SUMMARY OF THE THESIS ENTITLED

APPLICATION OF ACTIVITY THEORY IN SEMANTIC WEB FOR DYNAMIC BUSINESS ENVIRONMENT

A Thesis Submitted to
Babu Banarasi Das University
for the Degree of

Doctor of Philosophy

in

Computer Science and Engineering

by
Diwakar Yagyasen

Under the Supervision of
Dr. Manuj Darbari

Department of Computer Science & Engineering
School of Engineering
Babu Banarasi Das University
Lucknow 226 028 (U.P.) India

February, 2016

Certificate of the Supervisor

This is to certify that the thesis entitled "**Application of Activity Theory in Semantic Web for Dynamic Business Environment**" submitted by **Mr. Diwakar Yagyasen** for the award of **Degree of Doctor of Philosophy** in Computer Science and Engineering from Babu Banarasi Das University, Lucknow (Uttar Pradesh), is a record of authenticate work carried out by him under my supervision. To the best of my knowledge, the matter embodied in this thesis is the original work of the candidate and has not been submitted elsewhere for the award of any other degree or diploma.

Dr. Manuj Darbari
Professor
Department of Computer Science & Engineering
Babu Banarasi Das University,
Lucknow

DECLARATION

I, hereby, declare that the work presented in this thesis entitled "**Application of Activity Theory in Semantic Web for Dynamic Business Environment**" in fulfillment of the requirements for the Degree of Doctor of Philosophy, Department of Computer Science and Engineering, Babu Banarasi Das University, Lucknow is an authentic record of my own research work carried out under the supervision of Dr. Manuj Darbari.

I also declare the work embodied in the present thesis is my original work and has not been submitted by me for any other degree or diploma of any university or institution.

(Diwakar Yagyasen)

ACKNOWLEDGEMENTS

I would like to express my special appreciation and thanks to my guide Dr. Manuj Darbari, Professor, Department of Computer Science & Engineering, Babu Banarasi Das University, Lucknow for his exemplary guidance, monitoring and constant encouragement throughout the course of this thesis. He consistently steered me in the right direction whenever he thought I needed it. He encouraged my research, allowing me to grow as a research scientist. His advice on both research as well as on my career have been priceless.

Profound gratitude goes to Dr. Akhilesh Das Gupta ji our Honorable Chairman and Dr. A. K. Mittal, Vice Chancellor, Babu Banarasi Das University, for their constant support, encouragement under this aura of academic excalibur, their enormous experience, their sincerity towards their duty which have truly inspired me leaps and bounds.

I must not forget to appreciate the extreme patience and support of my Mother Dr. Manorama Gupta during this work. I am really thankful to her. I also owe this research work to my wife Sneha and loving daughters Miss Siddhima and Miss Ritika who was extremely affectionate throughout.

Last but not the least I would like to thank almighty God, my father Late Sri Ram Kumar Gupta and friends and colleagues for their constant encouragement without which this assignment would not be possible.

DIWAKAR YAGYASEN

PREFACE

The overall research approach is of qualitative nature and thus applies research methods from the social science. In this context, the purpose of the research would be classified as descriptive research where a detailed and highly accurate correlation between Activity Theory and Web Semantics has to be established with respect to Business Dynamics. The purpose of the research is to build a theoretical framework / following an inductive approach as opposed to applying theory / pursuing a deductive approach.

The scope description of the thesis explained that the focus is choosing Web-Ontology based on their capabilities and qualities and applying it for business Process modeling.

Thereby creating semantic model as an artifact and the context related to Activity oriented Business Modeling (AOM) needs to be applied.

The important find outs are below:

1. Evolutionary Algorithms are well suited for solving Multiobjective optimization problem and it leads to a burning research area called "Evolutionary Multiobjective Optimization".

2. To analyse the cascading effect of change in business process due to change in

environmental variables.

3. Several issues have been proposed in EMO, like non-dominated sorting, niching mechanism, elitism, diversity and convergence.

4. During multiobjective optimization of conflicting objectives two major requirements are: diversity and convergence towards the approximate true pareto-front.

5. We extend the above problem in sugar manufacturing process. The main objectives identified in sugar manufacturing process involve, Firing control of boilers, Sugarcane recovery and Transportation and delivery mechanism.

6. The above problem highlights the area of sugar manufacturing sector where we have used the concept of multiobjective optimization and semantic web to develop the ontology of sugarcane and its linkage with the sugar industry.

7. The thesis presents a novel idea of creating sustainable framework using the concept of context of things and ubiquitous environment. The two entities are integrated using ontology which provides a ubiquitous web service environment for sugarcane industry.

On the basis of above discussion a new framework has been proposed & four papers are published. The proposed ontology links the sugar mills' website with the enhanced capability of web service and tries to relate the context depending on situation of thing. The sugar mills' website is linked with control centre, which receives the signals from sensor informing about the current state of sugarcane yield, and the raster image highlights the yield density to be processed in three stages. The paper presents a novel idea of creating sustainable framework which using the concept of context of things and ubiquitous environment. The two entities are integrated using ontology which provides a ubiquitous web service environment for sugarcane industry.

List of Publications:

1. Diversity and Convergence Issues in Evolutionary Multiobjective Optimization: Application to Agriculture Science. IERI Procedia, Elsevier, Volume 5, 2013, pp 81-86, (with Manuj Darbari)

2. Transforming Non-Living to Living: A Case on Changing Business Environment. IERI Procedia, Elsevier, Volume 5, 2013, pp 87-94, (with Manuj Darbari).

3. Application of Semantic Web and Petri Calculus in Changing Business Scenario. In *Modern Trends and Techniques in Computer Science*, Volume 285, 2014, pp 517-528. Springer International Publishing (with Manuj Darbari).

4. Quantification of Dynamic Business Environment by Development of Ontology using Task Reduction. *International Journal of Scientific Engineering and Research*, Volume 6(1), 2015, pp 283-288, (with Manuj Darbari).

TABLE OF CONTENTS

Page No.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Today's dynamic business environment comprised from participants like business industry, production unit, service providers, suppliers, customers etc., where every entity is responsible for profitability of overall business and every entity is dynamic in nature. To support this business environment, business application software need to adapt to significant changes for maximizing profit. By defining well defined shared semantics and ontologies for dynamic business, the business applications can fulfil the requirements for dynamism, which is very crucial in current business environments, which is done by translating the key aspects of semantic web technologies into business terminologies. The thesis aims to develop a framework, which can use Semantic Web technologies as a tool for various dynamic business operations. For translation of dynamic business aspects into semantic notation, the activity theory framework is utilized which gives a robust framework and can handle various dynamic business scenarios.

1.2 BACKGROUND

Semantic Web technologies are migrating to key technology to resolve the problems of interoperability and integration within the heterogeneous world of ubiquitously interconnected systems with respect to the nature of components, standards, data formats, protocols, etc. It provides excellent communication link between individuals extending the features of Semantic Web technologies for heterogeneous environment.

The world-class competitiveness of enterprises strongly depends, in the future, on their ability to rapidly set-up, and maintains, virtual, networked enterprise structures. In fact, managing the semantics of business-to-business interaction may be the most challenging task in integrated e-business value chains, and there is more and more evidence that Semantic Web technology to mitigate such problems.

1.2.1 Design issues of Semantic web for Dynamic Business Environment

This thesis outlines in detail an approach for adaptation of heterogeneous Web resources into a unified environment as a first step toward interoperability of smart industrial resources, where distributed human experts and learning Web services are utilized by various devices for self-monitoring and self-diagnostics. The proposed General Adaptation Framework utilizes a potential of the Semantic Web technology and primarily focuses on the aspect of a semantic adaptation (or mediation) of existing widely used models of data representation to RDF-based semantically rich format. To perform the semantic adaptation of industrial resources, the approach of two-stage transformation (syntactical and semantic) is elaborated and implemented. The environment will provide automatic discovery, integration, condition monitoring, remote diagnostics, and cooperative and learning capabilities of the heterogeneous resources to deal with maintenance problems.

Maintenance (software) agents will be added to industrial devices, which are assumed to be interconnected in a decentralized peer-to-peer network and which can integrate diagnostic services in order to increase the maintenance performance for each individual device. The semantics (metadata), which are intended to be added to the data that describe corresponding industrial resources, include knowledge about their state, condition, and diagnosis in temporal and contextual space. Further, the semantically reached resource descriptions will be used as input to decision-making components.

1.3 PROBLEM STATEMENT

The objective can be summarized in the following steps:

1. The first step covers the identification of issues relating to business dynamics.
2. Second step includes the use of Activity Theory concept in business modeling.
3. The third step is to analyze the heterogeneous environment of business activities and relating it to web semantics.
4. The fourth step covers the integration of various web services and finding the optimal solution.

1.4 RESEARCH OBJECTIVE AND MAIN CONTRIBUTION

Our framework presents a methodology to detect and react to the exceptional changes that can be in a Web application is proposed. It first classifies these changes into behavioural (or user-generated), semantic (or application), and system exceptions. The behaviour exceptions are driven by improper execution order of process activities. For example, the free user navigation through Web pages may result in the wrong invocation of the expired link, or double-click the link when only one click is respected.

The semantic exceptions are driven by unsuccessful logical outcome of activities execution. The system exceptions are driven by the malfunctioning of the workflow-based Web application, such as network failures and system breakdowns.

It then proposes a modelling framework that describes the structure of activities inside hypertext of a Web application. The hypertext belonging to an activity is broken down into pages, where are identified within an activity. The framework consists of three major components: Identification stage, Alternate Path selection stage, and Execution stage. The Identification stage identifies and store the exceptions in information flow. The Alternate Path selection stage identifies the alternate paths for the broken information paths.

The Execution stage identifies a set of recovery policy to resolve break in information flow. A framework is presented to manage the business instruction sets and its evolution in service-oriented architecture. It uses several features to handling the running instances under the old protocol. These features include impact analysis and data mining based migration analysis. The impact analysis is to analyze how protocol change impacts on the running instances. It will be used to determine whether ongoing conversations are migrateable to the new protocol or not. The data mining based application is done in case where the regular impact analysis cannot be performed. Service interaction logs are analyzed using data mining techniques. It then uses the result to determine whether a conversion is executable or not. The work mainly deals with dynamic protocol evolution. We focus on automatically modifying the composition of Web services in a heterogeneous environment handling dynamic business changes.

Semantic Web service technologies have been proposed to automate the usage of Web services, such as its discovery and composition by adding semantic markup information to a service’s WSDL description. The input, output, and operations of a Web service are annotated with machine-understandable information. Such semantic annotations can help software agent software agents discover and invoke a Web service but it has a drawback that it mainly focuses on the semantics within a service description. We propose a framework that plays a key role in automating the process of change management. The central of the semantic support is a Web service ontology that has a hierarchical structure, which leverages the advantages of current approaches and addresses their limitations.

1.5 THESIS ORGANIZATION

The thesis has been organized into six chapters.

- Chapter 1 is the introduction of the proposed work that includes the basics of semantic web, Multi-objective optimization and business dynamics extensions. This includes the research objectives and major contribution and thesis organization.
- Chapter 2 is the literature survey on of Semantic Interoperability, Semantic Heterogeneity, Semantic Annotation, Web Agents, and Evolutionary Algorithms (EAs) & Evolutionary Multiobjective Optimization (EMO).
- Chapter 3 provides the reader with background material to provide a basis for understanding subsequent chapters. Chapter is divided into three parts the first part deals with brief overview of Activity Theory and its applications. This chapter also describes ontology and Semantic Web concepts and Web Agents for Semantic Web.
- Chapter 4 focuses on development of a framework providing context aware ubiquitous environment focusing on achieving coherency between business dynamics and sugar and Web ontology highlighting the coordination in an heterogeneous environment. In order to model the above situation we have used coordination of Ubiquitous systems for efficient sugar recovery in Sugar mill.
- Chapter 5 introduces the techniques used in framework for Dynamic Business environment, which includes decision support system, supply chain management and also how supply chain dynamics and work breakdown strategy (WBS) works into Semantic Web. Next part deals with Quantification of Dynamic

Business Environment by Development of Ontology using Task Reduction.

- Chapter 6 is the verification of the model using Petrinets. Then it shows application of Semantic web and Petri Calculus in changing Business Scenario showing how Petrinets and Semantic web can be linked together.
- Chapter 7 brings to a conclusion of the work done and leaves the future scope of the research work of this thesis.

CHAPTER 2

LITERATURE SURVEY

As the development of the Web, more and more Information Systems start to be implemented using Web technology. Some Web technologies, e.g. Semantic Web technologies, are introduced into Information Systems development. On the other hand, the Web applications also benefit from traditional Information Systems theories and practices. COEUR-SW (Concepts On Enhancing, Understanding and Representing the Semantics on the Web) triangle in Sølvberg, Hakkarainen, Brasethvik, Su, Matskin and Strasunskas (2002) discloses how an Information Systems on the Semantic Web comply with the semiotic triangle in Ogden, Richards, Malinowski and Crookshank (1946). COEUR-SW triangle is developed based on the semiotic triangle in the IS group at IDI3. In the COEUR-SW triangle (Figure 2-1), a concept in a Universe of Thought (UoT) is related to an uttered symbol in a Universe of Language (UoL), the symbol is related to a referent in a Universe of Structure (UoS), and the referent is related back to the concept. The concept, the symbol and the referents are related to a context in the Universe of Discourse (UoD).

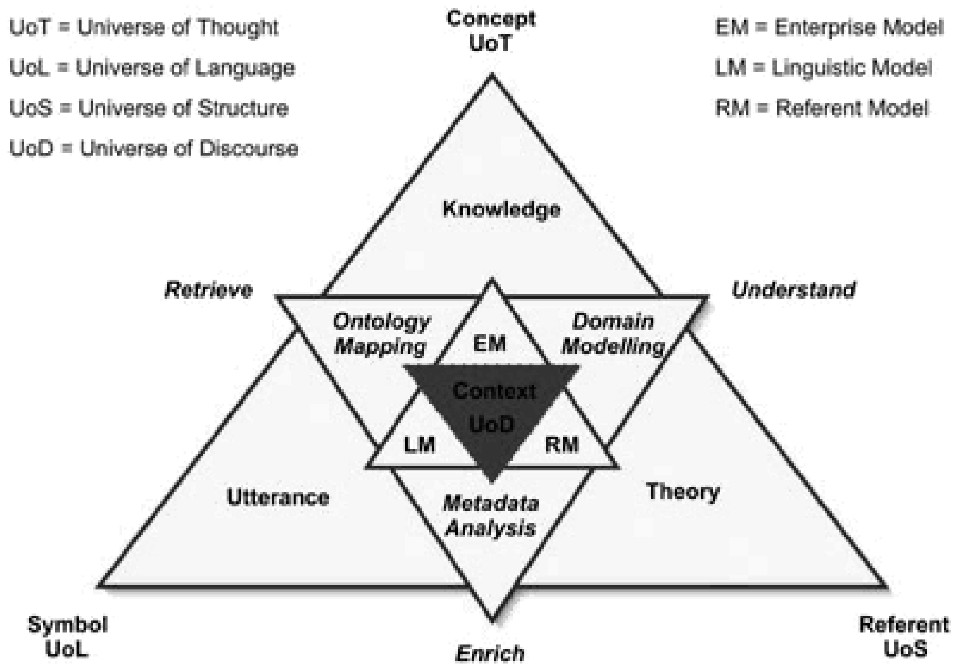

Figure 2-1: COEUR-SW Triangle

COEUR-SW triangle defines three roles of the Web in Information Systems in Sølvberg et al. (2002):

- The role of the dominant medium for information dissemination;
- The role of the tool for information compilation;
- The role of the evolving information repository.

2.1 SEMANTIC INTEROPERABILITY

Interoperability is the ability of two or more systems or components to exchange information and to use the information that has been exchanged in Geraci, Katki, McMonegal, Meyer, Lane, Wilson, and Springsteen (1991). Interoperability is a broadly used term, encompassing many of the issues impinging upon the effectiveness with which diverse information resources might fruitfully co-exists. The issues can be defined for different purpose, such as, semantics.

The main obstacle of semantic interoperability is semantic heterogeneity of the information to be exchanged. Common understanding of semantics and standardization of semantic representation are usually concerned as the solutions tackling the semantic heterogeneity to achieve semantic interoperability.

2.2 SEMANTIC HETEROGENEITY

Semantic heterogeneity is usually distinguished from syntactic heterogeneity and structural heterogeneity in the database community in Manolescu, Florescu and Kossmann. (2001), Galanis, Wang, Jeffery and DeWitt (2003), Lenzerini (2002), Levy (1999), Levy, Rajaraman and Ordille (1996). Syntactic heterogeneity is concerned with the heterogeneity of data formats. Standardizing data formats is taken as an approach to solve syntactic heterogeneity problems. For example, XML is used as a standard format for all forms of Web-accessible data. Structural heterogeneity is associated with different data models, data structures or schemas, e.g. relational and object-oriented database models. An example of the solutions for structural heterogeneity is that RDF based on XML syntax provides a unified way to structure information sources or object models for Web-based information exchange in Frank Manola. When two information sources are modeled in a same format by applying a same modeling methodology, there still might be semantic heterogeneity problem. Semantic heterogeneity can be identified according to the different types of conflicts in Spaccapietra, Parent, and Dupont (1992):

- **Semantic conflicts.** They are more focused on overlapping region of the sets.
- **Descriptive conflicts.** Descriptive conflicts include naming conflicts due to homonyms and synonyms in Batini and Lenzerini (1984),

Navathe, and Gadgil (1982), attribute domain, scale, constraints, operations, etc.

- **Structural conflicts.** Such structural conflicts are different from structural heterogeneity.

2.3 SEMANTIC ANNOTATION

The goal of empowering computer systems with semantic interoperability rests on the desirability of computer systems being able to find information and to use it for purposes that the original creator of the information did not anticipate. This goal of flexible information reuse requires some degree of understanding of the information, which in turn requires that the information be encoded in some standard fashion that is interpreted identically by all systems using that information. As a shared model of what the information represents, ontologies are usually used to achieve the level of understanding. Semantic annotation is an approach to link ontologies to the original information sources.

Annotation is the extra information associated with a particular point in a document or other piece of information. For semantic annotation, the extra information is meaning definitions of the concepts used in a document. The meaning definitions are in most cases represented in ontologies. Annotation can be in the form of comments, or in the form of metadata. Metadata is data about data and it is used to facilitate the understanding, use and management of data. Machine-manipulable annotations are often organized as metadata, which is also the format of semantic annotations. There are a number of alternative approaches regarding the organization, structuring, and preservation of annotations. In Wetzstein, Ma, Filipowska, Kaczmarek, Bhiri, Losada, ... and Cicurel (2007), semantic annotation of process models is concerned as a prerequisite of the vision of Semantic Business Process Management, which is very close to our proposal.

The executable process models can be partly generated from the conceptual business process model, which indicates there are underlying links between business process models and executable Web services. Although the work has just initiated and it is still an ongoing project, it shares the same vision with ours, i.e. semantic annotation can be also concerned as an alternative approach to achieve the semantic interoperability of semi-structured sources such as business process models, in spite of semantic annotations of unstructured sources (e.g. textual documents) and structured sources (e.g. WSDL described Web services). Efforts on the semantic enrichment of enterprise models by semantic annotations are also put by TG4 (Task Group 4: Semantic Enrichment of Enterprise Modeling, Architectures and Platforms) in EU project INTEROP (Interoperability Research for Networked Enterprise Applications and Software) in INTEROP, in which the main achievable targets are the semantic interoperability for model exchange, model transformation and model traceability. As the contemporary work, our research shares some similar objectives and available technologies.

The simplest knowledge discovery mechanism is based on the traditional query/answer paradigm, where each part acts as both client and server, interacting with other nodes directly sending queries or requests, and waiting until receiving an answer. This is only possible if the domains are previously known to each other or if a collaboration relationship has already been established between them. When this is not the case, the discovery of knowledge is affected by the dynamism of the system. Some nodes join and some nodes leave the network, at any time. Besides, each domain is responsible for its own knowledge representation and management, because there are no a-priori agreements regarding ontology language nor granularity. A typical example of such a system is the Semantic Web in Breitman, Casanova and Truszkowski (2007).

In open distributed systems, several nodes (domains), probably distributed among different organizations, need resources and information (i.e. data, documents, services) provided by other domains in the net. Therefore, an open distributed system can be defined as networks of several independent nodes, having different roles and capacities. In this scenario, a key problem is the dynamic discovery of knowledge sources, understood as the capacity of finding knowledge sources in the system about resources and information that, in a given moment, better response the requirements of a node request in Castan, Ferrara, and Montanelli (2006). Searching on the Semantic Web differs in several aspects from a traditional web search, especially because of the structure of an online collection of documents in the Semantic Web, which consists of much more than HTML pages. The semantics associated to the languages for the Semantic Web allows the generation of new facts from existing facts, while traditional databases just enumerate all available facts.

Traditional search machines do not try to understand the semantics of the indexed documents. Conventional retrieval models, in which the retrievals are based on the matching of terms between documents and the user queries, is often suffering from either missing relevant documents which are not indexed by the keywords used in a query, but by synonyms; or retrieving irrelevant documents which are indexed by unintended sense of the keywords in the query.

Instead, search agents for the Semantic Web should not only find the right information in a precise way, but also should be able to infer knowledge and to interact with the target domain to accomplish its duty in Peis, Herrera-Viedma, Hassan-Montero, and Herrera (2003). In the next subsection, a proposal for a knowledge source discovery agent is presented.

2.4 WEB AGENTS

Web agents are complex software systems that operate in the world wide web, the Internet, and related corporate, government, or military intranets. They are designed to perform a variety of tasks from caching and routing to searching, categorizing, and filtering. The ideal web agent needs to satisfy four requirements:

- *Communicative*: able to understand the user's scope, preferences and constraints. A problem with the most well known search engines is that they have no knowledge of the domains of interest. Solutions to this problem usually involve the use of ontologies, a formal definition of a body of knowledge. The most typical type of ontology used in building agents involves a structural component. It is essentially a taxonomy of class and subclass relations coupled with definitions of the relationships between the terms of the ontology. If an ontology is well structured and uses a machine readable vocabulary, it allows a software program to manipulate the terms used in the ontology, terms that make sense to users who understand this information. A software program can manipulate terms that the user understands. Software components, like agents promote this communication, by reflecting users' needs, preferences and constraints.
- *Capable*: able to take actions rather than simply provide advice. Either we refer to an autonomous agent or a multi-agent system, the agent needs to be capable not only to recommend but also to take action in order to fulfill tasks assigned by a user. It needs to take actions so to make things simpler and automated for the user. For this scope, it is necessary the agent to be able to overcome the syntactic elements in order to extract the semantic elements of a web page. Here is one of the major problems for the web agents. While agreement is starting to emerge, a lot of engineering is still to be done to encode information about Internet sources and about finding the appropriate way to manipulate them.
- *Autonomous*: able to act without the user being in control the whole time. A truly effective web agent or a web multi-agent system, needs to be able to take some sort of action and work for the user. The key to

autonomy is finding the right level for the task at hand. The autonomy is a difficult programming task, because it is very dependent on the characteristics of the domain a program is operating. The level of autonomy also depends on the group of users that use the proposed service and their constraints. It is difficult to predefine what a specific user would like to take action for and what he/she wants to be done automatically without knowing any details.

