

# LEGAL ONTOLOGY DRIVEN LEGAL SERVICE PROCESS MANAGEMENT



By

RAZIK KARIAPPER AHMADH RIFAI KARIAPPER

Thesis submitted to the University of Sri Jayewardenepura for the  
award of the Degree of Master of Philosophy in Computer  
Science on 2015

## DEDICATION

இதற்கு,

மாதிரி காரணங்களை மறக்க வேண்டும் !!!

அர்ப்பணிப்பு,

எனது அன்பான உம்மாவுக்கும் வாப்பாவுக்கும் !!!

Dedication,

To

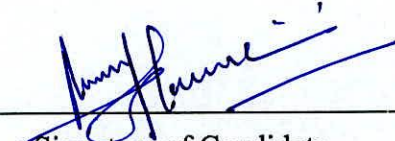
**My beloved Mum and Dad !!!**

### DECLARATION BY THE CANDIDATE

The work described in this thesis was carried out by me under the supervision of Dr. Prasad M Jayaweera and a report on this has not been submitted in whole or in part to any university or any other institution for another Degree/Diploma.

17.09.2015

Date

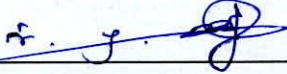


Signature of Candidate

**DECLARATION BY THE SUPERVISOR**

I/We certify that the above statement made by the candidate is true and that this thesis is suitable for submission to the University for the purpose of evaluation.

17/09/2015  
Date

  
Signature of Supervisor

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature of Supervisor

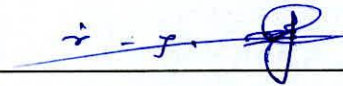
\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature of Supervisor

**RECOMMENDATION BY THE SUPERVISOR**

I/We certify that the candidate have made all the required corrections, additions and amendments to the thesis taking into account the comments and suggestions made by examiners.

17/09/2015  
Date

  
Signature of Supervisor

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature of Supervisor

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature of Supervisor

## TABLE OF CONTENT

TABLE OF CONTENT .....	i
LIST OF FIGURES .....	v
LIST OF TABLES .....	vii
LIST OF ABBREVIATION .....	viii
ACKNOWLEDGEMENT .....	x
ABSTRACT .....	xii
1. INTRODUCTION .....	01
1.1. General Background .....	01
1.2. Research Domain.....	02
1.3. Problem statement and Research Questions .....	03
1.4. Research Goal.....	04
1.5. Research purpose .....	05
1.6. Research Methodology .....	06
1.6.1. Research approach .....	06
1.6.2. Research procedure .....	08
1.6.2.1. Design an artifact .....	08
1.6.2.2. Problem relevance .....	08
1.6.2.3. Design evaluation.....	09
1.6.2.4. Research contribution.....	09
1.6.2.5. Research rigor .....	09
1.7. Disposition.....	10
1.8. Dissemination of Research Contribution.....	11
2. LITERATURE AND RELATED WORK.....	12

2.1. The ontology.....	12
2.2. Why and where ontology is being used.....	13
2.3. Classification of ontologies .....	15
2.4. Categories of ontology.....	18
2.5. Some selected popular ontology framework models in industries .....	19
2.5.1. Tove ontology .....	19
2.5.2. UMM (Unified Modeling Methodology .....	20
2.5.3. e3 value .....	22
2.5.4. REA ontology model .....	23
2.6. A short survey of selected popular legal ontologies.....	24
2.6.1. McCarty's LLD.....	25
2.6.2. Stamper's NOR.....	25
2.6.3. Valente's LFU.....	26
2.6.4. Van Kralingen's and Visser's FBO.....	27
2.7. Existing DEMO methodology .....	28
2.8. Enterprise ontology.....	29
2.9. Notion of DEMO ontological model & enterprise engineering .....	31
2.10. The system development process .....	33
2.11. The DEMO methodology .....	34
2.11.1. Modeling theory .....	34
2.11.2. Operation axiom.....	34
2.11.3. Transaction axiom.....	36
2.12. Development of DEMO aspect models .....	38
2.12.1. Construction model.....	39

2.12.2. Process model .....	39
2.12.3. State model.....	40
2.12.4. Action model.....	40
2.13. DEMO extended .....	40
2.14. Modification needed .....	41
3. LSPM FRAMEWORK .....	44
3.1. Overall Approach of LSPM.....	44
3.2. Development of LSPM.....	45
3.3. CAP framework.....	46
3.4. Aspect models.....	46
3.5. Analysis and synthesis.....	47
3.5.1. Performa-informa-forma analysis .....	48
3.5.2. Coordination-actors-production analysis .....	50
3.5.3. Transaction pattern synthesis .....	51
3.5.4. Result structure analysis.....	53
3.5.5. Construction synthesis .....	54
3.5.6. Organization synthesis .....	55
3.6. Aspect models.....	55
3.6.1. Interaction model .....	55
3.6.2. Process model .....	59
3.6.3. Action model.....	69
3.6.4. State model.....	75
3.6.5. Data model .....	77
3.6.6. Interstriction model .....	80



4. THE SUPPORTIVE ENHANCEMENTS .....	84
4.1. Decision point at the standard transaction model .....	84
4.2. Recognizing the duplicated actor role .....	85
4.3. Direction of the information access link.....	86
4.4. Data access level toward data model .....	86
4.5. Introduction of DEMO-E transaction meta-model .....	87
5. DISCUSSION .....	89
5.1. General Discussion .....	89
5.2. Application and Evaluation .....	95
6. CONCLUSION AND RECOMMENDATION .....	97
6.1. General conclusion .....	97
6.2. Direction for future research and recommendation .....	99
6.2.1. Concept model for the life of the case .....	99
6.2.2. Automate the system.....	99
6.2.3. Structure re-engineering.....	99
REFERENCE.....	101
ANNEXURE.....	109

## LIST OF FIGURES

Figure 1 Basic classification of an ontology .....	16
Figure 2 REA basic model .....	23
Figure 3 The system design process.....	32
Figure 4 Ontology system construction .....	33
Figure 5 The system development process .....	34
Figure 6 Graphical representation of the operation axiom.....	35
Figure 7 Representation of a coordination act .....	36
Figure 8 Basic transaction pattern.....	36
Figure 9 Basic pattern of a transaction.....	37
Figure 10 Standard transaction pattern .....	38
Figure 11 The ontological aspect models .....	39
Figure 12 LSPM approach .....	44
Figure 13 The ontological aspect model of DEMO-E .....	47
Figure 14 Result structure chart of case filing .....	54
Figure 15 Global ATD of case filing at District court .....	56
Figure 16 Detailed ATD of District court (case filing).....	58
Figure 17 Standard transaction pattern of DEMO-E.....	61
Figure 18 PSD I of District court (case filing).....	62
Figure 19 PSD II of District court (case filing) .....	63
Figure 20 Standard transaction diagram for tendering.....	65
Figure 21 Standard transaction diagram for payment.....	65
Figure 22 Standard transaction diagram for binding.....	66
Figure 23 Standard transaction diagram for decision .....	66

Figure 24 OFD of District Courts (case filing) .....	76
Figure 25 Data Transaction Meta-Model.....	77
Figure 26 Coordination level data access .....	78
Figure 27 Data Model of District Courts (case filing) .....	79
Figure 28 ABD of District Courts (case filing).....	81
Figure 29 OCD for District Courts (case filing) .....	82
Figure 30 Data access model.....	87
Figure 31 DEMO-E transaction meta-model .....	87

## LIST OF TABLE

Table 1 Transaction Result Table .....	52
Table 2 Transaction results and types .....	52
Table 3 Combined Transaction and Result Table .....	53
Table 4 Transaction and actor roles .....	54
Table 5 TRT of case filing .....	56
Table 6 Information use table of District Courts (case filing) .....	69
Table 7 OPL of District Courts (case filing) .....	76
Table 8 BCT of District Courts (case filing) .....	80

## LIST OF ABBREVIATION

ABD	Actor bank diagram
AM	Action model
ATD	Actor transaction diagram
C-act	Coordination act
C-fact	Coordination fact
CM	Construction model
DEMO	Dynamic Essential Modeling of Organization
DEMO	Design and Engineering Methodology for Organization
DEMO-E	DEMO extended
DT	Data Transfer model or Data access model
IAM	Interaction model
ICT	Information Communication Technology
ISM	Insterstriction model
IUT	Information use table
LSPM	Legal Service Process Management
OFD	Object fact diagram
OPL	Object property list
ORM	Object role modeling
P-act	Production act
P-fact	Production fact
PM	Process model
PSD	Process structure diagram
RUT	Result used table

SM	State model
WOSL	World ontology specification language

## ACKNOWLEDGEMENTS

I would like to take this opportunity to offer my hearty thanks and gratitude to my academic supervisors, Dr. Prasad M Jayaweera Senior Lecturer, University of Sri Jayawardenepura and Professor Hans Weigand, Tilburg University, Netherlands whose supervision and guidance are important contributing factors to this research work success. Their continuous motivation, deep interest in this research and encouragement helped me to achieve breakthroughs and to overcome many of the problems in this research work. Their meticulous reviews of report and suggestions have been invaluable.

I extend my sincere thanks to Sabaragamuwa University of Sri Lanka, which has been supporting me in many ways such as providing paid leave and necessary information with positive communication along with administrative and nonacademic staff members.

I would like to share my sincere thanks to HETC project and the crews of the project for supporting me in financial means, especially my heartfelt thanks to Prof. LL Rathnayaka, Prof. Munasinghe, Dr. Munasinghe, Mr. Nawarattna and Ms. Yohani for their valueless services.

I would also wish to extend my gratitude to Tilburg University for enrolling me as a visiting researcher for a certain period of time. I will be failing my duty if I miss to thank Ms. Sandra, the HRM executive at Tilburg University for her wonderful guidance and kind assistance towards me. Meanwhile, I like to take this moment to share my

credits with department of economic for providing well equipped office room and the facilities along with essential guidance of Ms. Alice, secretary to the department.

I would like to convey my sincere thanks to Vice Chancellor of Sabaragamuwa University of Sri Lanka Prof Chandana P Udawatta along with Registrar Mr. MF. Hibathul Careem for providing handfull encouragement and also to convey my sincere thanks to Prof. Mahinda S Rupasinghe for the supports given towards my studies during the period he served as Vice Chancellor to the Sabaragamuwa University of Sri Lanka.

Further I extend my gratitude to my beloved parents Razik Kariapper and Hajara Razik Kariapper who were passed away during my research work, my sister Dr. Shaikila Issadeen, and my brothers Mr. RM. Nowshad Kariapper, Mr. RM. Azmie Kariapper and my brother in law Dr. A Izzadeen.

Next, I would like to extend my sincere thanks to Mr. MSK. Sharik Kariapper, Mrs. Sulaiha Kalantharlebbe, Mr. MM.Alikhan, Mrs. KL. Siyana Alikhan, Mr.KLM. Rameez, Dr. KLM.Rayeez, Eng. KLM.Razeen and Rev. Mou.KLM.Ziyanudeen.

At last but not at least, my wife Mrs. Silmiya Rifai Kariapper and my kitties Athaf Kariapper, Kashif Kariapper and Imadh Kariapper are also acknowledged and remembered within as well as outside the context of this work for their love and encouragements that made me to successfully complete the Master of Philosophy.



# **LEGAL ONTOLOGY DRIVEN LEGAL SERVICE PROCESS**

## **MANAGEMENT**

**RK AHMADH RIFAI KARIAPPER**

### **ABSTRACT**

Process management and optimization play vital role in almost all industries regardless of different domains. The process managements involves with designing the process and applying the engineering principles on it in order to reduction of complexities in the selected system in the domain interested. The process optimization involves with reduction of time in between processes and possibility to automate the processes. The legal domain is one of the complex domains in Sri Lanka since more number of collaborative actors; interoperable issues within the system; complex legislation laws and rules; and difficult Courts procedures. This research project was initiated and completed in order to develop an ontology framework model for the service process management at the District Courts in the legal sector. The ontology framework model was developed with five aspect models namely action model, process model, object model, data model and construction model. These all five aspect models deal with five different views and construction of the ontology framework model. The development model is named as Design and Engineering Modeling of Organization – Extended (DEMO-E). The backbone of the DEMO-E is the “transaction” (which is the complete cycle of one or more coordination acts and a production act to produce an either material or immaterial output as the result). DEMO-E is a theory to do a construction and operation of any enterprises. The intended result is provided through the standard pattern of the transaction (it is series of communication act in order to achieve an output which cause an effect on social world) via series of communication acts. The legal

domain is rich in more communicational agenda and more number of participants in a single case. A transaction meta-model was developed in order to apply on the mile stone events and get the relevant model for the transaction in both within the selected case filing area and other stages of the case in legal sub domains. A case must pass around four basic stages such as filing, hearing, case moving or transferring when necessary and verdict. The case filing major transaction it has many sub-transactions namely case tender, necessary payments, binding and decision of the judge collectively implies as case filing. The said each sub transactions obey the generic transaction stages including both coordination acts such as request, promise, state and accept and production act such as execution act. In fact all five different ontology aspect models map the conceptual design of legal process management with easy understanding.

Further, DEMO-E can be successfully applied to the legal domain due to optimal constructional pattern and the optimal operational acts. More than this, the DEMO provides re-engineering and re-designing options to the designers. Thus the judicial Courts procedure and the organization can be restructured for the optimal output and it leads to break the barrier of complexity and inter-operable issues

## **CHAPTER ONE**

### **INTRODUCTION**

This chapter is to introduce the background, motivation of the work presented in this thesis and an overview of the research methodology which was used during my research work. Also, the chapter contains a briefing of the theoretical basis for the proposed approach and lists working relation to the approach that has been published in different forums.

#### **1.1. General Background**

Information and Communication Technology (ICT) is used everywhere regardless the domain and the platform. It supports the activities such; creation, storage manipulation, identification and communication of information together with related method. In other way the ICT enables us to record, store, process, retrieve, and transmit information. As the result, the communication between two or more end users or stakeholders are become well-fixed with accuracy, efficiency and secured. Meanwhile, the application of ICT is spanning from general data entry level to high tech intelligent system. Either the data or information is playing vital role between two systems in order to transfer the information to react with the input for long term sustainability.

Sri Lankan judicial system has rich heritage from different laws from different jurisdictions countries and legal systems who ruled Sri Lanka earlier nearly five hundred years. The rulers enforced their own laws to Sri Lankan legal system. It resulted to have more complex and mix of laws on us. With the effect of this complex, the judicial system is one of the most complex and complicated systems in Sri Lanka.

The judicial system is complicated since, no proper document maintenance; huge number of pending cases, one case may depend on other case (s), more number of collaborative actors, complex ethnic specific rules, hard legislation rules and complex procedures of the Courts proceedings. In general it takes lengthy time in between filing a case and getting the verdict for it. It is necessary to minimize the time period in between case filing and the verdict in order to reduce the unwanted complication of the case and the wastage of the time of the stakeholders (viz. plaintiff, defender, lawyer, witnesses and court officers) who ever involve in the case.

Researchers provide different solutions to minimize the time period in between two different stages, such as activity automation, combining the activities, reduction of the unnecessary activities, ignoring some activities, preparing user friendly rules and flexibility in each activity.

## **1.2. Research domain**

I would like to summarize here the domain of interest and the scope of this research. This research is about relating ontology framework in the state sectors. My scope of interest is in developing an ontology framework for case filing at the District court in legal system in Sri Lanka. I have decided to take the Kalmunai District Courts in order to work with since; District Courts are the starting level of courts hierarchy of Sri Lankan legislation system, divorce case filing is one of the four case types which the District Courts accepts and less structured with complex procedures, Kalmunai District Courts location is rich with multiethnic community surroundings, and the easy access to the researcher from his home town (permanent residence). Furthermore Courts of

Appeal and High Courts are other options to file the divorce cases if the verdict of District Courts regarding the divorce case is not satisfied by the plaintiff.

### **1.3. Problem Statement and Research Questions**

The enterprise use different ontology models in order to show the virtual structure of an organization, inter relationships and dependencies. Indeed, it extends to maintain their flexibility, accuracy and managing workflow of an organization. It is identified the necessity of a well-structured method to reduce the complexity of the design and increase the understandability. Traceability is one of the other important factors to be considered since it supports and helps to either organize the work properly or re-organize the work and rectify error occurring locations of the design. The research and motivations on the enterprise ontology models can be found in Enterprise Ontology.

The research question is going to be:

*Unavailability of ontology framework to facilitate Legal Service Process Management (LSPM) in efficient and productive manner.*

The main question can be divided into following sub questions.

*Question 1: develop the LSPM framework by the aspect models.*

The aspect models are well structured models which can provide the good structure of an enterprise without making any gap in any streams of the domain interested.

*Question 2: development and enhancement of the aspect models develop the LSPM framework for the legal domain*

The way the ontology framework developed must show the efficient way to the legal processes happening in the District court system. Each meta-model or aspect model has its own goal and the way of working toward the ontology framework in order to support the LSPM.

*Question 3: application of these aspect models to the legal domain*

The selected legal domain is one of the complex systems in this country since the collective composition of the legislation procedures as mentioned earlier in this chapter. The concept and the goal are two important factors that are used to develop the ontology framework model. An ontology framework model is developed in the intension of sharing common understanding of the working system in a selected domain. The previous statement would be described in the chapter two as well.

#### **1.4. Research Goals**

The main goal of this research work presented in thesis can be formulated as follows. It is important to develop the goal from our main research question. According to that, the main goal of this research is *“to develop an ontology framework model to support service process management of the District Courts in Sri Lanka”*.

Even the main goal can be divided in two following sub goals.

*Sub goal 1: construct / enhance a method to develop ontology framework model to support service process management of the District Courts in Sri Lanka:*

The sub goal can be the answer to the question 1 and question 2. This method should be used to develop the meta-models in order to achieve the final ontology framework model.

*Sub goal 2: apply the aspect-models in order to develop the LSPM ontology framework:*

The aspect-models constructed using the particular methodology would be used to test the model to support service process management of the District Courts in Sri Lanka. This sub goal would be answer for the question 3.

### **1.5. Research Purpose**

The target users of the proposed method and model are information system developers, ontology programmers, stakeholders of the District Courts, higher rank administrators of the legal department and ministry of justice including policy makers and decision makers. The intention of the proposed ontology framework model is;

- To facilitate and making sure of smooth flow of District court activities.
- To optimize the work flow in the District court system
- To prepare supreme ontology model by using meta-models in order to apply the proposed ontology framework model, to be applied in different types of Courts in Sri Lanka.

## 1.6. Research Methodology

### 1.6.1. Research Approach

The research is the systematic and continues investigation in to the study based on a scientific method in order to discover new artifacts or theory. In the information system pool most of the researches support to improve the effectiveness, efficiency of an organization by application of information technology.

The Information System (IS) researches can be conducted by two important paradigms namely behavioral science and design science(von Alan et al., 2004). In brief the design science creates and evaluates IT artifacts with the goal of solving organizational problem. The design science includes more and more quantitative evaluations such as mathematical proofs, analytical simulations and quantitative comparisons.

And the behavioral science Behavioral science is defined as the rich phenomena that emerge from the interaction of people, organizations, and technology may need to be qualitatively assessed to yield an understanding of the phenomena adequate for theory development or problem solving(Klein and Myers, 1999).The design science approach is focused mainly with following components.

**Awareness of a problem:** Understanding and awareness of a problem may reach us in different ways including dynamic development in industries or in an acknowledged subject. Referring an allied discipline also makes the opportunity for applying the new findings to the research worker's field. The result of the phase is a proposal either formal or informal for a new dimension of research effort.



**Suggestion:** This phase is fairly important and located after the proposal phase. The importance of this phase measured with creativity wherein new functions are suggested based on a novel creation of either existing or combination of new and existing elements. This phase assure that ideas should be remaining non-repeatability. The creative step is important in all researches leads to the suitable constructs towards the research scenario along with an appropriate research design.

**Development:** The chosen tentative design is recommended to be enhanced further and implemented to get the more or less correct result. The chosen method, approach, acts, algorithm, software tools used, and the technique of implementation may different from project to project.

**Evolution:** With the appearance of the artifacts the testing stages starts. The artifacts are tested with different test cases and criteria. The results should be recorded accordingly. Meanwhile the deviations also stated and explained with justification.

**Conclusion:** This is the phase where the final result of the research would be opened to the result space. So in a way it can be said that, this phase is the conclusion of the research aimed.

## **1.6.2. Research Procedure**

In this section I would like to describe more about the design science procedure since it was used in this research. The design science research process is done by certain guidelines in order to enhance and validate the research.

### **1.6.2.1. Guideline One: (Design an artifact)**

Design science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation (von Alan et al., 2004) according to the guideline one, we can propose following artifacts to achieve our goals.

