in EPC model



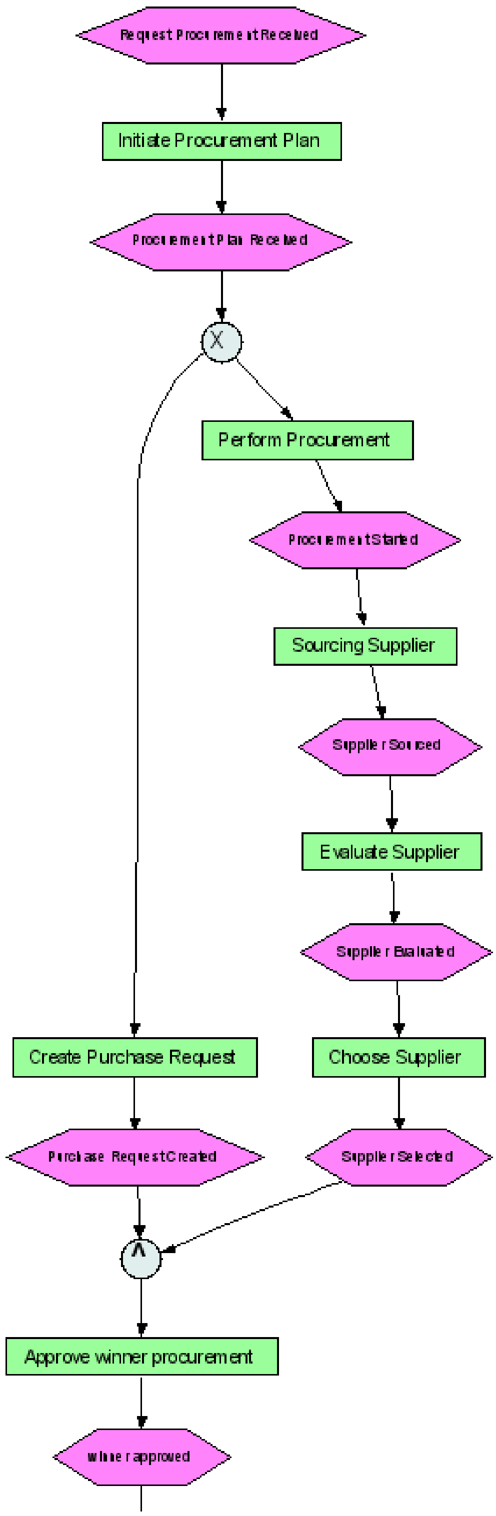
Appendix 15 Workflow 3 in EPC

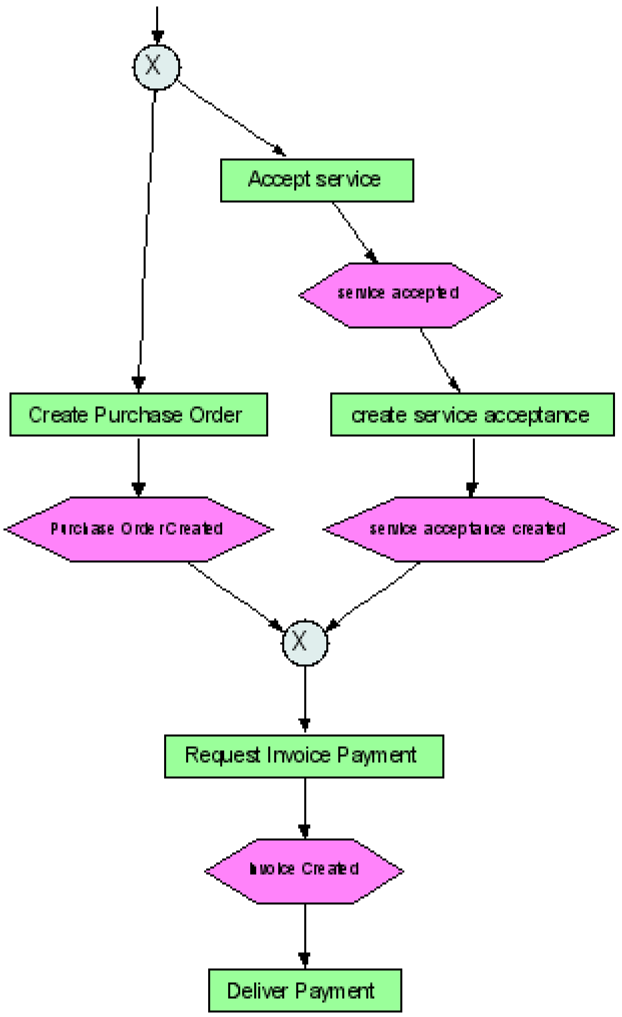


Appendix 16 Workflow 5 in EPC

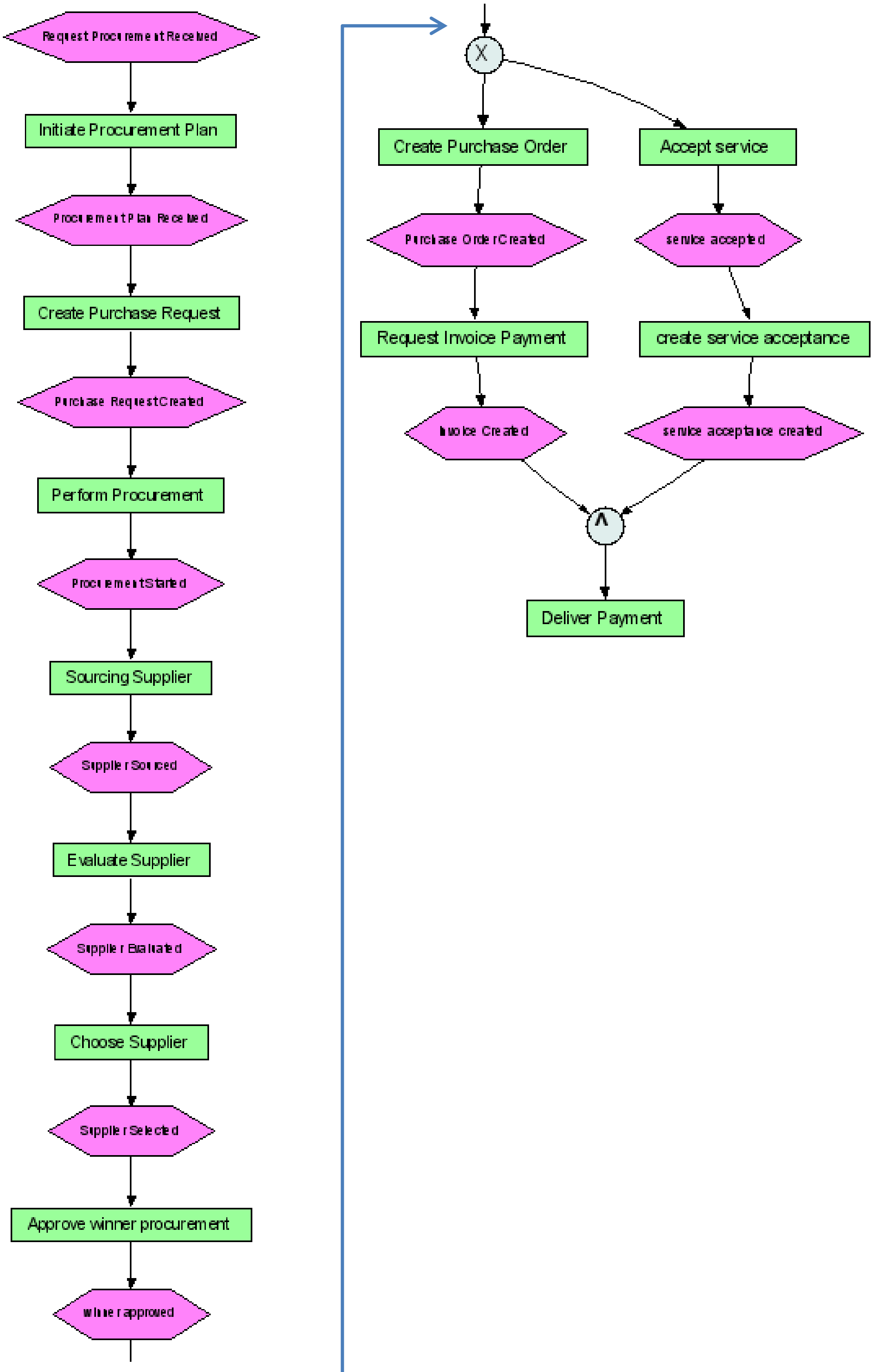


Appendix 17 Workflow 4 in EPC

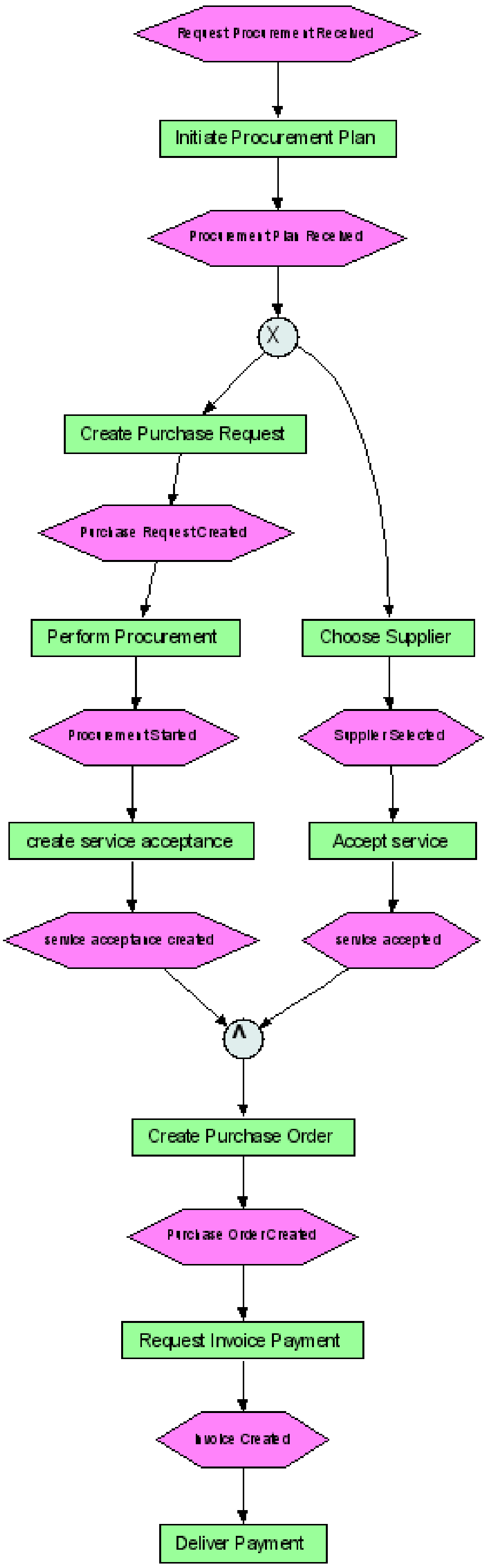

Initiate Procurement Plan
Perform Procurement
Sourcing Supplier
Evaluate Supplier
Create Purchase Request
Choose Supplier
Approve winner procurement



Appendix 18 Workflow 6 in EPC model



Appendix 19 Workflow 7 in EPC model



Appendix 20 Workflow 8 in EPC model

BIODATA

Rigga Widar Atmagi. He was born in March 2, 1990 and was originally from Surabaya, Indonesia. He likes music, travelling, and charity giving. Adaptable, eager to learn something new, hard-worker, and open-minded were his strengths.

He finished bachelor degree in Informatics from Institut Teknologi Sepuluh Nopember in 2012. He receives Fast-Track Scholarship for his master degree in Informatics from Institut Teknologi Sepuluh Nopember finished in 2013. He conducts his research devoted to Workflow Repository for Managing Workflow behind ERP Complexities. In addition to that, his research interests embrace Workflow Management and Workflow Retrieval. He wants to dedicate his expertise for Indonesia.

In addition, he was a professional system developer. He runs his own start-up company (i.e., rapid Develop), focusing on web application design and development. He frequently develops web-based application, information system, as well as, profiling web site for various organizations. He believes that the best of people were those that bring the most benefit to the rest of humanity. He could be contacted through his email (rigga.atmagi@gmail.com).