- *Adaptive*: able to learn from experience about both its tasks and about its users preferences. It is very important for the system to succeed its goal by meeting user's criteria and not overstep his/her constraints. To do that it needs to "understand" user's preferences on a particular application environment. A web agent with a predefined number of pages to visit is limited and needs to be able to adapt its behavior based on a combination of user feedback and environmental factors. Enabling autonomy is not an easy programming task, particularly because each system depends on the context of the area that operates. The ways of achieving such a behavior are many. Theories from machine learning, collaboration filtering etc. can be adapted.

2.4.1 Types of agent

There are following agent types (Maalal and Addou, 2011):

- *The Reactive Agent*- this type of agent is not clever by itself, but it has ability to understand the environment and act accordingly. For example robot motion agent convert their sensory data to motion vectors for avoiding obstacles and move to the goal.
- *The cognitive agent* – it is able to carry out activities in a flexible and intelligent manner. This type of agent learns from experience, and respond accordingly to changes in the environment. It shows goal oriented behavior and also communicate and co-operate with other

agents.

- *The intentional agent or BDI (Belief, Desire and Intention)* – this is an intelligent agent which shows human intelligence and human perspective, using mental concepts such as knowledge, beliefs, intentions, desires, choices, commitments.
- *The rational agent* – This agent is task oriented i.e. it acts to achieve success for the particular task it is assigned for. This agent should run when the possible objective for that task is desired with a particular measure of performance.
- *The adaptive agent* – This agent should adapt changes according to the environment. These changes may require changing its knowledgebase and its objectives.
- *The communicative agent* – This agent communicate information with all other agents, and this information may be its own or it may be transmitted by other agents.

2.4.2 Agent Communication Languages

For modeling our systems we have used the concept of Agent Communication Model, An agent communication language is defined at three levels: the lower level, which specifies the method of interconnection; the middle level, which specifies the format or syntax; and the top level, which specifies the meaning or semantics. An agent communication language has three components: the outer and inner languages and the vocabulary. The 'outer' language is the language that is used in order to express the primitives, i.e. the performatives that an agent is permitted to use in communicating with other agents. It defines an 'envelope' format for using messages and is used by the agent to explicitly state the intended illocutionary force of a message. The 'inner' or 'content' language (syntax) is the (logical or representation) language which is used to write the message itself. In other words, it is the syntax used for the message. This layer allows for knowledge sharing. The vocabulary describes the domain of discourse in terms of concepts and their relationships and prescribes meaning to the terms used, i.e. it is the semantics. Labrou and Finin (1998) identified a number of features as essential for a good ACL. First of all, a good ACL should be declarative and syntactically simple. Secondly, the ACL needs to have a well-defined set of primitives and also needs to distinguish between the communicative language which expresses the primitives, and the content language which expresses the message itself, although it should not commit to a content language. Thirdly, the semantics for the primitives should be clearly defined, preferably through a formal description, as in modal logic. Another desirable feature is that the implementation of the language should be efficient, both in terms of speed and bandwidth, with simple interfaces. Moreover, as networking is prevalent, a good ACL should support all of the basic connections, i.e. point-to-point, multicast and broadcast, and both synchronous and asynchronous communication. Given the heterogeneity of environments, programming languages and frameworks, a good ACL should provide tools for dealing with heterogeneity and support interoperability with other languages and protocols. Finally, an ACL should support reliable and secure communication among agents, including authentication facilities and error detection.

Our meta-model allows users to configure the models at run time. Modeling all functional requirements for the selection through our meta-model enables execution of the selection inside executable business process models. It also enables stakeholders to formally check if the requirements are correctly satisfied in the final software product before it is used. Web services are gradually taking root as a result of convergence of business and government efforts to make the Web the primary medium of interaction in Medjahed (2004). Furthermore, the maturity of XML-based Web service standards, such as SOAP and WSDL, are driving the rapid adoption of Web services in Curbera, Duftler, Khalaf, Nagy, Mukhi, and Weerawarana (2002) , Casati, Shan, Dayal, and Shan (2003).

In a short term composed service, both the business objective and Web service partnerships are temporary and short lived. Once the business objective is achieved, the partnerships between Web services are dissolved.

Ritchie, and Brindley (2007) proposes to examine the constructs underpinning risk management and explores its application in the supply chain context through the development of a framework. On the similar line Khan and Burnes (2007) found the risk in supply chain management. It shows that how number of key debates in the general literature on risk, especially in terms of qualitative versus quantitative approaches, which need to be recognized by those seeking to apply risk theory and risk management approaches to supply chains. . Until Now most web pages are limited together using the HTML that relates only to human understandability. As the user desires he/she can move from pages. All the process doesn't support any communication which is understandable to machines. Berners-Lee (1996) developed a concept of web pages that could be understandable by machines. The Semantic web has revolutionized the basic concepts of web information it has automated the process such that even the machine can understand. The advantage with this feature provides the basis for refinement in context knowledge. The supporting framework for this paradigm shift includes resource description framework (2002) and web ontology language (OWL) (2004). There has been lots of effort in the area of web ontology for Agricultural system. Some prominent work include the work done by (Islam and Piasecki, 2008) Ontology for hydrodynamics (2008). The development of ONT Agria for precision farming in Aqeel-ur-Rehman (2011) describes scalable service oriented agriculture ontology that consists of irrigation and fertilization on the similar lives Van Ittersum, Ewert, Heckelei, Wery, Olsson, Andersen ... and Wolf (2008) developed a component based framework SEAMLESS. It defines different levels describing about meta-information about Agriculture related entities. Paper by Sharahchandra and Lele (1991) highlights the issue relating to sustainable development of Natural Resources using Semantic. MOST which is a software for supporting knowledge base relating to water management in Agriculture domain. Agri QC by (Augustina, Mary and Mela, 2011) also supports Knowledge Management System (KMS) which is highlighted with classification of queries for the Agriculture Knowledge Management System (Agri KMS) .

2.4.3 Evolutionary Algorithms (EAs) & Evolutionary Multiobjective Optimization (EMO)

Population based search procedure is the major strength of Evolutionary Algorithms (EAs) and it makes us capable to get multiple solutions a single run. EAs are perfect with very large and complicated search spaces and highly applicable in multiobjective optimization problems. Evolutionary approaches, including genetic algorithms in Golberg (1989), genetic programming in Koza (1992), evolutionary strategies in Schwefel (1993) and evolutionary programming in Fogel (2009) are used to solve multiobjective optimization problem. It will be generating the research area called Evolutionary Multi-objective Optimization in Coello, Van Veldhuizen and Lamont (2002), Coello (2005), Coello, Pulido, and Montes (2005), Coello Coello (2006).

Different EAs have been developed for the purpose of multiobjective optimization. First generation of the algorithms is including NSGA (Non-dominated Sorting Genetic Algorithm) in Srinivas, and Deb (1994), NPGA (Niched Pareto Genetic Algorithm) in Horn, Nafpliotis and Goldberg (1994), Multi-Objective Genetic Algorithm (MOGA) in Fonseca and Fleming (1993). The major focus was on fitness sharing and niching integrated with Pareto Ranking.

CHAPTER 3

INTRODUCTION TO ACTIVITY THEORY AND SEMANTIC WEB

In this chapter, we endeavor to provide the reader with background material to provide a basis for understanding subsequent chapters. Chapter is divided into three parts the first part deals with Multi optimization techniques and second part deals with brief overview of Activity Theory and its applications, and finally meaning of Semantics web and its development and application.

3.1 A DESCRIPTION OF ACTIVITY THEORY

3.1.1 Modeling Using Activity Theory

Software Modeling practitioners have failed to model even the minutest details of the organization mainly because of the lack of established methods of operationalising concepts of Activity Theory within the system design process, filling the pragmatic vacuum by introducing a considerable number of challenges. The use of this theory for system design requires the justification of the method applied to operationalise the theory, together with a provision of clear evidence of mapping between Software Modeling and Activity Theory.

The recognition of significance for Activity Theory has been elaborated in our research and there is still growing needs to develop methods that can directly apply the concept of Activity Theory in Software Modeling.

The main focus of this work is to develop the Activity Oriented Model using various software tools. Our work mainly summarizes on Vygotsky's Theory, Leontev's Theory, Engestrom's eight steps Model and finally transformation of Activity Theory Models into Unified Modeling Language. In order to solve the above situational problems we have undergone various stages to prove our result.

Activity Oriented Models (AOM) can be executed as a part of the requirements elicitation process involving the study of work practices. The output of AOM based analysis is a report outlining the Activity Theory based conceptualization of Work Based Time Stamped models (WBTS) which plays a vital role in elements of system modeling.

3.1.2 Activity Theory: A Brief Introduction

Activity Theory is a philosophical framework for studying different forms of human behavior.

"Activity Theory is a philosophical and cross disciplinary framework for studying different forms of human practices as development processes, with both individual and social levels interlinked at the same time".

Activity theory is a framework for understanding both individual and collective aspects of human practices from a cultural and historical perspective. It has got following points relating to this thesis:

3.1.3 Engeström Model

Engeström and Ahonen (2001) expanded Vygotsky's original representation for mediated human behaviour 'mediational model'.

> *"Engestrom's approach extended Vygotsky's representation of mediated behaviour by producing a model that reflects both the collaboration and collective nature of human activity. In addition, Engestrom's approach also expanded Leontev's work by incorporating the 'subject' components, to represent those engaged in carrying out activity, also the division-of-labour component, to represent and make the various responsibilities of those engaged in activity explicit".*

His model was also known as 'Activity System' which is generally referred as 'Activity Triangle Model', [Figure 3-1] incorporating the components: Subjects, Object and Community with mediators of human activity, namely, Tools, Rules and the Division of labour.

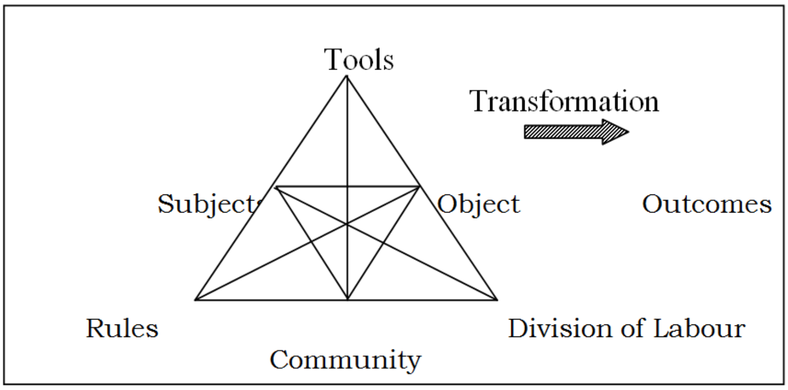

Figure 3-1: Activity Triangle Model

Components of the activity triangle are discussed below:

The '***object***' component portrays the purposeful nature of human activity, which allows individuals to control their own motives and behaviour when carrying out activity.

The '***subject***' component of the model reflects the individual and collective nature of human activity The main idea is to have a relationship with the object through tools.

The '***tool***' component gives the overview of mediational aspects of human activity through the use of both physical and psychological tools. Psychological tools are used to influence behaviour in one way or another.

The '***Community***' is the stakeholders in a particular activity or those who share the same overall objective of an activity.

The '***Rules***' are the set of guidelines under which community has to work.

The "***Division of Labour***" is defined as the work division which is done to avoid confusion..

The '***Activity System***' has several sub-activities that are interconnected such that disturbances or contradictions can occur *within* and *between* sub activities Operationalising Activity Theory

In Activity Theory, we have developed the principle of object oriented-ness, which refers to the need to focus on the 'Object' of activity when trying to transforming the 'object' into an outcome. It can be represented by simple Activity Oriented format developed specifically for Business Dynamics which is later embedded with multi-agent systems (Figure 3-2).

The principle of mediation plays a central role in the AT. An *activity* always contains various artifacts (e.g. instruments, signs, procedures, machines, materials, laws, forms of work, or organizations).

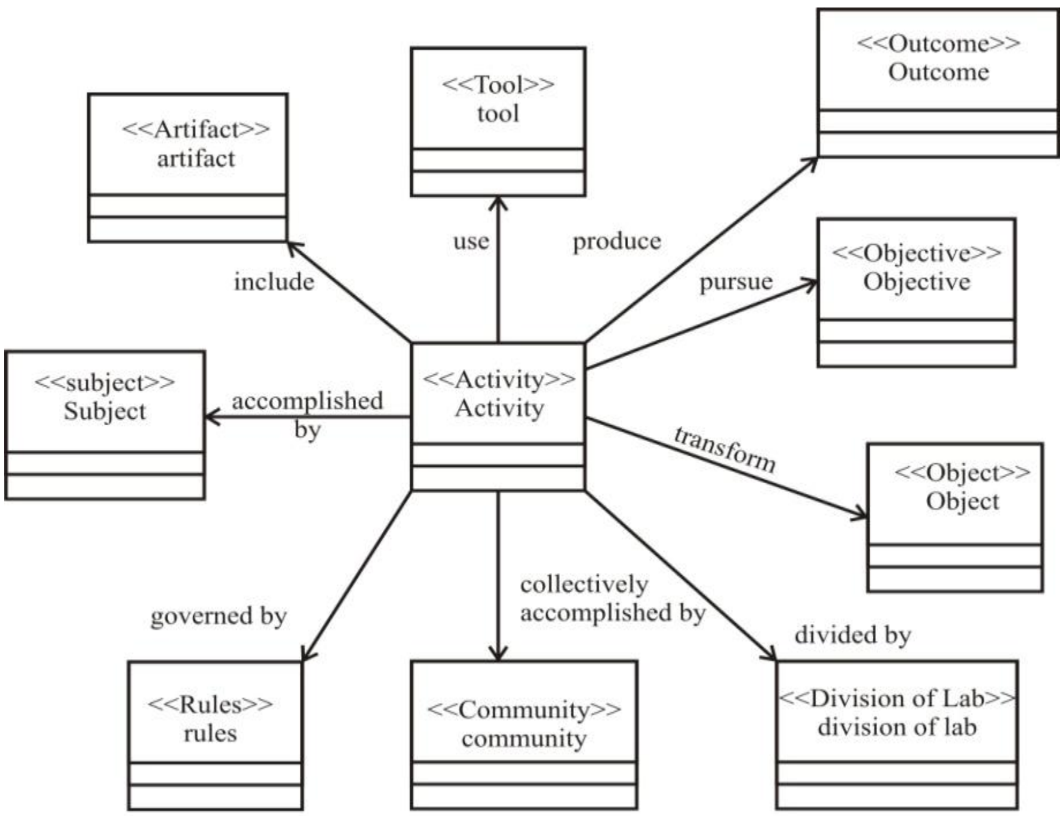

Figure 3-2: Object Oriented Activity Theory Model Using OPEN Framework

These artifacts have a mediating role: relations between elements of an *activity* are not directed, but mediated. *Tools* shape the way human beings interact with their context. A *tool* can be anything used in the transformation process of the *object*, including both material tools and tools for thinking. The relationship between *subject* and *community* is mediated by *rules* (e.g. laws, social conventions, or norms), and the relationship between *object* and *community* is mediated by the *division of labour* which establishes how the *activity* is distributed among the members of the *community*. The *division of labour* considers the role each individual in the *community* plays in the *activity*, the power each wields, and the tasks each is held responsible for.

In the next section, the implications of using activity theory as a research tool will be considered.

3.2 ACTIVITY THEORY AS A RESEARCH TOOL

This section considers the use of activity theory as a research tool from a number of perspectives and will include a brief overview of Activity Theory within educational research, a discussion of the use of activity as the participatory unit of analysis, and methodological implications for this inquiry.

According to Nardi (1996) Activity Theory can be considered as a powerful descriptive tool rather than a strongly predictive theory. It does not offer the tools and techniques required for research; instead the concepts of activity theory are to be applied to the specific subject under study.

Activity Theory has been a topic of great interest among various scholars doing research in the field of Human Computer Interaction as the computers can be viewed as tools that assist in performing human activity**.**

Activity theory provides a common lexicon to describe the organizational, technological and pedagogic perspectives proposed by in terms of subjects, tools, object and outcome, rules, community and division of labor. If we assume that the object of each activity system is an increase in the outcome of resources of the organization. The desired outcome of the organizational activity system is organizational sustainability. The technological activity system is largely represented by information technology specialists whose primary responsibility is for the health of the organizations information technology systems (both administrative and teaching) will desire technological sustainability.

3.3 PRINCIPLES OF ACTIVITY THEORY

3.3.1 Object Orientedness:

Human beings live in reality with a certain objective. The things that constitute this reality not only have the properties that can be considered objective scientifically but they have socially and culturally defined properties as well.

3.3.2 Internalization/Externalization:

Internal activities cannot be completely separated from external activities. They are reliant on each other as they transform into one another. By internalization we mean the conversion of external activities into internal ones. Internalization is a method by means of which people strive for potential interactions with reality without performing actual manipulation with real objects. Externalization transforms internal activities into internal ones. It converts internal activities into external ones. Externalization is essential when several activities are being carried out between a group of people and they need to be synchronized.

3.3.3 Mediation:

Activity Theory states that the human activity is mediated by tools. Tools are formed and transformed during the progress of an activity. The use of tools affects the nature of external behavior and the internal functioning of individuals.

3.3.4 Development:

It does not only mean the object of the activity but it can also be understood as the research methodology. It does not refer to the laboratory experiments; instead it stands for the formative experiments which merge active participation with monitoring changes of the participants.

The above principles must be considered as an integrated system as they are concerned with the various aspects of the whole activity.

3.4 ONTOLOGY

3.4.1 Introduction to Ontology

According to Gruber (1993) ontology can be defined as an explicit specification of conceptualization i.e. the structure of domain with possible restrictions. Ontology is the knowledge representation of model of the domain in particular modeling language. Ontology is state independent of the model.

An ontology describes multiple knowledge base, which describes specific state of affairs. An agent has its own knowledge base, and stores the knowledge in an ontology. For agent-to-agent communication, ontologies must be shared between them.

The term ontology was taken from philosophy. According to Webster's Dictionary an ontology is:

- a branch of metaphysics relating to the nature and relations of being
- a particular theory about the nature of being or the kinds of existence.

Although it is required from an ontology to be formally defined, there is no common definition of the term "ontology" itself. The definitions can be categorized into roughly three groups as in Guarino and Pierdaniele Giaretta (1995), Obitko (2001):

1. Ontology is a term in philosophy and its meaning is "theory of existence".

2. Ontology is an explicit specification of conceptualization in Gruber (1993).

3. Ontology is a body of knowledge describing some domain, typically common sense knowledge domain.

3.4.2 Ontology as a Specification of Conceptualization

According to Guarino and Giaretta (1995) conceptualization defines intensional semantic structure that encodes implicit knowledge of a domain and its specification is ontology (Figure 3-3). Hence ontology is logical theory that expresses conceptualization in some language. According to Gruber (1993), an ontology is a description of the concepts and relationships that can exist for an agent or a community of agents.

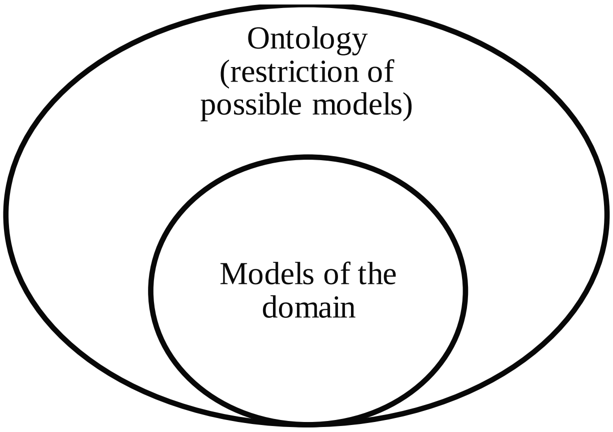

Figure 3-3 : Ontology expressing models for description of the domain.

An ontological vocabulary is such that when agents can commit to ontologies and so that knowledge can be shared between these agents. The knowledgebase is based on the specification of conceptualization, which is an abstract view of the problem domain for specific purpose. Every knowledge base, knowledge-based system or agent is committed to some conceptualization. The set of objects represented by the knowledge base is called the universe of discourse. The relationships among objects are reflected by representational vocabulary. In the context of Artificial Intelligence, we can describe the ontology by defining a set of representations. This ontology contains definitions of entities of universe of discourse (e.g. classes, relations, functions, or other objects). Formally, it can be said that an ontology is a statement of a logical theory in Gruber (1993).

3.4.3 Operations on Ontologies

Multiple ontologies may be used in a single application. For this some ontological operations are needed, as given in Sowa (2000), Noy and Musen (1999), Klein (2001), McGuinness, Fikes, Rice and Wilder (2000). These ontological operations are required for maintenance and integration of ontologies.

The operations are:

i. *Ontology Merge:*

It is creation of a new ontology by linking up the existing ones. The new ontology imports selected knowledge from the original ontologies so that the result is consistent. The new merged ontology introduces new concepts and relations from the original ontologies.

ii. *Ontology Mapping:*

It is the translation of one ontology into another one i.e. translation between concepts and relations.

iii. *Alignment:*

It is mapping process in both directions i.e. to modify original ontologies so that suitable translation exists. This makes possible to add new concepts and relations to ontologies.

iv. *Refinement:*

It is from ontology A to another ontology B such that primitive concepts of ontology A may correspond to non-primitive (defined) concepts of ontology B.

v. *Unification:*

It is the alignment of all the concepts and relations in ontologies. The inference in one ontology can be mapped to inference in other ontology and vice versa.

vi. *Integration:*

It is developing new ontology from previous ontologies, which allows translation between previous ontologies and interoperability between systems that uses previous ontologies.

vii. *Inheritance:*

An ontology A inherits everything from ontology B i.e. all concepts, relations and restrictions or axioms, and there is no inconsistency introduced by additional knowledge contained in new inherited ontology.

3.5 INTRODUCTION TO SEMANTIC WEB

The World Wide Web commonly known as Web as the biggest information construct has had much growth since its beginning. Even though it is not synonymous with Internet but it is the most essential constituent of the Internet that can be considered as a techno-social system that is able to communicate with the humans based on the technological networks. The techno-social system essentially refers to the system that improves the human communication, cognition and co-operation. The Cognition is the essential pre requisite to interact and the pre condition to co-operate. Therefore, we can state that co-operation requires communication and communication requires cognition. The Web is the leading transformable construct of information and its thought was given by Tim Burners Lee in 1989. There has been a lot of progress in the area of the Web and its associated technologies in the past two decades. We can say that Web 1.0 is a web of cognition, Web 2.0 is a web of interaction, Web 3.0 is a web of co-ordination and Web 4.0 is a web of amalgamation of the four generations of Web since the advent of Web.

3.5.1 Web 1.0

The Web 1.0 is the foremost generation of the Web created in 1989 by Tim Berners-Lee. He recommended developing a worldwide hypertext space in which any network reachable information can be accessed. It can be considered as the read-only Web as well as the system of cognition. It is stationary and to some extent mono-directional. The Businesses can offer brochures or catalogs to demonstrate their productions with the Web and the people can read them and get in touch with the businesses. The websites comprise static HTML pages that are updated occasionally. The major objective of the websites is to issue the information for anybody at anytime and set up an online occurrence. Those websites were not dynamic and might be considered as leaflets only. The users of these websites can just access these websites with no contributions or impacts and the connecting structure was also very pathetic. The central protocols of Web 1.0 were HTML, HTTP and URI.