1. A method to design meta models
2. An enhanced method to develop the ontology framework model

### **1.6.2.2. Guideline two: (problem relevance)**

The design-science research focuses to develop technology-based solutions and relevant business problems (von Alan et al., 2004). In our case the ontology framework will free the complexity of the design complex and workflow complications at the District Courts (case filing) since the construction of the artifacts aimed to change the phenomena. The thesis especially focuses the new method to be applied in order to model an error free ontology framework in the judicial system of Sri Lankan District Courts especially.

### **1.6.2.3. Guideline three: (Design evolution)**

It is important to have good evaluation since this component gets the position as crucial component(von Alan et al., 2004). The quality of the thesis proposed model would be tested using functionality, usability, flexibility, simplicity, completeness, accuracy, reliability and efficiency. In our research we use the scenarios as qualitative method.

Scenario 01: tender the case

Scenario 02: payments

Scenario 03: binding

Scenario 04: Decision making

### **1.6.2.4. Guideline four: (research contribution)**

The effect design science research should end up with the valuable research artifacts either new from the scratch level or the enhancements of the existing artifacts. The first goal of the research is to construct a method to develop ontology framework model to support service process management of the District Courts in Sri Lanka. This result is taken by publication 1 and publication 2.

Our second goal is to apply the meta-models in order to develop the ontology framework is satisfied with our 3<sup>rd</sup> and 4<sup>th</sup> publication.

### **1.6.2.5. Guideline five: (research rigor)**

In design science research, rigorous methods should be employed during development and evaluation of the artifact(von Alan et al., 2004).

## **1.7. Disposition**

Chapter two discusses the overview of the related literature required for the work. A brief overview about ontologies, existing ontologies, the existing DEMO method along with some aspect models are described under the chapter two.

The chapter three discusses about the modeling methodology of the ontology framework model for the case filing at the District Courts. The methodology is called DEMO-E (DEMO-extended). This chapter includes many different aspects of the research such as introduction of the data model is key conceptual design apart from other four aspect models such as action model, process model, state model and construction model. Further, introduction of couple of meta-models are discussed in depth level.

The enhancement made to the DEMO is discussed under the chapter four. Further the description is extended to five important enhancements and the advantages of the enhancements.

The chapter five describes the discussion made of this research. All advancements, advantages, working effort, and major challenges faced during the cause of the research.

At last the chapter six is set up and elaborated to describe the overall conclusion and future recommended work directions of this research. The necessary annexure is following the chapter six.

### **1.8. Dissemination of Research Contribution**

The different parts of the research results from work related to the thesis have been accepted published in several forms. These publications are listed below.

1. KARIAPPER, R. A. R. & JAYAWEERA, P. M. A Comparison of Selected Legal Ontologies for the Design of Legal Process Optimization Ontology. *International Journal of Scientific Researches*, 3, 862-865.
2. KARIAPPER, R. K. A. R. & JAYAWEERA, P. M. A Survey on Legal Ontologies for the Development of Process Optimization Ontology. International Conference on Multidisciplinary Approaches, 2014 Colombo, Sri Lanka. University of Srijayawardenepura, 121
3. KARIAPPER, R. K. A. R. & JAYAWEERA, P. M. Process Marting Approach for Legal Record Archiving. International Research Conference on Humanities and Social Sciences, 2013. University of Sri Jayewardenepura, Sri Lanka, 18
4. KARIAPPER, R. & JAYAWEERA, P. M. Dynamic Essential Modeling of Organization (DEMO) Towards the Legal Domain. Second International Conference on Interdisciplinary Legal Studies 2015, Available at SSRN 2607921, 2015. Unique Conferences Canada - 2015, 40-45

## CHAPTER TWO

### LITERATURE AND RELATED WORK

It is very important that one should understand the under-lying fundamentals of the problem domain in order to deliver a better solution for the existing and potential problems. In this topic the survey is extended by examining the different ontology models, ontology principles, advantages of the ontology models, how the ontology models have been used in different domains and the variations of the models.

#### 2.1. The Ontology

There are many definitions proposed for the word of “ontology” depends on the subject and the authors of the scientific papers. We would like to state most relevant definitions of the ontologies from the various literature available from the different domains including more or less closer to the ontologies in this thesis. Ontologies are conceptual models of a specific domain(Mommers, 2010). Gruber defines the ontology in a more specific way, “specification of a conceptualization and a more specifically, as a description of concepts and relations that exist for an individual or community of individual” (Gruber, 1993). Another definition of an ontology is “a shared understanding of some domain of interest”(Uschold and Gruninger, 1996). Ontology is stated as metal level description of the model under construction (van Heijst, 1995). In abstract, ontology is a model of reality of the world and the concepts in the ontology must reflect this reality. Ontology is broadly used for sharing common understanding of the structure of information among people or software agents(Gruber, 1993), enabling reuse of domain knowledge(Noy and McGuinness, 2001), making domain assumption explicitly or in other word domain theory development, separating domain knowledge

from the operational knowledge, analyze the domain knowledge (McGuinness and Wright, 1998), system development and system documentation furthermore, the ontology is playing vital role on different domains (viz. medical, engineering, mathematical, IT, financial, and social).

## **2.2. Why and Where Ontology is being used**

The ontology design is applied in many popular areas (Artificial Neural Network, Artificial Intelligence, Medicine, Engineering, Computing and etc...) since the goal is to produce spectacular goal oriented solutions. These ontology models and the frame works would be applied for following reasons; to share common understanding of the structure of information among people or software agents, to enable reuse of domain knowledge, to make domain assumption explicit, to separate domain knowledge from the operational knowledge, and to analyze the domain knowledge.

Sharing the common understanding of the structure of information between people or software agents is one of the general goals in developing ontologies (Tu et al., 1995, Gruber, 1993). The above stated goal is understood by the following scenario. Suppose many different web stores or sites may comprise with either medical information or supply medical service through internet. If the above stated medical stores or web sites share and publish the same inherent ontology they all apply, leads other agents can take and combine information from these different stores or sites. The computer agents can utilize this combined information to response user queries or make the data as input to other applications (Gruber, 1993).

Enabling reuse of domain knowledge widely accepted in the ontology researches and the ontology application development. The ontology is developed after a vast area of any domain studied with the adequate number of models gathered in different domains in any particular perspective or problem stated. The developed ontology may be used in different domains too. This simply stated as the ontology can be reused. Moreover, it is possible to combine several existing ontologies in order to work with the huge domain interested when it is possible. Meanwhile, it is also possible to reuse a general ontology, such as the United Nation Standard Product and Service Code (UNSPSC<sup>1</sup>), and extend it to describe the domain of interest (Gruber, 1993).

Implementation of assumption that we decided in a domain can be developed with the use of domain knowledge. Meanwhile, it is possible to re-implement with the change of domain knowledge or accepting new materials to the domain as regular changes. A programming language is another assumption in a domain which is called as hard-coding assumption, which is very difficult to understand and the changes made in the codes are hardly accepted from the base level users. Moreover, precisely stated specs of domain knowledge are very helpful for fresh end users who should understand the perspective of the domain interested. (Gruber, 1993, Noy and McGuinness, 2001).

Extracting the domain knowledge from the operational knowledge is one of the other advantages of any ontology. It can be described that, a task of setting up a product from its components with the respect of mandatory requirements and develop a program or an algorithm which guides to setting up the above should be independent of the products

---

<sup>1</sup> United Nation Standard Product and Service Code is taxonomy of products and services for use in e-commerce.



and components themselves (McGuinness and Wright, 1998). Therefore ontology can be developed of PC-components and use the algorithm to setup the machines. Even, there is a possibility to apply the same algorithm to be applied in different situation regardless of the domain (Noy and McGuinness, 2001).

One of the usages of the ontology is analyzing domain knowledge with the available specs. Formal analysis is much suitable for this purpose (McGuinness and Wright, 1998).

### **2.3. Classification of ontologies**

Ontologies can be classified according to the various views and the application domains (Guarino, 1997). The above stated is classified the ontology according to the two dimensions: their level of detail and their level of dependence on a particular task or point of view. On the first dimension he describes about the detailed ontology which refers the meaning vocabulary. Meanwhile, a simple ontology can be developed with underlying common concept with the objective of sharing the resources such as either online or offline applications (dictionaries). In the second dimension, the ontologies classified with the level of dependence and which is shown below in the figure 1. Meanwhile there is another concept of reference ontology (Andersson et al., 2006) is to derive most of the concepts are taken from one of the original ontologies, and additional necessary concepts are added in order to enhance it.

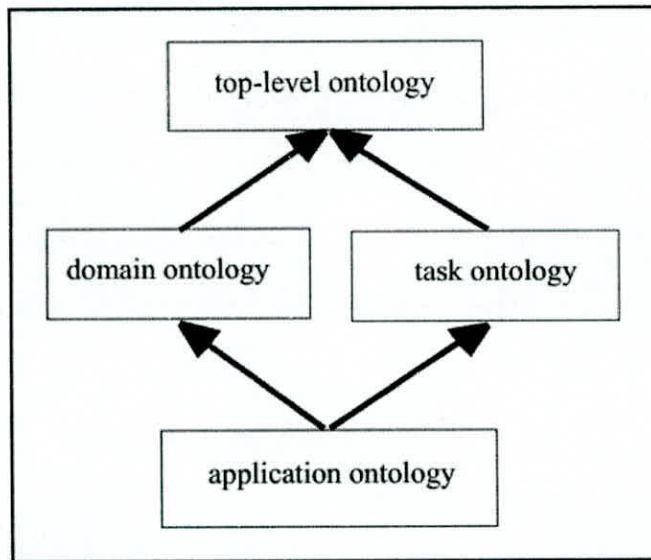


Figure 1 basic classification of ontology (Noy and McGuinness, 2001)

### ***Top-level ontologies***

Top-level ontologies describe very general concepts like space, time, matter, object, event, action, etc., which are independent of a particular problem or domain: it seems therefore reasonable, at least in theory, to have unified top-level ontologies for large communities of users(Guarino, 1997, Mascardi et al., 2007).

### ***Domain ontology***

Domain ontologies and task ontologies describe, respectively, the vocabulary related to a generic domain (like medicine, or automobiles) or a generic task or activity (like diagnosing or selling), by specializing the terms introduced in the top-level ontology(Guarino, 1997, Kaiya and Saeki, 2006, Velardi et al., 2001).

### ***Application Ontology***

Application ontologies describe concepts depending both on a particular domain and task, which are often specializations of both the related ontologies. These concepts often

correspond to roles played by domain entities while performing a certain activity, like replaceable unit or spare component (Tu et al., 1995).

***Upper ontology:***

It is a model of the common objects which can be applied to the most of the domain ontologies available. It consist of a glossary which contains the terms and affiliated object descriptions which are used in different domain sets, concepts supporting development of ontology (Mascardi et al., 2007, Hoehndorf, 2010).

***Representation ontology or Meta ontology:***

It is a description of primitives such as concept; attributes; and relations used by the knowledge representation languages (Gruber, 1993, Guarino, 1997, Yudelson et al., 2005, Wikipedia, Van Inwagen, 1998).

***Process ontology:***

It is used to describe the structure of the processes in an enterprise (inputs, outputs, constraints, sequencing information, involved in business or engineering processes) (Braidotti, 2006, Guizzardi et al., 2008, Liao et al., 2003).

***Organizational ontology:***

This ontology provides a description of artifacts and, actors. (Fox et al., 1998, Filipowska et al., 2009).

***Gellish Ontology:***

It is a combination of domain ontology and upper ontology (Wikipedia, Van Renssen, 2003)) (Van Renssen, 2005).

***Domain specific ontology:***

It provides the additional information specific to an organization from given domain. A domain ontology (or domain-specific ontology) models a specific domain and it represents part of the world. The concept including rules and the meaning is very much domain specific and since that, different ontologies created with different languages, intended usage, and perception are more complex and pretty much incompatible to merge to make as a single ontology. I could notice in my literature works that the merging different ontologies as single as whole is still in crawling stage (Wikipedia, Kietz et al., 2000, Mukhopadhyay et al., 2007).

**2.4. Categories of Ontology**

The ontology can be categorized in to two according to the aspect of information technology as Generic Enterprise Ontology (GEO) and Deductive Enterprise Ontology (DEO)(Jayaweera, 2004).GEO is a collection of concepts, and its relationships of a selected enterprise. The DEO is its ability to automatically deduce answers to many “common sense” questions varies from general basic common sense questions to intelligence level communicational questions.

## **2.5. Selected Popular Ontology Framework Models in Industries**

In this section, some popular ontology frameworks which gain more popularity and prominent in the industry are discussed.

### **2.5.1. Tove Ontology**

It is about to present logical framework activities, states, and time in enterprise integration architecture. The tove deals with first-order logic and identifies the problems as occurrence of actions. And the framework focusses in structured supply chain management and enterprise engineering. Moreover it provides the path for applying of theories of action and time. (Gruninger and Fox, 1994).

The TOVE<sup>2</sup> project is to develop an ontological framework model for Enterprise Integration (EI) based on and suited for enterprise modeling (Fox et al., 1998). This project was initiated with the four main goals respectively; a) create a shared representation of the enterprise that each agent in the distributed enterprise can jointly understand and use b) define the meaning of each description c) implement the semantic in set of rules which enable TOVE to automatically derive the answer to many “common sense” questions towards the enterprise, and d) define a symbology for a concept in a graphical context. The above said four goals were reached by four different approaches indeed. The first goal was reached the development of reference model for the enterprise which provides data dictionary concepts which are common for range of enterprises. The second goal was reached with defining a generic level representation based on “terminological logic”. Third goal was reached by defining a set of axioms

---

<sup>2</sup>Toronto Virtual Enterprise project

which can provide more common sense answers, in all three different layers such as representation, generic and application as well. The fourth goal was reached by the defining of the symbols/ graphic representation of the terms. The extension of the TOVE project was TOVE enterprise model development by the approach of “ontology engineering”. The first generation of TOVE is with the approach of “extracting rules from experts” (Fox et al., 1998, Fox, 1998). The success and competence of the TOVE developed the model for computer manufacturer and an aerospace engineering firm with the knowledge engineering background along with computer integrated manufacturing (Fox et al., 1998, Fox, 1998).

### **2.5.2. The UMM (Unified Modeling Methodology)**

The UMM is a Unified Modeling Language (UML<sup>3</sup>) modeling approach to design the business services which collaborate the business partners and provides the business justification for the service to be implemented in a service-oriented architecture (SOA) developed by UN/CEFACT<sup>4</sup>.

The UMM can be understood by its information and documentations. The UMM is an incremental business processes and information model construction methodology which provides different levels of granularity. The UMM provides the conceptual framework to communicate common concepts such as; industry expert, business expert, business stakeholder, business domain, business domain, business process, business

---

<sup>3</sup> The UML is a general purpose modeling language, which is designed to provide standard way to visualize the design of the system.

<sup>4</sup> United Nations Center for Trade Facilitation and Electronic Business

collaboration, business process activity, business interaction activity, business information, business collaboration domain and business collaboration model with open edi-scenario (Ahlsén et al., 1994, Veijalainen, 1992).

A UMM business collaboration model comprises of Business Domain View (BDV) the partitioning of business domain into business areas, process areas, and business processes. This view establishes the business context of the process which is a precursor to evaluating the likelihood of finding reusable, previously defined, process descriptions or terminologies, the Business Requirements View (BRV) the view of a business process model that captures the business scenarios, inputs, outputs, constraints and boundaries for business processes and their interrelationships within business process collaborations, the Business Service View (BSV)- the view of a business process model that specifies the component services and agents and their message (information) exchange as interactions necessary to execute and validate a business collaboration, and the Business Transaction View (BTV) the view of a business process model that captures the semantics of business information entities and their flow of exchange between roles as they perform business activities(Folmer and Bastiaans, 2008, Huemer et al., 2008, (TMG), 2006).

REA (Recourse-Event-Agent) ontology was also used to model the above project. The REA ontology is a specification of the declarative semantics involved in business collaboration (or more generally in a business process). In its most simple form without a high degree of precision, REA can be portrayed as a UML class diagram with associations and generalizations relating the object classes.

### 2.5.3. e<sup>3</sup> Value

e<sup>3</sup> project was initiated with the goal of uplifting the interaction between the business enterprises in a wide level by JaapGordin and Hans Akkermans(Gordijn and Akkermans, 2001, Wieringa and Gordijn, 2005, Andersson et al., 2006). This project resulted many artifacts and findings by focusing their main goal. We are benefited with their findings such as e<sup>3</sup> family includes e<sup>3</sup>value, e<sup>3</sup>control, e<sup>3</sup>strategy, e<sup>3</sup>services and e<sup>3</sup>alignments and toolsets. In elaborating; e<sup>3</sup>value is developed to bridge the gap between business and IT groups, particularly for development of e-business systems. Further, it was brought on requirement engineering and the conceptual modeling technique from the information systems, e<sup>3</sup>control pattern was developed to keep the business networks sustainable in abstract, e<sup>3</sup>strategy was developed to settle the positions problem of business enterprise, e<sup>3</sup>service was developed to find the techniques and software tools for designing online services though value models. The e<sup>3</sup>value ontology is widely used by many business models for its various advantages.

The e<sup>3</sup>value is constructed on three important base level viewpoints namely, business value viewpoint, business process viewpoint and system architecture viewpoint. Basically e<sup>3</sup> value concept is based on actor, value exchange, value activity and value objects. This project mainly focused to develop a network business model consisting of actors and the items they exchange. In this project the success theme is notion of economic value, how objects are created, exchanged and consumed in a multi-actor network. This project widely uses the Unified Modeling Language (UML) as the modeling language since the light weight of that language and other feasible facilities. The challenging victory of this project is the e<sup>3</sup> value ontology. It consists of main



components as actor, value object, value port, value interface, value exchange, value offering, market segment, composite actor and value activity.

Meanwhile the e<sup>3</sup>ontology is widely spread due to its number of different approaches such as light weight, conceptual, semi-formal, graphical, multi viewpoint, scenario and economic value aware approaches. In conclusion the e3 project is succeeded with its error free e<sup>3</sup> ontology.

#### 2.5.4. REA Ontology Model

The Resource-Event-Agent (REA) framework in (McCarthy, 1979, McCarthy, 1982) is a specific format of representation of an exchange. The REA deals three important aspects of exchanges: the *events*, the *resources* that are subject of the exchanges, and the participating *agents* in one side. The write opposite of the above set is the duality concept of the REA ontology model. The basic concept is given below in the figure 2.

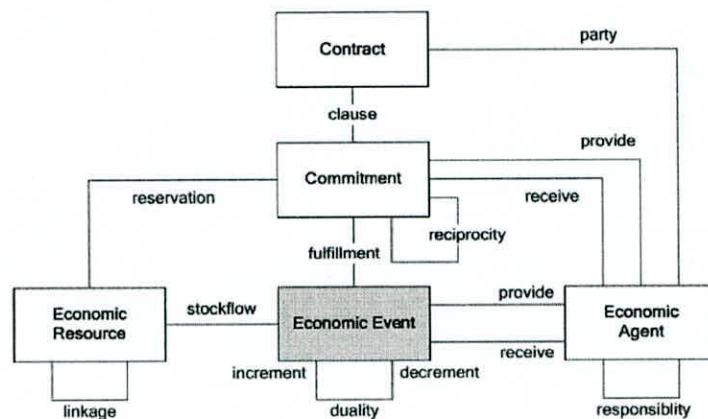


Figure 2 REA basic model (McCarthy, 1982)

*Economic Resource*: It has utility for economic agents, and is something users of business applications want to plan, monitor, and control on products and services; money; raw material; labor tools; and services.

*Economic Agents:* An individual or organization (customers, vendors, employees, and enterprises) capable of having control over economic resources, and transferring or receiving the control to or from other individuals or organizations.

*Economic Event:* It (sales of goods, labor acquisition, and provision and use of services) represents either an increment or a decrement in the value of economic resources that are under the control of the enterprise.

## **2.6. A short survey of selected popular legal ontologies**

Several studies have been done in order to formulate a good ontology in different conceptual schemes so far. It is clear by referring many literatures in legal domain; most of the studies have been focused to develop ontology for knowledge acquisition (viz. knowledge transfer, queries, and domain knowledge development), system development and domain theory development. The third one is still in crawling stage compared to earlier two. This thesis focuses on surveying four legal ontologies in this section and the main focus is on how far these ontologies can support the process optimization concept (which will be later developed). The followings are four popular ontologies in legal domain knowledge system development.

