DESIGN AND DEVELOPMENT OF AN ONTOLOGY BASED MULTI-AGENT VIRTUAL ENTERPRISE SYSTEM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BAHRAM LOTFI SADIGH

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
MECHANICAL ENGINEERING

FEBRUARY 2015

Approval of the thesis:

DESIGN AND DEVELOPMENT OF AN ONTOLOGY BASED MULTI-AGENT VIRTUAL ENTERPRISE SYSTEM

submitted by **BAHRAM LOTFI SADIGH** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Mechanical Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Tuna Balkan
Head of Department, **Mechanical Engineering** ________________

Prof. Dr. Metin Akkök
Supervisor, **Mechanical Engineering Department, METU** ________________

Prof. Dr. S. Engin Kılıç
Co-supervisor, **Manufacturing Engineering Dept., Atılım Uni.** ________________

Examining Committee Members:

Prof. Dr. Mustafa İlhan Gökler
Mechanical Engineering Department, METU ________________

Prof. Dr. Metin Akkök
Mechanical Engineering Department, METU ________________

Assoc. Prof. Dr. Melik Dölen
Mechanical Engineering Department, METU ________________

Prof. Dr. Can Çoğun
Mechanical Engineering Department, Başkent Uni. ________________

Assist. Prof. Dr. A. Murat Özbayoğlu
Computer Engineering Department, TOBB ETU ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: BAHRAM LOTFI SADIGH

Signature :

ABSTRACT

DESIGN AND DEVELOPMENT OF AN ONTOLOGY BASED MULTI-AGENT VIRTUAL ENTERPRISE SYSTEM

Sadigh, Bahram Lotfi

Ph.D., Department of Mechanical Engineering

Supervisor : Prof. Dr. Metin Akkök

Co-Supervisor : Prof. Dr. S. Engin Kılıç

February 2015, 210 pages

Major developments in computers and information technologies, enable industrial and mechanical engineers to establish new net based, virtual collaboration platforms for enterprises. Benefiting from virtual enterprise platform enterprises will be able to combine their resources and capabilities on project based collaborations meanwhile protect their independent mainstream policies and secure their secret information. This concept is called virtual enterprise(VE). Virtual Enterprise (VE) is a collaboration model between multiple business partners in a value chain. The VE model is particularly feasible and appropriate for Small and Medium Enterprises (SME) and industry parks containing multiple SMEs that have different vertical competencies. One of the main targets of this research is to create an Ontology based Multi Agent Virtual Enterprise (OMAVE) System to prepare an appropriate platform for collaboration between technology start-ups in techno-parks and SMEs in Organized Industrial Zones in order to produce high value added high-tech products. OMAVE aims to help SMEs to shift from classic trend of manufacturing part pieces towards high-tech, innovative and research based products. In this way and to reach this goal a new semantic data infrastructure to enhance Re-Configurability and Flexibility of virtual enterprise systems has been developed.

In order to support flexibility in Virtual Enterprise business processes and enhance its integration to enterprises' available manufacturing systems (e.g. MRP) an ontology based domain model of VE system has been established. OWL DL semantic data structure of VE by defining concepts, axioms, rules and functions in VE system has been developed. TDB data store

to keep VE data and information in form of triples developed. SPARQL semantic RDF query language is used to handle and manipulate data on developed system data store. This architecture supports structure flexibility for developed VE infrastructure and improve reusability of data and knowledge in VE life cycle. To establish a multi agent based partner selection platform different agent types have been developed. These agents collaborate and compete with each other to select the most appropriate partner for the forthcoming VE project consortium. The agent based auctioning platform is coupled with a Fuzzy-AHP-TOPSIS multi criteria decision making algorithm to evaluate incoming bids from agents and rank proposals in each iteration. It is also important to notice that here, agents interaction's semantic is provided by an agent ontology. This agent ontology provides concepts, properties and all message formats for agents to settle a common language in interactions between agents. Implementing concurrent engineering, collaborative design and Product Life Cycle Management (PLM) concepts by integrating Dassault systems web based CATIA/ENOVIA V6 design and PLM tools to OMAVE system. To test and verify these achievements a case study to produce a test product by using developed OMAVE tools is established. This test product manufactured by contributions of SMEs from OSTIM organized Industrial Zone Aviation and Defense Cluster.