3.5.2 Web 2.0

The term Web 2.0 was officially defined by Dale Doughtery, Vice President of O'Reilly Media in the year 2004. Tim O'Reilly defines Web 2.0 as follows:

"The Web 2.0 is the trade revolution in the computer business due to shift to the Internet as a platform, and an effort to realize the protocols for success on that fresh platform. Principal among those rules is this: Develop applications that bind network effects to get enhanced, the more people use them".

The Web 2.0 is also referred as the people-centric web, wisdom web, people-centric web and participative web. Through writing as well as reading, the Web might turn into bi-directional. The Web 2.0 is platform where the users can depart a lot of the controls they have been utilized to in the Web 1.0. We can say it also as, the users of the Web 2.0 have additional interaction with a reduced amount of control. The creative reuse, flexible design, updates, modification collaborative content creation were facilitated by Web 2.0. The one of the exceptional characteristics of Web 2.0 is to maintain collaboration and to assist to collect collective intelligence instead of Web 1.0. The major services and technologies of Web 2.0 comprise Really Simple Syndication (RSS), blogs, tags, wikis, mash ups, tag clouds and folksonomy. The Developers make use of the three fundamental development ways to design the applications of Web 2.0: XML (AJAX) and Asynchronous Java Script, Google Web Toolkit and Flex.

3.5.3 Semantic Web or Web 3.0

The Semantic Web represents the next major evolution in connecting information. John Markoff of the New York Times christened Web 3.0 as the third generation of the Web in 2006. Starting from, as a methodology utilized as machine interpretable data by means of the new generation software, the Web 3.0 is also referred by its assumed name as the Semantic Web. It has grown itself into a group of standards that support open data formats and at the same time processes the information that emphasizes information instead of mere processing. The major goal of Web 3.0 is to describe a structure of the data and offer its connecting so that it is easy to find out, automate, reuse and integrate the data across a variety of applications. The layered architecture was proposed by Tim Berners-Lee for the Semantic Web as shown in Figure 3-4.

The major thought behind the semantics in the Web 3.0 was the formation of Web content through not using the natural language but a type of script that can be understood and gauged by the software agents to allow them to discover, share or combine information more efficiently and effortlessly, meeting the initial stepping stone towards smart applications. The central aim of Web 3.0 technology is to support the users of web to add information in manners so that computers can comprehend, process and trade. These developments in the Web technology would allow Web application to carry out a variety of tedious jobs such as collating information from mixed sources and efficiently support users to explore related information as per their needs.

It facilitates the data to be connected from a source to any other source and to be understood by the computers in order to perform gradually more sophisticated jobs on our behalf. The Semantic Web is a net of information connected such a manner that can easy to process by machines, on a worldwide scale. We could imagine it as being a proficient way of showing the data as a globally linked database or on the World Wide Web.

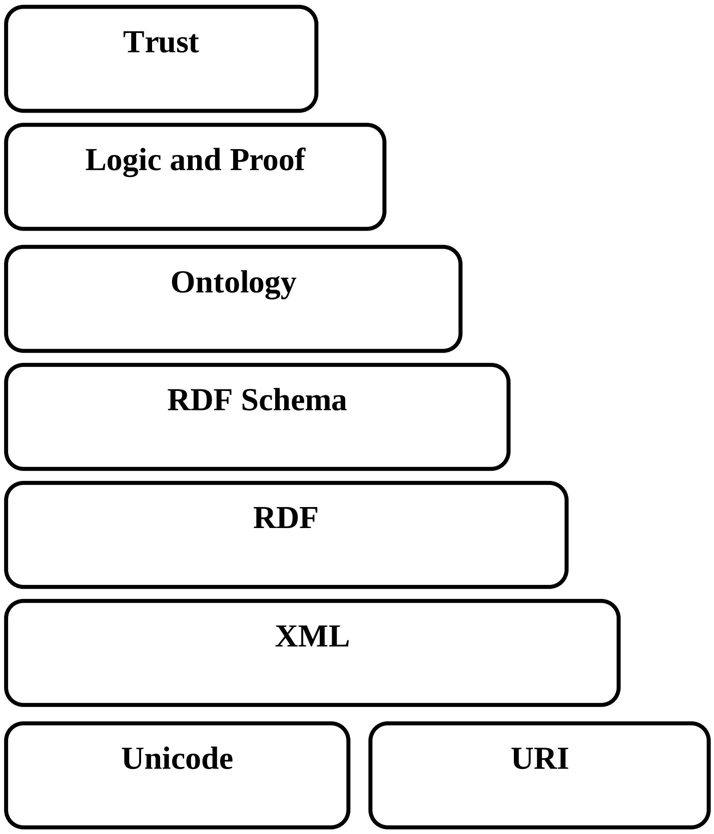

Figure 3-4: Semantic Web Layered Architecture

Tim Berners-Lee, who is the inventor of the World Wide Web, HTTP, HTML and URIs, was first thought up about the semantic web. A dedicated team of people at the World Wide Web Consortium is operational to advance, expand and regulate the system, and a lot of languages, tools, publications; and so on, have by now been developed. Though, Semantic Web technologies are yet in their initial stage, and although the prospect of the project in common appears to be bright, there seems to be slight compromise about the characteristics and likely direction the early Semantic Web.

Semantic Web is usually developed on the syntaxes, which utilize Uniform Resource Identifiers (URIs) to signify data, generally in triples based structures: that is several triples of URI data that could be stored in the databases, or interchanged on the World Wide Web by means of a set of exacting syntaxes designed particularly for the assignment. These syntaxes are known as "Resource Description Framework" syntaxes (RDF).

3.5.4 URI (Uniform Resource Identifier) And Unicode

A Uniform Resource Identifier (URI) is merely a Web identifier similar to the strings starting with "ftp:" or "http:" that we frequently come across on the World Wide Web. Anybody can make a URI, and the possession of them is visibly delegated, so they figure a perfect base technology with which to develop a global Web on the top of. Actually, the World Wide Web is such kind of thing: anything that has a Uniform Resource Identifier (URI) can be considered as "on the Web".

3.5.5 RDF-Resource Description Framework

The Resource Description Framework (RDF) is a standard model for data exchange on the World Wide Web. The RDF has characteristics that support data merging even if the original schemas vary, and it particularly supports the development of schemas over time with no requirement of all the data consumers to be altered.

The RDF extends the connecting structure of the World Wide Web to employ URIs to define the relationship among things and the two ends of the link (this is generally referred to as a “triple”). By means of this simple model, it facilitates structured and semi-structured data to be integrated, uncovered, and distributed across the diverse applications.

This connecting structure makes a labeled, directed graph, where the edges represent the defined link among two resources, given by the graph nodes. The above discussed graph view is the easiest probable mental model for the RDF and is frequently utilized in easy to follow and understand the visual explanations. The RDF language namespace prefix is usually rdf: and is (syntactically) defined at http://www.w3.org/1999/02/22-rdf-syntax-ns#. RDF vocabulary includes the following elements:

rdf:type – it is a predicate used to state that a resource is an instance of a class

rdf:XMLLiteral – the class of typed literals (i.e., of XML literal values)

rdf:Property – the class of properties (i.e. binary relations that are used as predicates in triples)

rdf:Alt, rdf:Bag, rdf:Seq – containers of alternatives, unordered containers, and ordered containers

rdf:List – the class of RDF Lists

rdf:nil – an instance of rdf:List representing the empty list

rdf:Statement, rdf:subject, rdf:predicate, rdf:object – used for reification. RDF reification allows to disassemble a statement (triple) to its parts and to use the whole statement or parts of the statement as a part of other triples. The whole triple can then be treated as a resource which allows to make assertions about the statement.

For example, for the statement

: john : has : cat

the RDF reification is as follows:

[a rdf:Statement;

rdf:subject :john;

rdf:predicate :has;

rdf:object :cat] .

To summarize, RDF triple is a triple $< subject; predicate; object >$ where *subject* can be URI or b-node, *predicate* can be URI, and *object* can be URI, b-node, or literal. RDF graph is a set of RDF triples. Formal semantics for RDF is defined using model theory and is available in Hayes (2004).

3.5.6 RDF Schema (RDFS)

RDF is extended by RDF Schema (Brickley and Guha, 2004) (RDFS). RDFS extends vocabulary of RDF by adding taxonomies of classes and properties. In addition, some elements definitions is extended by RDFS. RDFS sets the domain and range of properties; it also relates the RDF classes and properties into taxonomies using the RDFS vocabulary.

The following example shows RDFS vocabulary, showing taxonomy of classes and properties and usage of range and domain of properties:

```
@prefix : <http://www.example.org/sample.rdfs#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>        .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
:Sugarcane      rdfs:subClassOf :Crops.
:Loams          rdfs:subClassOf :Soil.
:ClayeyLoams rdfs:subClassOf :Soil.
:hasBagasse     rdfs:range :Crops;
                rdfs:domain : Crops.
:hasMoisture    rdfs:subPropertyOf :hasBagasse.
:Farm1          a : Sugarcane.
                :hasMoisture :80.
:Field1         a :Loam.
```

Classes shows the group of resources. All resources can be divided into groups of classes. Classes are also resources, so they are identified by URIs. Properties are the description of classes. The class members are instances of the classes, which is stated using the rdf:type property. A set of instances may represent extension of the class. Two different classes may contain the same set of instances.

In RDFS a class may be an instance of a class. All resources are instances of the class rdfs:Resource. All classes are instances of rdfs:Class and subclasses of rdfs:Resource. All literals are instances of rdfs:Literal. All properties are instances of rdf:Property. The rdfs:subClassOf (subclass-of relation) and rdf:type (instance-of relation) for all RDFS classes are shown in the Table 3-1: RDFS Classes.

The list of classes defined by RDFS is shown in the Table 3-1: RDFS Classes.

Table 3-1: RDFS Classes

Element	**Class of**	**rdfs:subClassOf**	**rdf:type**
rdfs:Resource	all resources	rdfs:Resource	rdfs:Class
rdfs:Class	all classes	rdfs:Resource	rdfs:Class
rdfs:Literal	literal values	rdfs:Resource	rdfs:Class
rdfs:Datatype	datatypes	rdfs:Class	rdfs:Class
rdf:XMLLiteral	XML literal values	rdfs:Literal	rdfs:Datatype
rdf:Property	properties	rdfs:Resource	rdfs:Class
rdf:Statement	statements	rdfs:Resource	rdfs:Class
rdf:List	lists	rdfs:Resource	rdfs:Class
rdfs:Container	containers	rdfs:Resource	rdfs:Class
rdf:Bag	unordered containers	rdfs:Container	rdfs:Class
rdf:Seq	ordered containers	rdfs:Container	rdfs:Class
rdf:Alt	containers of alternatives	rdfs:Container	rdfs:Class
rdfs:Container Membership Property	rdf:_1… properties expressing membership	rdf:Property	rdfs:Class

Properties in RDFS are relations between subjects and objects in RDF triples, i.e., predicates. All properties may have defined domain and range. Domain of a property states that any resource that has given property is an instance of the class. Range of a property states that the values of a property are instances of the class. If multiple classes are defined as the domain and range then the intersection of these classes is used. Ranges and domains for RDFS properties are summarized in the Table 3-2: RDFS Properties

Table 3-2: RDFS Properties

Element	Relates	rdfs:domain	rdfs:range
rdfs:range	restricts subjects	rdf:Property	rdfs:Class
rdfs:domain	restricts objects	rdf:Property	rdfs:Class
rdf:type	instance of	rdfs:Resource	rdfs:Class
rdfs:subClassOf	subclass of	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	subproperty of	rdf:Property	rdf:Property
rdfs:label	human readable comment	rdfs:Resource	rdfs:Literal
rdfs:comment	human readable label	rdfs:Resource	rdfs:Literal
refd:member	container membership	rdfs:Resource	rdfs:Resource
rdf:first	first element	rdf:List	rdfs:Resource
rdf:rest	rest of list	rdf:List	rdf:List
rdf:_1, rdf:_2, . . .	container membership	rdfs:Container	rdfs:Resource
rdf:seeAlso	further information	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	definition	rdfs:Resource	rdfs:Resource
rdf:value	for structured values	rdfs:Resource	rdfs:Resource
rdf:object	object of statement	rdf:Statement	rdfs:Resource
rdf:predicate	predicate of statement	rdf:Statement	rdfs:Resource
rdf:subject	subject of statement	rdf:Statement	rdfs:Resource

The taxonomy of classes is formed by property rdfs:subClassOf, taxonomy of properties is formed by property rdfs:subPropertyOf.

3.5.7 WEB ONTOLOGY LANGUAGE (OWL)

The Web Ontology Language OWL (W3C Recommendation, 2004) extends RDF and RDFS. OWL is designed for use by applications that need to process the content of information instead of just presenting information to humans in McGuinness and van Harmelen (2004). Building upon RDF and RDFS, OWL provides more machine-interpretable semantics by defining additional vocabulary along with formal semantics. OWL builds on Description Logics, which is a restriction of First Order Logic. OWL provides three increasingly expressive sublanguages: OWL Lite, OWL DL (Description Logics), and OWL Full. Each of these sublanguages is an extension of its simpler predecessor. Compared to the other two sublanguages, OWL DL is often chosen as the ontology modeling language because of its capacity of fair semantics expressiveness and inference. Most available OWL reasoners support OWL. An OWL ontology usually consists of classes, properties, instances of classes, and relationships between these instances. Instances of classes in OWL are called individuals. OWL classes are described through "class descriptions", which can be combined into "class axioms" in Dean, Schreiber, Bechhofer, van Harmelen, Hendler, Horrocks, ... and Stein (2004).

3.5.8 Purpose

The major goal of the Semantic Web is driving the development of the existing Web by enabling users to discover, distribute, and integrate information without difficulty. Humans are able of utilizing the Web to perform jobs like locating the Estonian translation for "twelve months", reserving a library book, and locating for the minimum price for a DVD. Though, machines can’t complete all of these tasks with no human direction, as web pages are developed to be read by humans, not equipments. The semantic web is an idea of information that can be eagerly interpreted by machines, so that machines can execute more of the tedious jobs involved in locating, integrating, and acting upon the information on web.

Semantic Web, as initially envisioned, is a system that facilitates machines to "recognize" and react to complex requests by humans based on their sense. This kind of "understanding" requires that the related information sources be semantically ordered.

Several view this as a mixture of semantic web and artificial intelligence (AI). The semantic web will educate the computer about what the data means, and this will grow into artificial intelligence that can use that information.

3.5.9 AGENTS AND THE SEMANTIC WEB

Software agents are successfully used in various types of applications (Nicholas, Sycara and Wooldridge, 1998; Weiss, 1999; Michal, et al. 2006; Michael, 2009). The main obstacle in agent-based system is sharing and distributing knowledge (Hendler, 2007). For sharing and distributing knowledge OWL ontologies are incorporated within agent based systems (Michal et al., 2012; Anjalee, et al., 2005; Walton, 2006; Youyong, et al., 2003).

According to (Burstein et. al. 2005; Huhns 2006; Huhns et. al. 2006; LUCK et. al. 2006; Singh and Huhns 2005) software agents play an important role in semantic web services.

3.5.10 SENSOR WEB

According to Jet Propulsion Lab (JPL) at NASA (Delin and Jackson, 2001) the Sensor Web is:

”... a system of intra-communicating spatially distributed sensor pods that can be deployed to monitor and explore new environments.”

The revised version (Delin, 2006):

”... is a type of wireless network of sensors that acts as a single, autonomous macroinstrument. It is a temporally synchronous, spatially amorphous network, creating an embedded, distributed monitoring presence.”

So, the Sensor is a network composed of sensors, which are designed to achieve a desired goal. There may be various isolated sensor networks which may be separately deployed and controlled.

A broader definition is given by Liang et al. (2005).

”The Sensor Web is a revolutionary concept towards achieving a collaborative, coherent, consistent, and consolidated sensor data collection, fusion and distribution system. The SensorWeb can perform as an extensive monitoring and sensing system that provides timely, comprehensive, continuous and multi-mode observations.”

According to Open Geospatial Consortium (OGC), a global industry consortium representing over three hundred organizations, define (Botts, Percivall, Reed and Davidson, 2007) a sensor network as:

”... a computer network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations.”

and then define a Sensor Web as:

”... a web accessible sensor networks and archived sensor data that can be discovered and accessed using standard protocols and application programming interfaces (APIs).”

For monitoring and understanding our natural environment sensor web (Figure 3-5) is essential part for making decisions, which are based on type of sensor information collected from the natural environment.

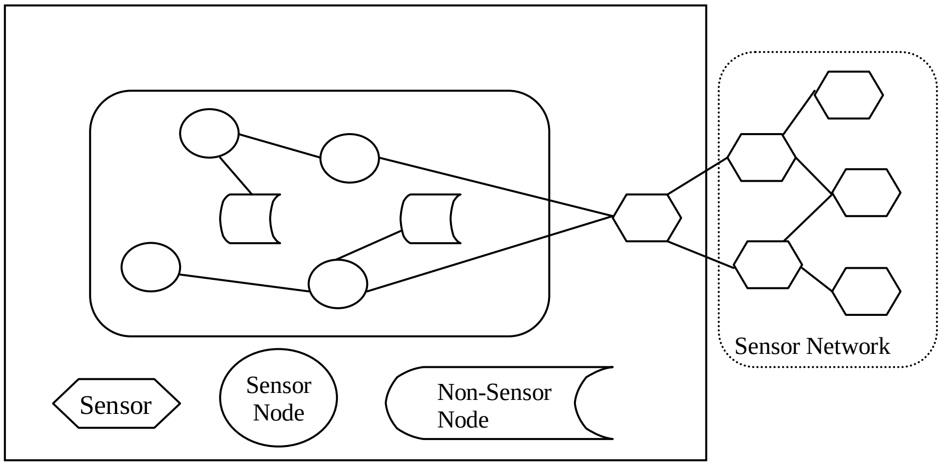

Figure 3-5: Sensor Web

3.5.11 AGENT COMMUNICATION

The sensor nodes are part of sensor agents, which uses semantic web languages for communication. As described in (Obitko. and Marik, 2004, Obitko, and Snasel, 2004), RDF resource represents an object, and RDF statement represents a proposition. An action instance can be described by its act, actor, and parameters, which is modeled by an OWL class.

For proper communication, agents must have following characteristics:

- *Deliver and receive messages* – there must be an agreement at the physical and network layers between agents, so that they can send and receive objects of strings, which is a message.
- *Parse the messages* – agents must be able to parse messages at the syntactic level, to correctly decode the message to its parts, which may be message content, language, sender etc.
- *Understand the messages* – a shared or explicitly defined ontology must

be available for the agents, so at the semantic level, the parsed symbols must be understood, which is required to decode the information contained in the message.

Each agent has to know something about a domain he is working in and also has to communicate with other agents. An agent is able to communicate only about facts that can be expressed in some ontology. This ontology must be agreed and understood among the agent community (or at least among its part) in order to enable each agent to understand messages from other agents.

The ontology agent (OA) need to provide some or all of the following services (Fipa, 2001):

- discovery of public ontologies in order to access them
- help in selecting a shared ontology for communication
- maintain (e.g. upload, download, or modify) a set of public ontologies
- translate expressions between different ontologies and/or different content languages
- respond to queries for relationships between terms or between ontologies
- facilitate the identification of a shared ontology for communication between two agents

Every OA must be able to perform these tasks for agent communication.

The ontology agent may not provide some of the services above.

3.6 CONCLUSION

This chapter gives a brief overview of the various technologies such as Activity Theory, Multi Agent Systems and Semantic Web that will be applied later in developing an Adaptive Business Environment framework.

CHAPTER 4

DIVERSITY AND CONVERGENCE ISSUES IN AGRICULTURE SCIENCE

4.1 APPLICATION OF EVOLUTIONARY OPTIMISATION TO AGRICULTURE SCIENCE

Many conflicting objectives are handled by multi objective optimization. In this situation, one cannot derive one solution, which is optimizing all the objectives simultaneously. Now, it is required to find multiple trade-off solutions. Evolutionary Algorithms are well suited for solving multiobjective optimization problem and it leads to a burning research area called “Evolutionary Multiobjective Optimization” by Zhou, Qu, Li, Zhao, Suganthan and Zhang (2011). The first development in the area of EMO has been started with the development of algorithm VEGA (Vector Evaluated Genetic Algorithm) by Schaffer in 1984. Several issues have been proposed in EMO, like non-dominated sorting, niching mechanism, elitism, diversity and convergence.

We extend the above problem in sugar manufacturing process. The main objectives identified in sugar manufacturing process involve,
$Firing\ control\ of\ boilers\ \rightarrow 'C_B'$, $Sugarcane\ recovery\ \rightarrow 'Sr'$ and

$Transportation\ and\ delivery\ mechanism \rightarrow 'TD_m'$

4.1.1 Firing control of boilers (C_B)

It consists of all the parameters required to control the speed of boiler thereby eliminating the waste of firing material like bagasse. The signals are being sent from Central Mill Control through Supervisory Control And Data Acquisition (SCADA). SCADA gets the signal from the Main Control Center which is a block receiving signals from sensor Mesh and Raster Scan images. Signal control is being achieved by DNP3 protocol of SCADA to computing relay which governs the steam fire.

4.1.2 Sugarcane recovery (S_r)

Sugarcane recovery depends on several small parameters like Method of Planting, Seed Rate, Spacing between sugarcane, Organic Measures, Maturity and Harvesting.

4.1.3 Transportation and delivery mechanism (TDm)

TD_m refers to delivery of sugarcane without waiting, as more is the waiting time more will be drying of sugarcane, hence less will be recovery. During start of the production process boiler is run at maximum speed, therefore sugarcane arriving at the mill should be thick and should have high juice value so that boiler energy is full converted into juice.