1. LLD: Language for Legal Discourse (McCarty, 1989)
2. NOR: Norma Formalism (Stamper, 1991, Stamper, 1996)
3. LFU: Functional Ontology of Law (Valente, 1995)
4. FBO: Frame Based Ontology (van Kralingen, 1997, Visser and Bench-Capon, 1996, Bench-Capon)

### **2.6.1. McCarty's LLD:**

McCarty proposed a language for legal disclosure by considering the basement of any ontology is language. If we narrow down McCarty's concept, the basic components of LLD is constructed in three elements called atomic formulae, rules and modality respectively (McCarty, 1989). The atomic formulae can be described as building relations between basic objects. "A distinction is made between count terms (tangible; company, actor) and mass terms (intangible; value, cash, stock). Rules are formed by connecting atomic formulae with logical connectives. The combination of atomic formulae and rules become as first order logical expression. Modalities are second order expressions. Modality deals with time, event and actions. Here state changes are realized by the events. According to the LLD, actions bridge (relation) the actors and events. LLD supports four modality operators namely permitted, forbidden, obligator and enabled. Ultimately the LLD is a language for the ontology but not an ontology even though it shows generic conceptualization of legal domain (Visser and Bench-Capon, 1996, Visser and Bench-Capon, 1998, McCarty, 1989, McCarthy, 1979). The common language is focused here in order to maintain the understanding the general quarries in legal domain. This model is well supported for terms and condition used legal domain. This LLD was not modeled for the workflow or process management concept.

### **2.6.2. Stamper's NOR:**

In contrast to McCarty's LLD, Stamper proposed a new concept which was against the use of traditional logic (symbolic representation) especially in legal domain, since the traditional logic suffers from weak connection with real world. Stamper proposed

NORMA formalism(Stamper, 1991) though his project LEGOL(Stamper, 1980) to fill the gap of McCarty's concept. Stamper stressed that the entities should be described by their behavior instead of individuality or any truth value of it. The NOR is conceptualized with the concept of agent, behavioral invariants and realizations(Stamper, 1991, Stamper, 1996). The agent (viz. person, team, enterprise, social agents and nations) plays major role and by the experience it shows the results to the real world through actions by the help of responsibility. Stamper tries to describe the invariant behavior of entity is constant during cause of any actions. In other words behavioral invariant is a situation which does not affect the entity or its behavior. And the realization is a state change of an agent due to the actions performed(Stamper, 1991, Stamper, 1996). Though NOR put effort to formalize the real world mapping still suffers from the conceptual reality. The NOR is still with uncovered process flows with managerial concepts. Limitation of the generalization due to the language is another disadvantage of the said ontology.

### **2.6.3. Valente's LFU:**

Valente proposed first acceptable ontology towards the functional perspective of the legal system as functional ontology of law. It is possible to abstract from the Valente's LFU that, the legal system is a tool, which can influence the social world. Valente strongly dealt with formation knowledge system that lead him to demonstrate six main legal knowledge areas (viz. normative, world or legal abstract model, responsibility, reactive, meta-legal and creative) in his LFU(Valente, 1995). We would like to summarize all of his knowledge as follows. Normative knowledge defines a standard of social behavior i.e. behavior of the people in society. World knowledge is descriptions

of the legal domain possibly the behavioral description of the system and stakeholders. Thus, Valente declared the world knowledge as a Legal Abstract Model (LAM). The said LAM consists two parts namely; definitional knowledge describes the definitions of legal norms (viz. concept, relations, case, situation and conditions) and causal knowledge describes dynamic part such as behavior of people in society. Responsibility knowledge describes the responsible of the people. Reactive knowledge shows the reaction to be taken in a situation. Meta-legal knowledge is the knowledge of a legal knowledge, most of the time reference knowledge. The creative knowledge in LFU refers the creation of non-existing entity ever before when it is needed (Valente, 1995). Moreover Valente fails to deal with process flows and the forming frameworks in order to show the creative knowledge or meta knowledge.

#### **2.6.4. Van Kralingen's and Visser's FBO:**

The main focus of FBO is to reduce the task dependencies in the legal knowledge system. The FBO is a mixture of legal ontology and statute-specific ontology. The FBO stands in good position since some part of the knowledge design is still reusable in sub legal domain (knowledge base systems) (Visser and Bench-Capon, 1996, Bench-Capon). According to the legal ontology; it consists of three parts norms, acts and descriptions respectively. As we can see the norms is general rules and it is built with eight components (viz. norm identifier, norm type, a promulgation, scope, conditions & applications, norm subject, legal modality and act identifier). Act represents a behavior which can affect the real world. State changes, events and processes are stated under the acts and which comprised with set of components (viz. act identifier, promulgation, scope, agent, act type, modality of means, modality of manner, temporal aspects, spatial

aspects, circumstantial aspects, cause of the action, aim of the action, intentionality of an action and final state (Visser and Bench-Capon, 1996). Concept description is defined as the meaning of the different concepts in the legal domain and the concept description comprised with seven elements (viz. concept to be described, concept type, priority, promulgation, scope, conditions under which a concept is applicable, and enumeration of instances of the concept). As we discussed above generic legal ontology the elements are very much generic to the legal domain in FBO leads the elements stated above can be reused. The statute-specific ontology would be used under a specific sub domain since it describes and suits only within a sub domain. It cannot be used in another sub domain since the knowledge of the sub domain is different than other sub domain(Visser and Bench-Capon, 1996). One of the big advantages of this ontology some parts or terms can be used in legal sub domain with strong vocabulary concepts. This ontology is also strongly towards to the language and collection of legal vocabulary specific but not with the focus of the modeling organizational flow or work flow concept.

## **2.7. Existing DEMO Methodology**

This section introduces the methodology adapted to the research conducted. Most of the information system researches follow the design science and action theory as mentioned earlier in the chapter one. This thesis is mainly focusing on design science proofing that, the application of existing knowledge to apply in a new system and resulting either solution or new artifact. This chapter is divided in to two parts; first is to describe the currently available DEMO methodology and the second part would be DEMOE (DEMO-Extended).

## 2.8. The Enterprise Ontology

It defines, that the ontology (or ontological model) of an enterprise is defined as an understanding of its operation, that is completely independent of the realization and the implementation of the enterprise (Uschold and Gruninger, 1996, Uschold et al., 1998).

It is essential and deeply realized of a conceptual model in order to cope up with the current and the future challenges. The said conceptual model should be coherent, comprehensive & complete, consistent, modular, objective and concise and demonstrates the operational factors of the enterprise model and further the properties of the conceptual model is described here (Dietz, 1999, Dietz, 2006, Reijswoud and Dietz, 1999)

**Coherent:** The enterprise model considers an organization to consist of a coherent layered integration (Dietz, 2006, Dietz, 2001) of three aspect-organizations: the B-organization (business), the I-organization (information) and the D-organization (document). These organizations have series of supportive norms, in which the I-organization supports the B-organization and the D-organization supports the I-organization. Every organizational change must be reflected on one of any aspect-organizations. The coherent integration of the three aspect-organizations makes (re-) designing and (re-) engineering manageable.

**Consistent:** Employed four aspect models of DEMO (Dietz, 2006) are categorized under the same meta-model. Therefore the mutual consistency of the perspectives is naturally safeguarded (and for example easily implemented in a supporting tool). A

sudden organizational change may affect in all aspect models. Through the mutual consistency of the models the correlation between the processes, the data and the organization is always clear. The impact of changes in one of the perspectives on the other perspectives is always fully and directly visible; there are no surprises, in the form of unanticipated consequences, during the implementation process(Dietz, 1999, Dietz, 2006).

**Comprehensive and Complete:** The DEMO-transaction is a universal pattern of coordination acts which lead the creation of a production fact. It is the generic building block for all business processes. This transaction can be used for designing of different business processes with the assurance of no information will be overseen regardless of the interested domain. The actions, especially the promise and the acceptance are carried out by unexpressed agreement and are rarely supported by IS or workflow management systems (Dietz, 1999, Dietz, 2006). DEMO supplies a complete and clear definition of competences, authorities and responsibilities for organizational implementation and relevant information for information structure which are necessary to the actors.

**Modular:** The combination of a transaction and the actor role that is its executor, constitutes the 'molecular' building block of organizations, whereas the transaction steps in the universal transaction structure are the 'atoms'. A business process is a tree structure of transactions. This modular structure offers a perfect alignment with component-based system development and it offers the ideal starting point for decisions about splitting and allying organizations(Dietz, 1999, Dietz, 2006). The redesign and



restructuring of business processes can be implemented always still remains comprehensible and manageable without affecting the organization structure.

**Objective:** DEMO models are objective. It provides reproducible models, which are independent from the 'modelers'. Since all irrelevant information are refused, the DEMO delivers compact and truthful models (Dietz, 1999, Dietz, 2006).

**Consistent:** The aspect models / Meta models are free from contradictions on set of models. Thus conceptual aspect models are well balanced and covering broad area of understanding including functional and organizational areas in an enterprise(Dietz, 1999, Dietz, 2006).

**Concise:** The way of modeling should be compact and succinct in order to express correct and exact message to the user without irrelevant matters. The property “essential” to the enterprise model is important since it shows fully independent from the way the model is realized and implemented The essence is highly stable and always up-to-date as it only shows which products (services) are delivered and what the structure of their corresponding business processes is, but not how these are implemented (Dietz, 1999, Dietz, 2006).

## **2.9. The Notion of DEMO Ontological Model & Enterprise Engineering**

The DEMO shows strong ontological notion of the focused system with broad description. Especially the ontological notion of the DEMO focuses on construction and operation of the decided system as well. This ontological notion of the system is

perfectly adequate for building and changing system (re-engineering). It is more clear if it precisely stated that, ontological system notion defined as “something is a system if and only if it has the following properties such as; composition (set of elements of some category), environment (set of elements of the same category), production (the elements is the composition produce things that are delivered to the environment– goods or services), and structure (set of bonds among the elements). Meanwhile the composition, environment and structure collectively called construction of the system. Most of the times white box model (Dietz, 2001) obeys the ontological definition since it captures construction and the operation of the system. By the construction it clearly demonstrates the components and their interaction relationships.

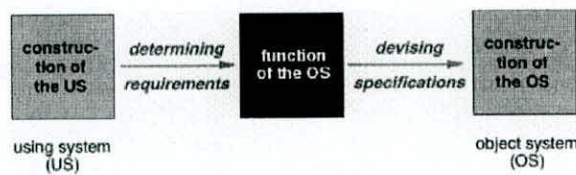


Figure 3 The system design process (Dietz, 1999)

More than the practice side of enterprise engineering, it focuses at studying enterprises or organizations in a multidisciplinary and engineering-driven way. In spite of these efforts, the current literature on enterprise engineering consists merely of best practices, without an integrating theory and without a clear definition of the field.

The figure 3 illustrates apparently how the design process is being done. There are two systems being engaged with while the design process is done, called using system (US) and object system (OS) respectively(Dietz, 2006). According to the literature an ontology model of the system can be independent in its implementation. Structural or aspect models are playing vital role of describing overall structure and workflow of any

information systems. The below figure-4 shows the way ontology is being used to construct a system.

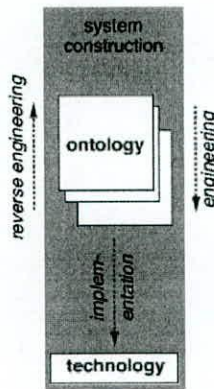


Figure 4 Ontology system constructions (Dietz, 1999)

The following definition is useful when understanding further about the methodology. Re-engineering a system means redoing the engineering process, starting from the ontological model, which may or may not have been changed as a result of redesigning the system (Dietz, 2001). The process of reconstructing the higher level models from the implementation model is known as reverse engineering (Dietz, 1999, Dietz, 2006, Dietz, 2001).

## 2.10. The system development process

The ontological framework model consists of actor roles, coordination acts/facts and production acts/facts. Therefore figure 5 is suitable architectural design with actor technology, communication technology and production technology. The actor technology can be represented by either actor roles or the systems. Meanwhile the system also represents the role of a human on a selected end point. The communication technology consists of series of communication acts in between two actor roles consists

of data logical work and info logical works. The production act would be handled by the human since it may be either material or immaterial production. This thesis focuses the immaterial production since legal domain is rich in decision making.

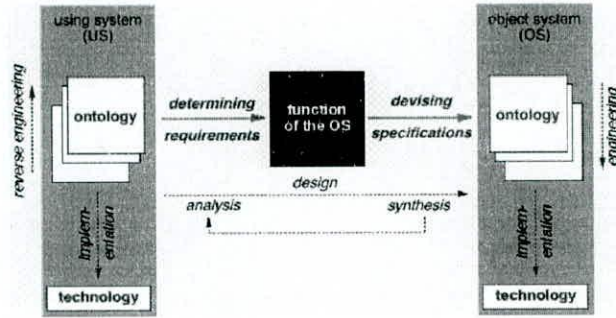


Figure 5 The system development process (Dietz, 1999)

The system to be developed is shown in above figure 5 as OS. It is meant to be used by the US and is thus going to support the US. The support of the US by the OS is called a way of realization of the US.

## 2.11. The DEMO Methodology

### 2.11.1. Modeling Theory

The DEMO is an acronym that has had several long forms in the course of time, starting with “**D**ynamic **E**ssential **M**odeling”. The current one is “**D**esign and **E**ngineering **M**ethodology for **O**rganization” (Dietz, 2006, Dietz and Habing, 2004, Reijswoud and Dietz, 1999).

### 2.11.2. Operation Axiom

We would like to introduce the DEMO methodology with the basic theory called “ $\Psi$  theory”(Dietz, 2006, Krouwel and Op’t Land, 2011), which is all about the

organizations. The symbol “ $\Psi$ ” is said as PSI: Performance in Social Interaction, the paradigm on which the theory is founded. The  $\Psi$  theory does justice to the fact that organizations are social systems, while at the same time providing a rigorous engineering type of framework for understanding them.

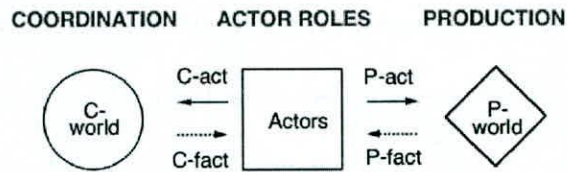


Figure 6 Graphical representation of the operation axiom (Dietz, 1999)

The  $\Psi$  theory consists of mainly three components; actor role, coordination and production. An actor role always deals with both production world (P-world) and the coordination world (C-world). The  $\Psi$  theory is shown in figure 6 given above. By performing production acts (P-act), the subjects contribute to bringing about the goods and / or services to the environment in the form of either material or immaterial. A subject enters into and complies with commitments towards each other regarding the performance of production is called as coordination acts (C-act). A subject in its fulfillment of an actor role is called an actor. The illustration of the coordination act shown in the figure 7.

The coordination act is described with flow of some sub activities initiated from performer to addressee focusing on producing an important production with the intension. The intension can be described as request, promise, question and assertion.

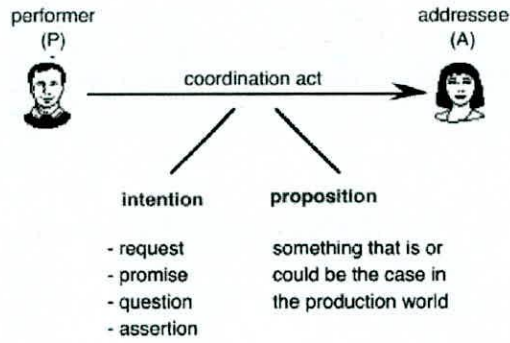


Figure 7 Representation of a coordination act (Dietz, 1999)

### 2.11.3. Transaction Axiom

It is the second axiom of the DEMO. The transaction is clearly elaborated here with well explained way in different level. A transaction is “a sequence of coordination acts between two actor roles that are aimed at achieving a well-defined result concerning a P-act/fact through three different phases such as order phase (O-phase), execution phase (E-phase) and result phase (R-phase)(Dietz, 2006). The person who initiates the transaction is called as initiator and the person who execute it as executer. A basic transaction pattern has four coordination acts and a production act namely request, promise, accept and execute (Dietz, 2006). The basic transaction pattern is illustrated in the figure 8 given below.

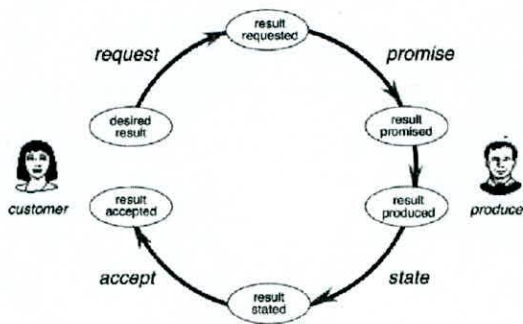


Figure 8 Basic transaction pattern (Dietz, 1999)

In addition to the above diagram the coordination acts such “request” and “promise” fall under the O-phase. The “state” and “accept” fall under the R-phase. The real execution is classified under the E-phase.

It is possible to show more depth level of the full positive transaction pattern diagram.

The transaction pattern is shown below in figure 9.

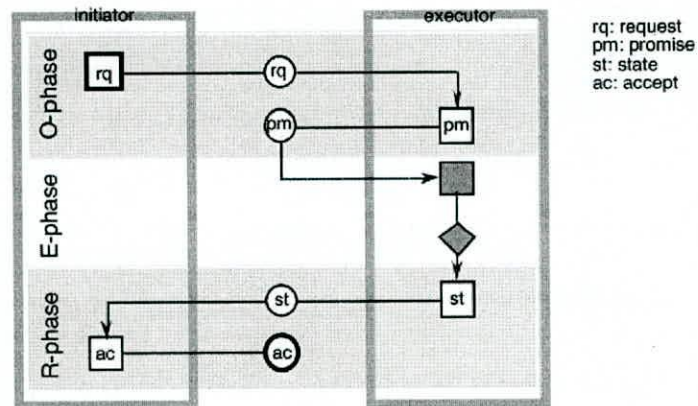


Figure 9 Basic pattern of a transaction (Dietz, 1999)

In the above figure 9, white box represents a C-act type and white disk represents a C-fact type. A gray box represents a P-act type and a gray diamond represents a P-fact type. The initial C-act is drawn with a bold line, as is the terminal C-fact. In the background, the three phases are indicated. Every act and fact belongs to one of them. The gray-colored frames, denoted by “initiator” and “executor” represent the responsibility areas of the two partaking actor roles. As advancement, the basic transaction pattern can be extended to the standard transaction pattern with the decision state like accept; decline; or reject, shown in figure 10 is given below.

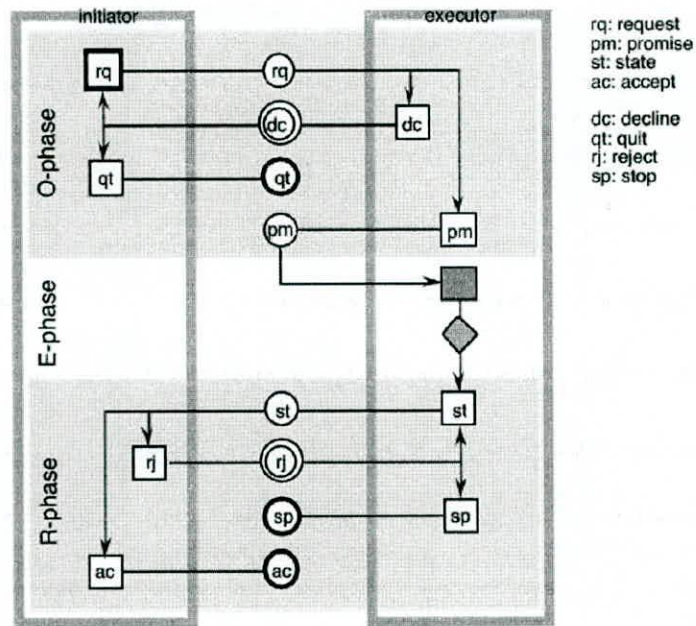


Figure 10 Standard transaction pattern (Dietz, 1999)

In the above diagram, it is clearly illustrated that there are more stages than the basic transaction pattern such as “decline”, “quit”, “reject” and “stop” respectively. More over either the basic transaction pattern or standard transaction pattern can be extended with cancellation patterns. The DEMO is rich in situational cancellation pattern. The cancellation pattern rises with different communication act stages such as; “request”, “promise”, “state” and “accept”; and it describes as request-cancellation, promise-cancellation, statement-cancellation and accept-cancellation pattern respectively. It summarize that, a transaction can be cancelled four different places during the cause of the transaction pattern life cycle.

## 2.12. Development of DEMO Aspect Models

A set of ontology aspect models can be developed by the application of DEMO methodology. This DEMO methodology is mainly focusing on four aspect models namely action model (AM), process model (PM), state model (SM) and construction



model (CM). In depth level the construction model can be developed in two different models called interaction model (IAM) and interstriction model (ISM) respectively. The brief description of all of these aspects models are shown in figure 11 given below.