Keywords: Virtual-Enterprise, Multi Criteria Decision Support System, Multi-agent Systems, Ontology Based Model, semantic web framework

ÖZ

ONTOLOJİ TABANLI ÇOK-ETMENLİ SANAL FABRİKA SİSTEMİNİN TASARIMI VE GELİŞTİRİLMESİ

Sadigh, Bahram Lotfi
Doktora, Makina Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Metin Akkök
Ortak Tez Yöneticisi : Prof. Dr. S. Engin Kılıç

Şubat 2015 , 210 sayfa

Bilgisayar ve enformatik dünyasından hızla gelişen teknolojileri, sanayi ve makina mühendisliği alanında ortak çalışma ve işbirliği için yeni bir çığır açmaktadırlar. Sanal dünyada veri aktarımı, işbirliği firmalar içinde çok faydalı ve güvenli bir ortamda hazırlamaktadır. Firmalar kendi genel üretim, finansal politikalarını ve hayati bilgilerini etkilemeyecek şekilde yeni proje bazlı ortaklıklara girip ve bundan faydalanabilirler. Böyle bir platformun oluşması halinde bu ortaklık türü ***'Sanal Fabrika'*** olarak adlanmaktadır. Sanal fabrika Küçük ve Orta Büyüklükte İşletmeler (KOBİ'ler) ve Organize Sanayi Bölgeler (OSB) ler için daha hayati önem taşımaktadır. Sanal fabrika aynı zamanda üretici firmalar ve araştırma ve geliştirme alanında faaliyet gösteren firmalar ve enstitüler arasında bir köprü rolu oynayarak yeni yükske teknoloji içeren yüksek katma değerli ürünlerin üretilmesine de vesile olabilmektedir. Böylece KOBİlerin ürün kalitesini ve ürün düzeyini de yükselmiş olacaktır. Veri ve sistem çerçevesi açısından esneklik kazandırılması için ontoloji tememlli bir alan modeli ve buna bağlı bir anlamsal veri modeli geliştirilmiştir. Geliştirilmiş olan bu model üzerinden üçlü veri tabanı olarak tabi edilen TDB veri deposu geliştirilmiştir. TDB veri deposundaki verileri yönetmek ve bu veriler üzerinde işlemlerin yapılması içinde SPARQL (anlamsal veriler üzerinde sorgulama yapabilmek için geliştirilen özel bir sorgulama dili) kullanılmıştır. Bu yapı daha önce yapılmış olan sanal fabrika yapıların aksine verilerin tekrar kullanabilirliğini arttırarak, aynı zamanda istenilen yapısal değişikliklere de imkana sağlamaktadır. Sanal fabrikanın esnek olamamsından dolayı kullanılamaz hale gelmesine böyle bir yapı bir çare olarak gözükmektedir.

Bu araştırmada sanal fabrika literatürüne yapılmış olan bir diğer katkı, sanal fabrikanın oluşum aşamasında üye seçimi için farklı bir çok etmenli üye seçimi yapısının sunulmasıdır. Bu yapıda farklı türde etmenler bir biri ile ortaklaşa çalışarak veya rekabet ederek yeni sanal fabrika projesi konsorsiyumu için en iyi üyelerin seçmesini hedeflemektedir. Bu yapıyla iç içe bir bulanık-AHP-TOPSİS çok kıstaslı karar destek sistemi de kurulmuştur. Bu yapı etmenlerin her tur tekliflerinin sunmasıyla çalışarak yeni bir sıralanmış frima listesini sunmaktadır. Bunu belirtmekte yarar vardır ki etmenler bu sistemde geliştirilmiş olan bir ayrı etmen ontolojisi ile çalışmaktadırlar. Bu ontoloji etmenler için anlamlı olan kavramlar, özellikler ve mesaj içeriliği şablonlaronı tanımlamak için kullanılmaktadır. Aynı zamanda etmenler veriler ve bilgi için sanal fabrika alan ontolojisini kullanmaktadırlar. Eşzamanlı mühendislik, ortak ağ tabanlı tasarım ve ürün yaşam döngü yönetimi sistemleri kavramı, Dassault Systems tarafından geliştirilmiş olan CATIA ve ENOVIA V6 sistemlerinin Sanal Fabrika sistemi üzerinde kurulması ile hayata geçirilmiştir. Araştırmada geliştirilmiş olan araçları ve sistemin çalışmasını değerlendirmek ve doğrulamak amacıyla örnek bir ürünün ONÇESAFAnın araçları ve yapısını kullanarak üretilmesi gerçekleşmiştir. Örnek ürün OSTİM havacılık ve savunma sanayi kümelenmesinde (OSSA)da yer alan firmaların katkıları ile üretilmiştir.

Anahtar Kelimeler: Sanal Fabrika, Karar Destek Sistemi, Çok-Etmenli Sistem, Ontoloji Temelli Model

To My Lovely Mother

ACKNOWLEDGMENTS

My deepest gratitude is to my advisor, Prof. Dr. S. Engin Kılıç. I have been amazingly fortunate to have an advisor who gave me the freedom to explore on my own, and at the same time the guidance to recover when my steps faltered. His patience and support helped me overcome many crisis situations and finish this dissertation.

I would also like to express my gratitude and appreciation to my supervisor Prof. Dr. S. Metin Akkök for his guidance, encouragement, and continual personal and academic support.