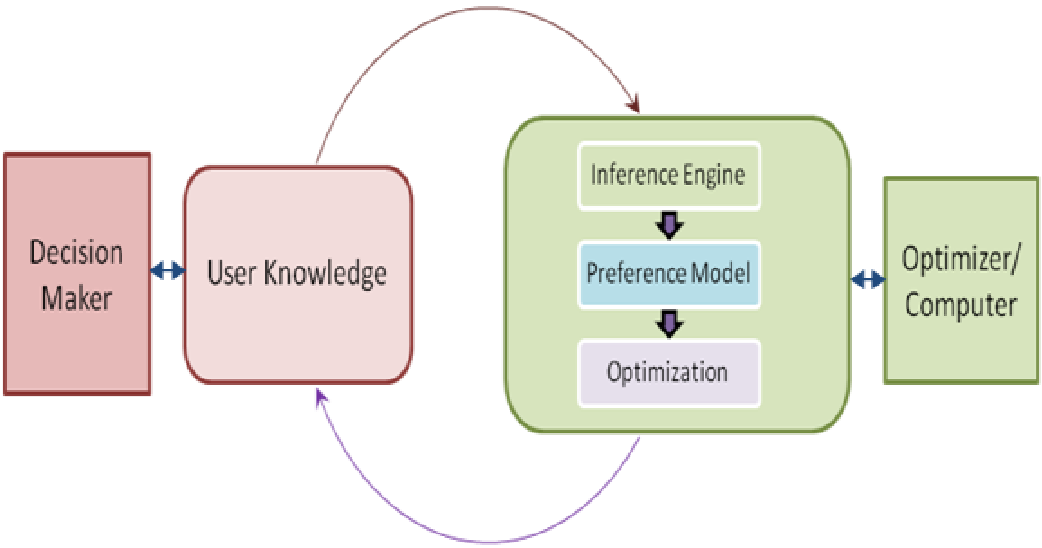

Figure 4-1: IMO Learning Cycle (Adapted from V. Belton et al., 2008)

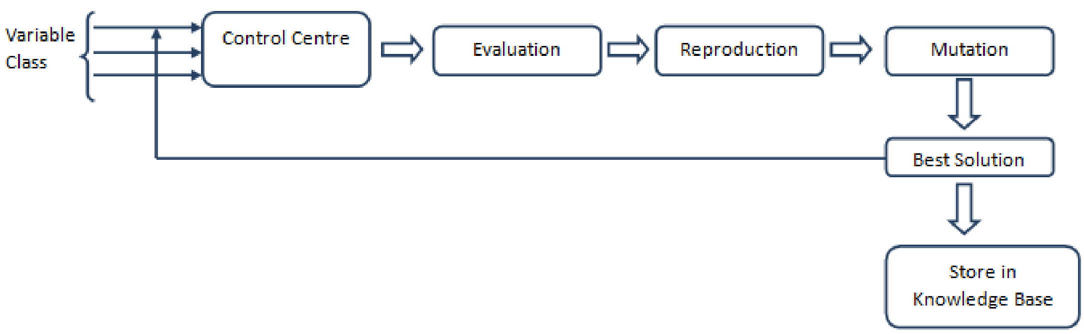

Figure 4-2: Interactive Learning & Storage Process

4.2 EMO INFERENCE MODEL

In order to achieve higher quality sugar we need to have all these parameters optimised. Since it is a continuous process, we use Interactive Multiobjective Optimization (IMO) and apply it on learning perspective. Learning here relates to "Farmer Learning" and "Machine Learning" relating to sensor networks and SCADA control system. The learning cycle of IMO (V. Belton et al.,

2008) has Decision Maker (DM) which provides information about the preferences needed from the point of view of generating a reference model and plurality of value functions are generated. The Decision Maker uses the User Knowledge and process it on the basis of three basic parameters: Inference Engine, Preference Model, Optimization of knowledge rules based on these parameters C_B, S_r, TD_m respectively.

In order to develop a mathematical model consider a decision, 'D' consisting of decision classes Cl= $\{Cl_1, Cl_2, \ldots, Cl_m\}$; in case of sugar industry we have three basic classes $\{C_B, S_r, TD_M\}$.

In order to apply interactive learning to the above classes we need to apply multi-criteria sorting and learning. Under these three basic classes we have three subclasses. The condition attributes are criteria and decision classes arranged in upward or downward manner depending on the current situation. If the dominance of these classes is to be identified we use "if-then" rules as described by the maximization function below:

$$\begin{array}{ll} \text{Maximize } f_{o/p}(x) & ;o/p=(1,2,\ldots,M) \\ \text{subject to } TD_m>=0 & ;TD_{m\,min}<TD_m<TD_{max} \\ \qquad C_B>=0 & ;C_{B\,min} < C_B < C_{B\,max} \\ \qquad S_r>=0 & ;\ Sr_{min} < S_r < Sr_{max} \end{array}$$

Solution vector 'x' of three decision variables needs to be identified.

In order to proceed with the process the classes needs to be iterated and randomly combined by changing by changing their values from "min" to "max". The two values of classes are then mutated to generate a good solution. The second part of the work includes extension of the work using Ontology creation and Internet of things for linking sugarcane production and sugar generation.

4.3 REFINING BUSINESS DYNAMICS OF SUGARCANE INDUSTRY USING ONTO INTERNET OF THINGS

India is the second largest producer of sugarcane in the world. During last two decades there are number of sugar mills that have cropped up in entire North India and Western India. Using combination of ontology, sugarcane, farming, GIS and internet of things with integration of SCADA we will be proposing a framework that can fine-tune the sugarcane farming system and raw sugar processing. To develop the coherency between the two entities we focus of Sugarcane Farming System (SFS).

The framework for application of activity theory in Semantic Web for Dynamic Business Environment is shown in Figure 4-3. In this model we use SAX (Simple API for XML) to parse the website in HTML to XML parser. We use XSLT (Extensible Style sheet Language Transformation) to convert XML to RDF. RDF streams are then stored in Knowledge Base. RDF is stored in Sesame open source framework for storage and querying RDF.

4.3.1 Sugarcane Farming System:

Organic Manures increases the soil health significantly, sensor probes (Figure 4-4, Figure 4-5) helps in determining the level of productivity of soil. The basic idea is to give information to the Control Centre, which finally passes the information through SMS to the farmer. For development of Soil Ontology and Sensor Ontology, there are ontologies like Lyra ontology system (H. Beck et al., 2010).

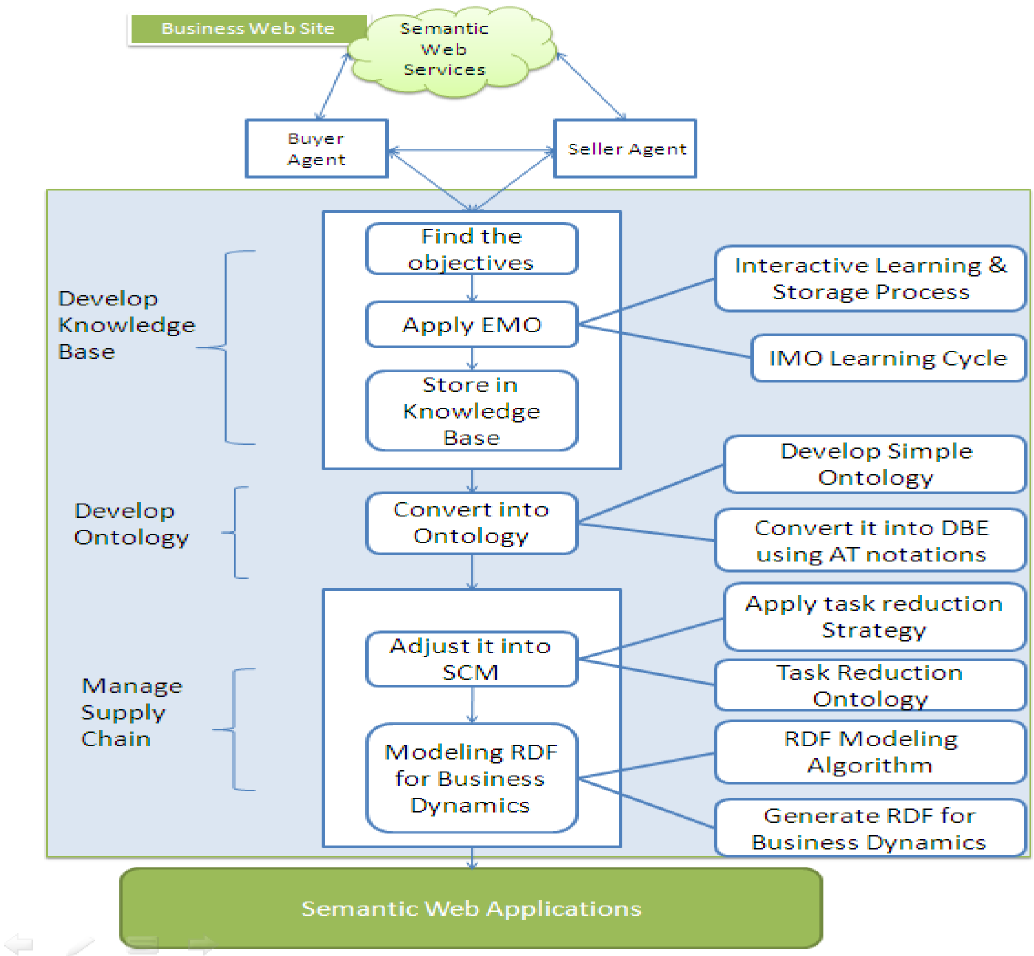

Figure 4-3: Framework for Application of Activity Theory in Semantic Web for Dynamic Business Environment (Web-Eco-AT framework)

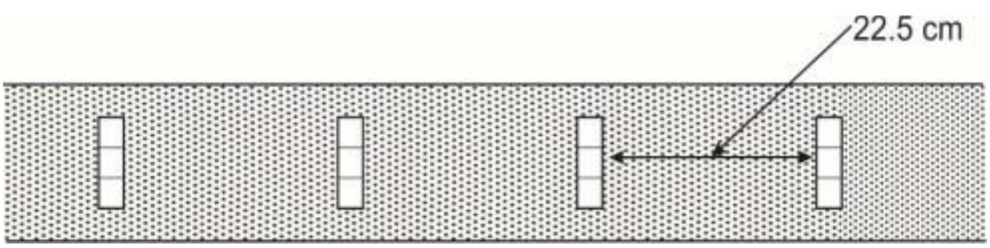

Figure 4-4: Ferrous set placement

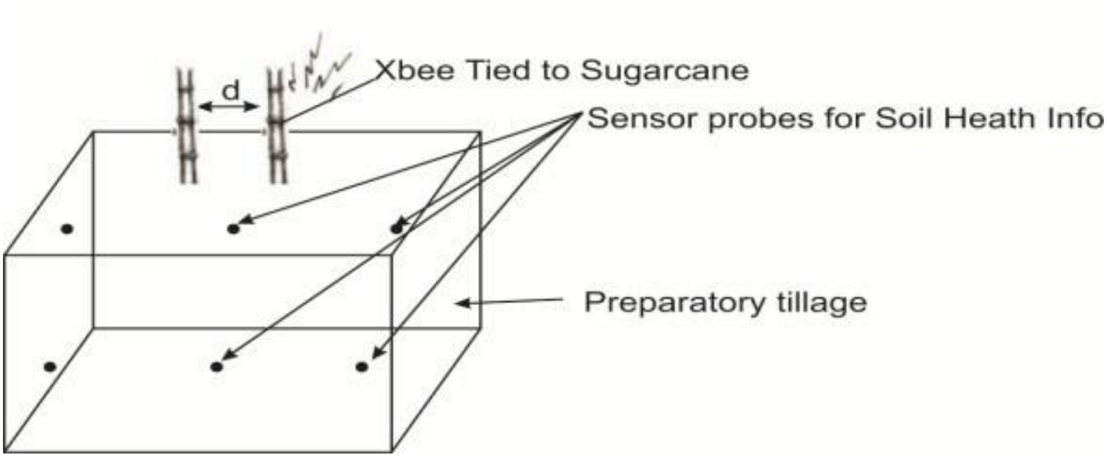

Figure 4-5: A soil Cell (adopted from H. Beeket al., 2010) with sensors and Xbee Transreceiver

4.3.1.1 Raw Sugar Processing:

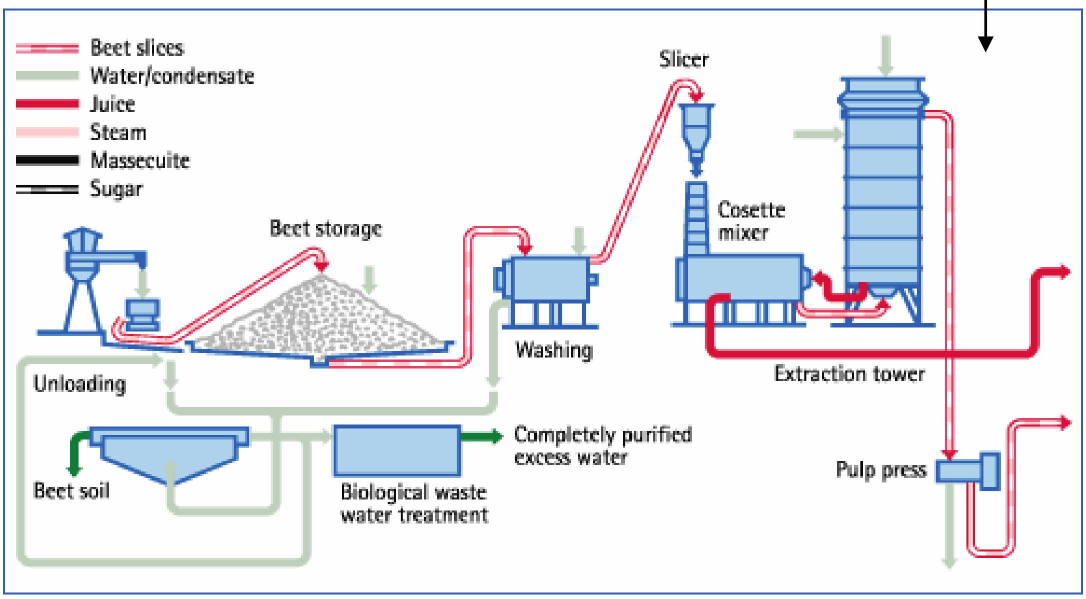

Figure 4-6: Raw Sugar Processing

As shown in Figure 4-6 the process starts with crushing of sugarcane followed by the process of shredding the cane billets into fiber which is conveyed to the milling train for juice extraction. The shredded can is fed into the crushing mill consisting of three large rollers. It then dewaters the residual cane fiber, which is used as a boiler fuel. Finally the steps like Filtration, Evaporation and Crystallization are done. The main idea is to regulate the Boiler according to the harvest based on ontology. We will be able to automatically adjust the selling through Resource Description Foundation (RDF) i.e. creation of Knowledgebase.

4.3.2 Control Centre:

The main task of control centre as shown in Figure 4-7 is to receive signals from various sensors/ XBee[1] and pass it so the SCADA system to monitor the control of boiler accordingly. Along with processing of signals, it uses the techniques of Raster GIS to find out the harvest quality of sugarcane as shown in Figure 4-8.

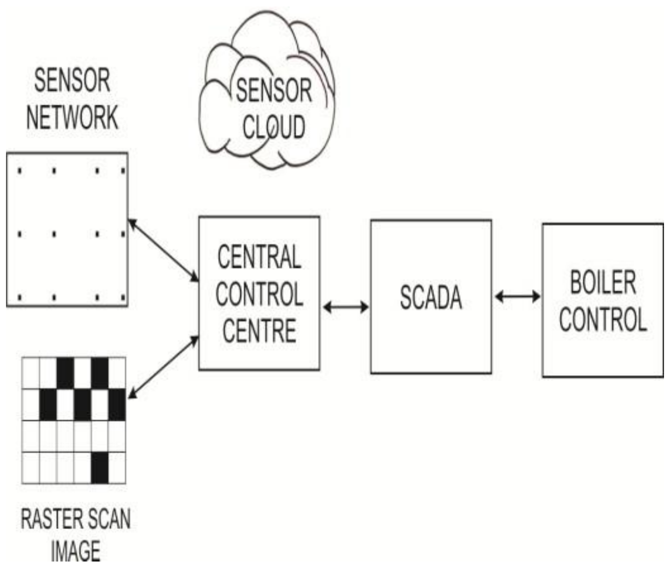

Figure 4-7: Block Schematic of Flow Control

[1] XBee is the brand name for Digi International for a family of form factor compatible radio modules.

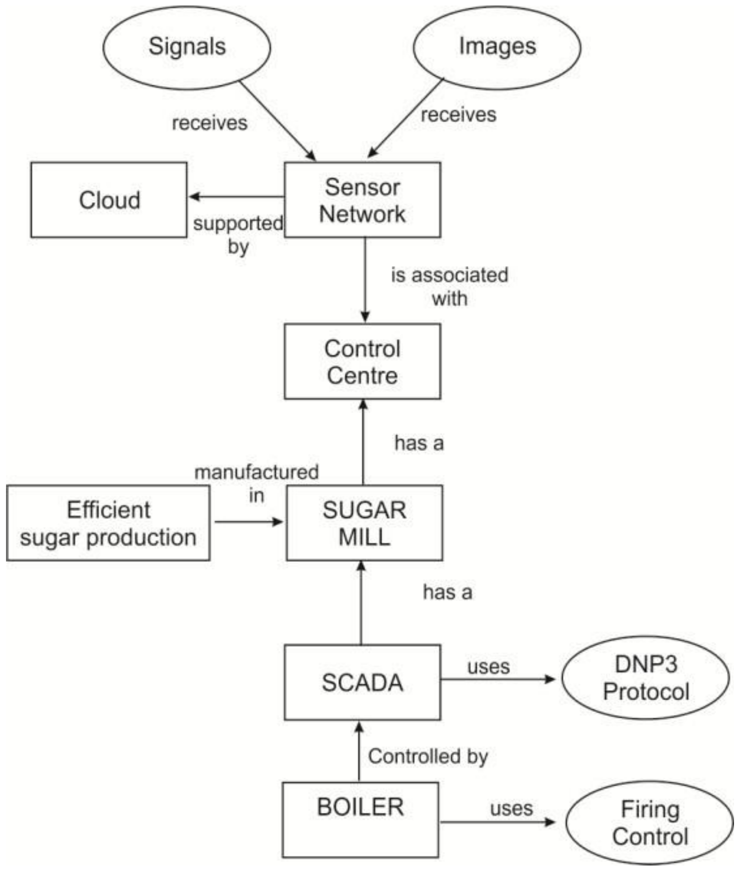

Figure 4-8: A Simple Flow Control Ontology

4.3.3 Development of Ontology:

An ontology file (Beck, 2008; Cuske, Dickopp and Seedorf, 2005) is an RDF description that contains all possible predicates for an application. It consists of vocabulary to define the domain and range of a predicate. By the help of ontology we can define the range to indicate that the predicate is of a particular data type. The ontology is a rhythmic taxonomic organization which allows categorization of objects. From these taxonomic graphs, we write RDF for control centre. Ontologies are nowadays providing a formal representation of objects and their relationships. The above flow control can be formalised by developing taxonomy. We extend the simple ontology in RDF schemas using the basic principle. Ontology is a triplet consisting of:

1. *Concepts = {Control Centre, SCADA, Boiler, Cloud, Sugar Mill, Sensor Network}*

   it is a set of concepts of real world objects.
2. *Roles = {Controlling, Sensing}*
3. *Inheritance = {Sensor Network, Raster Images, Cloud, DNP3, Firing Controller Circuit}*

```
<rdf: RDF
xmlns: rdf = "http://www.w3.org/1999/02/22-rdf-ns#"
xmlns: rdfs = "http://www.w3.org/2000/01/rdf-schema#"
xmlns: owl = "http://www.w3.org/2002/07/owl #"
xmlns: dc = "http://sugarcane.org/dc/elements/1.1/"
xmlns: sugarcane = "http://www.linkeddatatools.com/sugarcane#">
<! OWL class definition- sugarcane Process Control --->
<owl: class rdf: about ="http://www.linkeddatatools.com/sugarcane# sugarmill">
<rdfs: label> sugar mill </rdfs: label >
</owl: class >
<! - - OWL Sub class definition - Controlcentre - - >
<owl: class rd?: about = "http://www.linkeddatatools.com/sugarcane# Control
Centre">
<! - - Control Centre is a subclassification of Sugarmill - - >
<rdfs: subclass of rdf: resource =" "http://www.linkeddatatools.com/sugarcane#
control centre"/>
<rdfs: label> Sensor Network </rdfs: label>
< rdfs: comment> Sensor Networks receives signals and Images </rdfs: comments>
</owl: class > <! - OWL Subclass Definition - SCADA - - >
< OWL: Class rdf: about = "http://www.linkeddatatools.com/sugarcane#
sugarmill"/>
<rdfs: subclass of rdf: resource = " "http://www.linkeddatatools.com/SCADA#
DNP3">
< rdfs: comment > DNA3 is a protocol of SCADA to transmit signals> </ rdfs:
```

comment>

</owl: class > <! - - OWL subclass Definition - Boiler - - >

< rdfs: label > firing control </rdfs: Label >

< rdfs: comment > firing control of Boiler is done by signals from SCADA . </rdfs: comment>

</rdf: Description > </rdf: RDF >

The above ontology links the sugar mills' website with the enhanced capability of web service and tries to relate the context depending on situation of thing. The sugar mills' website is linked with control centre which receives the signals from sensor informing about the current state of sugarcane yield and the raster image highlights the yield density to be processed in three stages as in Figure 4-9.

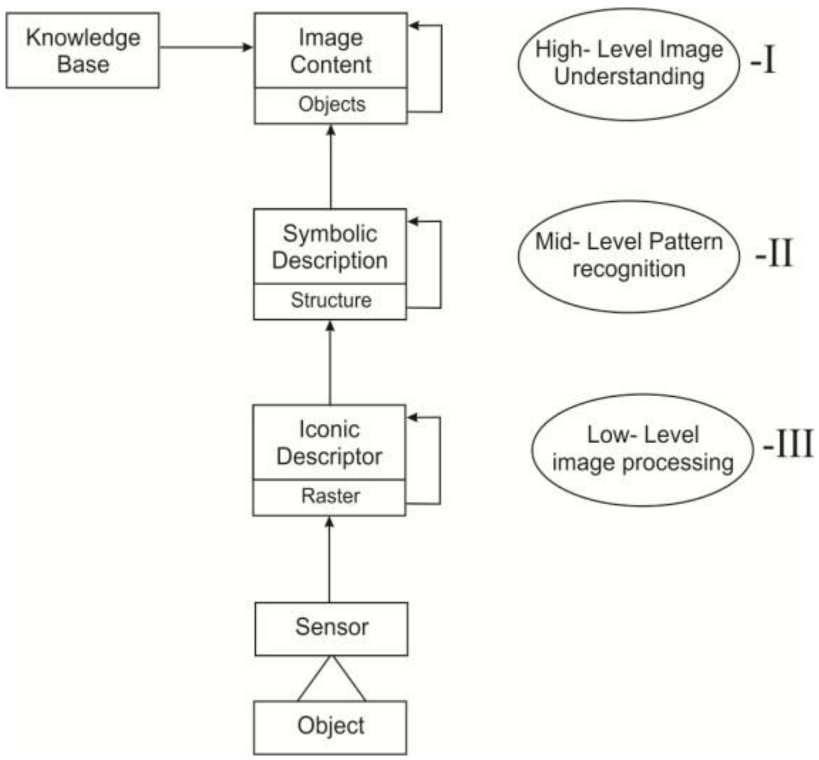

Figure 4-9: Information extraction (Adapted from Ehlers et al. 1989)

The cluster showing the good harvest support dark green color intimating the control centre to pass on the information to Mill during early start of the crushing. Secondly the message received through Xbee via Tera term on windows operating system needs to be sent through DNP3 protocol. XBee Tera term receives signals from other XBee arranged in Mesh Network as in Figure 4-10. API format for Remote Transfer has the following fields starting with Delimiter, Length and Frame - specific data. DNP3 message passing between device and control centre is achieved via public switched telephone network (PSTN).The whole automation process is being achieved by the help of Smart web service start with context of things service, domain knowledge service, which provides an update of new developments in sugarcane farming and end up in energy efficient control of Boilers. It provides a sustainable method of achieving the overall efficiency of the system.