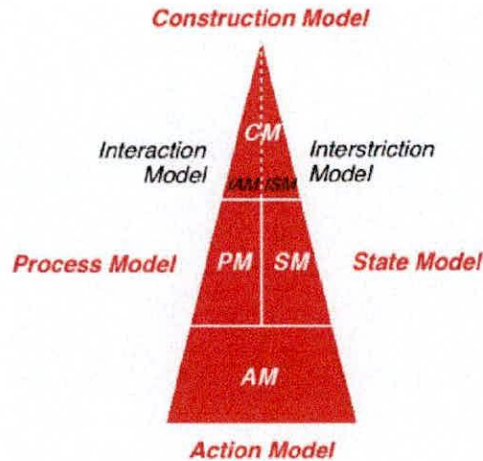


Figure 11 The ontological aspect models(Dietz, 2001)

### 2.12.1. Construction Model

The construction model of an organization specifies its composition, environment, and structure. Both composition and environment are set of actor roles fulfilled by human individually or collectively. The interaction structure of an organization consists of the transaction types in which the identified actor roles participate as initiator or executor. It is said as an Actor Transaction Diagram (ATD) and a Transaction Result Table (TRT).

### 2.12.2. Process Model

The process model (PM) defines the state space and the transition space of the C-world and thus sequences of states in the coordination world. A C-fact and its causing C-act are collectively called a process step. The PM is partially based on the information defined on the CM concerning which actor roles perform the C-acts. The coordination acts performed by each role are designated the responsibility areas of that actor role.

The process model is further described with the help of process state diagram (PSD) and information use table (IUT).

### **2.12.3. State Model**

The state model (SM) is the specification of the state space of the P-world. It consists of object classes, fact types, and result types, as well as the derived laws which hold. An SM is expressed with an Object-Fact Diagram (OFD) using Object Role Modeling. The SM also includes an Object Property List (OPL) that is able to specify fact types that are mathematical functions.

### **2.12.4. Action Model**

The action model (AM) is the most comprehensive aspect model and integrates the concepts defined on the other aspect models (PM, SM, and CM). Therefore, the other models can be derived from the AM. This model describes the rules that apply to every transaction and would be applied to each and every actor roles. In general a pseudo-algorithmic language is used to specify the action rules of the organization interested.

## **2.13. The DEMO Extended**

The said methodology in previous section shows some incompleteness and is incompatible while we apply it in the legal domain. Therefore we intended to propose the enhanced version of the DEMO methodology as DEMO-E in order to overcome the incompleteness and the incompatibility. The base of the methodology is more or less equal but we found many places it should be remade or reengineered. I would like to point out some places where it should be enhanced. The DEMO method mainly focuses

on the business domain. Thus, it needs significant modification in order to apply in the legal domain with more complicated interactions. The development of the DEMO-E will be discussed in the next chapter.

#### **2.14. The Modifications Needed**

The following breaches were found in the current DEMO methodology while it is being used in legal domain and these should be restored in order to develop the DEMO-E methodology.

1. Duplication of the actor role is not mentioned, which makes it very hard to identify whether any particular actor role has been used earlier or not specially in a huge aspect model diagram (process model). It can be described by the case filing concept. In case filing the external actor role (lawyer) would be interacting with the courts in different places such as tending case, payment to the courts, receiving the approval from the judge, case hearings, and receiving the verdict. In this complex situations it is hard to point out the actor role is already available in the process structure diagram or not. So it is necessary to state the duplication of the actor roles throughout the process structure diagrams.
2. It is not clear whether a data set is being under retrieval or storing/ writing in the data store or data warehouse specially Actor Bank Diagram (ABD). The designer of the ontology cannot understand the data from the data bank under read or write since this step is much needed for the either programmer or designer.

3. It is a big gap that, we are not able to see a separate data flow from any of these aspect models. The complexity of the legal system is much interconnected with data flow. The status of the data being processed is necessary to in order to proceed to the next level of transaction correctly and precisely.
4. The standard transaction pattern lacks of decision points (decision visibility). According to the standard transaction diagram one action can be accepted and rejected at the same time. In the legal system it is not possible to happen like both decisions at the same time. Meanwhile this situation makes the programmer confuse and the programmer would be facing big difficulties while they try to program on it.
5. The data flow has only been stated in the Construction Model (CM). The data should be stated precisely since the data shows huge issues in legal system. Combining the data flow with another model is hard to identify the data flow as well all the depth level data flow will not be shown at the combined data flow with another model.
6. There is no single concept model, which can be applied in any situation. In the legal system there are many actions happening toward the verdict. It is necessary to produce a Meta model / concept model to apply in any level in the case from the filing up to verdict. Apart from this, it is more useful if there is a Meta model for the transaction, which can be applied others cases too.

7. Cancellation patterns can be reduced. In business oriented culture there may be many places where cancellation patten can be applied, but the service oriented structure like legal system the cancellation pattern should be very limited.
8. There are no decision points while we work with cancellation patterns.
9. In the legal domain anywhere we cannot find the “refuse” state. Not like the business enterprise, the legal system there is no state like refuse in the complete case filing scenario.

The above said problems were considered while the ontology aspect models and an abstract goal model are being modeled.

## CHAPTER THREE

### LSPM FRAMEWORK

This chapter proposes the development of the LSPM framework through DEMO-E modeling method. The LSPM framework was modeled according to the description of the case filing scenario at the District Courts. The case filing procedure is attached the annex I.

#### 3.1. Overall Approach of LSPM

As mentioned earlier in this thesis, the hybrid methodology was adopted. The enterprise ontology framework model for legal domain illustrated in this section in a series of steps including five aspect models and the way the models are being modeled with necessary basic theoretical framework. The following figure 12 shows the approach of the designing the framework for Legal Service Process Management (LSPM).

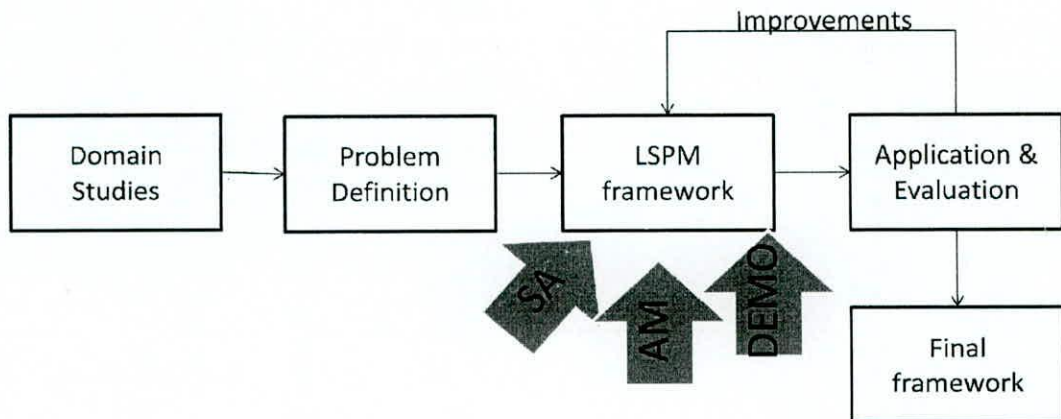


Figure 12 LSPM framework approach

At the initial the legal domain bounded within the research area (case filing and District Courts) was thoroughly studied in order to find the problems regarding the process management which is been practiced at the District Courts. Then the problems of Legal

Service Process Management listed (in the chapter one the problem definition is clearly stated). The overall LSPM based on Speech Act (SA) (Jayaweera, 2004), which describes about the communication activities among the different processes; Aspect Model (AM) which is to be developed in this same chapter under 3.4 sub section; and DEMO, which is explained in the chapter two under section 2.11.. The development of the LSMP framework is described by integration of Synthesis and Analysis, aspect models and the DEMO framework. The final framework was accepted by evaluating the proposed LSPM framework against test cases with the improvements.

### **3.2. Development of LSPM**

The LSPM framework was developed with three basic analyzing (activity classification, actor role classification and result structure classification) and three syntheses (process pattern, construction and organization) as base level actions. On top of that, in order to do the analysis and synthesis the Coordination-Actor-Production (CAP) framework was applied in order to identify the activities that belong to coordination world or production world with the actor role involvement. Further, LSPM comprise of the standard transaction patterns that would be utilized to understand and design communication between actor roles in legal collaborations. Finally ontology aspect models were developed.

The LSPM framework is comprised with following sub-components as listed below

- CAP framework
- Development of standard transaction pattern
- Analysis and synthesis

- Aspect models

### **3.3. CAP framework**

The CAP framework consists of mainly three components; actor role, coordination and production. An actor role always deals with both production world (P-world) and the coordination world (C-world). By performing production acts (P-act), the subjects contribute to bringing about the goods and / or services to the environment in the form of either material or immaterial. A subject enters into and complies with commitments towards each other regarding the performance of production is called as coordination acts (C-act). A subject in its fulfillment of an actor role is called an actor.

### **3.4. The Aspect Models**

There are five important aspect models available in order to describe the total framework of the legal domain case filing scenario namely; action model (AM), process model (PM), state model (SM), data transfer or data access model (DT) and construction model (CM). The CM is constructed with interaction model (IAM) and interstriction model (ISM) respectively. The definition of each model is stated in the previous chapter except data transfer model. The data transfer model promotes to the ontology programmer and the system developer to identify the data flow and the level of use. The figure 13 illustrates the five ontological aspect models.



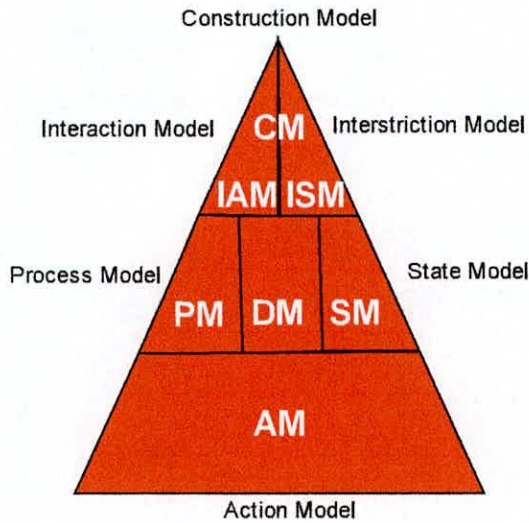


Figure 13 The ontological aspect model of DEMO-E

It is important that, the general elicitation method to acquire the basic for a correct and complete set of aspect models of enterprise ontology like DEMO-E based on three analysis and three synthesis steps which are stated below.

### 3.5. Analysis and Synthesis

1. *The performa-informa-forma analysis:* All available pieces of knowledge are divided in three sets, with respect to the distinction axiom; ontological action, info-logical action and data-logical action respectively.
2. *The coordination-actor-production analysis:* The Performa items are divided into C-acts/results, P-acts/results, and actor roles, according to the operation axiom. It is very much transparent since the three kinds are well distinguished in textual descriptions.
3. *The transaction pattern synthesis:* The transaction pattern, according to the transaction axiom, is juxtaposed over the results so far, as a template, in order to cluster them into transaction types. Next, for each transaction type, respected result

type is rightly and exactly derived. The Transaction Result Table (TRT) can now be produced

4. *The result structure analysis:* According to the composition axiom, every transaction type of which an actor in the environment is the initiator who is the medium for the delivery of the result to the environment. In general, the (internal) executor of the transaction type is initiator of one or more other transaction types, and so on. The transaction results can be viewed as components of the end result.
5. *The construction synthesis:* For each and every transaction type, initiating and executing actor role(s) are identified founded on the transaction axiom. And which leads to development of the Actor Transaction Diagram.
6. *The organization synthesis:* The identification of the construction is broken down in to both organization to be analyzed and its environment. The Actor Transaction Diagram can be set up with the organization synthesis.

### **3.5.1. Performa-Informa-Forma analysis**

The performa-informa-form analysis is the first step of the development of the DEMO-E model for the development of LSPM framework. This analysis can best be done by coloring the appropriate parts of the descriptions: **red** for Performa items, **green** for Informa items, and **blue** for Forma items.

*The general scenario of filing a case is described as given below. The plaintiff meets the plaintiff lawyer for the **consultancy** regarding to file a case (divorce case). Meanwhile lawyer fee also will be discussed as well. The lawyer **prepares the set documents such as motion, proxy, plaint, summons, and precept to fiscal** in order to **tender** a case with the **information given by the plaintiff**. The lawyer **starts to write the plaint with the clear***

*description with in which ground the case can be tendered. After series of discussions and preparation of necessary documents, client/ plaintiff signs proxy (appointment of an attorney-at-law) in order to give authority to the plaintiff lawyer for the institution of actions and precedence of the case. Plaintiff lawyer signs the motion and plaint. Soon the plaintiff lawyer tenders the motion, proxy, plaint, documents, summon, precept to fiscal along necessary payments (stamp duty, Courts fee) which is already paid of the request of Courts clerk to the clerk, who is designated for the purpose. Lawyer files (hand over) above said documents and stamp duty payment receipt to the Courts clerk. The clerk checks all the documents and enters the case number and name of the parties in the case register. Once the file is been registered, the clerk sends the documents to the binder to binds as a case record. The bound document (case record) is sent to the chief clerk (registrar). Registrar checks the case record and forwards it to the judge for the acceptance and issue the summon on defendant. Judge peruses the case record and checks for the validity of it. If the judge satisfies with the case record he accepts and judge issues the summon by a way of order or decision, accordingly the registrar signs the summons and direct the fiscal to serve summon together copy of the plaint and relevant documents which were filed by the plaintiff's lawyer to the defendant. Meanwhile a copy of all above stated documents will be served to the plaintiff lawyer too. The case filing process terminates with the submission of the summon to necessary parties.*

In the above paragraphs the analysis have been done in order to differentiate the activities due to ontological production works, info logical and data level works.

### 3.5.2. The Coordination-Actors-Production analysis

As mentioned earlier in this chapter the Performa items are divided into C-acts/results, P-acts/results, and actor roles, according to the operation axiom. A suitable way to perform this analysis is to draw a small box or disk or diamond over the pieces of text that are marked red in the previous performa-informa-forma analysis. These shapes, respectively, indicate an actor role, a C-act/result, or a P-act/ result. In electronic documents, it is more convenient to enclose a piece of text indicating an actor role between the brackets “[“and “]”, a piece of text indicating a C-act/ result between the brackets “(“and “)”, and a piece of text indicating a P-act/ result between the brackets “<” and “>”. In addition, to avoid any confusion, the enclosed pieces of text are underlined

The general scenario of filing a case is described as given below. *The plaintiff meets the plaintiff lawyer for the **consultancy** regarding to file a case (divorce case). Meanwhile lawyer fee also will be discussed as well. The lawyer prepares the set documents such as motion, proxy, **plaint**, summons, and precept to fiscal in order to <tender>a case with the information given by the plaintiff. The [lawyer] (starts) to write the **plaint** with the clear description with in which ground the case can be tendered. After series of discussions and preparation of necessary documents, client/ plaintiff signs proxy (appointment of an attorney-at-law) in order to give authority to the plaintiff lawyer for the institution of actions and precedence of the case. Plaintiff lawyer signs the motion and **plaint**. Soon the plaintiff lawyer tenders the motion, proxy, **plaint**, documents, **summon**, precept to fiscal along necessary <payments> (stamp duty, Courts fee) which is already paid of the (request) of Courts clerk to the clerk, who is designated for the*

*purpose. Lawyer files (hand over) above said documents and stamp duty payment receipt to the Courts clerk. The clerk checks all the documents and enters the case number and name of the parties in the case register. Once the file is been (registered), the clerk sends the documents to the [binder] to <binds> as a case record. The bound document (case record) is sent to the chief clerk (registrar). Registrar checks the case record and forwards it to the judge for the acceptance and issue the summon on defendant. Judge peruses the case record and checks for the validity of it. If the judge satisfies with the case record he accepts and [judge] issues the summon by a way of order or <decision>, accordingly the registrar signs the summons and direct the fiscal to (serve) summon together copy of the plaint and relevant documents which were filed by the plaintiff's lawyer to the defendant. Meanwhile a copy of all above stated documents will be (served) to the plaintiff lawyer too. The case filing process terminates with the submission of the summon to necessary members.*

As shown in the above scenario we have identified the actors, production facts and coordination activities.

### **3.5.3. The Transaction Pattern Synthesis**

The major activity of this steps is to cluster the identified coordination acts/facts and production acts/facts in to transactions. Next, for every transaction type, the result type is correctly and precisely formulated. To be fully sure, one can check the identification of a transaction by validating the existence of one or more C-acts/results, either by studying the case description, or by asking the people who are involved in carrying through transactions of this type. Only if in the description of the case, there is no hint at

all of the existence of a transaction, can one forget its modeling. The Transaction Result Table (TRT) can now be produced.

Transaction No	Transaction Description
T01	tender
T02	payment
T03	binding
T04	decision

Table 1 Transaction Result Table

These transactions show the main jobs of the system; however, this has to be validated in the following steps. It is necessary to add the list of generated result type according to each transaction type. This step is being done for couple of reasons; one is that only a precise formulation can be taken as the starting point for constructing the State Model and this is the last chance to eliminate those transactions that are not B-transactions, but I- or D-transactions, or are not transactions.

According to the above mentioned transactions the following resultant types will be produced. As we can see, all these transactions happen towards to achieve a case acceptance and are called this as "C".

Result	Result type
R01	tendering of C has been tendered
R02	payment of C has been paid
R03	binding of C has been performed
R04	decision of C has been made

Table 2 Transaction results and types

It is important and possible to get the common and final table by joining table-1 and table-2 given above

Transaction No	Transaction description	Result No	Result Type
T01	tender	R01	tendering of C has been tendered
T02	payment	R02	payment of C has been paid
T03	binding	R03	binding of C has been performed
T04	decision	R04	decision of C has been made

Table 3 combined transaction and result table

In summary the detail of the above table would be important and be used everywhere.

### 3.5.4. The Result Structure Analysis

According to the composition axiom, every transaction type of which an actor in the environment is the initiator may be conceived as delivering and end result to the environment. In general, the (internal) executor of this transaction type is initiator of one or more other transaction types, and so on. The results of these cascaded transactions can be viewed as components of the end result. The practical way to perform this step is to read the description of a case to look for phrases in which dependencies between production acts or results are expressed. This step of the method helps in determining the causal and conditional relationships between transactions.

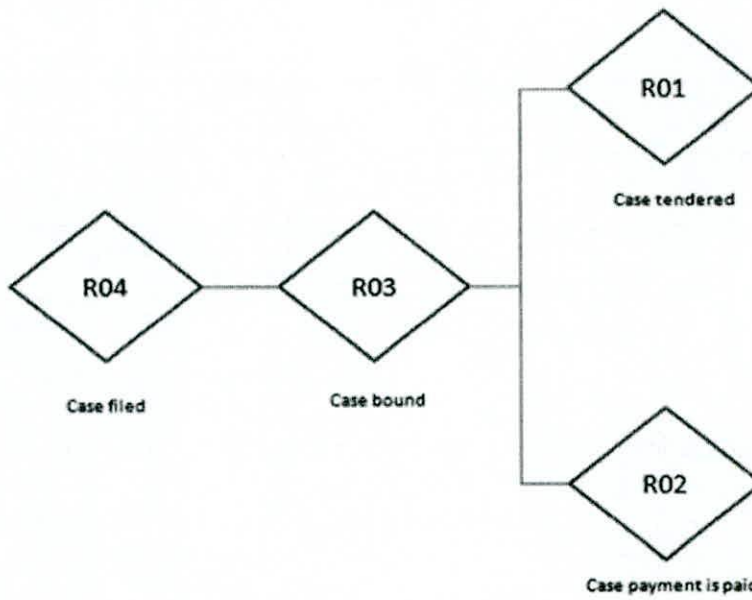


Figure 14 Result structure chart of case filing

It is clear from the above figure 14 that the T04 (decision) depends on T03 (case binding) and the T03 (case binding) depends on both T01 (tendering) and T02 (payment). With the identification of the dependencies this part is concluded.

### 3.5.5. The Construction Synthesis

Under construction synthesis, the actor roles are identified that serve as the initiator and/or the executor of the transaction types that have been found in the previous steps. The crispinet (Dietz, 2006) representation plays major role in order to develop the diagrams / models here after. The given table (table#) describes the initiator and the executor.

Transactions	Actor Role Initiator	Actor Role Executer
Tender	Plaintiff lawyer	Clerk / Registrar
Payment	Clerk / Registrar	Plaintiff lawyer
Binding	Clerk / Registrar	Binder
Decision	Registrar/ Binder	Judge

Table 4 Transaction and actor roles



### **3.5.6. The Organization Synthesis**

This last step of the method and all about identifying and deciding about what belongs to the kernel of the organization and what belongs to the environment, as well as what constitutes the interface between them. This synthesis will be done with the ontology aspect models follows.

## **3.6. Aspect Models**

In this section all five aspect models are explained with the ontology models for case filing scenario in the District court of Sri Lankan judicial system.