My co-advisor, Assist. Prof. Dr. Hakkı Özgür Ünver, has been always there to listen and give advice. I am deeply grateful to him for the long discussions that helped me sort out the technical details of my work.

My sincere appreciation is extended to Assoc. Prof. Dr. Erdoğan Doğdu and Assist. Prof. Dr. A. Murat Özbayoğlu for their invaluable supports and advises on the system modeling and decision making algorithms.

I would also like to thank my committee members, Prof. Dr. Mustafa İlhan Gökler, Prof. Dr. Can Çoğun, Assoc. Prof. Dr. Melik Dölen and Assist. Prof. Dr. A. Murat Özbayoğlu for serving as my committee members even at hardship. I also want to thank you for your brilliant comments and suggestions, thanks to you.

Many thanks to my dear friend Ural Uluer for his supports that helped me overcome setbacks and stay focused on my graduate study.

I would like to express my special thanks to my dear friends Shahrzad Nikghadam and Marieh Moranezhad for their continuous encouragement, help, patience and moral support.

I would also like to extend my thanks to Mr. Orhan Aydın from OSTIM Technology management and Prof. Dr. Sedat Çelikdoğan, Çiğdem Tambağ, Tuna Çağlar Gümüş and Mustafa Zeynel Dağlı from OSTIM Technology for their support and contributions in this project.

Special thanks to Mr. Mithat Ertuğ and Münir Ertuğ from ELEKTRO MEKANİK GEREÇLER ENDÜSTRİSİ (EMGE) for their technical support during this project.

I would also acknowledge Turkish Ministry of Science, Technology and Industrial Ministry for their technical and financial support for Virtual Enterprise project.

Finally, I would like to express my special gratitude to my family for their endless support, sacrifices, patience, and love throughout my life. I am proud to have such a supportive and lovely family.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBRV	Abbreviation
ACL	Agent Communication Language
AHP	Analytic Hierarchy Process
AMS	Agent Management System
ANN	Artificial Neural Network
ANP	Analytic Network Process
API	Application Programming Interface
ASELSAN	Askeri Elektronik Sanayi, Military Electronic Industries
BRIC	Block like Representation of Interactive Components
CIMOSA	Computer Integrated Manufacturing Open System Architecture
CNP	Contract Net Protocol
CORBA	Common Object Request Broker Architecture
CSV	Comma Separated Values
DBP	Distributed Business Processes
DEA	Data Envelopment Analysis
DF	Directory Facilitator
DSS	Decision Support System
EBNF	Extended Backus-Naur Form
ER	Entity Relationship
ERP	Enterprise Resource Planning
EU	European Union
FaCT	Fast Classification of Terminologies)
FFBD	Functional Flow Block Diagrams
FIPA	Foundation for Intelligent Physical Agents
FISEA	Framework for Inter Sensing Enterprise
GERAM	Generalised Enterprise Reference Architecture and Methodology
GIOP	General Inter-ORB Protocol
GLOBEMEN	Global Engineering and Manufacturing in Enterprise Networks

GRAI/GIM	Graphs with Results and Actions Inter-related Integrated Methodology
GRDDL	Gleaning Resource Descriptions from Dialects of Languages
GUI	Graphical User Interfaces
ICT	Information and Communication Technologies
IDEF	Integration Definition for Function Modeling
IIOP	Internet GIOP
IMTRG	Integrated Manufacturing Technologies Research Group
ISO	International Standard Organization
JADE	Java Agent Development Platform
JDK	Java Development Kit
JSON	JavaScript Object Notation
JVM	Java Virtual Machine
KQML	Knowledge Query Manipulation Language
LGPL	Lesser General Public License
MAS	Multi Agent Systems
MCDM	Multi Criteria Decision Making
MES	Manufacturing Execution System
METU	Middle East Technical University
MTS	Message Transport Service
OEM	Original Equipment Manufacturers
OKBC	Open Knowledge Base Connectivity
OMAVE	Ontology based Multi Agent Virtual Enterprise
OMG	Object Management Group
ORB	Object Request Broker
OSI	Open Systems Interconnection
OSMOS	Open Systems for inter-enterprise information Management in dynamic virtual Organizations
OSSA	OSTIM Savunma Sanayi Kumelenmesi, OSTIM Defense Industry Cluster
OWL DL	Web Ontology Language Description Logic
OWL	Web Ontology Language
PERA	Purdue Enterprise Reference Architecture
PERT	Program Evaluation Review Technique
PLM	Product Life cycle Management