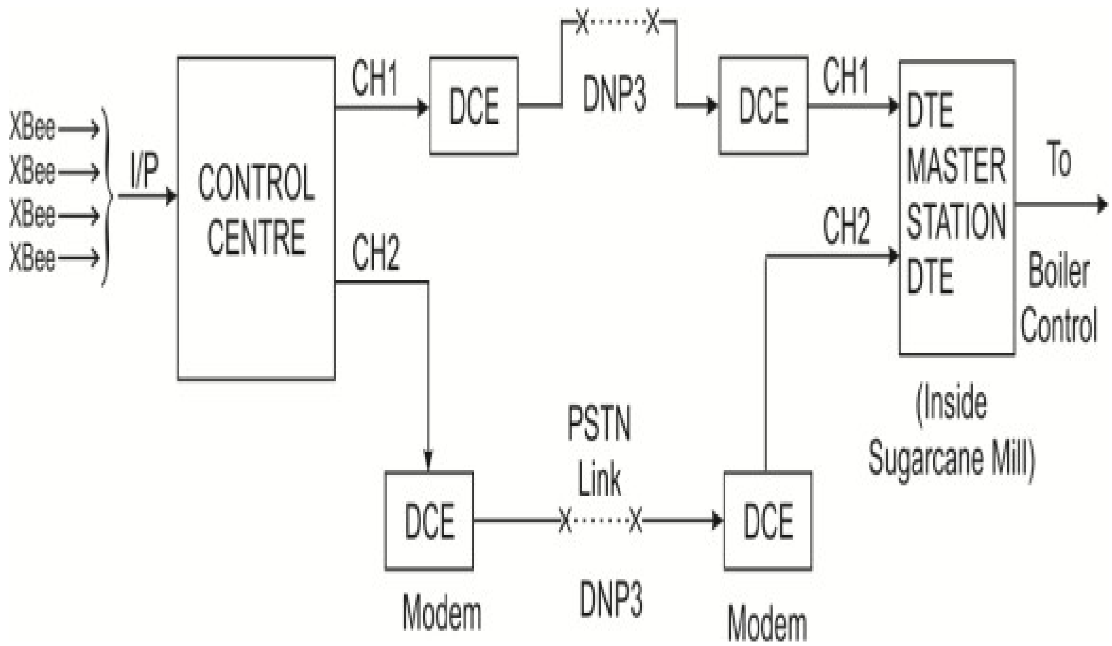

Figure 4-10: Message Passing between XBee to Boiler Control

4.4 CONCLUSION

The chapter presents a novel idea of creating sustainable framework which using the concept of context of things and ubiquitous environment. The two entities are integrated using ontology, which provides a ubiquitous web service environment for sugarcane industry.

CHAPTER 5

FRAMEWORK FOR DYNAMIC BUSINESS ENVIRONMENT

5.1 INTRODUCTION:

This chapter is divided into two parts, first part covers the development of architecture for dynamic business environment using the concept of Activity Theory and economic principles, and the model is extended further by taking Supply Chain Dynamics and Work Breakdown Strategy (WBS) into Semantic Web.

> *"The semantic web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."*

-Tim Burners-Lee, James Hendle, On Lassila.

The birth of semantic web in Sheu; Yu, Ramamoorthy, Joshi.and Zadeh (2010) came from use of Natural language in indexing descriptions from the document text. The polymorphic form of word makes this task very difficult in making the search ambiguous. Although there are number of refinements being done in natural language but none of the method were able to analyze information extraction and information retrieval without any ambiguity.

Fensel (2005) identified three-level solution to the problem of developing intelligent applications on the web, they are stated as:

- *Information Extraction*: It supports wrapping technology for uniform extraction of information.
- *Processable Semantics*: It deals with capturing information structures as well as meta-information about the nature of information.
- *Ontologies*: This is the process of converting taxonomies generated into resource description format that reflects the consensual and formal specification of the domain.

The main purpose of semantic web is to enable machine understandable web knowledgebase. Literal meaning of semantics is to study about the linguistic i.e. about the language. If we are to develop semantics for a model it is termed as Model Semantics. In order to deal with Semantic Heterogeneity in Sheu et al. (2010), Benatallah, Medjahed, Bouguettaya, Elmagarmid and Beard (2000), Sheth and Miller (2003) we use Ontology based information integration. It provides a common understanding of particular domain. It considerably eliminates ambiguity and provides a common understanding of the meaning of terms, thereby providing an excellent semantic interoperability. There are various approaches to employ the ontologies, in general three most popular ontologies are: single-ontology approach, multiple-ontology approach and hybrid ontology approach.

Paper by Deshpande, Ives and Raman (2007) describes an Adaptive Query Processing Engine. We are inspired by this work on Adaptive Question and have implemented it on modeling dynamic business behavior. SAWSDL-iMatcher in Wei, Wang, Wang and Bernstein (2011) helps us in building Ontology for economic production and costing. The paper by Wei et al. (2011) describes the development of matching strategies which are suitable for different tasks and contexts.

The concept of handling exceptional changes was stated by Brambilla, Ceri, Comai and Tziviskou (2005), it very well describes the dynamic workflow-driven work application, for example the user may land-up to a link which is an expired link.

We have extended the concept of Howse, Stapleton, Taylor and Chapman (2011) on “Visualising Ontologies: A Case Study”. Ellis and Keddara (2000) focuses on Modeling Language to Support Dynamic Evolution within Workflow Systems. The paper highlights the concept of modalities of changes, including change duration, change time frame. The concept of Activity Theory in Leontiev (1981) has been adopted from Kuttik and Engetrom which highlights the use of Activity Triangle in modeling a system.

5.2 DECISION SUPPORT SYSTEM

The basic idea is to provide a framework which can support sustainable environmental impact of goods and processes, beginning from extraction of raw materials to the final movement of end product.

The evolution of supply chain is very gradual starting from Mazumdar and Balachandran (2001) supporting three stage evolutionary process. Later on Ballon (2007) divided SCM into three periods, Past, Present and Future. Foremost advantage of SCM is its close collaboration with internet market and industry. Internet market is a platform where buyer and seller exchange information, do the transaction to satisfy both of them. The concept of Multiagent was started by (Fazel Zarandi et al., 2007) stating "the main interest of managers is to ensure that the overall cost is reduced and operations among various system are interchanged through coordination". Fox, Barbucaen and Teigen, 2000 also supported Agent Application in distribution, collaboration, autonomy and intelligence.

5.3 INTRODUCTION TO SUPPLY CHAIN MANAGEMENT

A supply chain is a network of facilities and distribution options that performs the functions of procurement of materials, transformation of these materials into intermediate and finished products, and the distribution of these finished products to customers. Supply chains in Nishat Faisal et al. (2006), Cucchiella et al. (2006), Tang (2006), Wu, Blackhurst and Chidambaram (2006), Goh, Lim and Meng (2007), Ritchie and Brindley (2007), Khan.and Burnes. (2007), Li and Chandra (2007), Jüttner. (2005) exist both in service and manufacturing organizations. Realistic supply chains have multiple end products with shared components, facilities and capacities. Usually, business unities along a supply chain operate independently, having their own objectives that are often conflicting with each other. Therefore, an essential condition to the success of a company is the conception of a strategy for coordinating the several business unities in a supply chain, leading to an effective management at strategic, tactical and operational levels. The efficiency of a supply chain is influenced by several factors, such as: stock management, production planning, production costs, scheduling and distribution strategies, and customer-specific demand, among others. Planning and modelling the production, stocking and distribution systems of a supply chain is an important support for decision making in a competitive market. According to Chaffey (2006) the several approaches for modelling and optimisation of a supply chain can be classified into five classes:

- Project of the supply chain
- Integer-mixed programming optimisation
- Stochastic programming
- Heuristic methods
- Simulation-based methods.

Modeling a supply chain has the following two purposes: to analyse the dynamics of the supply chain so as to identify strategies that minimise its dynamics Siddiqui, Darbari and Bansal (2012), Dhanda, Darbari, and Ahuja (2012), Darbari, Singh, Asthana, Prakash and Kendra (2010); and to validate an accurate model that represents a supply chain. Conventional numerical optimisation methods in supply chain design can get trapped in local maximum due to hill-climbing. Several problems of supply chain optimisation arise from difficulties in applying calculus-based analytical methods to parameter optimisation under constraint conditions.

Further, the objective functions needed in these numerical methods must be 'well-behaved'.

In most organizations, supply chain planning is the administration of supply facing and demand-facing activities to minimize mismatches, and thus create and capture value requires a cross-functional effort. A framework on supply chain planning in SCM is shown in Figure 5-1.

Figure 5-1: Basic Supply chain planning steps

Supply chain planning is focused on synchronizing and optimizing multiple activities involved in the enterprise from procurement of raw materials to the delivery of finished products to end customers Genetic algorithms and artificial neural networks have been applied to derive optimal solutions for collaborative supply chain planning. Work Breakdown Strategy is used for development of ontology (Figure 5-2).

The basic idea of intelligent Agent is to develop a framework supporting knowledge conceptualization and representation. Since supply chain and its linking is a very complex issue, we use the concept of Task Reduction in solving this problem. A fundamental and powerful concept of problem solving and simple representation states that if 'P' is a problem than it can be reduced into simpler problems $P_1, \ldots, P_n$. The solution $S_1, \ldots, S_n$ of the problems $P_1, \ldots, P_n$ can be combined into a solution S of P.

Minsky 1985 described intelligent agent as:

"An expert system that perceives its environment and finally draw inferences, to realize the set of goals."

Task reduction plays a crucial role in development of Multi Agent Systems in Uma, Prasad and Kumari (1993) described the various phases of Multi Agent Systems:

- Problem decomposition
- Sub-problem decomposition
- Sub-problem solution
- Sub-problem integration

The term "Task" is used to represent problem and for reduction of any task to any sub-task, it is essential to pinpoint the element of knowledge focusing on problem solving issues of the "Task".

In this methodology, the required problem solving steps consists of:

- Name of task
- Question relevant to the solution
- Sub Task or solutions identified by proceeding task

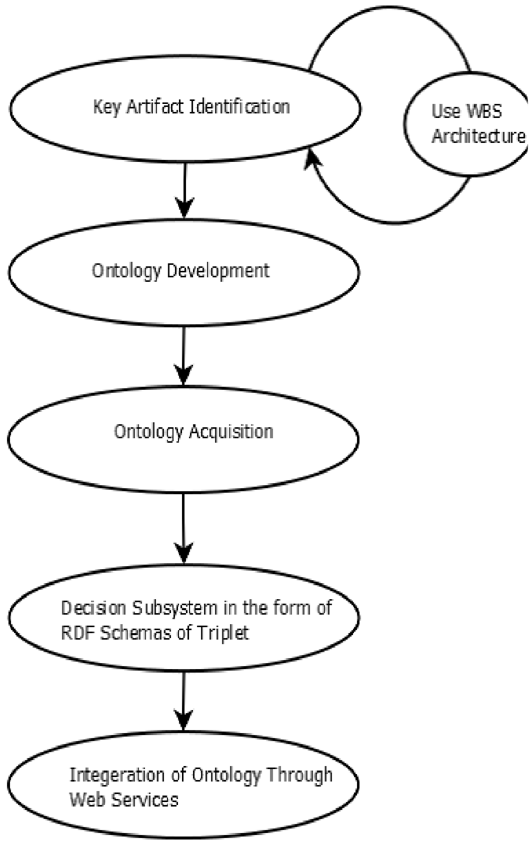

Figure 5-2: Work Breakdown Strategy in Ontology

Each task has its associated solution, and these solutions are integrated from bottom to top. This integration of solution moves up to form a final solution i.e. S1 (Figure 5-3)

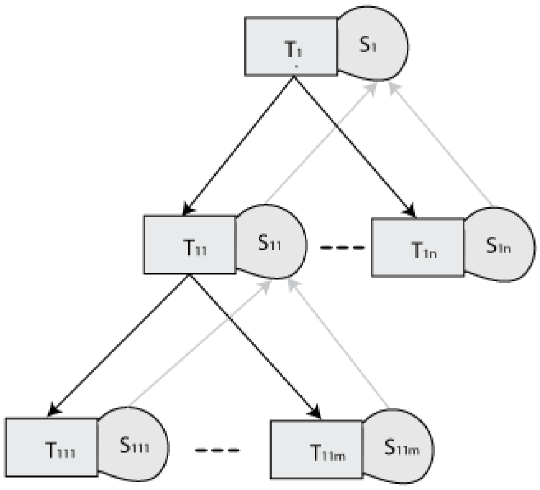

Figure 5-3: Task reduction diagram (Adapted from Michael Bownran, 2007).

In order to streamline the process of Internet Marketing and Supply Chain we use Porter's value chain creation. Porter value chain links the final selling of the product with a relationship in product with a relationship in product design concept.

5.3.1 Primary Activities of Porter's value Chain

- *Inbound logistics*: It refers to the inventory being procured from the vendors.
- *Operations*: All the activities/process involved in generation of final product is a part of Operation.
- *Outbound logistics*: All the distribution channels available for selling the finished goods like distribution centres, wholesalers, retailers or customers.
- *Marketing and Sales*: It deals with all the marketing mix is for promoting the product in the market.
- *Services*: All the supporting activities which is a part of the after sales and services is a part of services.

All these activities provide an excellent "competitive advantage" defined by Porter.

The protocols for interconnectivity provided through Supply Chain Management can be defined as in Figure 5-4. The Agent lifeline in sequence diagram (Figure 5-5) defines the period of buyer and seller exists which is decided according to XOR operator (Figure 5-6).

It depicts the condition that only one output is allowed. The single Agent lifeline which ends up only after completion of particular operation is shown by interleaved protocols. In order to verify any operation we use two kinds of action parameters <<proactive>> and <<reactive>>. For example, the buyer can only purchase the product when it is available in the stock.

$$\{[public][Stock = n]\ Buying\ Process\ [(Item, lists)][Stock\ =\ n - 1]\}$$

Now we have to work on business dynamics, the first step will be to find out the Task (Question-Answer) and then derive an Ontology which can be improved as required (Figure 5-7). Finally we are developing a Solution Tree which is a Formalized Solution set. The elaborated tree of the entire solution can be designed as in Figure 5-8.

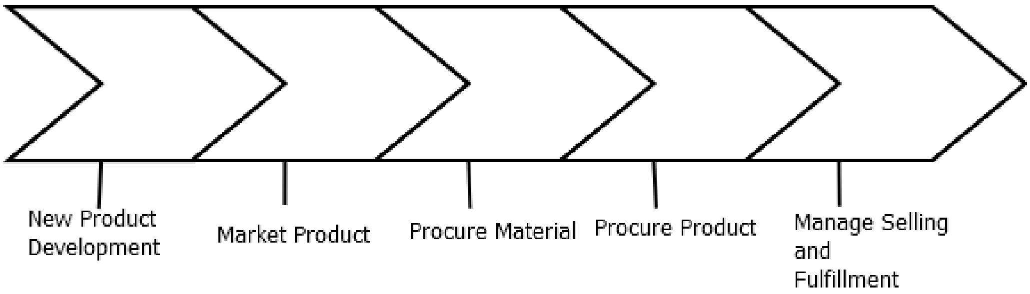

Figure 5-4: Porter's value chain model

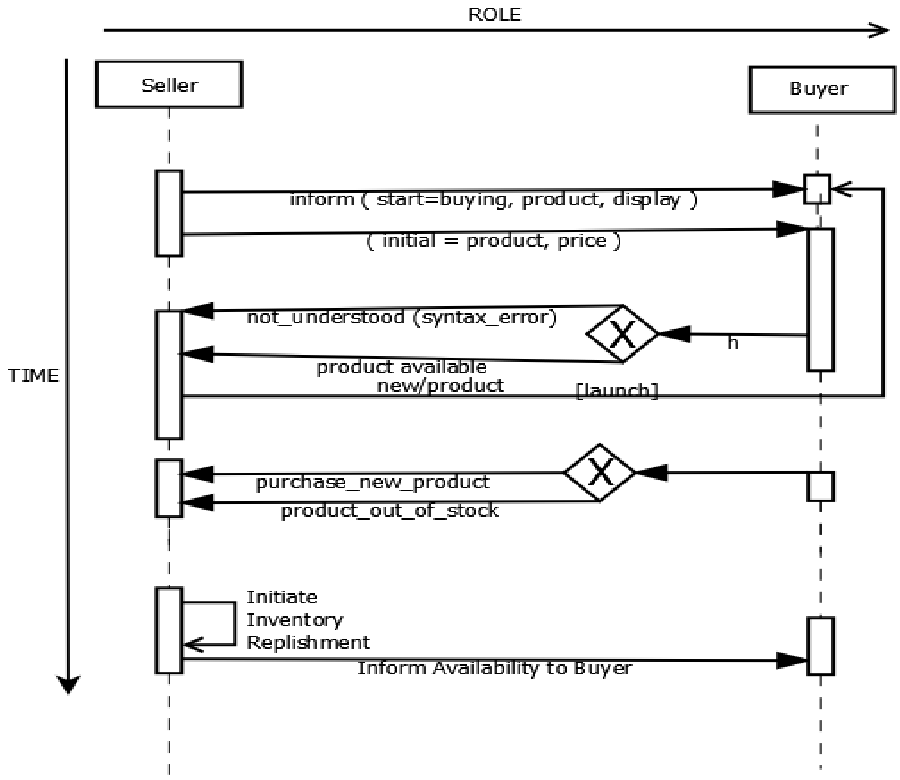

Figure 5-5: Protocol diagram of buyer and seller

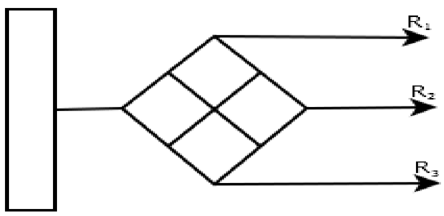

Figure 5-6: XOR operator

We can summarize the entire process of Task Reduction Ontology as:

$T_{11}Q_{11}\{(A_{111}, OI_{111}, T_{111})\ldots(A_{11m}, OI_{11m}, T_{11m})\}$

We are able to generate language to logic translation of integration ontology with transition state as:

$$[K_T = \{T_T Q_T (A_T, OI_T, S_T)\}]$$

$$Smooth\ flow\ of\ Product\ (T_1) = \{Coordination\ of\ Buyer\ Ontology\ and\ Seller\ Ontology\}$$

The task reduction in Dhanda, Darbari and Ahuja. (2012), Darbari, Asthana, Ahmed and Ahuja (2011), Darbari, Singh, Asthana, Prakash and Kendra, D. (2010), ontology generation method is best method for finding the solution for complex problem. It defines logical set of steps that reduces it into less complex problem.

Figure 5-2 is graphical plot of breaking down ontology development into two steps: The first step deals with initial domain modeling while second step deals with initial ontological development. Using WBS in Darbari et al. (2010), Kant, Singh, Darbari, Yagyasen and Shukla (2014), Darbari, Srivastava and Medhavi (2009) we are able to identify Ontological specification, state information and expert reasoning. The new information is represented in the form of Question, Answer and Transition.

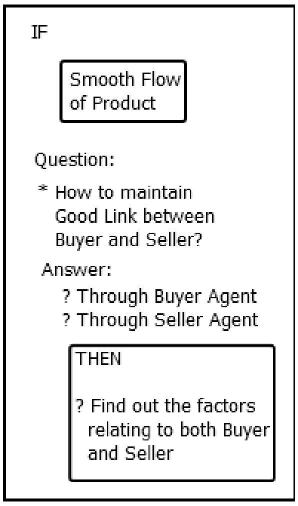

Figure 5-7: Algorithm for Task Flow

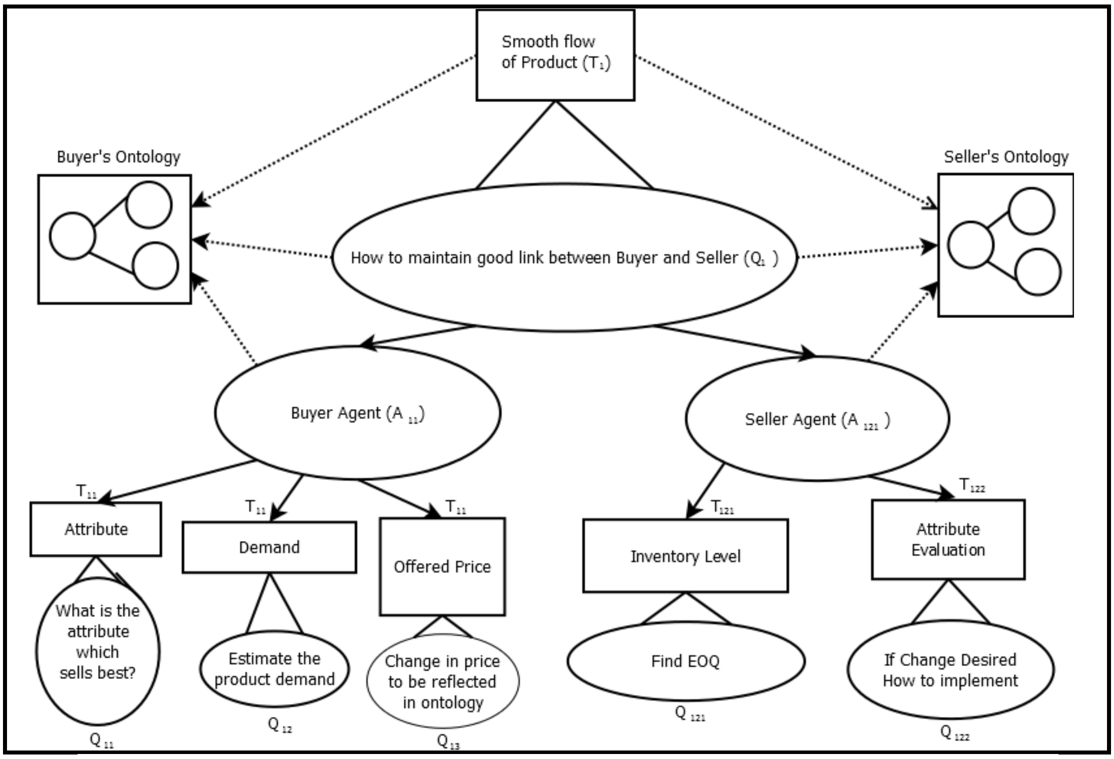

Figure 5-8: Buyer Seller Task Reduction Ontology

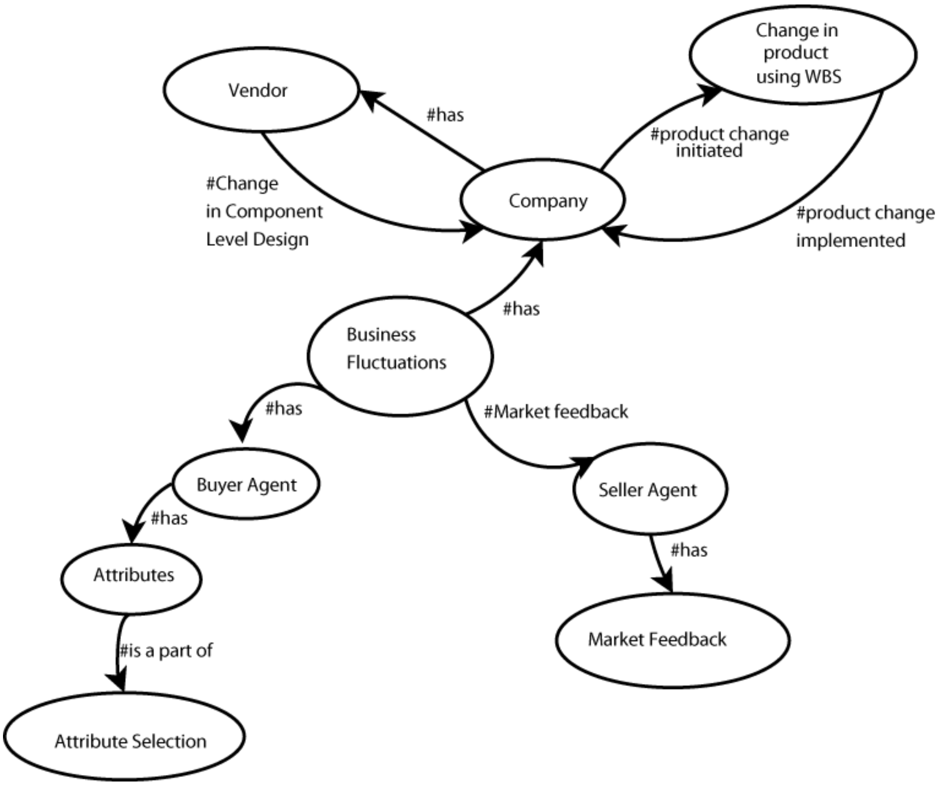

Figure 5-9: RDF for Business Dynamics

To represent the dynamic business environment using semantic web we convert them into Activity Theory Notation and generate it with Resource Description Framework of Economic Activity. The mapping of Activity Theory and Semantic Web for Dynamic Business Environment can be shown as in Table 5-1