### **3.6.1. Interaction Model (IAM)**

The construction model (CM) of an organization specifies its composition, environment, and structure, according to the system definition. The composition and the environment are both a set of actor roles. By norm, the environmental actor roles as composite actor roles, even though that an actor role is elementary since, in general it is unclear whether an environmental actor role is elementary or composite. Further, the kernel of the selected system is seen as a global composite actor role. The resulting CM is referred as the global CM of an enterprise. The CM in which the kernel contains only elementary actor roles is referred the detailed CM. The boundary clearly distinguishes the set of all (relevant) actor roles into the composition and the environment. The IAM is a part of the construction model (CM), and which can be modeled by modeling actor transaction diagram (ATD) global, actor transaction diagram (ATD) detail and the transaction result table (TRT). And moreover this ATD contains the transaction types in which the identified actor roles participate as initiator or executor.

In order to draw the first diagram of ATD as global ATD it is necessary to create the table called transaction result table (TRT) of the court case filing. The TRT table 5 is given below.

Transaction type	Result type
T01 case tender	R01 tender C has been started
T02 payment of fee	R02 payment C has been paid
T03 case record binding	R03 binding C has been performed
T04 decision	R04 decision C has been made

Table 5 TRT of case filing

Now it is possible to draw the global ATD since we have identified the external actor roles, kernel of the court, initiators, executers and the transactions. The global ATD is illustrated below in the figure 15.

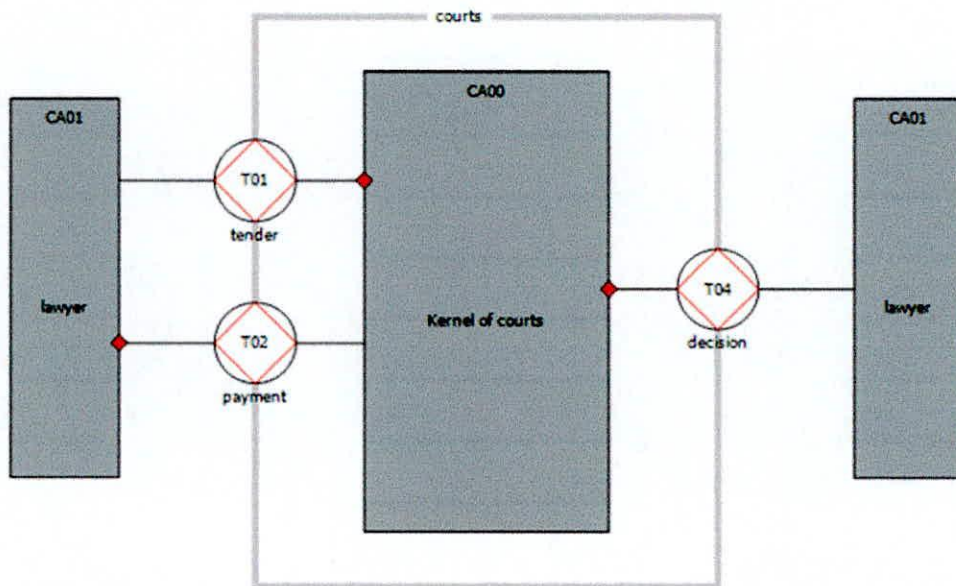


Figure 15 Global ATD of case filing at District court

CA – composite actor role    Txx – Transactions

Figure 15 illustrates that, the diagram in a very abstraction mode without showing much detail such as internal transaction except the interface transactions. The names of the actor roles are not a formal part of the construction diagram but for the easiness it is just used. CA00 is the first composite role of the court system and which is called the kernel of the Courts. The kernel system may be initiator or executer of the transaction but not both. The grey line with the text of "court" is the system boundary where whatever the transactions being done inside the grey line is called internal transaction and the actor roles which are inside the grey line is called internal actor roles. Outside the grey line is the opposite of the above stated sentences as external actor roles and the external transactions (interface transaction). Mostly the interface transactions are called the external transaction but in general it is said as interface transactions. The transactions T01, T02 and T04 are called transactions (interface transactions). There are two different kinds of links are found in between an actor role and a transaction. The link starting with the small red spot from an actor role is called executor actor role. As example for tender (T01), decision (T04) and payment (T02); the court kernel and lawyer are executors respectively. Meanwhile the link starting without the red spot in the beginning is called as initiator for the transaction. In the above diagram CA01 (lawyer) acts as initiator of T01 and T04; and CA00 (Courts kernel) is for T03.

The next part of the global actor transaction diagram is to design the detailed actor transaction diagram. The detail transaction diagram is shown below in figure 16.

ශ්‍රී ජයවර්ධනපුර විශ්වවිද්‍යාලය, ශ්‍රී ලංකාව  
UNIVERSITY OF SRI JAYEWARDENEPURA, SRI LANKA

Department of Computer Sciences  
Faculty of Applied Sciences  
University of Sri Jayewardenepura  
10250 Nugegoda  
Sri Lanka



eMail: prasad.jayaweera@gmail.com  
URL: prasadjayaweera.pbworks.com  
Office Phone: +94 (0)112758909  
Mobile Phone: +94(0)718301599  
Fax: +94(0)112802914

September 17<sup>th</sup>, 2015

Dean

Faculty of Graduate Studies

University of Sri Jayewardenepura

Gangodawila

Through;

Head/Department of Computer Science

Forwarded  
through  
20/15/17

**Viva - Voce Examination" RKAR Kariapper**

This is to certify that RKAR Kariapper has substantially addressed all corrections, additions, and amendments in accordance with the comments and suggestions received by his examiners in his final version of M.Phil. Dissertation.

Thanks!

P. Prasad M. Jayaweera

PhD & PhL (Stockholm) and BSc (Colombo)

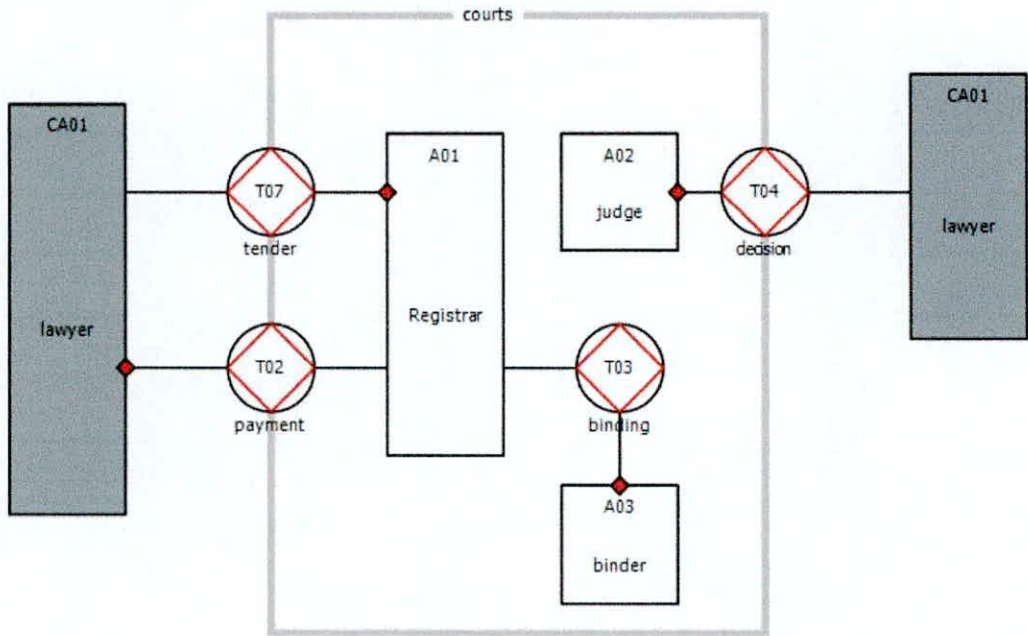


Figure 16 Detailed ATD of District court (case filing)

In that above diagram, there are mainly three parts such as tender and payments; binding; and decision making. The detailed ATD explains the elementary actor roles and internal transactions. Here the Courts kernel CA00 (composite actor role) is converted to the basic elementary roles such as registrar role (A01), binder (A02) and judge (A03). We can see there is a new transaction called T04 (binding) is out there. The binder (A02) will be executor for the transaction T03 (binding) and the A03 (judge) is the initiator of the same transaction T03. In abstract; one could apply a finer-grained product structure, as we suggested result structure analysis above. This can always be done for material production, down to the level of the nuts and bolts. Regarding nonmaterial products, like being paid or being completed of a purchase, the product structure is always simple, typically only one level deep.

### 3.6.2. The Process Model

The process model (PM) of an organization shows state space and transition space of the C-world. Furthermore it describes the set of possible and allowed series of states in the C-world. Each transition in the C-world results a C-result since there is a strong 1:1 relationship between C-result and causing C-act. A C-result and its causing C-act are collectively called as process step. A process model is expressed in a Process Structure Diagram (PSD) and an Information Use Table (IUT). The symbol of the C-act type (a small box), from which it is created. This is possible since any C-act type has exactly one C-fact type as its result. A similar reasoning holds when pushing the small diamond of the P-fact type onto the small box of the P-act type. The combined symbols represent process steps. By recalling the DEMO it is mentioned that, the process state diagram should not be used with standard transaction pattern but the basic transaction pattern. In the standard transaction pattern only we can see the rejection, refuse or cancellation for any reasons. Meanwhile as I mentioned in the figure 10 in the previous chapter the decision point is not mentioned in the standard transaction pattern. Before process to the PSD it is necessary to make clear picture on the decision points.

If we take the standard transaction pattern of the DEMO (figure 10), it is clearly illustrated an initiator request with the request acts and the executor receives that as request fact. Then the executor commits both declining and promising acts. This is the big weakness of the DEMO since with the diagram either a good programmer or ontologist cannot go to the next step by keeping the both acts at a time without decision point. *Here we introduce an empty diamond symbol in order to overcome the confusion stated.* So if the decision point is available in between the request fact and the

executor actor role then the executor makes the selection to move which path; either “promise” or “decline”. Further, the same problem occurs when the executor states his finishing his execution the initiator should accept it. But in the standard transaction diagram (figure 10) when we see that, again the initiator accepts and rejects the production result. The above said two activities cannot be happened in parallel. Again in order to overcome this confusion we introduce the same *empty diamond symbol* as decision point for taking the either of the decision but not both. The fixed standard transaction pattern is shown in the figure 17.

Figure 17 clears the confusion on decision points. According to the DEMO-E standard transaction pattern the initiator sends the request for the dealing as request fact and the executor decides whether the requested to be refused or promised. This can be applied again when the executor sends the product as state act, the initiator receives as state fact and initiator is a stable position to decide whether the product can be accepted or rejected but not both. And this added simple symbol definitely makes impact in the DEMO technology.

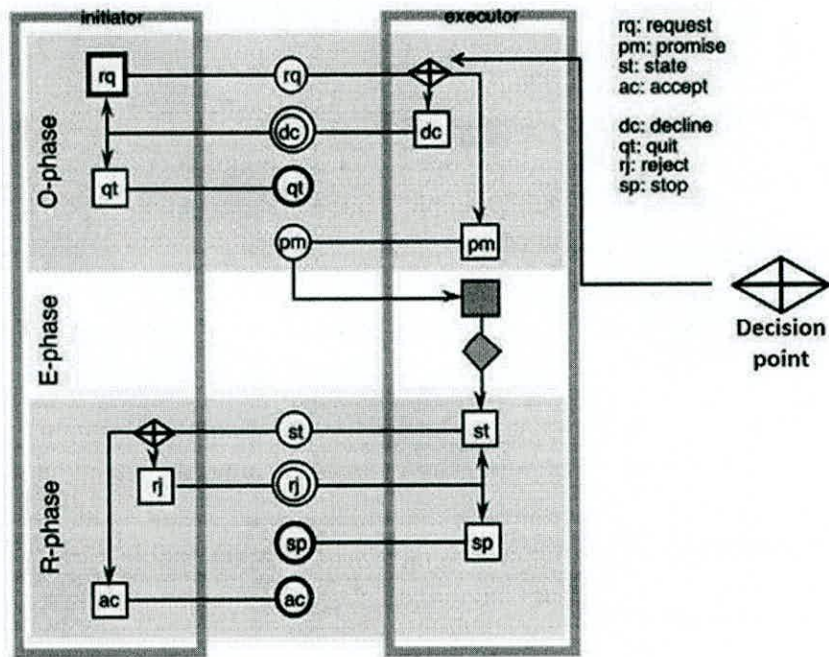


Figure 17 Standard transaction pattern of DEMO-E

In the above diagram it is clear that, both initiator and the executor can make good decision without confusion. Thus, programmers need not to search for another ontologist in order to understand the hardcore of the ontology model. Meanwhile it is important either the initiator or executor cannot take the both decision (dc and pm) at the same time.

Next we will get in to the PSD with basic transaction pattern. The figure 18 shows the transactions, elementary actor roles, composite actor roles, coordination activities (viz. request, promise, state and accept) and production acts. It is up to the designer to break the PSD in to more than one part for the easiness and the paper size.



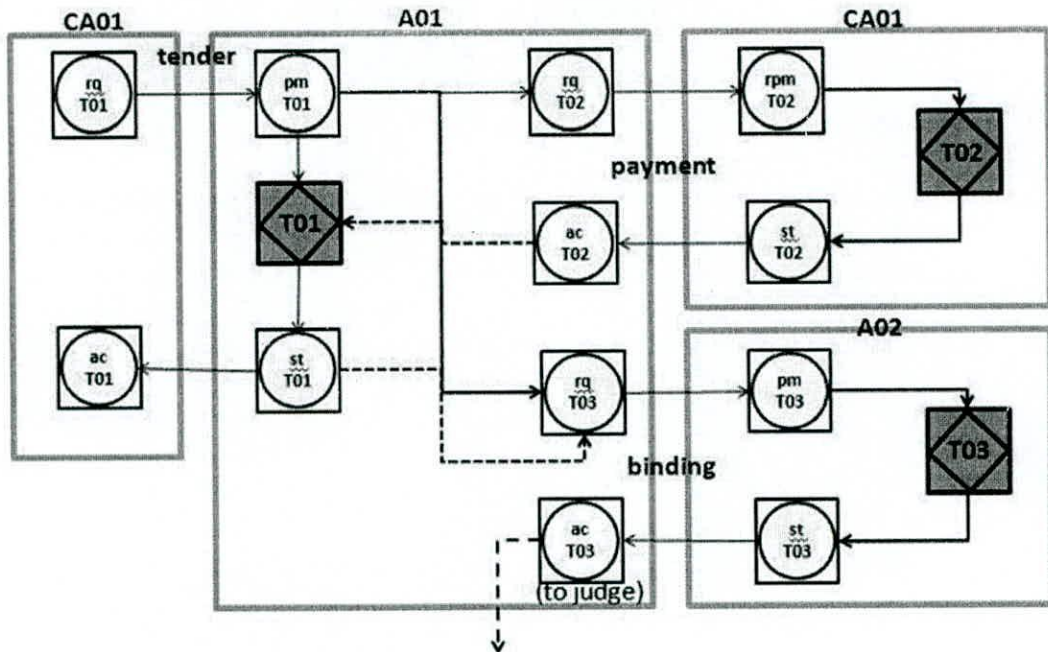


Figure 18 PSD I of District court (case filing)

According to the diagram given above in figure 18 we could see there are two parts in the case filing processes. Thus, we can draw accordingly. The PSD also become two parts according to the figure 18 in the ATD detailed. The part one of the PSD figure 18 illustrates the tendering, payment and binding processes. Meanwhile, the part two of the PSD shown below in figure 19.

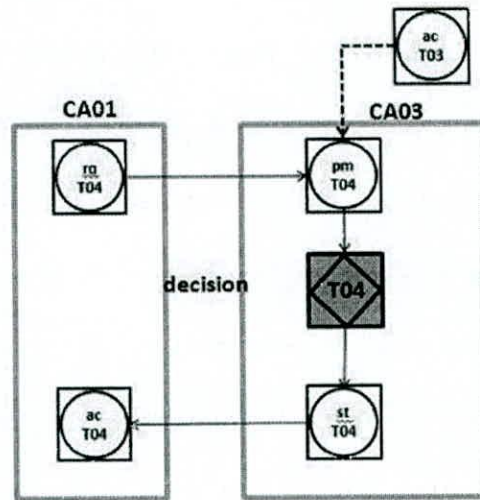


Figure 19 PSD II of District court (case filing)

As one can see in the figure 18, the standard steps in a transaction are connected to each other by a causal link. If we take the figure 18, composite actor CA01 initiate the transaction by requesting (rqT01) the transaction tender (T01) to tender the case documents. Actor role A01 (registrar), promise (pmT01) the transaction and initiate the payment transaction (T02) by the request (rqT02) to the composite actor role (CA01). Meantime, A01 waits for the T02 ends in order to finish the transaction (T02). Actor role (CA01) promises (pmT02) and start the execution of transaction (T02). Soon after the execution of T02, composite actor role (CA01) states (stT02) the result of the transaction T02. Actor role A01 receives the status (stT02) and accepts (acT02). The acceptance of T02 is necessary to process the execution of T01. So we can see in the figure 17, the acT02 and execution T01 is connected with the conditional link but not the causal link. With the acceptance of T02, the T01 is being executed and states the status (stT01). While requesting the rqT02 another request (rqT03) also initiated by the actor role A01. The executed status (stT01) of T01 is necessary in order to process the request to actor role A02. Again we can see in the figure 18, the conditional link goes

from stT01 to rqT03. Actor role A02 promise (pmT03) transaction and execute (T03). The status (stT03) is the result after end of the T03. The status (stT03) binding transaction is accepted by the actor role A01. Even though that status (stT03) accepted by the A01, that acceptance is *delegated* to the actor role A03. The figure 19 shows the second part of the PSD. Indeed a lawyer expects decision from a judge of the court regarding the filing of a case but not the final judgment in this stage. Thus we can state that, the diagram illustrated in figure 19, as main diagram since decision of the judge is vital part of the Courts system. If we closely look the figure 19, the composite actor role (CA01) requests (rqT04) the actor role A03 to produce the decision. So the actor role A03 needs to promise (pmT04) but in order to promise the transaction T04, the previous all the steps must be completed. That is why, the link between acT03 and the pmT04 is a conditional link. Once the actor role A03 accepts the statement of the actor role A02 then actor role A03 promises (pm/T04) and it goes for the execution of the T04. Soon after transaction the result is stated (st/T04) to the composite actor role CA01. By end, the composite actor role CA01 accepts (ac/T04). With the acceptance of T04 the case filing procedure comes to an end.