Table 5-1: MAPPING OF AT, DBE AND SEMANTIC WEB

SIMULATION MODEL VARIABLE	DYNAMIC BUSINESS ENVIRONMENT	WEB SEMANTICS
Activity	Change in demand	Adjust in Taxonomy
Object	Dropping of Product	Drop in schema tree
Subject	Product being Manufactured	Generation of schema
Outcome	Change in overall cost of production	Customization in Web Semantic
Objective	To cut down cost	Triggering Changes
Tool	Effective techniques like JIT, SCM	Protégé
Community	Organisation / Market	Organisation

"Morolo" which manufactures mobile phones of various segments and sells them through their portal as well as direct selling through retailers. Due to sudden change in income (because of pay commission) there is sudden rise in market demand, price needs to be increased by the expansion of production shown by the graph as: (Figure 5-10 & Figure 5-11)

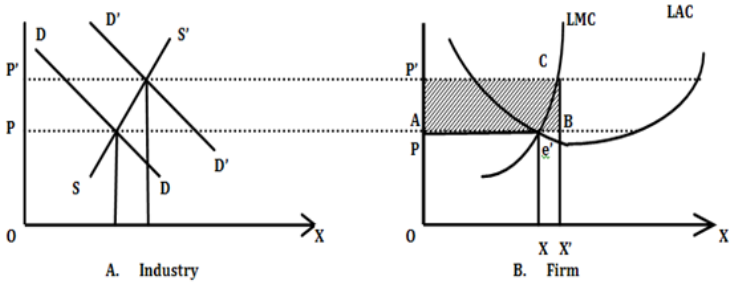

Figure 5-10: Impact of Demand Curve shift and its relation with LMC and LAC

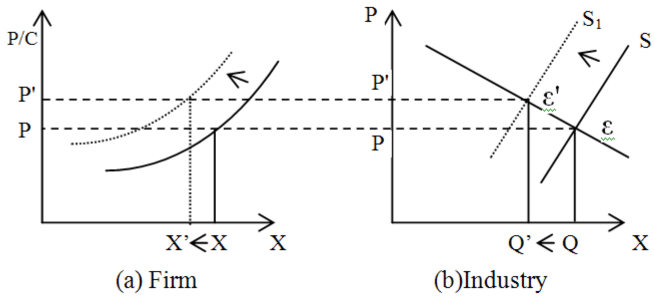

Figure 5-11: An increase in variable cost due to entry of new entrants

Due to excess profit many new competing firms enters into the mobile market. The influx thus causes the cost to come down and industries may start adding new models of mobile from the profits earned. This effect forces the firm to bring about functional changes; which include adding, removing and replacing functionality. To add further, addition of new entrants in the market increases the variable costs because of the competition by entry of new entrant, labour and raw material prices also increased finally causing shift in Average Variable Cost (AVC) and the quality supplied by the firm also changes from X to X' shown in Figure 5-11.

Under the condition, new market equilibrium the number of firms will be same but the quantity and the above situation brings about changes at the functional level: For example, mobile company may need to outsource some type of assembly to lower its cost of production.

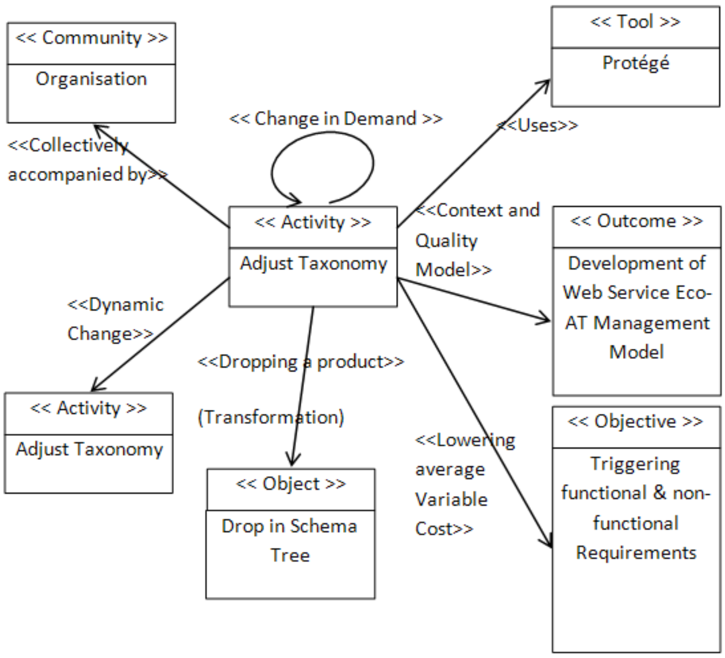

Figure 5-12: AT – Business – Semantic Web Framework

In order to do it has to make necessary changes in the RDF (Resource Description Framework). The pre stage of RDF development requires developing a relationship model using Activity Theory, economics and Web Semantics.

Figure 5-12 shows a common framework showing a link between Activity Theory notations being combined with Dynamic Organisational changes and Semantic Terminology. The framework provides a guideline in developing the Resource Description Framework (RDF), the close linkage with Activity Theory terms provides Human factors in developing Functional changes.

5.3.2 Functional Changes

For the first step the intended service should provide the specified operations. An example under this scenario includes manufacturing some part of the product from vendor, which needs to be linked with online delivery checking.

Secondly, adding a service like finding a "Point of Sale" in particular area by giving the zip code.

5.3.3 Non Functional Changes

Next step is to identify the non-functional changes like context and performance changes as stated by Liu, X., Akram, S., & Bouguettaya, A. (2011) in "Change Management for Semantic Web Services". A customer purchases a mobile in India and moves to Italy then he wants the same support by the company in the Italy. While developing the Change Dynamics Taxonomy website owner should have an option that by just giving the UID of Mobile phone Italy people should the necessary service to it. Similarly for the online payment of various spares which the user from countries may be expecting. In view of "Morolo", we would develop an RDF covering every aspect of organizational dynamics.

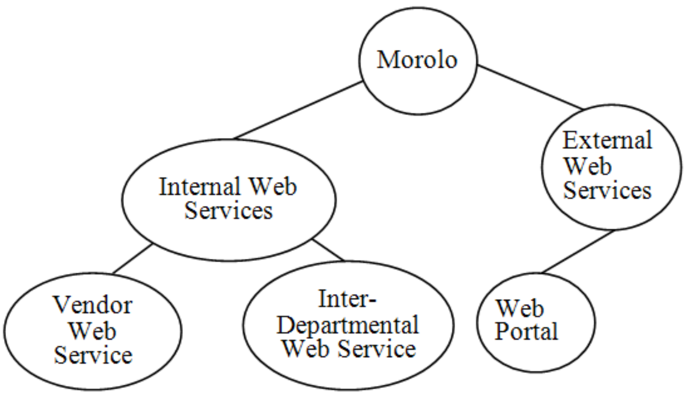

Figure 5-13: An Ontology Tree

The above (Figure 5-13) Ontology Tree gives the basic Ontology framework based on node structure we use the text editor to develop an RDF (Figure 5-14) where the collaboration with web services is realized using Activity Theory Notation.

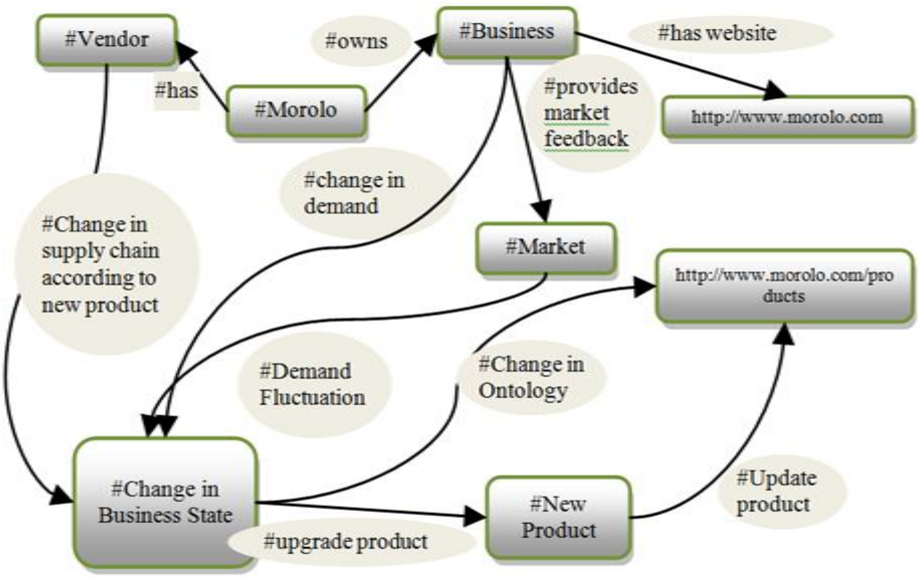

Figure 5-14: Basic RDF graph of "Morolo"

5.4 SUPPLY CHAIN DYNAMICS AND WORK BREAKDOWN STRATEGY (WBS) INTO SEMANTIC WEB

A real intelligent agent is proving to be very useful in the real world. It supports the features of building an expert system, which is helping in problem solving complex problems. This paper provides ontological Agent collaborations of Internet Marketing; it supports expert knowledge based features in finding the right product from the large repository of product database. This can only be possible if the product is readily available with the supplier. SCM (Supply Chain Management) provides holistic approach acting as a bridge between major business functions and business process within and across companies. Information system provides an effective backbone in managing the supply chain using softwares like ERP and Decision Support System.

Rao and Goldsby (2009) review the growing literature examining SCRM and to develop a typology of risks in the supply chain. The above process can be converted into necessary RDF (Figure 5-9) showing a combination of Business fluctuations in Prasad et al. (2013) in the form of inward demand flow and external product development. Any change can be formed by the help change in RDF modeling, which can be shown by the Algorithm I as:

Algorithm I: Change in RDF modeling

```
1:    Search(A, OI,T) {A is a service node answer, OI is a set of Ontology of
      required operation or data, T is another set of transition}

2:    if (OI, T = = A) return OI; {match of the requisite Ontology is found}

3:    if (OI, T <> ∅) {Current node does not cover the required set}

4:    N=Search for other sub node = Children_database(r); for all n∈N
      OI=(AT); {The search will be performed on the external Ontobase}

      If (N<>NULL) return OI initial;

      Return NULL;
```

The above algorithm Yagyasen and Darbari (2015) starts with generation of search for the appropriate solution. The WBS strategy searches all the possible outcomes and integrates it with final result, generating the required ontology.

The RDF trees will be optimized using the concept Evolutionary Semantic Web Analysis. The algorithm we will apply to generate real time selection is NSGA-II in Deb et al. (2002). In order to solve for the selection criteria we model the RDF in an isomorphic tree representing RDF. The http://mobile.com URL and its relationship with suppliers as the lower part of isomorphic tree. (Figure 5-15)

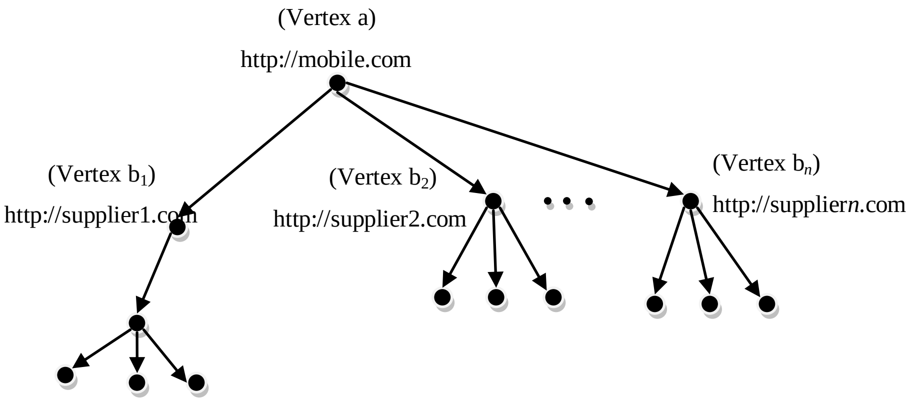

Figure 5-15: Isomorphic RDF Tree of http://mobile.com

Supplier_Selection_NSGA-II

(***N'*** members evolved ***g*** generation to solve for optimized supplier)

1. Initialize population ***P'*** *//based on optimistic supplier Location and Time*

2. Generate Random Population-Size ***N'*** from list of suppliers.

3. Evaluate Distance and Time of Delivery.

4. Assign Rank based on the best selected supplier.

5. Generate Child Population.

6. Binary Tournament Selection.

7. Recombine and Mutate.

8. for i=1 to ***g*** do

```
        for each Parent and Child in Population do
            Assign Rank (Level) based on Pareto.
            Generate sets of nondominated vectors.
            Assign the solutions based on crowding distance
                between suppliers. (Loop)
        end for
        Select points (elists) on the lower front (with lower rank) are
        outside crowding distance
        Create next generation
        Binary Tournament selection
        Recombination and Mutation
    end for
    end Algorithm Supplier_Selection_NSGA-II
```

The above algorithm finds the best supplier which is optimized for timely delivery.

5.5 CONCLUSION:

Our proposed technique readout a standard Owl XML file expressing semantic network ontology designed in Protégé, and transforms into an equivalent frame based knowledge source in domain specific language. A case on mobile manufacturing firm have been adopted in order to validate the proposed technique by verifying the number and names of the nodes/frames, the number and type of attributes and methods for each node/frame and the type of relationship between any of the two nodes/frames were identical in the input and the corresponding output problem of organisation using voluminous content of information which is to be shared in a proper format. It uses the concepts of economic principles and Activity Theory for conversion of information for organizational usage by using ”Psychological Aspect of Human” covered in Activity Theory. The use of Semantic Web helps in development of various web services.

CHAPTER 6

FORMAL VERIFICATION OF THE SEMANTIC FRAMEWORK USING PETRINETS

6.1 NEED FOR FORMALISATION

The rapid development in the software technology by using various modeling technique has made the software development very fast and easy. However, modeling of large-scale and complex software systems, is much more difficult and error prone. This is due to the fact that techniques and tools for assuring the connecters and reliability of software system lag far behind the increasing growth in size and complexity of software systems. The developed software should possess important properties of reliability: e.g. connecters, robustness, performance and security.

Studies have shown that most of the cost of software development and maintenance stems from the design defects. The Formal verification method developed by us helps to identify the defects at early stages, such as requirement and design phase of the Software life cycle.

For more than twenty-five years, some people have touted formal methods as the best means available for developing safe and reliable digital systems.

One well-known researcher expressed this attitude succinctly when he wrote about software engineering that: "It is clear to all the best minds in the field that a more mathematical approach is needed for software to progress”.

6.2 APPROACHES OF FORMAL VERIFICATION

In this chapter we will be focusing on developing a framework of business process model by using the concept of factor of production. We will be developing ontology based semantic annotation to achieve the semantic interoperability. The purpose of developing a semantic annotation is to build a knowledge base for automatic semantic-based discovery of services. In developing the semantics for an enterprise, we convert it into Activity Theory notation, which provides taxonomy for the above framework.

Generally, it consists of exploring all states and transitions in the model, by using smart and domain-specific abstraction techniques to consider whole group of states in a single operation and reduce computing time.

Most of the theory was developed in 60's and 70's but not fully exploited till 20-30 years. Figure 6-1 shows the gradual advancement in the field of formal verification.

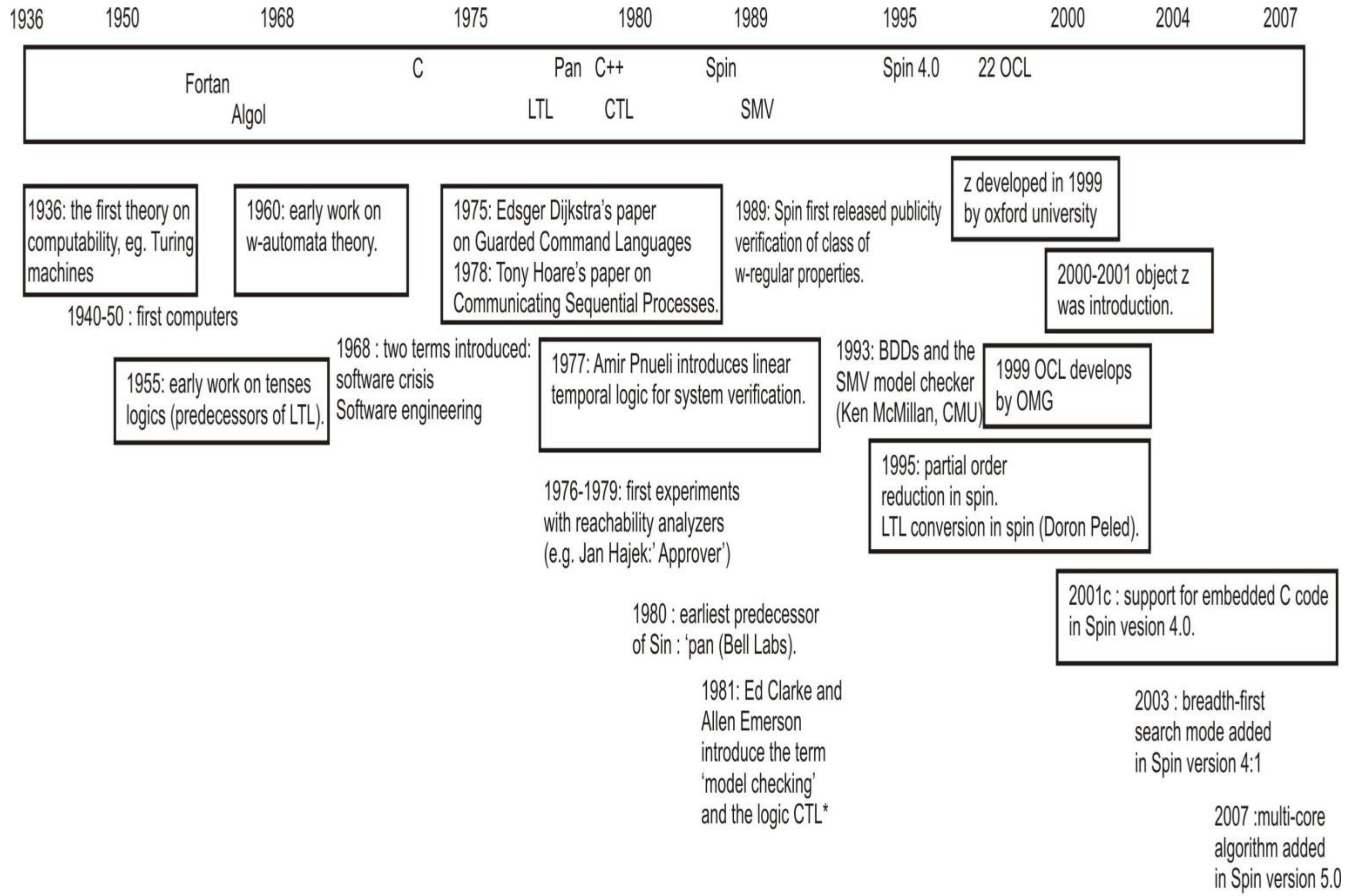

Figure 6-1: Era of Formal Verifications

6.3 VALIDATION OF THE MODEL

The verification of the above framework for dynamic business environment and ontology change we use Petrinets as a tool to model the framework and validate it.

6.3.1 Petrinets

Petrinet are a well founded process modeling technique. It was invented by Carl Adam Petri in the sixties since then petrinets have been used to model and analyze all kinds of processes. In nineties petrinet has been extended with color, time and hierarchy. Although there are workflow techniques available but Petrinet has certain edge over then like:

- Formal semantics
- Graphical Nature
- Expressiveness
- Vendor Independence.

6.3.2 Sequence Pattern of Petrinets

A sequence pattern contains two or more ordered activities that are performed sequentially, i.e. an activity starts after a previous activity has completed. This pattern is easily implemented by means of the basic Petri Net constructs: for each activity a transition is created and the transitions are connected with each other by means of arrows and places.

In order to verify our model we have used Petrinet Simulator 2.0. It consists of an area known as PetriNet Document. PetriNet Document is consisted of three views: PetriNet Editor view, Description view and Response view.For our simulation we have used Response view (Figure 6-4 & Figure 6-5).

Figure 6-4 represents simulation of ANDing approach where the similar activities are grouped together depending on the validity condition. Any change in market feedback will result in "Change in Component Level Design" which needs to be supported by "Ontology Change" as it has to be supported by change in vendors and other process systems. The Figure 6-4 can be further extended to Figure 6-5 where each and every process is discussed in detail, till a final outcome in the form of change in consumer liking is being designed.

6.3.3 Petrinet Calculus

We can represent the entire RDF graph using Petrinets as it will verify the basic workflow model. Petrinets (Murata, 1989) are used for modeling process. It was invented by Carl Aelam Petri in sixties. The main justification of using Petrinets is its strong foundation on formal semantics, its pictographical nature (Figure 6-2), expressiveness and strong foundation to formal verification of process. The classical Petrinet is a directed bipartite graph with two nodes types called places and transitions. The nodes connected via directed arcs. The simple definition of Petrinet is a triple(P, T, F).

P is finite set of places.

T is finite of Transitions ($P \cap T = \emptyset$)

$F \subseteq (P\ X\ T) \cup (T\ X\ P)$ is a set of Arc. (Flow relations)

A place $\boldsymbol{p}$ is called an *input place* of transition $\boldsymbol{t}$ iff there exists a directed arc from place $\boldsymbol{p}$ to $\boldsymbol{t}$.

A transition $\boldsymbol{t}$ is said to be *enabled* iff each input place $\boldsymbol{p}$ of $\boldsymbol{t}$ continues to have at least one token.

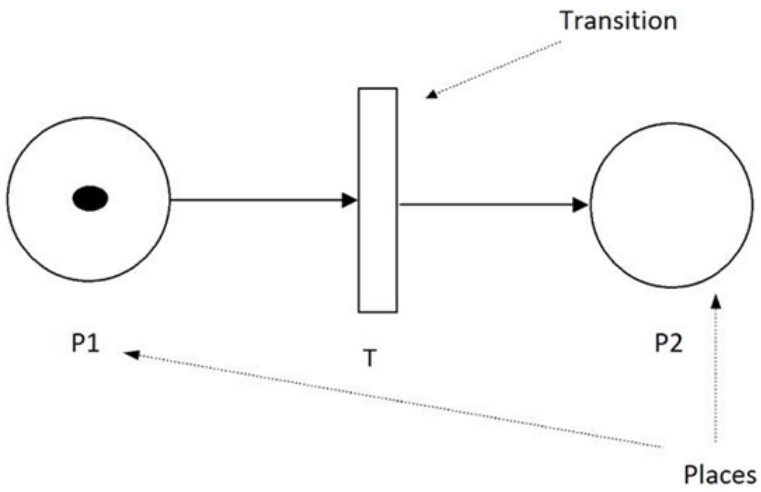

Figure 6-2: A simple Petrinet

If transition $\boldsymbol{t}$ fires, then $\boldsymbol{t}$ consume one token from each input place $\boldsymbol{p}$ of $\boldsymbol{t}$ and produces one token in each output place $\boldsymbol{p}$ of $\boldsymbol{t}$.

The initial marking is given as $\{1, 0, 0, 0\}$ corresponding to $\{t_1,\ t_2,\ t_3,\ t_4\}$. The above model can be analysed by using:

- Reachability graph
- Place and transitions graph
- Simulation

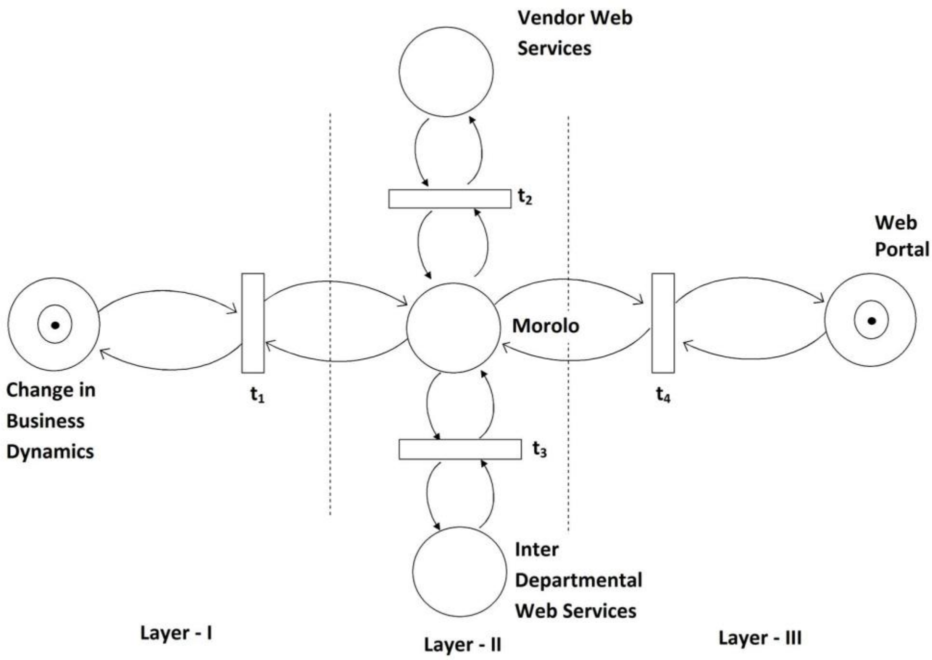

Figure 6-3: Layered structure of functional flow of information

The generalized layered structure of functional flow of information for Morolo is shown in Figure 6-3.

It provides concise and readable textual notations of the graphical model. The format represents the basic steps:

Step 1: $(cbd1 * \; cbd1@) \rightarrow [wpd1] \; \rightarrow (cbd2) \rightarrow [wpd2] \rightarrow \quad [* \, cbd1]$

Step 2: $(cbd1 * \; cbd1) \rightarrow [wpd1] \; \rightarrow (cbd2@) \rightarrow [wpd2] \rightarrow \quad [* \, cbd1]$

Step 3: $(cbd1 * \; cbd1) \rightarrow [wpd1] \; \rightarrow (cbd2) \rightarrow [wpd2] \rightarrow \quad [* \, cbd1@]$

where:

cbd = ***change in business dynamics.***

wpd = ***web portal dynamics***

These steps can be expanded for real time traffic mode. The process is represented in three layered swim lanes, it starts with "Change in Business Dynamics (cbd)" showing the change in Business which may happen for various reasons like change in product design, change in demand etc. In order to reflect this change we have linked it with "Morolo" which acts as a central agent and provides a dynamic control of the situation by changing the "vendor web services (vws)" and "interdepartmental web services (iws)". (Figure 6-3)

$$(cbd@ * RDU \{\leftarrow [wpd1 * wpd1] \leftarrow (cpd1 * cpd1) \leftarrow [t_1 * \ t_1]$$
$$\leftarrow (Morolo * Morolo)\}$$
$$\leftarrow \{[t_2 * \ t_2] \leftarrow (vws * vws@), \leftarrow [t_3 * \ t_3] \leftarrow (iws * iws@)\}$$
$$\leftarrow \{(Morolo * Morolo)[t_4 * \ t_4] \leftarrow (wpd2)\}\{* cbd\})$$

The above algebraic form represents the movement of token in Petrinets. The main purpose of representing it into algebraic form is its easy convertibility into PNML Notations, which can be converted into XML file. It is embedded with Eclipse to support the linking with HPSIM2.0. XML file can be directly converted into RDF notation.

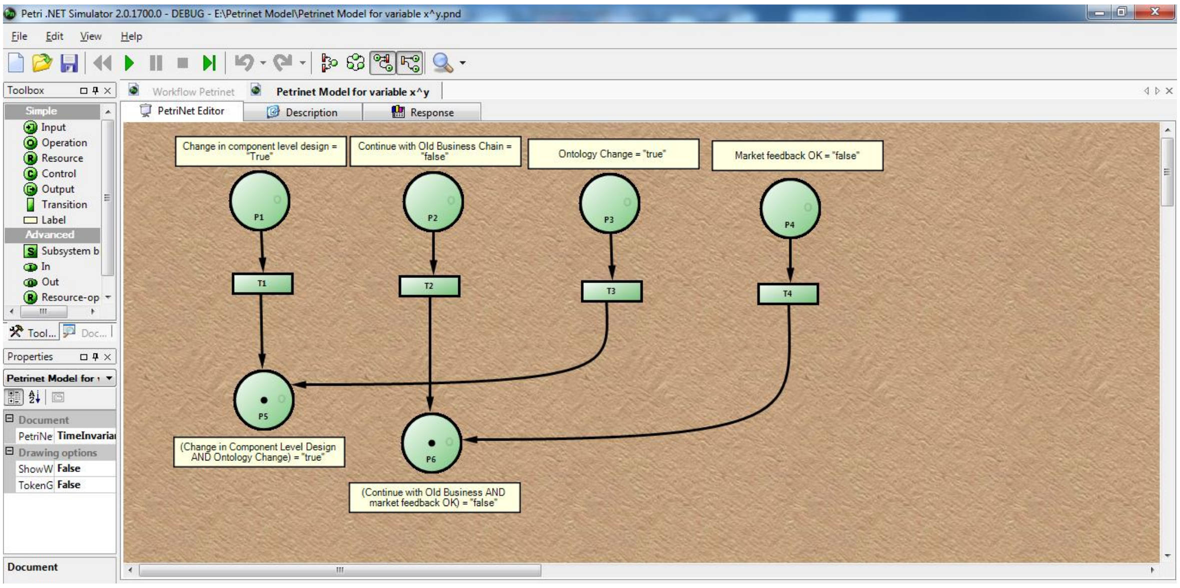

Figure 6-4: Petrinet X^ Y Interpretation

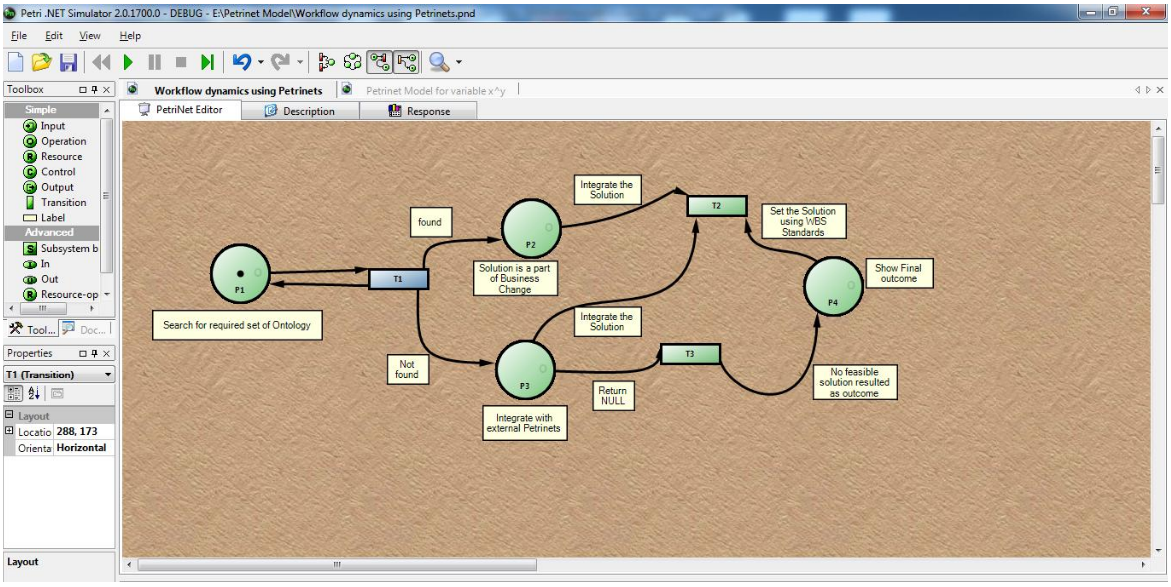

Figure 6-5: Change in Ontology in response to Business Dynamics

6.3.4 RESULTS

Using the concept of Petrinets we are able to formalize the situation. Future scope of our work will focus on extension of our work for automatic generation of business rules in terms of Ontology and Multi-Agent Language for a particular business domain.

Petri nets is a graphical language focusing on modeling of the workflow primitives. In contrast with many other process modeling techniques, Petrinet provides best source of modeling any system. The techniques vary from informal techniques to formal techniques such as process algebra's are event-based.

Petri-net theory can be used as an enhanced tool to model the complex workflow systems.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

The thesis bring to a conclusion by highlighting the novel idea of creating sustainable framework which using the concept of context of things and ubiquitous environment. Various entities are integrated using ontology, which provides a ubiquitous web service environment.

Semantic network is one of the knowledge representation technique used for communication between knowledge engineer and user. It lacks logical completeness and exactness. Due to the uncertainty of information about nodes and links in semantic networks problems arise in inferring and queering of knowledge. There is an object oriented layout of knowledge representation that can be modified with slot filling capability and procedural attachments. This thesis proposes a solution to extract knowledge from semantic networks ontologies in a semi-automatic way of knowledge transformation into an inferring suitable knowledge source.

Our proposed technique readout a standard Owl XML file expressing semantic network ontology designed in Protégé, and transforms into an equivalent frame based knowledge source in domain specific language. Various case examples like sugar industry and mobile manufacturing firm have been adopted in order to validate the proposed technique by verifying the number and names of the nodes/frames, the number and type of attributes and methods for each node/frame and the type of relationship between any of the two nodes/frames were identical in the input and the corresponding output problem of organisation using voluminous content of information which is to be shared in a proper format.

It uses the concepts of economic principles and Activity Theory for conversion of information for organisational usage by using ”Psychological Aspect of Human” covered in Activity Theory. The use of Semantic Web helps in development of various web services. The economic principle adopted in the model provides a dynamic change in web services which is due to change in production cost/dropping of a particular product.

The newly proposed system architecture, transformations function and system simulator design is the novel contributions of the research to perform the required transformation.

The system performance has been tested on nine case examples; three each from the small, medium and large classes. The results were verified by Petrinet-calculus, proving the validity of transformation and the objectives of the research.

There are few concerns and tasks that should be taken in further interpretation in future to carry enhancement in knowledge transformation process such as; to make sure the possibility to transform all the knowledge sources to their suitable and desired format. The re-usability and modification of the same system will be helpful as stepping stone for any kind of business logic.

The proposed system architecture can be revised after modifying its features. Upcoming advanced automation tools can be used to modify the functionality of the simulator. The system maintenance issues can be handled in a better way and the storage structure can be upgraded.

Multi Agent provides a strong base in the area of field of Software Engineering in order to deal with complexity of software systems. The architectural design of Multi Agent system is an attempt to formally verify the Real Time problem. Using the above approach we are able to model and implement all type of dynamic business situations including static ontology generation and dynamic ontology according to business rules. Using the concept of Petrinets we are able to formalize the situation.

In addition, this model can be extended by applying Visual Cognitive Language (VCL) linking it with Web-Eco-AT framework for changing Business Environment.

REFERENCES

1. Agarwal, V., Chafle, G., Dasgupta, K., Karnik, N., Kumar, A., Mittal, S., Srivastava, B., Synthy, A. (2005), System for end to end composition of web services. *Journal of WebSemantics*, 3(4), 311–339.

2. Aqeel-ur-Rehman, S. Z. (2011). ONTAgri: scalable service oriented agriculture ontology for precision farming. In 2011 *International Conference On Agricultural And Biosystems Engineering* vol 5 (pp. 1-2).

3. Ardagna, D., & Pernici, B. (2007). Adaptive service composition in flexible processes. *Software Engineering*, IEEE Transactions on, 33(6), 369-384.

4. Baghdadi, Y. (2005, August). A Web services-based business interactions manager to support electronic commerce applications. In Proceedings of the 7th international conference on *Electronic commerce* (pp. 435-445). ACM.

5. Batini, C., & Lenzerini, M. (1984). A methodology for data schema integration in the entity relationship model. *Software Engineering*, IEEE Transactions on, (6), 650-664.

6. Beck, H. (2008). Evolution of database designs for knowledge management in agriculture and natural resources. *Journal of Information Technology in Agriculture*, 3(1), 23.

7. Beck, H., Morgan, K., Jung, Y., Grunwald, S., Kwon, H. Y., & Wu, J. (2010). Ontology-based simulation in agricultural systems modeling. *Agricultural Systems*, 103(7), 463-477.

8. Benatallah, B., Medjahed, B., Bouguettaya, A., Elmagarmid, A. K., & Beard, J. (2000, September). Composing and Maintaining Web-based Virtual Enterprises. In *TES* (pp. 155-174).

9. Ben-Tal, A. (1980). Characterization of Pareto and lexicographic optimal solutions (pp. 1-11). *Springer Berlin Heidelberg*.

10. Berners-Lee, T. (1996). WWW: Past, present, and future. *Computer*, 29(10), 69-77.

11. Booth, D., Haas, H. , McCabe, F., Newcomer, E., Champion, M., Ferris, C., and Orchard, D. (2004) Web Services Architecture. Technical report, *W3C*, Retrieved from w.w3.org/TR/2004/NOTE-ws-arch-20040211/.

12. Bosworth, A., & McKusick, M. K. (2003). A Conversation with Adam Bosworth. http://www.acmqueue.org/.

13. Botts, M., Percivall, G., Reed, C., and Davidson, J., OGC sensor web enablement: Overview and high level architecture:version: 3.0. Tech. rep., Open Geospatial Consortium, OGC, December 28 2007. Available from: http://portal.opengeospatial.org/files/?artifact_id=25562.

14. Brambilla, M., Ceri, S., Comai, S., & Tziviskou, C. (2005, May). Exception handling in workflow-driven Web applications. In Proceedings of the 14th international conference on World Wide Web (pp. 170-179). ACM.

15. Breitman, K., Casanova, M. A., & Truszkowski, W. (2007). Semantic Web: Concepts, Technologies and Applications: Concepts, Technologies and Applications. Springer Science & Business Media.

16. Burstein, M., Bussler, C., Zaremba, M., Finin, T., Huhns, M. N., Paolucci, M., Sheth, A. P., and Williams, S. A semantic web services architecture. IEEE Internet Computing 9, 5 (2005), 72–81.

17. Burstein, M., Bussler, C., Zaremba, M., Finin, T., Huhns, M. N., Paolucci, M., Sheth, A. P., And Williams, S. A semantic web services architecture. IEEE Internet Computing 9, 5 (2005), 72–81.

18. Casati, F., & Shan, M. C. (2001). Dynamic and adaptive composition of e-services. Information systems, 26(3), 143-163.

19. Casati, F., Shan, E., Dayal, U., & Shan, M. C. (2003). Business-oriented management of Web services. Communications of the ACM, 46(10), 55-60.

20. CASTAN, S., Ferrara, A., & Montanelli, S. (2006). Dynamic knowledge discovery in open, distributed and multi-ontology systems: techniques and applications.

21. Chaffey, D. (2006) E-Business and E-Commerce Management, 3rd edn. Financial Times/Prentice Hall, Harlow.

22. Chan, T. M., Man, K. F., Kwong, S., & Tang, K. S. (2008). A jumping gene paradigm for evolutionary multiobjective optimization. *Evolutionary Computation*, IEEE Transactions on, 12(2), 143-159.

23. Cieniawski S E (1993) An Investigation of the Ability of Genetic Algorithms to Generate the Tradeoff Curve of a Multiobjective Groundwater Monitoring Problem Master's Thesis, University of Illinois at Urbana-Champaign.

24. Coello Coello, C. A. (2006). Evolutionary multi-objective optimization: a historical view of the field. *Computational Intelligence Magazine*, IEEE, 1(1), 28-36.

25. Coello, C. A. C. (2005). Recent trends in evolutionary multiobjective optimization. In *Evolutionary Multiobjective Optimization* (pp. 7-32). Springer London.

26. Coello, C. A. C. C., & Pulido, G. T. (2001, January). A micro-genetic algorithm for multiobjective optimization. In *Evolutionary Multi-Criterion Optimization* (pp. 126-140). Springer Berlin Heidelberg.

27. Coello, C. A. C. C., & Pulido, G. T. (2001, January). A micro-genetic algorithm for multiobjective optimization. In *Evolutionary Multi-Criterion Optimization* (pp. 126-140). Springer Berlin Heidelberg.

28. Coello, C. A. C., Van Veldhuizen, D. A., & Lamont, G. B. (2002). *Evolutionary algorithms for solving multi-objective problems* (Vol. 242). New York: Kluwer Academic.

29. Coello, C. C., Pulido, G. T., & Montes, E. M. (2005). Current and future research trends in evolutionary multiobjective optimization. In *Information Processing with Evolutionary Algorithms* (pp. 213-231). Springer London.

30. Cucchiella, F., & Gastaldi, M. (2006). Risk management in supply chain: a real option approach. *Journal of Manufacturing Technology Management*, 17(6), 700-720.

31. Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., & Weerawarana, S. (2002). Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI. IEEE *Internet Computing*, 6(2), 86-93.

32. Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., & Weerawarana, S. (2002). Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI. IEEE *Internet Computing*, 6(2), 86-93.

33. Cuske, C., Dickopp, T., & Seedorf, S. (2005). JOntoRisk: An ontology-based platform for knowledge-based simulation modeling in financial risk management. In *European Simulation and Modeling Conference*.

34. Dan Brickley and R.V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation, 2004. http://www.w3.org/TR/rdf-schema/.

35. Darbari, M. , Medhavi, S. , & Srivastava, A. K. (2008). Development of effective Urban Road Traffic Management using workflow techniques for upcoming metro cities like Lucknow (India). *International Journal of Hybrid Information Technology*, 1(3), 99-108.

36. Darbari, M., & Dhanda, N. (2010). Applying constraints in model driven knowledge representation framework. *International Journal of Hybrid Information Technology*, 3(3), 15-22.

37. Darbari, M., Asthana, R., Ahmed, H., & Ahuja, N. J. (2011). Enhancing the capability of N-dimension self-organizing petrinet using neuro-genetic approach. *International Journal of Computer Science Issues* (IJCSI), 8(3).

38. Darbari, M., Singh, V. K., Asthana, R., Prakash, S., & Kendra, D. (2010). N-Dimensional Self Organizing Petrinets for Urban Traffic Modeling. *IJCSI*.

39. Darbari, M., Srivastava, A. K. , Medhavi, S. (2009). Formal Verification of Urban Traffic System using the concept of Fuzzy Workflow System (FWS). *Journal of International Technology and Information Management*, 1(18), USA.

40. Dean, M., Schreiber, G., Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., ... & Stein, L. A. (2004). OWL web ontology language reference. *W3C* Recommendation February, 10.

41. Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation*, IEEE Transactions on, 6(2), 182-197.

42. Deborah L. McGuinness, Richard Fikes, James P. Rice, and Steve Wilder. The Chimaera Ontology Environment. In Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI2000), 2000.

43. Delin, K. A. The sensor web: Distributed sensing for collective action. Sensors Online (July 2006).

44. Delin, K., and Jackson, S. The sensor web: a new instrument concept. In Proc. SPIE Symposium on Integrated Optics, 20-26 Jan. 2001, San Jose, CA (2001), vol. 4284, pp. 1–9.

45. Deshpande, A., Ives, Z., & Raman, V. (2007). Adaptive query processing. *Foundations and Trends in Databases*, 1(1), 1-140.

46. Dhanda, N., Darbari, M., & Ahuja, N. J. (2012). Development of Multi Agent Activity Theory e-Learning (MATeL) Framework Focusing on Indian Scenario. *International Review on Computers & Software*, 7(4).

47. Ell, B., Vrandečić, D., & Simperl, E. (2011). Labels in the Web of Data. In *The Semantic Web–ISWC 2011* (pp. 162-176). Springer Berlin Heidelberg.

48. Ellis, C., & Keddara, K. (2000). ML-DEWS: Modeling language to support dynamic evolution within workflow systems. *Computer Supported Cooperative Work (CSCW)*, 9(3-4), 293-333.

49. Engeström, Y. (2007). Putting Vygotsky to work: The change laboratory as an application of double stimulation. The Cambridge companion to Vygotsky, 363-382.

50. Engeström, Y., & Ahonen, H. (2001). On the materiality of social capital: An activity-theoretical exploration. *Information systems and activity theory*, 2(55), 73-55.

51. Erickson, M., Mayer, A., & Horn, J. (2001, January). The niched pareto genetic algorithm 2 applied to the design of groundwater remediation systems. In *Evolutionary Multi-Criterion Optimization* (pp. 681-695). Springer Berlin Heidelberg.

52. Erickson, M., Mayer, A., & Horn, J. (2001, January). The niched pareto genetic algorithm 2 applied to the design of groundwater remediation systems. In *Evolutionary Multi-Criterion Optimization* (pp. 681-695). Springer Berlin Heidelberg.

53. Fensel, D. (2005). Spinning the semantic web: bringing the World Wide Web to its full potential. MIT Press.

54. FIPA. Ontology Service Specification, 2001. http://www.fipa.org/specs/fipa00086/index.html.

55. Fogel, L. J. (2009). Artificial Intelligence through simulated evolution, John Wiley, New York.

56. Fonseca, C. M., & Fleming, P. J. (1993, June). Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In *ICGA* (Vol. 93, pp. 416-423).