The above showed aspect model (process model diagram) is modeled with only basic transaction pattern. But it should be modeled using standard transaction pattern since any one can understand the transaction concept well. The following illustrations clearly show the standard transaction between different actor roles.

tendering – standard transaction diagram

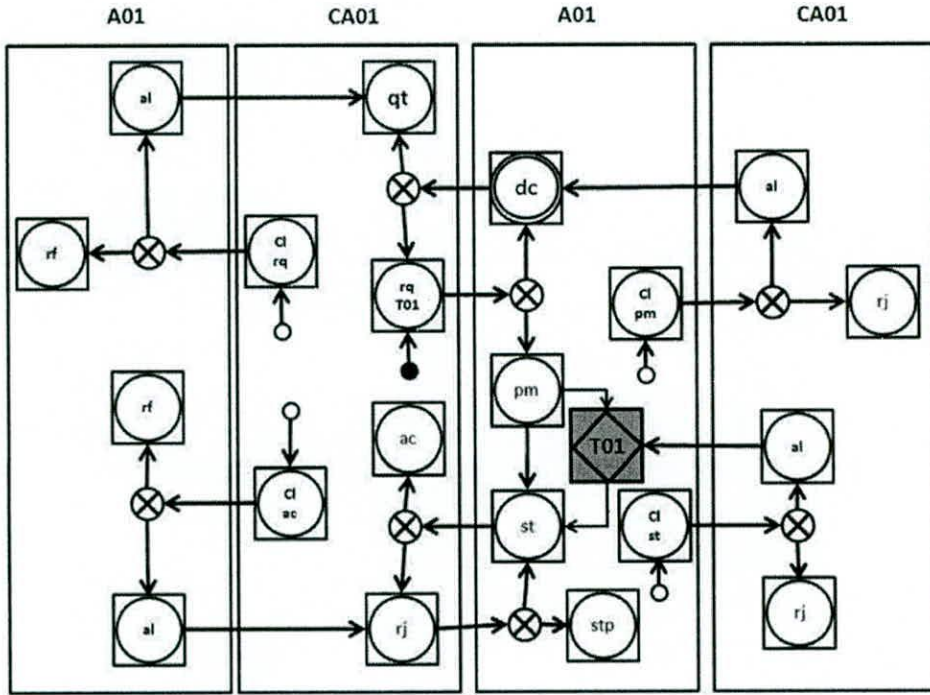


Figure 20 Standard transaction diagram for tendering

payment – standard transaction diagram

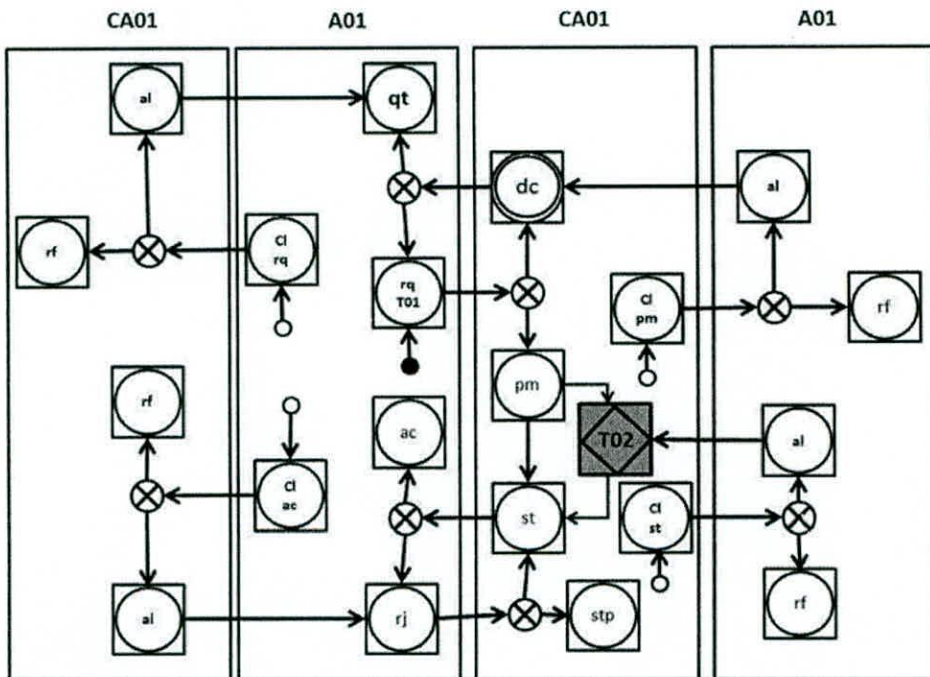


Figure 21 Standard transaction diagram for payment

*binding – standard transaction diagram*

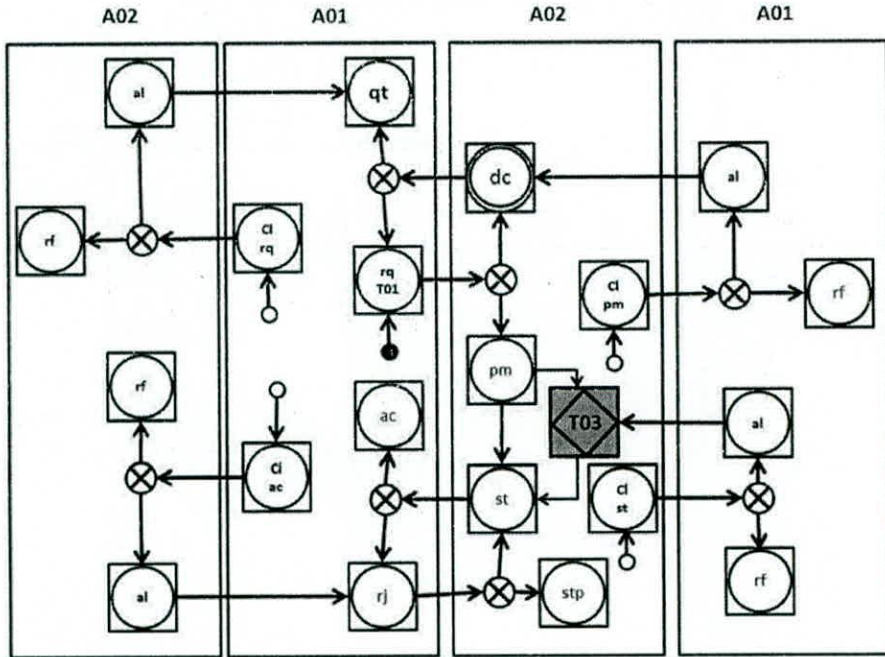


Figure 22 Standard transaction diagram for binding

*decision – standard transaction diagram*

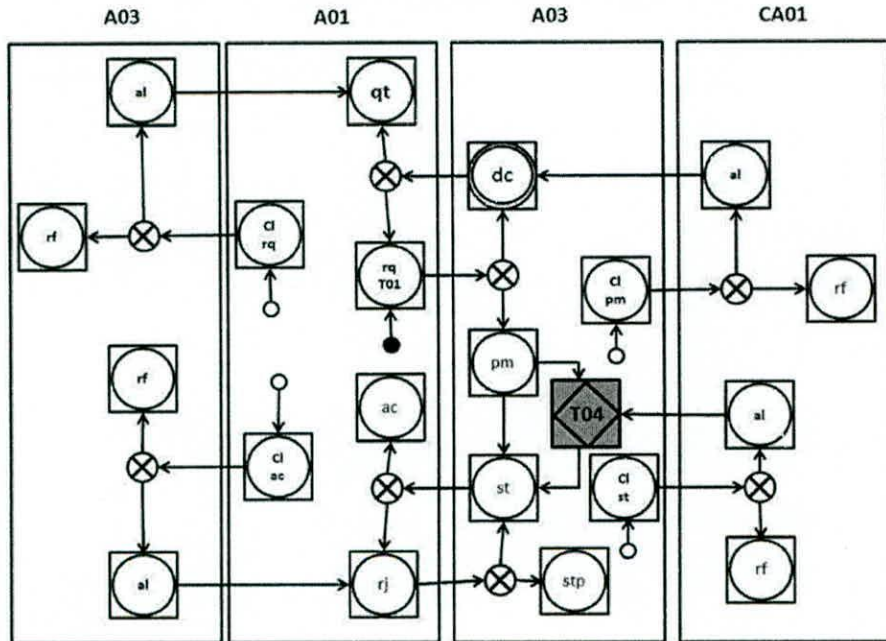


Figure 23 Standard transaction diagram for decision

A01: actor role 01    A02: actor role 02    A03: actor role 03

al: allow    rj: reject    pm: promise    rf: refuse    dc: decline    st: state

CA01: composite actor role    cl: cancel    ac: accept    rq: request

Basically a coordination action can find different ways in order to obey the certain condition. The request act is posted to the next following actor. The next actor has two choices before the execution takes place. The first choice would be decline (dc) where the necessary conditions are not met or, the promise state (pm) where the conditions are acceptable. If the request finds the path as decline (dc), then the request re-sent to the requesting person. At the initiating actor has two choice of declined request; either the request is made as re-request or the requesting actor role quite from that requesting action. It is possible to an executor to cancel the promise. Then the cancel-promise request is sent to the initiator. Initiator has two options either to allow (al) the cancel-promise request or refuse (rf) it. Once the cancel-promise is allowed by the initiator, then the executor sets as the decline state to the initiator. The cancel-promise request is refused then the process finds the normal path.

Once the request is promised then, it is sent to the execution state. Soon after the execution, the result is set state (st) stage. It is possible to the executor to cancel the state (st) in order to re-correction of the execution. Then the cancel-state request is sent to the initiator for the approval. The initiator can decide in to two ways either to allow (al) it or reject it. If it is allowed then the request is sent to the execution point and it will be re-executed. If the cancel-state rejected, then the request finds the ordinary path.

Once the result of the transaction is stated by the executor then the transaction is completed once the initiator accepts it. If not, the result is rejected and sent to the executor again. Executor has two options either to re-do the process and set the state as stated (st) or to stop (stp) the current transaction.

On the other hand an initiator can cancel the request. If he wants to cancel the request the initiator has to send the cancel-request to the executor. Executor either allows it or rejects the cancel-request. If the cancel-request is accepted by the executor, then the initiator quits (qt) from the transaction. If the cancel-request is refused by the executor then the transaction finds the normal path.

There is another possibility to cancel the accepted state. It is said that, the initiator launches the cancel-accept request to the executor. Executor decides either to refuse or accept it. If it is refused then that is the ordinary path of the transaction. If the request is accepted by the executor then, the initiator rejects (rj) the result leads either to stop the transaction or redo the result and state it again by the executor.

In a summary the standard pattern gives more detail about the “acceptance”, “cancellation”, “stop”, “refuse” or “rejection” on the different level of requests.

As this thesis suggested at the initial level, the process model (PM) is constructed with two components such as Process Structure Diagram (PSD) and Information Use Table (IUT). According to the DEMO model this IUT can be completed only after the state model (SM) has been completed.

Object, Class, Fact type or Result type	Process Step
CASE FILING	T01/rq, T02/rq, T03/rq, T04/rq
PERSON	T01/rq
payment has been paid	T02/rq,
binding has been performed	T03/rq
decision has been made	T04/rq
CASE TYPE	T01/rq
case data	T01/rq

Table 6 Information use table of District Courts (case filing)

### 3.6.3. The Action Model

The Action Model (AM) is the most descriptive and base level model. Its structure shows the atomicity on the ontological level. Indeed other three aspect models (PM, SM, and CM) are derived from the AM, even though the order of formation of the models differs. The AM of an organization consists of a set of action rules. The action rules guides the actor to do the actions accordingly. There are certain cases which actor role needs to deviate from an action rule. Eventually, the responsible remains to the actor for his or her acting. The pseudo algorithmic language is used for the action rules for the organization. The following informal definition rules are applied to the pseudo algorithmic language.

1. An action rule is enclosed by an on-no bracket pair. This on clause, as it is called formally, specifies the actions that are being dealt with in the first line, and what actions have to be taken for dealing with it in the other lines.
2. Conditional responses (choices) are represented by an if clause, enclosed in an if-fi bracket pair



3. If there is more than one choice, the second and following ones are preceded by the symbol “€”.
4. Every choice consists of the condition, which is checked to be true, followed by the symbol “≥”, followed by the action(s) to take.
5. Repeated actions are specified by **do** clause, enclosed in a **do-od** bracket pair. In the first line (after do) the number of repetitions is specified, usually, in an indirect way, like using variables, the values of which are inspected at runtime.

According to the guideline, the action rules for each actor along with the transactions are made.

*Action rule for the actor role A01:*

**onrequested** T01 (C) **with** tender (new C) = T

**if not** <necessary documents are available>**AND**<payment is done>

≥ decline T01 (C)

€ <necessary documents are available>**AND**<payment is done>

≥ **promise** T01 (C)

**no**

Description: This action rule specifies what has to be done if the agendum to be dealt with is a T01 being requested. At initial, the generation of the entity is being done with the type of case. We have selected to process with the divorce case filing at the District court of Kalmunai (flow chart of the case filing is under annexure III). Then the actor role A01 checks the necessary documents (motion, proxy, necessary payments, with stamp duty along with stamp, plaint, necessary evidence documents such as marriage

certificate; birth certificate including parents and children; medical certificate; police report; report from the family counselor; summon; copy of complaint and field documents; precept; and stamp together envelope. If the composite actor role (CA01) fails to submit either the above said necessary documents or the payment to the Courts then the transaction one (T01 – tendering) will be declined or rejected by the Courts. Or the necessary documents and the payment to the Courts are accepted by the actor role one (T01) then the T01 will be promised by the A01. And the rest of the orders are come with the promise of the request made by the composite actor role CA01.

**onpromised** T01 (C)

request T02 (C)

**no**

Here once the actor role promised the transaction one (T01) then the actor role A01 requests for the Courts payment. Thus, the actor role expects the correct payment in order to precede the T01 after T02 completes by the composite actor role CA01.

**onstated** T02 (C)

**if not**<payment is acceptable>

        ≥ decline (T02)

**if**<payment is acceptable>

        ≥ accept (T02)

**no**

Soon after the execution of T02, the result would be stated (st/T02). With the appearance of st/T02, actor role A01 checks whether the payment is made correctly (amount) or not. If the payment is validated and acceptable by the actor role A01, then actor role A01 accepts the T02.

**on**accepted T02 (C)

execute (T01)

state T01 (C)

**no**

Once the payment transaction (T02) is accepted, the actor role A01 executes the T01 and state (st/T01) the result to the initiator of the T01.

**on**stated T01 (C)

request T03

**no**

With the display (st/T01) of transaction (T01), the actor role A01 requests (rq/T03) for the binding transaction to the executor.

**on** stated T03 (C)

**if not**<bound case document correct>

≥reject T03 (C)

**if**<bound case document correct>

≥ accept T03 (C)

**no**

Soon after the binding transaction, the binding role posts (st/T03) the result to the initiator (A01) of the transaction (T03). If the bound case record is correct and acceptable then that transaction (T03) is accepted (ac/T03) otherwise it is rejected (rj/T03).

*Action rule for actor role A02:*

onrequested (T03)

**if not**<necessary case documents in order>

≥ decline T03 on (C)

**€** (necessary case documents in order>

≥ promise T03 (C)

**no**

These set of rules belongs to the actor role A02. Actor role A02 denotes the binder of the case document as the case record. With the request (rq/T03), the actor role (A02) checks the case documents for the completeness. If the case documents are complete then actor role promise (pm/T03) the transaction (T03) else, declines (dc/T03) the transaction (T03).

onpromised T03 (C)

execute T03

state T03 (C)

**no**

If the transaction (T03) is promised (pm/T03) by the actor role A02, soon the execution of the transaction (T03) happens. End of the successful transaction (T03) the result (case record) is stated (st/T03).

*Action rule for actor role A03:*

**on**requested T04 (C)

**ifnot** accepted T03 (C)

≥ decline T04 (C)

**€**promise T04 (C)

**no**

The actor role A03 denotes the judge of the District Courts. If the transaction (T03) is accepted (ac/T03) by the actor role A03 then the new transaction (T04) is promised (pm/T04) else the new transaction T04 will be declined (dc/T04).

**on**promised T04 (C)

execute T04

state T04 (C)

**no**

Once the transaction is promised (pm/T04) by the actor role, then the execution of transaction (T04) starts. End the execution the result would be posted (st/T04).

#### **3.6.4. The State Model**

The State Model (SM) of an organization is the specification of the state space of the P-world. The SM is described with specifying the object classes, fact types, and the result types, as well as the existential laws that hold. It is built up with an Object Fact Diagram (OFD) and an Object Property List (OPL). The OPL is a table where specifying fact types that is proper (mathematical) functions, and of which the range is a set of values. The fact type in an OPL is called as properties of object classes. The OFD is designed by World Ontology Specification Language (WOSL). It is possible to get object classes, fact types, and result types, and all pertaining existential laws along with all derivation rules, in the action rule (Action Model) specifications. Thus, the contents of both the OFD and the OPL of an organization are completely determined by its action model. This makes the state model of an organization more or less closely related to the objective model. Apart from the above said factor we need to include the information items which are relevant for the operation of the organization. This is in sharp contrast to the current practice in requirements engineering, in which the information wishes of users are collected.

With the explanation presented above, we can develop the state model of the case filing at the District Courts. Prior to the development of it is necessary to present the object classes, fact types, and pertaining existential laws. The said properties are displayed in the table-7 given below.

property type	object class	type / scale
date of tender	CASE FILING	DATE
deliver address	PERSON	ADDRESS
divorce case	CASE TYPE	TEXT

Table 7 OPL of District Courts (case filing)

In the above figure 24 given below, there are three core categories namely PERSON, CASE FILING and CASE TYPE. These three categories almost cover the whole filing processes in different places and ways. There are two fact kinds called F01 and F02 respectively. F01 denotes that, case filing C is the case type of CASE FILING has a person P which is the type of PERSON. The person P depends on case filing C since, it is necessary that, if there are any case filing there should be a person but each person need not to have the case filing. And the other fact type F02 denotes that, each case filing (C) has a case type (T). The dark dot is states the dependency law and the small dash on the C is unicity law. There are four result types called R01, R02, R03 and R04 respectively which are already mentioned in previous chapters.

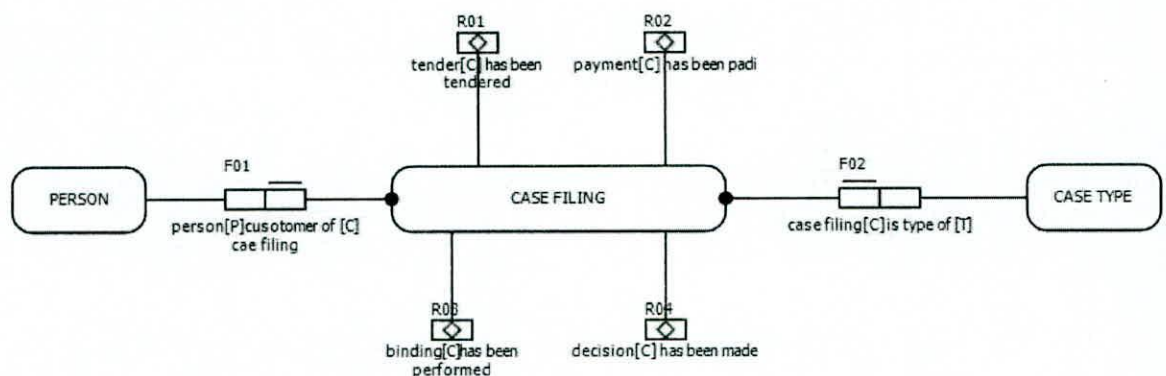


Figure 24 OFD of District Courts (case filing)

### 3.6.5. The Data Model

The next aspect model is data model, which describes how the data is used during the transaction even sub level transactions such as (request, promise, execute, state and accept). Later in the interstriction model, we can find the description about the databanks (storages) which are used to either for retrieval purpose or storing purpose. The access of the databanks are very much occur all coordination levels except acceptance act since it is just a matter of acceptance the result of the transaction interested. Coordination acts such as request, promise, and state access (read or write) the databanks massively. The meta-model of the data model is given below in the figure 25.

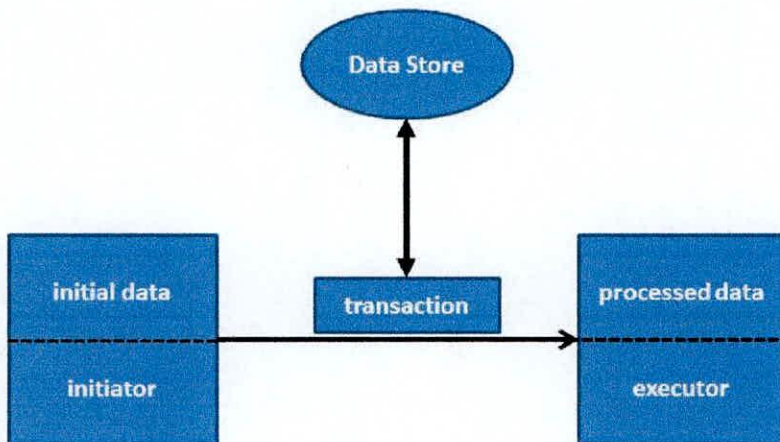


Figure 25 Data transaction meta-model

The data meta-model simply describes the way the data is being transferred from one destination to another with the involvement of actor roles and the transactions along with place where the data is accessed (read or write). The solid arrow line denotes the direction where transfers happen. And the dashed line denotes the conditional data links (transfers). The double arrow denotes that, the data access direction either way or both, it totally depends on the transaction being processed.



The data model can be traversed in to one level depth. If we take the total transaction as we know there are two major acts namely production and coordination. There are four coordination acts involve for a single transaction to be completed. The following model describes the coordination level data access (read or write) level.

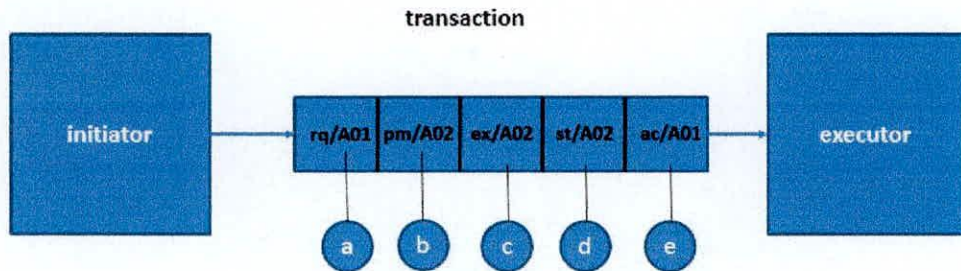


Figure 26 Coordination level data access

As mentioned in the earlier paragraph, there are couple of actor roles namely initiator and executor. In the above figure 26, A01 denotes the initiator and A02 denotes the executor. This diagram illustrates in which place exactly the data is accessed (either stored or retrieved) during the transaction in between two actor roles. There are five different stages of the transaction including two acts (viz. production & coordination). The “a”, “b”, “c”, “d”, and “e” denotes the data stores (data banks) which is accessed by the relevant acts. This access is either retrieval or store.

The diagram given below shows the depth level illustration how the data is being moved and what are the data stores is accessed by the both transactions and the actor roles. If we take on the figure 27 given below, the initiator of the case filing CA01 actor role initiate the transaction T01 with the data d0. When the transaction begins the transaction T01 needs either read or write the data from the CPB01, CPB02 and CPB04. Meanwhile the transaction T02 also starts in parallel to the T01. T02 needs to access the

CPB01 and CPB04 for the execution and the payment data is d01. T01 starts to execute when the result of the transaction T02 reaches. In T02, payment data is sent to the execution of the transaction T01. The payment data (d01) and the case tendering data (d0) is collected and processed. End of the transaction T01, the resultant data would be case document data (da). The data (da) is sent to the binder (A02) to do the transaction T03. The binder collects the information (data) and bind as the case record; but the data still remains same as case record (da). The case record data (da) is sent to the actor role A03 in order to get the approval for the case to be filed. Then the end of the transaction T04, actor role A04 delivers the order either for the approval or reject. The data turns to the new direction as decision data (db).

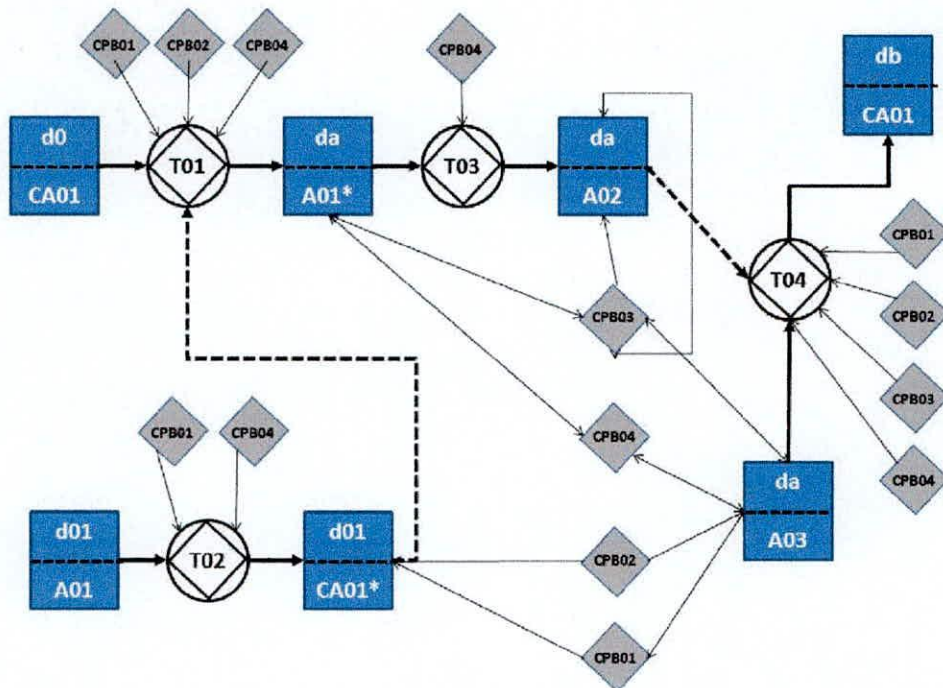


Figure 27 Data model of District Courts (case filing)

Meantime, the data banks are shown in the above figure 26. CPB01, CPB02, CPB03 and CPB04 are data banks (data stores) and defined by client data, judicial laws (case

law and statute law), Courts data and case data respectively. The arrow toward the data banks denotes for the retrieval and towards the databanks denotes storing information to the data store.

### 3.6.6. The Interstriction Model

The Interstriction Model (ISM) is just below the level of the Ontology Construction Model (OCM) and which consists of Actor Bank Diagram (ABD) and Bank Content Table (BCT). Here the transaction symbol is now interpreted as the combination of the production bank and the coordination bank. More than the Interaction Diagram (IAM) here additionally the information links added in between different objects including actor roles and the transactions along with external composite production banks (CPB). In order to model the Actor Bank Diagram (ABD), it is necessary to prepare the Bank Content Table (BCT). The BCT consists of object classes, fact types or result types and the identified production banks. The table 8 given below is the BCT of the case filing system.

object classes, fact type or result type	p-banks
tender has been started	PB01
payment has been paid	PB02
binding has been performed	PB03
decision has been made	PB04
PERSON	CPB01
delivery address	
CASE TYPE	CPB02
CASE FILING	CPB03
case data	CPB04

Table 8 BCT of District Courts (case filing)

According to the bank content table (BCT) and the Interaction Model (IAM) we can model the interstriction model. The Actor Bank Diagram (ABD) model is illustrated below in the figure 28.

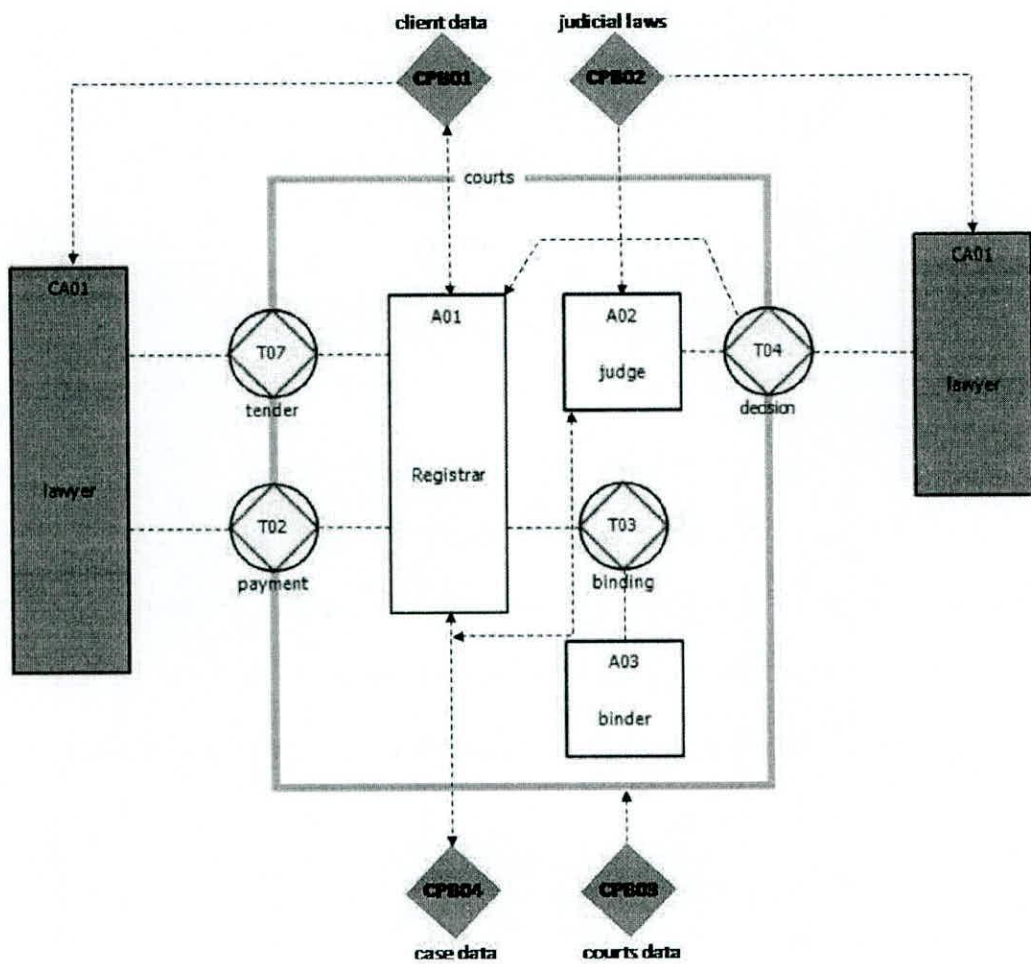


Figure 28 ABD of District Courts (case filing)

The ABD describes about the information links, information flow, and the data banks where the data is either stored or retrieved apart from the interaction model which was described early in this chapter. Here the data flow is additionally enhanced in order to understand the data flows from and to. Earlier DEMO does not include this data flow. It was the big gap in between the ontologists and the programmers. But in this Actor Bank

Diagram (ABD), we proposed to display the information links flow as retrieve, store or both. The client data Composite Production Bank (CPB01) issued by the actor role CA01 and A01. The Case Data Bank (CPB04) is accessed and stored by actor role A01 and A03. Legal rule data bank (CPB02) is accessed by the actor roles A03 and CA01. Meanwhile common Courts data bank (CPB03) is accessed by all Courts staff involving the case filing system. Further, actor role A01 needs to retrieve the information of the decision transaction T04 in order to get the relevant information. The Ontology Construction Diagram is the final version of the whole ontology aspect model is shown below figure 29.

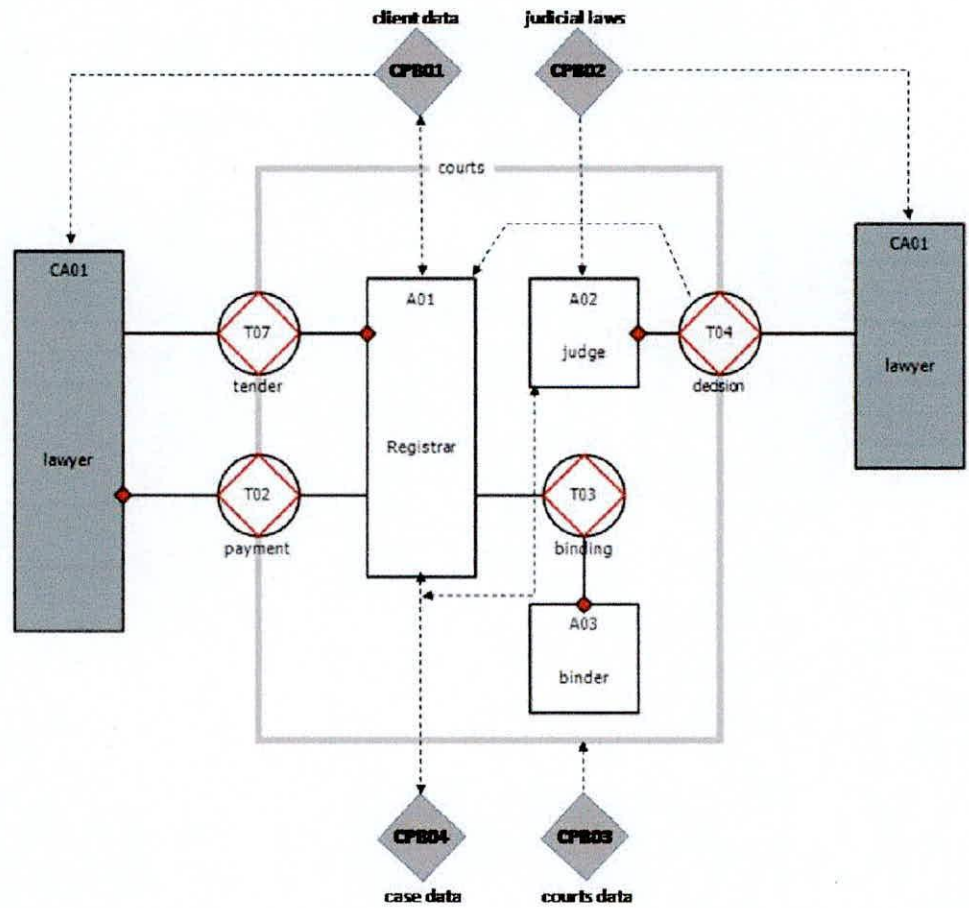


Figure 29 OCD for District Courts (case filing)

We can gather necessary information from the ontology construction diagram (OCD) of the case filing model above illustrated in the figure 29. It is clear that the main difference in between the ABD and the ontology construction diagram is appearance of the initiator and executor links. By recalling previous interaction mode, the red spotted beginning is the executor and the other end is the initiator of the transaction.

## CHAPTER FOUR

### THE SUPPORTIVE ENHANCEMENTS

This chapter describes the enhancements which were applied during the course of development of the ontology model for the case filing at the District Courts in Sri Lanka. During the application of the DEMO in the legal domain, we had very significant problems in different places that we could not answer with the existing DEMO ontology. These enhancements were introduced and used in DEMO-E in order to fill the gap of DEMO while it is being used. It is described in different sub headings below.

#### **4.1. Decision point at the standard transaction model**

As this thesis discussed in the previous chapter (chapter three), there was a big gap in the decision point at the standard transaction pattern. The given figure 10 describes the activities happening in between the request coordination act and the acceptance coordination act. Once a programmer examines the standard transaction diagram he is not able to understand the four important decision making points in earlier DEMO methodology as given below.

1. In the order phase, initiator request for the transaction. According to the standard transaction diagram the promise (pm) and decline (dc) happens at the same time.
2. Again in the order phase soon after the decline (dc) of the request made by the executor, the initiator decides to do the both request (rq) again and quieting (qt) from the transaction not or condition.

3. Soon after the execution of the request made by the initiator, the executor states (st) about the transaction, which is done. Then the initiator again does the same mistake by both accepting (ac) and rejecting (rj) the transaction.
4. Once the statement of the execution is rejected by the initiator, then the executor roll back the result and does the both acts stating again and stopping (sp) the transaction.

Furthermore it is clear that, if there is a branch of two ways, it is impossible to do the transaction in the both ways at the same time. It is the hard mistake and should be amended by introducing the decision symbol in order to clear the confusion among the both ontologist and the ontology programmers. By applying the decision symbol when necessary it clearly describes that either of the activities can be carried out. This is one of the major improvements made by the thesis during the cause of the research.

#### **4.2. Recognizing the duplicated actor role**

In general this actor role duplication seriously affects the Process Structure Diagram (PSD). Suppose of the PSD is a huge and complicated model with many processes then the either programmer or the viewer of the ontology experience very much difficulties to find whether the particular actor role is used somewhere else either before the current level of inspection or after the current level of inspection. We introduced an asterisk (“\*”) symbol on the side top of the actor role which is going to be duplicated. By this any user can understand the particular actor role going to be duplicated either two or more places. Therefore the viewer can scan the diagram with the positive mindset of expecting the duplication symbol appeared actor role.



### **4.3. Direction of the Information access link**

One of the next major gaps in the existing DEMO while it was being applied to the legal domain is, especially in the latter part of the aspect model formation. The Ontology Construction Diagram (OCD) is the last and informative abstraction ontology model in DEMO. When we analyze the OCD there are some information links make link in between external composite production bank, actor roles, and transactions. The last two models such as actor bank diagram (ABD) and the OCD deal with the information links which is mentioned. The information link used to show the data transaction. But in the current DEMO ontology it is shown but only the information links but not the direction of data flow. It is hard to follow which direction the data flows between data banks or actor roles or transactions. And it is hard to identify whether the data is either retrieved or stored during the actor role's actions. We enhanced the links by providing the direction of the data flows. It is untied the doubtful states of the data flows.

### **4.4. Data Access Level toward Data Model**

If we take any transaction it is hard to find in which state the data is being used either storing or retrieval or both. The newly introduced data model is giving fine solution in order to over this problem. The data model describes about the transaction, actors and the data store access with either read or write or both. With the data model one can easily understands that, in which level the data can be accessed. The further detail is given in the chapter 03 under the title of data model.

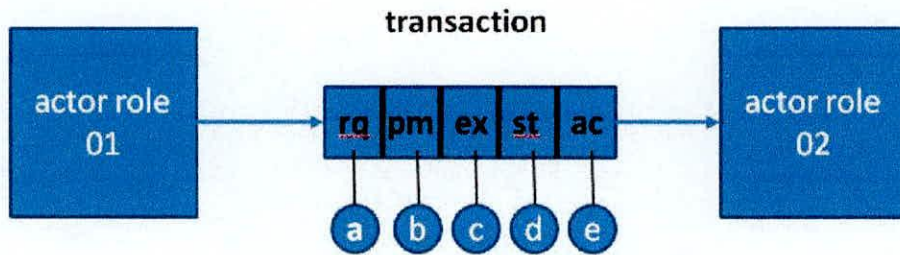


Figure 30 Data access model

#### 4.5. Introduction of DEMO-e Transaction Meta Model

The introduction of demo-e transaction meta-model is highly supportive to the users of the demo-e since it abstracts the way of usage, the outputs, and interactions. This meta-model can be instantiated and generate the demo or demo-e. This model can be applied any transaction kind and it perfectly matches to the situational scenario. The model is given below in the figure 31.

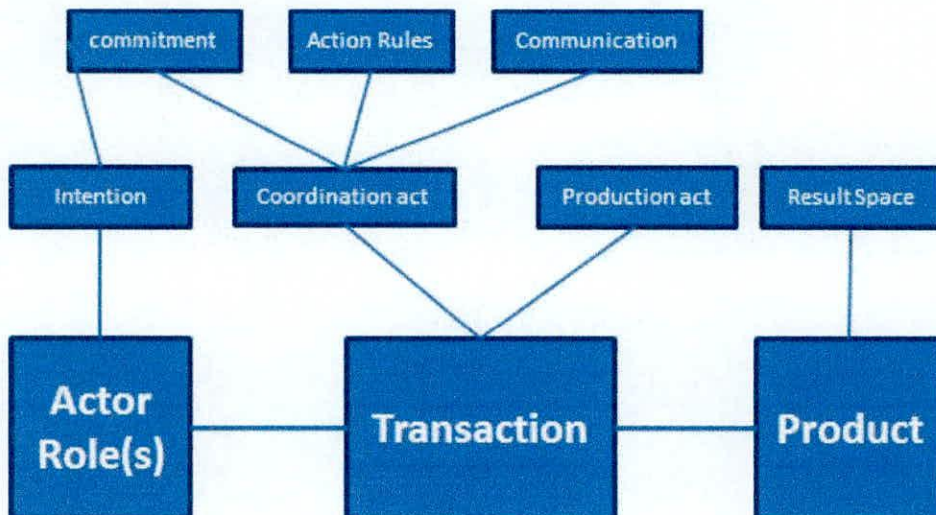


Figure 31 DEMO-E transaction meta-model

Any complete transaction has three components such as actors, theme transaction and product. Actor role actively participates with the intension and the commitment. The

real transaction is with the major two components namely production act and coordination act. The coordination act is fulfilled with commitment, communication and action rules. Product or the result is the output of the transaction. This is easier when one tries to apply the model in order to populate the transaction model.

The above stated enhancements are very useful in order to reduce complexity and confusion; and easiness to design ontology framework for the legal domain.

## CHAPTER FIVE

### DISCUSSION

This chapter discusses the proposed and developed LSPM and DEMO-E along with the aspect models with the respect of research question.

#### 5.1. General Discussion

The discussion is clearer by recalling the research question. The research question is “*Unavailability of ontology framework to facilitate LSPM in efficient and productive manner*” The research question is answered by the invention LSPM through DEMO-E, with the development of the aspect models namely Interaction Model (IAM), Process Model (PM), Action Model (AM), Data Model (DM) and Construction Model (CM) with inclusive of Interstriction model (ISM). Each model serves different part of the research questions. As this thesis mentioned the research question can be divided in to three sub questions.

The first sub question (*develop the LSPM framework by the aspect models*) is answered by the development of the all aspect models with the DEMO-E methodology ending up with the construction model (called Ontology Construction Model) shown in figure 29 in the third chapter which supposed to be the ontology model of the Courts case filing scenario.

And the third question (*application of these aspect models to the legal domain*) is fulfilled by applying all these models in the case filing scenario at the District Courts. Filling a case passes many different steps including tendering, payments, binding and

decision as the main transactions. These different transactions can be viewed by all five aspect models in different viewpoints. The rules and regulations to the actor roles at the Courts is viewed by the Action Model (AM), where each actor role has to follow certain Courts procedures in order to do their dedicated work perfectly. These rules are shown in the action model. The overall abstracted picture of the dealt system is viewed by the interaction model with the Actor Transaction Diagram (ATD) and transaction result table (TRT) by showing the actor roles, relevant transactions, the system boundary, and whether the actor role acts as initiator or executor. The state model describes the state of the selected system both input state, result state and necessary object classes by the object fact diagram (OFD) and Object Property List (OPL). The Data Model (DM) shows the transaction of the data and the level of access by the actor roles. It is very important to the ontology designers and the ontology programmers to know the flow of the data in order to reduce the confusion. And finally the total ontology of the case filing scenario at the District Courts is illustrated by the Ontology Construction Diagram (OCD) along with the InterstrictionModel (ISM) and the Actor Bank Diagram (ABD). More than the IAM, the ISM shows the data access (either read or write) from the external data banks (data stores)

The second question (*development and enhancement of the aspect models develop the LSPM framework for the legal domain*) is answered by the application of these aspect models in the legal domain and the benefit (process management) gained from the aspect models by the following descriptions.

Altogether, we modeled five ontology aspect models by applying the DEMO-E to the case filing scenario at the District Courts of Republic of Sri Lanka. The existing DEMO methodology is mostly suit to the e-business community but not in legal domain since the legal domain system shows many deviations from general e business systems; and the e-business modelers. We made the DEMO-e with enhancements in order to develop LSPM framework in the legal domain. Next few paragraphs briefly describe about the outcomes of the all aspect models that we modeled (ontology framework model) for the legal case filing.