57. Fonseca, C. M., & Fleming, P. J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary computation*, 3(1), 1-16.

58. Fouad, A., Phalp, K., Kanyaru, J. M., & Jeary, S. (2011). Embedding requirements within model-driven architecture. *Software Quality Journal*, 19(2), 411-430.

59. Fourman M. P. (1985). Compaction of symbolic layout using genetic algorithm. Proc. 1st Int conf genetic algorithms. p. 141–53.

60. Frank Manola. Towards a web object model. http://www.objs.com/OSA/wom.htm.

61. Galanis, L., Wang, Y., Jeffery, S. R., & DeWitt, D. J. (2003, September). Locating data sources in large distributed systems. In Proceedings of the 29th international conference on Very large databases-Volume 29 (pp. 874-885). VLDB Endowment.

62. Geraci, A., Katki, F., McMonegal, L., Meyer, B., Lane, J., Wilson, P., ... & Springsteel, F. (1991). IEEE standard computer dictionary: Compilation of IEEE standard computer glossaries. IEEE Press.

63. Goh, M., Lim, J. Y., & Meng, F. (2007). A stochastic model for risk management in global supply chain networks. *European Journal of Operational Research*, 182(1), 164-173.

64. Golberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addion wesley, 1989.

65. Gottschalk, K., Graham, S., Kreger, H., & Snell, J. (2002). *Introduction To Web Services Architecture* IBM System journal.

66. He, J., Zhang, Y., Huang, G., & Cao, J. (2012). A smart Web service based on the context of things. *ACM Transactions on Internet Technology* (TOIT), 11(3), 13.

67. Hendler, J. Where are all the intelligent agents? IEEE Intelligent Systems 22, 3 (2007), 2–3.

68. Horn, J., Nafpliotis, N., & Goldberg, D. E. (1994, June). A niched Pareto genetic algorithm for multiobjective optimization. In *Evolutionary Computation*, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on (pp. 82-87).

69. Howse,J., Stapleton, G., Taylor, K., and Chapman,P.(2011). Visualizing Ontologies: A Case Study, *The SemanticWeb - ISWC 2011*, Lecture Notes in Computer Science Volume 7031, pp 257-272.

70. http://www.w3.org/standards/

71. Huhns, M. N. A research agenda for agent-based service-oriented architectures. *In Cooperative Intelligent Agents* (2006), pp. 8–22.

72. Huhns, M. N., Singh, M. P., Burstein, M., Decker, K., Durfee, E., Finin, T., Gasser, L., Goradia, H., Jennings, N., Lakkaraju, K., Nakashima, H., Parunak, H. V. D., Rosenschein, J. S., Ruvinsky, A., Sukthankar, G., Swarup, S., Sycara, K., Tambe, M., Wagner, T., and Zavala, L., *Research directions for service-oriented multiagent systems*. IEEE Internet Computing 9, 6 (November 2005), 65–70.

73. Hwang, S. Y., Lim, E. P., Lee, C. H., & Chen, C. H. (2008). Dynamic web service selection for reliable web service composition. *Services Computing*, IEEE Transactions on, 1(2), 104-116.

74. Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From SHIQ and RDF to OWL: the making of a Web Ontology Language. J. Web Sem., 1(1):7{26, 2003.

75. Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean. SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C Submission, 2004. http://www.w3.org/Submission/SWRL/.

76. International Electrotechnical Commission. IEC 61131-3 | Ed. 2.0 | English: Programmable controllers |- Part 3: Programming languages, 2003. http://webstore.iec.ch/webstore/webstore.nsf/artnum/029664.

77. INTEROP. Interoperability research for networked enterprises applications and software. http://interop-noe.org/.

78. Islam, A. S., & Piasecki, M. (2008). Ontology based web simulation system for hydrodynamic modeling. *Simulation Modelling Practice and Theory*, 16(7), 754-767.

79. Jakob, W., Gorges-Schleuter, M., & Blume, C. (1992, September). Application of Genetic Algorithms to Task Planning and Learning. In PPSN (pp. 293-302).

80. Jennings, Nicholas R., Katia Sycara, and Michael Wooldridge. "A roadmap of agent research and development." Autonomous agents and multi-agent systems 1.1 (1998): 7-38.

81. John F. Sowa. Knowledge Representation | Logical, Philosophical and Computational Foundations. Brooks/Cole, 2000.

82. Jones, D. R., Perttunen, C. D., & Stuckman, B. E. (1993). Lipschitzian optimization without the Lipschitz constant. *Journal of Optimization Theory and Applications*, 79(1), 157-181.

83. Jordan, D., Evdemon, J., Alves, A., Arkin, A., Askary, S., Barreto, C., ... & Yiu, A. (2007). Web services business process execution language version 2.0. OASIS standard, 11, 10.

84. Jüttner, U. (2005), "Supply chain risk management" *International Journal of Logistics Management*. Vol. 16, No.1, pp. 120-141.

85. Kant, G., Singh, V. K., & Darbari, M., Yagyasen, D. & Shukla, P. K. (2014). Legal semantic web- A Recommendation System. *International Journal of Applied Information Systems* 7(3). pp. 16-20.

86. Khan, O.and Burnes, B. (2007), " Risk and supply chain management: creating a research agenda" *International Journal of Logistics Management*, Vol. 18, No. 2, pp. 197 – 216.

87. Klyne, G., & Carroll, J. J. (2006). Resource description framework (RDF): Concepts and abstract syntax.

88. Knowles, J. D., & Corne, D. W. (2000). Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2), 149-172.

89. Konak, A., Coit, D. W., & Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering & System Safety, 91(9), 992-1007.

90. Koza, J. R. (1992). *Genetic Programming: On The Programming of Computers by Means of Natural Selection* (Vol. 1). MIT press.

91. Kursawe, F. (1991). A variant of evolution strategies for vector optimization. In Parallel Problem Solving from Nature (pp. 193-197). Springer Berlin Heidelberg.

92. Kuutti. K (1996), "Activity Theory as a Potential Framework for Human-Computer Interaction Research" Context and Consciousness: Activity Theory and Human-Computer Interaction. MIT Press, Cambridge.

93. Labrou, Y., & Finin, T. (1998). *Semantics for an agent communication language* (pp. 209-214). Springer Berlin Heidelberg.

94. Laclavik, Michal, et al. "Agentowl: Semantic knowledge model and agent architecture." Computing and Informatics 25.5 (2012): 421-439.

95. Laliwala, Z., Desai, A., Chaudhary, S., & Allam, A. (2008, July). Why context, content and contract are key for dynamic service selection. In Services-Part I, 2008. IEEE Congress on (pp. 281-288). IEEE.

96. Langdon, C. S. (2003). The state of Web services. Computer, 36(7), 93-94.

97. Laumanns, M. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm.

98. Lee, T. B. (1999). Weaving the Web. São Francisco: Harper.

99. Lenzerini, M. (2002, June). Data integration: A theoretical perspective. In Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (pp. 233-246). ACM.

100. Leontiev. A. N. (1981, Russian original published 1947),"*Problems of the Development of Mind*". English translation, Moscow, Progress Press.

101. Levy, A. Y. (1999). Combining artificial intelligence and databases for data integration. In *Artificial Intelligence Today* (pp. 249-268). Springer Berlin Heidelberg.

102. Levy, A., Rajaraman, A., & Ordille, J. (1996). *Querying heterogeneous information sources using source descriptions*.

103. Li, X., and Chandra, C. (2007), "A knowledge integration framework for complex network management", *Industrial Management & Data Systems*, Vol. 107, No. 8, pp. 1089-1109.

104. Liang, S. H. L., Croitoru, A., and Tao, C. V. A distributed geospatial infrastructure for sensor web. Computers & Geosciences 31, 2 (2005), 221–231.

105. Liu, X., Akram, S., & Bouguettaya, A. (2011). *Change Management for Semantic Web Services*. Springer Science & Business Media.

106. Liu, Y., & Guihua, N. (2007, June). Implementation of supply chains coordination using semantic web service composition. In Service-Oriented Computing and Applications, 2007. SOCA'07. IEEE International Conference on (pp. 249-254). IEEE.

107. Louis, S. J., & Rawlins, G. J. (1993, June). Pareto Optimality GA-Easiness and Deception. In *ICGA* (pp. 118-123).

108. Luck, M., Mcburney, P., Shehory, O., and Willmott, S. Agent technology roadmap: A roadmap for agent based computing, 2005. Available from: http://www.agentlink.org/roadmap/index.html [cited 28 May 2006].

109. Maekawa, T., Yanagisawa, Y., Sakurai, Y., Kishino, Y., Kamei, K., & Okadome, T. (2012). Context-aware web search in ubiquitous sensor environments. *ACM Transactions on Internet Technology* (TOIT), 11(3), 12.

110. Manolescu, I., Florescu, D., & Kossmann, D. (2001, September). Answering XML Queries on Heterogeneous Data Sources. In *VLDB* (Vol. 1, pp. 241-250).

111. Marek Obitko. Ontologies | Description and Applications. Technical Report 126, Gerstner Laboratory for Intelligent Decision Making and Control, Czech Technical University in Prague, 2001. http://cyber.felk.cvut.cz/gerstner/reports/GL126.pdf.

112. McBride, B. (2004). The resource description framework (RDF) and its vocabulary description language RDFS. In Handbook on ontologies (pp. 51-65). Springer Berlin Heidelberg.

113. McGuinness, D. L., & Van Harmelen, F. (2004). OWL web ontology language overview. W3C recommendation, 10(10), 2004.

114. Medjahed, B. (2004). Semantic web enabled composition of web services (Doctoral dissertation, Virginia Polytechnic Institute and State University).

115. Michalewicz, Z., & Janikow, C. Z. (1991, June). Handling Constraints in Genetic Algorithms. In ICGA (pp. 151-157).

116. Michel Klein. Combining and relating ontologies: and analysis of problem and solutions. In Workshop on Ontologies and Information Sharing, IJCAI'01, 2001.

117. Milanovic, M., Gasevic, D., & Wagner, G. (2008, September). Combining rules and activities for modeling service-based business processes. In Enterprise Distributed Object Computing Conference Workshops, 2008 12th (pp. 11-22). IEEE.

118. Murata, T. (1989). Petri nets: Properties, analysis and applications. Proceedings of the IEEE, 77(4), 541-580.

119. Nardi. B.A, Engestrom (1996), "*Activity Theory and Human-Computer Interaction*", Context and Consciousness: Activity Theory and Human-Computer Interaction. MIT Press, Cambridge (Mass.) pp 7-16 .

120. Natalya F. Noy and Mark A. Musen. SMART: Automated Support for Ontology Merging and Alignment. In Twelfth Ban® Workshop on Knowledge Acquisition, Modeling, and Management, 1999.

121. Navathe, S. B., & Gadgil, S. G. (1982, September). A Methodology for View Inegration in Logical Database Design. In Proceedings of the 8th International Conference on Very Large Data Bases (pp. 142-164). Morgan Kaufmann Publishers Inc.

122. Nicola Guarino and Pierdaniele Giaretta. Ontologies and Knowledge Bases - Towards a Terminological Clarification. In N.J.I. Mars, editor, Towards Very Large Knowledge Bases. IOS Press, Amsterdam, 1995.

123. Nicola Guarino. Formal Ontology and Information Systems. In Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, 1998.

124. Nishat Faisal, N M., Banwet, D. K., & Shankar, R. (2006). Supply chain risk mitigation: modeling the enablers. Business Process Management Journal, 12(4), 535-552.

125. Obitko M. and Marik, V., OWL Ontology Agent based on FIPA proposal. In Znalosti 2004, pages 226{237. Ostrava: VSB-TUO, 2004. ISBN 80-248-0456-5.

126. Obitko M. and Marik, V.,. Ontology Service Agent with Web Ontology Language. In EUMAS 2004 - Proceedings, pages 477-488, 2004.

127. Obitko, M. and Snasel, V.,. Ontology Repository in Multi-Agent System. In Artificial Intelligence and Applications - Volume I & II, pages 853-858. Calgary: Acta Press, 2004. ISBN 0-88986-404-7.

128. Ogden, C. K., Richards, I. A., Malinowski, B., & Crookshank, F. G. (1946). The meaning of meaning (pp. 109-38). New York: Harcourt, Brace & World.

129. OWL Web Ontology Language Overview, W3C Recommendation, 2004. http://www.w3.org/TR/owl-features/.

130. Patil, S., & Newcomer, E. (2003). ebXML and Web services. *Internet Computing*, IEEE, 7(3), 74-82.

131. Patra, P. (2010). Remote sensing and Geographical Information System (GIS). The Association for Geographical studies.

132. Patrick Hayes. RDF Semantics, W3C Recommendation, 2 2004. http://www.w3.org/TR/rdf-mt/.

133. Pechoucek, Michal, et al. "Agents in industry: the best from the AAMAS 2005 industry track." Intelligent Systems, IEEE 21.2 (2006): 86-95.

134. Peis, E., Herrera-Viedma, E., Hassan-Montero, Y., & Herrera, J. C. (2003). Ontologías, metadatos y agentes: recuperación" semántica" de la información.

135. Prasad, R., Darbari, M., & Yagyasen, D. (2012) Ontology-Based Knowledge Representation of Homeopathic Products. *International Journal of Scientific and Engineering Research* 5 (2), pp. 1276-1280.

136. Rangan, V. Kasturi, and Marie Bell. (Autumn 1998), "Case Study: Dell Online." *Journal of Interactive Marketing* 12(4).

137. Rao, S., and Goldsby, T. J. (2009), "Supply chain risks: a review and typology", The international *Journal of Physical Distribution & Logistics Management,* Vol. 20, No. 1, pp. 97-123.

138. Recker, J., Rosemann, M., Van Der Aalst, W., Jansen-Vullers, M., & Dreiling, A. (2006). Configurable reference modeling languages. Reference modeling for business systems analysis, 22-46.

139. Reijers, H. A., & Mendling, J. (2011). A study into the factors that influence the understandability of business process models. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 41(3), 449-462.

140. Resource Description Framework (RDF): Concepts and Abstract Syntax (2002). W3C Working Draft http://www.w3.org/TR/2002/WD-rdf-concepts-20021108/

141. Richardson, J. T., Palmer, M. R., Liepins, G. and Hilliard, M. (1989). Some guidelines for genetic algorithms with penalty functions. Proc. Third lnt. Con/. Genetic Algorithms, George Mason University, 4 June, pp. 191-197. Morgan Kaufmann, Fairfax.

142. Riichiro Mizoguchi and Mitsuru Ikeda. Towards Ontology Engineering. Technical Report AI-TR-96-1, The Institute of Scientific and Industrial Research, Osaka University, 1996.

143. Ritchie, B., & Brindley, C. (2007). Supply chain risk management and performance: A guiding framework for future development. *International Journal of Operations & Production Management*, 27(3), 303-322.

144. Ritzel B J, Eheart W and Ranjithan S 1994 Using genetic algorithms to solve a multiple objective groundwater pollution containment problem Water Resources Res. 30 1589–603

145. S. Maalal, M. Addou, "A practical application of a method of designing multi-agent systems based on the AUML language and the MDA approach", Proceedings of the Fourth Workshop on Information Technologies and Communication WOTIC"11, Casablanca, Morocco, p.104, 2011.

146. Saeedi, K., Zhao, L., & Sampaio, P. R. F. (2010, July). Extending BPMN for supporting customer-facing service quality requirements. In Web Services (ICWS), 2010 IEEE International Conference on (pp. 616-623). IEEE.

147. Schaffer, J. D. (1985, January). Multiple objective optimization with vector evaluated genetic algorithms. In Proceedings of the 1st International Conference on Genetic Algorithms, Pittsburgh, PA, USA, July 1985 (pp. 93-100).

148. Schwefel, H. P. P. (1993). Evolution and optimum seeking: the sixth generation. John Wiley & Sons, Inc.

149. Sharahchandra M. Lele, (1991). Sustainable development of natural resources. World development 19(6). Pagamon Press.

150. Sheth, A., & Miller, J. A. (2003). Web services: technical evolution yet practical revolution. IEEE Intell Sys, 18(1), 78-80.

151. Sheu, P.; Yu, H., Ramamoorthy, C. V., Joshi, A. K.and Zadeh,L. A. (2010). SEMANTIC COMPUTING, NJ: Wiley-IEEE Press.

152. Siddiqui, I. A., Darbari, M., & Bansal, S. (2012). Application of Activity Theory and Particle Swarm Optimization Technique in Cooperative Software Development. International Review on Computers & Software, 7(5).

153. SINGH, M. P., AND HUHNS, M. N. Service-oriented computing: Semantics, Processes, Agents. John Wiley & Sons Ltd, Chichester, England, 2005.

154. Singh, V. K., Darbari, M., Shankhdhar, G. K. (2014). Legal Semantic Web- A Recommendation System, *International Journal of Applied Information Systems* 7 (3), pp. 21-27.

155. Sølvberg, A., Hakkarainen, S., Brasethvik, T., Su, X., Matskin, M., & Strasunskas, D. (2002). Concepts of Enriching, Understanding and Retrieving the Semantics on the Web. ERCIM News, 51, 41-42.

156. Spaccapietra, S., Parent, C., & Dupont, Y. (1992). Model independent assertions for integration of heterogeneous schemas. The *VLDB Journal*, 1(1), 81-126.

157. Srinivas, N., & Deb, K. (1994). Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary computation, 2(3), 221-248.

158. Stuart E. Madnick. From VLDB to VMLDB (Very MANY Large Data Bases): Dealing with Large-Scale Semantic Heterogeneity. In Proceedings of the 21st VLDB Conference, 1995.

159. Sugarcane Information System (A User Guide), Department of Sugar Industries and Sugarcane Development, Uttar Pradesh. website: http://www.upcane.org/sis/

160. Sujanani, Anjalee, Pradeep Ray, N. Paramesh, and Ramprasad Bhar. "The development of ontology driven multi-agent systems: a case study in the financial services domain." In Proceedings of the IEEE EEE05 international workshop on Business services networks, pp. 1-1. IEEE Press, 2005.

161. Syswerda, G., & Palmucci, J. (1991, July). The Application of Genetic Algorithms to Resource Scheduling. In ICGA (pp. 502-508).

162. Tang, C. S. (2006). Perspectives in supply chain risk management. *International Journal of Production Economics*, 103(2), 451-488.

163. Thomas R. Gruber. A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, (2), 1993.

164. Thomas R. Gruber. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. In Nicola Guarino and Roberto Poli, editors, Formal Ontology in Conceptual Analysis and Knowledge Representation.Kluwer Academic Publishers, 1993.

165. Thomas R. Gruber. What is an Ontology? http://www-ksl.stanford.edu/kst/what-is-an-ontology.html.

166. Uma, G., Prasad, B. E., & Kumari, O. N. (1993). Distributed intelligent systems: issues, perspectives and approaches. Knowledge-Based Systems, 6(2), 77-86.

167. van der Aalst, W. M. (1998). The application of Petri nets to workflow management. *Journal of circuits, systems, and computers*, 8(01), 21-66.

168. van Ittersum, M. K., Ewert, F., Heckelei, T., Wery, J., Olsson, J. A., Andersen, E., ... & Wolf, J. (2008). Integrated assessment of agricultural systems–A component-based framework for the European Union (SEAMLESS). Agricultural systems, 96(1), 150-165.

169. Vaughan-Nichols, S. J. (2002). Web services: Beyond the hype. Computer, 35(2), 18-21.

170. Vinoski, S. (2002). Web services interaction models. Current practice. *Internet Computing*, IEEE, 6(3), 89-91.

171. Vogels, W. (2003). Web services are not distributed objects. *Internet Computing*, IEEE, 7(6), 59-66.

172. Vygotsky, L. S. (1997). The Collected Works of LS Vygotsky: Problems of the theory and history of psychology (Vol. 3). *Springer Science & Business Media*.

173. Vygotsky. L. (1978), Mind and Society, *Cambridge MA*, Harvard University Press.

174. W3C. Simple Object Access Protocol (SOAP). (2003). Retrieved from http://www.w3.org/TR/SOAP/.

175. W3C. Universal Description, Discovery, and Integration (UDDI). (2003). Retrieved from http://www.uddi.org, 2003.

176. W3C. Web Services Description Language (WSDL). (2003). Retrieved from http://www.w3.org/TR/wsdl, 2003.

177. Walton, C. Agency and the Semantic Web. Oxford University Press, Inc., New York, NY, USA, 2006.

178. Wei, D., Wang, T., Wang, J., & Bernstein, A. (2011). SAWSDL-iMatcher: A customizable and effective Semantic Web Service matchmaker. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(4), 402-417.

179. Weiss, G. (1999). Multiagent systems: a modern approach to distributed artificial intelligence. MIT press.

180. Wetzstein, B., Ma, Z., Filipowska, A., Kaczmarek, M., Bhiri, S., Losada, S., ... & Cicurel, L. (2007, June). Semantic Business Process Management: A Lifecycle Based Requirements Analysis. In *SBPM*.

181. Wilson, P. B., & Macleod, M. D. (1993, November). Low implementation cost IIR digital filter design using genetic algorithms. In *IEE/IEEE workshop on natural algorithms in signal processing* (Vol. 1, No. 4, pp. 1-4).

182. Wooldridge, Michael. An introduction to multiagent systems. John Wiley & Sons, 2009.

183. Workflow Management Coalition site http://www.wfmc.org

184. WSMO Working Group. Deliverable D2v1.2 Web Service Modeling Ontology WSMO, http://www.wsmo.org/TR/d2/v1.2/ (2005)

185. Wu, D., and Olson, D. (2008). Supply chain risk, simulation, and vendor selection. *International Journal of Production Economics*. 114 (2), 646–655.

186. Wu, T., Blackhurst, J., & Chidambaram, V. (2006). A model for inbound supply risk analysis. *Computers in industry*, 57(4), 350-365.

187. Yagyasen, D., & Darbari, M. (2014). Application of Semantic Web and Petri Calculus in Changing Business Scenario. In *Modern Trends and Techniques in Computer Science* (pp. 517-528). Springer International Publishing.

188. Yagyasen, D., & Darbari, M. (2015). Quantification of Dynamic Business Environment by Development of Ontology using Task Reduction. *International Journal of Scientific Engineering and Research*, 6(1), 283-288.

189. Yagyasen, D., Darbari, M., & Ahmed, H. (2013). Transforming non-living to living: a case on changing business environment. *IERI Procedia*, 5, 87-94.

190. Yagyasen, D., Darbari, M., Shukla, P. K., & Singh, V. K. (2013). Diversity and convergence issues in evolutionary multiobjective optimization: application to agriculture science. *IERI Procedia*, 5, 81-86.

191. Yang Li, Simon G. Thompson, Zhu Tan, Nick Giles, and Hamid Gharib. Beyond Ontology Construction; Ontology Services as Online Knowledge Sharing Communities. In International Semantic Web Conference, pages 469-483, 2003.

192. Yang, J. (2003). Web service componentization. *Communications of the ACM*, 46(10), 35-40.

193. Zhou, A., Qu, B. Y., Li, H., Zhao, S. Z., Suganthan, P. N., & Zhang, Q. (2011). Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation*, 1(1), 32-49.

194. Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *Evolutionary Computation*, IEEE transactions on, 3(4), 257-271.

195. Zou, Youyong, Tim Finin, Li Ding, Harry Chen, and Rong Pan. "Using Semantic web technology in Multi-Agent systems: a case study in the TAGA Trading agent environment." In Proceedings of the 5th international conference on Electronic commerce, pp. 95-101. ACM, 2003.