The Interaction Model (IAM) is very much abstracted ontology model to show up the brief summary of the selected system. The IAM shows the boundary of the organization, interface transactions with actor roles in the environment. And it is possible in a single view; we can see the initiators as well as the executers. This makes the IAM preeminently suitable for strategic alignment since it is ideal starting point for system redefinition, the most dramatic type of organizational change. Since the wholeness of the transaction displays the transaction pattern and facilitates the attention for customer care. Again the detailed IAM, it is very transparent to understand about the processes within the court system. In a way the IAM shows the invisible responsibility to the viewers. In the detailed IAM one subsequently identifies what has to be done in component transactions to fulfill the case filing at the District Courts. IAM shows the ontological units of competence, authorization and responsibility. In addition, the IAM can be fitted and one looks as the total model in a small piece of paper since the high level of abstraction.

If we take the process model (PM) of demo-e, it shows the high quality of viewing deep structure of the case filing procedures. Thus, this approach can be applied even in other sub legal domain needs since its independent behavior of its implementation toward the ontology framework. And also no distinction can be made between data logical, info logical and ontological actions. This unique property of the PM paves the way for a number of different interesting sub applications in the legal system. Apart from this, the PM facilitates discussions about the redesign, as well as the reengineering of Courts processes towards the process optimization by providing flexible redesigning structure. Further, this PM is straight forward structure to understand the process in the working system rather than any other technique. Thus, it makes easy to ontology programmers to work with. Even from the PM it is possible to start the requirements engineering regarding supporting information systems since the ontological requirements are clearly visible and can be re-engineered. Even the court processes can be promoted as object oriented development view. This model leaves no room for unnecessary requirements since the PSD and IUT cover almost all necessary requirements.

The complete action model (AM) is essential in order to enforce the rules belongs to actor roles and how the actor roles be reacted with the rules. The action rule is necessary to produce the correct state model. Thus it provides full account of operational decisions in the Courts processes leaving non-essential matters. The  $\Psi$  (PSI) theory highly supports to understand the action rules well.

The state model (SM) supports to develop and maintain the data dictionary of the District court case filing system since it delivers the stable core of such a data

dictionary. More than this, it is possible to get the error free best concept model and the ideas to construct the good concept model. We can easily find the categories in the state model and its properties such as tendering, payment, binding and decision as result types. Furthermore it is possible to simply find the data ownership since already we know the connections of the result types with the transaction types.

ISM adds to the benefits of the IAM is that it shows at the same level of compactness the complete 'passive' system structure, i.e., the information links between actor roles and banks. In addition, it contains the external banks to which the court system needs to have either read or write. The ontology construct model (OCM) is the most compact and informative diagram with actor roles, transactions, data access either read or write from external data banks (vitz. Courts rule, case data and etc.) and information links in between actor roles and transactions. By visualizing the total compact diagram (OCD), the ownership of the data is opened and transparent. i.e each fact is the result of a transaction, two actor roles such as initiator role and the executor role. Any one will be the owner of the fact. Furthermore it is visualized that, initiator always starts the transaction with the requesting act and the executor execute it by checking necessary conditions.

In addition to above discussion, an important turning point is the meta-model for the transaction which is mentioned in the chapter five. That transaction meta-model can be applied not only in legal sub domain but also in any domain with confidence since the condition of the meta-model is made common for all the domains.



In addition to the work we have done, we faced some constraints during the course of the research that was carried out. This ontology model is optimized in different places and one of the places is to deal with the case tending with the registrar (actor role A01). This is, in a way, to optimize the time constraint between the clerk role and the registrar role. This ontology model reduces the time of being the case document to be tendered on the table of the clerk. Since, in our ontology framework model, the client/ lawyer (composite actor role CA01) directly deals with the registrar actor role (A01) then there is a possibility of losing the job of the clerk. Meantime, if we take the clerk role, it can be delegated as the computer system. Thus, all the case documents would be updated or uploaded by the lawyer through a well-structured Courts case filing system. Even here also the clerical role would not be used. This is a big disadvantage for the clerk role since he /she may lose his/her job.

While, in the development of the cancellation pattern, some places the cancellation pattern is mismatched with the Courts procedure. But we have recommended additional cancellation patterns in different situations. This deviates from the procedure of the Courts but it is possible to be added in the Courts system. In conclusion, it is important that the ontology can be used for the betterment of the District Courts system.

## **5.2. Application and Evaluation**

A general case evaluation was done in order to compare the application of generic business process models and LSPM on legal system in different situations.

Generic business process models are applied in the binary collaborative parities whereas legal domain in most of the places engaged with multiple party collaborative actor roles.

The LSPM designed can be applied in both binary and multi-party collaborations.

Business enterprises are highly profit oriented organizations and the concept behind the generic models would be economic basis meanwhile, legal system is service oriented state organization. The main concept behind the LSPM is service oriented with non-profitable.

The business generic process models is high with business rules, which is fixed and pre-defined processes with no changes but in legal system on the fly it can be changed. Thus the generic BPM cannot be applied to the legal domain and LSPM would show the strong room for adaptive, flexible and dynamic processes.

The generic BPM cannot be applied certain stages in the legal domain which causes incompatible error such as decision as considered as a total transaction in legal system but in the business domain it cannot be applicable. The LSPM ensures to overcome the problem stated both material and immaterial outputs.

While applying the generic BPM in legal system it is possible to get the partial correctness but not fully acceptable result due to lack of flexibility, and the applicable

scenario. The LSPM set for both legal and many other domains including business sectors even.

The generic BPM can be applied only on the business domain since it is bundle with business oriented strict rules and business transaction laws. The legal domain is free from business transactions and generic PBM cannot be applied over the legal domain. The LSPM can be successfully applied on the legal system since it is specially redesigned for the legal sector and further it can be applied to the other domains too due to its flexible adaptiveness to the domain interested.

The legal processes are complicated and the suitable model should be applied in order to understand the process happening within the domain. It is getting worse and hard to understand processes if we apply generic business rules to the legal system. The LSPM simplifies the system and to see the system in five different angles with five aspect models.

## CHAPTER SIX

### CONCLUSION AND RECOMMENDATION

This chapter describes the milestones which were achieved during the research work and the major advantages of the ontology framework model of the case filing at the District Courts of Sri Lankan legal system in the general conclusion and the future recommendation is under direction for future research.

#### 6.1. General Conclusion

This research was set out to organize the Courts processes and processes optimization in order to reduce the time consumption between case (divorce) tendering and case acceptance at the District Courts of Sri Lanka. The Kalmunai District court was selected for my research work. The Courts proceedings were analyzed by direct visits to the Courts (hearings), meeting Courts staff, lawyers; related previous research findings; literature; books related to case filing; and judicial laws of the separations (divorce).

We have been noticed many mismatching concepts such as actor role identification, data transaction, and meta-model concepts when we applied the existing DEMO methodology. The new methodology called DEMO-E was introduced in order to overcome the problems stated above. With the introduction of DEMO-E methodology, modeling of ontology aspect models along with meta-model for the standard transaction and application of this ontology aspect models in the legal domain the research question and the all sub questions have been answered.

One of the research parts is to optimize the case process. One way of achieving the optimization is make the case filing automated system and still the necessary selected actor roles and the authentication with response cannot be demolished. But when we develop this concept model with the automated system there is fare possibility to lose the job of the Courts clerk and the role of the clerk will be delegated to the computer system. If the clerical actor role is removed or replaced by either automated system or direct dealing with registrar of the District Courts then registrar actor role needs to check each and every document since he /she fill the gap of clerical actor role. It may be additional work load to the actor role registrar.

Furthermore, the LSPM shows sound ontological base since it is well organized communication oriented design where the communication is one of the major area in the legal domain; having solid CAP framework with description of the activities happening two different worlds (coordination and production) with the actor roles engaging with the worlds; well-structured analysis and synthesis approaches to do the analyzing and synthesizing the results; with addition to the another aspect model called data model with existing aspect models leads to enhanced DEMO methodology to design the LSPM framework; and adoption and integration of different components ensures complete and correct LSP solution.

## **6.2. Direction for future research & Recommendation**

### **6.2.1. Concept model for the life of the case**

This research shows only a first part of the total life cycle of a case since a case has case filing, hearing, case move and verdict as its component. It is important to continue this development of ontology framework conceptual model to whole life of the case in future since it covers a complete case and necessary optimization processes in order to reduce both the complexity and life time of a case (time management).

### **6.2.2. Automate the System**

In addition to this, it is highly appreciated to convert the Sri Lankan judicial system in to an automated system to avoid unnecessary delays in overall cases. A new diversion can be made with the introduction and implementation of the “process marts” (Kariapper and Jayaweera, 2014, Kariapper and Jayaweera, 2013) to retrieve the process bundle and the case records through electronically rather than the manual processes.

### **6.2.3. Structure Re-engineering**

Concept model re-engineering is one of the major advantages in this DEMO-E methodology. It is possible for the conceptual modelers to design different conceptual design without changing the internal structure. It is recommended to direct the research towards the new conceptual legal ontology model without changing its structure in order to obtain the new structure according to the preference of the judicial system of the selected government of the country.

The necessity of an ontology framework model for the case filing at the District court of Sri Lanka was realized via the research question and it was successfully achieved by invention of DEMO-E methodology and the series of ontology aspect models.

## REFERENCE

(TMG), T. A. M. G. 2006. UMM Meta Model – Foundation Module Version 1.0.

AHLSÉN, M., PELKONEN, H. & WALSETH, S. 1994. *Concepts and notations for open-EDI scenarios*, Svenska Institutet för Systemutveckling.

ANDERSSON, B., BERGHOLTZ, M., EDIRISURIYA, A., ILAYPERUMA, T., JOHANNESSEN, P., GORDIJN, J., GRÉGOIRE, B., SCHMITT, M., DUBOIS, E. & ABELS, S. 2006. Towards a reference ontology for business models. *Conceptual Modeling-ER 2006*. Springer.

BENCH-CAPON, P. R. S. V. A. T. J. M. *Ontologies in the Design of Legal Knowledge Systems; Towards a Library of Legal Domain Ontologies* [Online]. Available: <http://ontology.buffalo.edu/FARBER/visser.html> [Accessed 13.05. 2014].

BRAIDOTTI, R. 2006. Posthuman, All Too Human Towards a New Process Ontology. *Theory, Culture & Society*, 23, 197-208.

DIETZ, J. L. 1999. Understanding and modelling business processes with DEMO. *Conceptual Modeling—ER '99*. Springer.

DIETZ, J. L. 2001. DEMO: Towards a discipline of organisation engineering. *European Journal of Operational Research*, 128, 351-363.



DIETZ, J. L. 2006. *What is Enterprise Ontology?*, Springer.

DIETZ, J. L. & HABING, N. 2004. The notion of business process revisited. *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*. Springer.

FILIPOWSKA, A., KACZMAREK, M., KOWALKIEWICZ, M., MARKOVIC, I. & ZHOU, X. Organizational ontologies to support semantic business process management. Proceedings of the 4th International Workshop on Semantic Business Process Management, 2009. ACM, 35-42.

FOLMER, E. & BASTIAANS, J. 2008. Methods for design of semantic message-based B2B interaction standards. *Enterprise Interoperability III*. Springer.

FOX, M., BARBUCEANU, M., GRUNINGER, M. & LIN, J. 1998. An Organizational Ontology for Enterprise Modeling. MIT Press.

FOX, M. S. 1998. The Tove Project, Towards the Common Sense Model of the Enterprise.

GORDIJN, J. & AKKERMANS, H. 2001. Designing and evaluating e-business models. *IEEE intelligent Systems*, 16, 11-17.

GRUBER, T. R. 1993. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5, 199-220.

GRUNINGER, M. & FOX, M. S. 1994. An activity ontology for enterprise modelling. *Department of Industrial Engineering, University of Toronto.*

GUARINO, N. 1997. Semantic matching: Formal ontological distinctions for information organization, extraction, and integration. *Information Extraction A Multidisciplinary Approach to an Emerging Information Technology.* Springer.

GUIZZARDI, G., DE ALMEIDA FALBO, R. & GUIZZARDI, R. S. Grounding Software Domain Ontologies in the Unified Foundational Ontology (UFO): The case of the ODE Software Process Ontology. *CIbSE*, 2008. 127-140.

HOEHNDORF, R. 2010. *What is an upper level ontology* [Online]. Available: <http://ontogenesis.knowledgeblog.org/740> [Accessed January 30 2014].

HUEMER, C., LIEGL, P., MOTAL, T., SCHUSTER, R. & ZAPLETAL, M. The development process of the UN/CEFACT modeling methodology. *Proceedings of the 10th international conference on Electronic commerce*, 2008. ACM, 36.

JAYAWEERA, P. M. A Unified Framework for e-Commerce Systems Development. *Business Process Pattern Perspective (BP3)*", PhD Thesis (ISBN, 2004. Citeseer.

KAIYA, H. & SAEKI, M. Using domain ontology as domain knowledge for requirements elicitation. *Requirements Engineering*, 14th IEEE International Conference, 2006. IEEE, 189-198.

KARIAPPER, R. & JAYAWEERA, P. M. 2014. Process Marting Approach for Legal Record Archiving. *Available at SSRN 2540870*.

KARIAPPER, R. K. A. R. & JAYAWEERA, P. M. Process Marting Approach for Legal Record Archiving. International Research Conference on Humanities and Social Sciences, 2013. University of Sri Jayewardenepura, Sri Lanka, 18.

KIETZ, J.-U., VOLZ, R. & MAEDCHE, A. Extracting a domain-specific ontology from a corporate intranet. Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning-Volume 7, 2000. Association for Computational Linguistics, 167-175.

KLEIN, H. K. & MYERS, M. D. 1999. A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS quarterly*, 67-93.

KROUWEL, M. R. & OP'T LAND, M. 2011. Combining DEMO and Normalized Systems for developing agile enterprise information systems. *Advances in Enterprise Engineering V*. Springer.

LIAO, L., QU, Y. & LEUNG, H. K. 2003. A software process ontology and its application.

MASCARDI, V., CORDÌ, V. & ROSSO, P. A Comparison of Upper Ontologies. WOA, 2007. 55-64.

REIJSWOUD, V. V. & DIETZ, J. L. 1999. DEMO modelling handbook. *Delft University of Technology, Dept. of Information Systems*.

STAMPER, R. 1980. LEGOL: Modelling legal rules by computer. *Computer Science and Law*, 45-71.

STAMPER, R. K. 1991. The Role of Semantics in Legal Expert Systems and Legal Reasoning\*. *Ratio Juris*, 4, 219-244.

STAMPER, R. K. 1996. Signs, information, norms and systems.

TU, S. W., ERIKSSON, H., GENNARI, J. H., SHAHAR, Y. & MUSEN, M. A. 1995. Ontology-based configuration of problem-solving methods and generation of knowledge-acquisition tools: application of PROTEGE-II to protocol-based decision support. *Artificial intelligence in medicine*, 7, 257-289.

USCHOLD, M. & GRUNINGER, M. 1996. Ontologies: Principles, methods and applications. *The knowledge engineering review*, 11, 93-136.

USCHOLD, M., KING, M., MORALEE, S. & ZORGIOS, Y. 1998. The enterprise ontology. *The knowledge engineering review*, 13, 31-89.

VALENTE, A. 1995. Legal Knowledge Engineering. *A Modelling Approach*, IOS Press, Amsterdam, Dissertation.

VAN HEIJST, G. A. C. M. 1995. The role of ontologies in knowledge engineering.

VAN INWAGEN, P. 1998. Meta-ontology. *Erkenntnis*, 48, 233-50.

VAN KRALINGEN, R. A conceptual frame-based ontology for the law. Proceedings of the First International Workshop on Legal Ontologies, 1997. Citeseer, 6-17.

VAN RENSSSEN, A. Gellish: an information representation language, knowledge base and ontology. Standardization and Innovation in Information Technology, 2003. The 3rd Conference on, 2003. IEEE, 215-228.

VEIJALAINEN, J. Issues in open edi. Systems Integration, 1992. ICSI'92., Proceedings of the Second International Conference on, 1992. IEEE, 401-412.

VELARDI, P., FABRIANI, P. & MISSIKOFF, M. Using text processing techniques to automatically enrich a domain ontology. Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001, 2001. ACM, 270-284.

VISSER, P. & BENCH-CAPON, T. The formal specification of a legal ontology. Proceedings of JURIX, 1996. 15-24.

VISSER, P. R. & BENCH-CAPON, T. J. 1998. A comparison of four ontologies for the design of legal knowledge systems. *Artificial Intelligence and Law*, 6, 27-57.

VON ALAN, R. H., MARCH, S. T., PARK, J. & RAM, S. 2004. Design science in information systems research. *MIS quarterly*, 28, 75-105.

WIERINGA, R. J. & GORDIJN, J. Value-oriented design of service coordination processes: correctness and trust. Proceedings of the 2005 ACM symposium on Applied computing, 2005. ACM, 1320-1327.

WIKIPEDIA. *Ontology (information science)* [Online]. Available: [http://en.wikipedia.org/wiki/Ontology\\_\(information\\_science\)#Domain\\_ontology](http://en.wikipedia.org/wiki/Ontology_(information_science)#Domain_ontology) [Accessed January 4 2014].

YUDELSON, M., GAVRILOVA, T. & BRUSILOVSKY, P. 2005. Towards User Modeling Meta-ontology. In: ARDISSONO, L., BRNA, P. & MITROVIC, A. (eds.) *User Modeling 2005*. Springer Berlin Heidelberg.

- c) Medical certificate
  - d) Police report
  - e) Report from the Family Counselor (If matrimonial dispute was referred to Mediation before the Family Counselor )
6. Summon
  7. Copy of plaint and filed documents
  8. Precept
  9. Stamp together envelop

**Payment to the courts:**

1. Filing fee ( papers to be used within the courts)
2. Stamp duty /Tax by the state
3. Service fee ( Process Server )
4. Binding fees

**Court proceedings:**

1. Plaintiff<sup>1</sup> meets the registrar to have a discussion or advice.
2. Plaintiff meets the known Lawyer and does the consultation regarding the divorce cases to be filed and get the advice and instructions how to precede the case and the Lawyer fees and charges too.
3. The motion<sup>2</sup>, proxy, plaint, summons, precept to fiscal are prepared by Lawyer based on the instruction given by the Client i.e Plaintiff.

---

<sup>1</sup> Plaintiff is the person who is submitting the petition against other partner in divorce cases especially  
<sup>2</sup> A formal proposal

4. Then the lawyer starts to write a plaint. With the clear discussion with plaintiff lawyer states in which ground he is going to file the case.
5. Proxy<sup>3</sup> (Appointment of an Attorney at Law ) is given ( Sign) by the Plaintiff to give Authority to the Lawyer for the institution of action and proceed.
6. Plaintiff Lawyer signs the motion and plaint.
7. Lawyer tender the motion ,proxy plaint documents summons precept to fiscal along stamped enveloped to the Court Clerk to obtain a Case No
8. Lawyer put the Case No. in all documents and pays the stamp duty to the Clerk who is designated for that purpose
9. Lawyer files ( Hand Over) above said documents and stamp duty receipt to the Court Clerk
10. The Clerk check all documents and enter the Case Number and Name of the Parties in the Case Register
11. the Clerk send the documents to the Binder to bind as a case record
12. after that the above said case record will be sent to Chief Clerk (Registrar)
13. Registrar will check and forward the case record to the Judge for the Acceptance and to issue the summons on the Defendant
14. Judge peruse the record and check , If he satisfied the case will be accepted and He will issue the summons by way of order, accordingly the Registrar will sign the summons and direct the fiscal to serve summons on the Defendant .
15. Registrar sign the summon and serve the summons to the Defendant together with copy of the Plaint and relevant documents which were filed by the Plaintiff's Lawyer.
16. With the service of the summon the case filing process terminates.

---

<sup>3</sup> Appointing a lawyer and give the authority to the lawyer to precede the case



## **Annexure II**

### **The major grounds of divorce**

1. Adultery
  - a) Man adultery
  - b) Female adultery
2. Impotency
3. Malicious Desertion
  - a) Sexual harassment
  - b) Drunkenness / drug addiction
  - c) Physical incapability
  - d) Discretion / abandonment
  - e) Imprisonment
  - f) Cruel treatment
  - g) Breach of promises
  - h) Fraud and cheat
  - i) Disease
  - j) No income or financial backing
  - k) Criminal activities
  - l) Spouse already married
  - m) Entered in to the marriage because of threat
  - n) Entered in to the marriage when the partner is unconscious or temporary disorder
  - o) Mental disorder
  - p) Incompatible of treatment

q) Culture and religion

r) Sexual issues

### **Annexure III**

The flow chart of the case filing is illustrated here.