PMS	Performance Measurement Systems
preDiS	Persistent Distributed Data Store
PRMA	Penalty and Rewards Management Agent
RDF/S	Resource Descriptive Language / Schema
RFLP	Requirement, Functional, Logical and Physical
SaaS	Software as a Service
SADT	Structured Analysis and Design Technique
SME	Small and Medium Size Enterprise
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
STEP	Standard for Exchange Product Model Data
SWRL	Semantic Web Rule Language
TAI	Turkish Aerospace Industry
TCI/IP	Transmission Control Protocol and the Internet Protocol
TDB	Triple Data Bases
TEI	Turkish Engine Industry
TOBB ETU	Tukriye Odalar ve Borsalar Birliği- Economy and Technology University
TOPSIS	Technique for Order of Preference by Similarity to Ideal Solution
TurkStat	Turkish Statistical Institute
UML	Unified Modeling Language
URI	Uniform Resource Identifier
USAF ICAM	US Air Forces Integrated Computer- Aided Manufacturing
VBE	Virtual Breeding Environment
VE	Virtual Enterprise
VERAM	Virtual Enterprise Reference Architecture and Methodology
VESSE	Virtual Enterprise System Search Engine
VITE	Vegetable Business Virtual Enterprise
W3C	World Wide Web Consortium
WAP	Wireless Access Point
WAP	Wireless Application Protocol
WFMA	Work Flow Manager Agent
XML/S	Extended Markup Language /Schema

CHAPTER 1

INTRODUCTION

1.1 Small and Medium Size Enterprises and Virtual Enterprise

Today's global markets are characterized by competition. In order to maximize both market share and profit, corporations strive to offer better products and/or services [87]. New advances in technology raise customer expectations; hence manufacturers are required to enhance product technology and applicability [101]. These highly dynamic market conditions put increasing pressure on suppliers and producers [20]. In order to remain competitive, manufacturers must maintain a wide range of choices and satisfy customer expectations as much as possible by developing new products, technologies or services with better quality at lower prices. This requires highly agile, flexible and reconfigurable enterprise infrastructures which are expensive, even for giant multi-national companies [17] [91].
Small and medium sized enterprises (SMEs) are one of the main economic pillars in developed and developing countries. SMEs account for about 67% of employment in the European Union's non-financial business economy; micro enterprises contribute about 34%, small enterprises about 20% and medium-sized enterprises about 13%. In addition to being a major source of employment in the European Union (EU), SMEs also have a positive effect on economic growth [63] [33]. According to Turkish Statistical Institute (TurkStat), 99.9% of total number of enterprises in Turkey are SMEs. SMEs contribution to Turkish economy is really considerable. As stated by TurkStat, 76% of employment and 63% of total revenue is provided by SMEs. SMEs proportion was 62.6% for exports in 2012 and 38.5% for imports in the same period [117].

Table 1.1 depicts SMEs and large enterprises contributions in employment in Europe. SMEs are playing a very important role from employment point of view [62]. In turbulent and unpredictable market conditions, it is very difficult for smaller enterprises to survive and maintain competitiveness alone. SMEs are typically highly specialized and lack complementary capabilities which preclude them from producing new high value added products or services. In order to survive in such a competitive environment, one of the solutions available to SMEs is collaboration [62]. Collaboration is a key enabler for multiple business partners (especially SMEs), empowering them to join their competencies in order to pursue common business targets and helping them manage turbulent business environments characterized by unpre-

Table 1.1: SMEs and Large Enterprises in Europe [73]

	Unit	SME	Large Enterprises	Total
Number of Enterprises	1000	20,415	40	20,455
Employment	1000	80,790	40,960	121,750
Persons employed per Enterprise		4	1,020	6
Turn over per Enterprise	Million €	0.6	255.0	1.1
Share of exports in turn over	%	13	21	17
Value added per person employed	€1000	65	115	80
Share of labor cost in value added	%	63	49	56

dictable market conditions and customer demands, shortened product life-cycles, and intense cost pressures [14, 19, 108].

A Virtual Enterprise (VE) is a temporary collaboration framework among multiple business partners in a value chain designed to reach business goals by sharing fundamental capabilities using information and communication technology (ICT) [87, 108]. The VE framework is particularly feasible and appropriate for SMEs located in industrial parks with other SMEs that have different vertical competencies. By cooperating within a VE framework, SMEs are able to combine their diverse competencies to develop new, higher quality products and reduce the effects of market turbulence [14, 87, 91].

However, enhancing product quality and creating innovative, technologically advanced, high value added products requires more than forming a collaboration network among multiple manufacturing SMEs [101]. Thus, the main target of this research is to examine how including high technology research and development (R&D) companies may increase the ability of a VE consortium to produce innovative, high value added, high technology products. By combining the production capability of manufacturers with the research capability of R&D companies in via a secure and trustworthy collaboration platform, it would be conceivable for SMEs to produce high value added products. At the same time, such a system would provide financial benefits for SMEs and increase their market competitiveness by shifting SMEs production capabilities from common manufacturing parts to high value added high technology products [107].

In 21st century, continued competitiveness by enterprises in the flat economic world depends on their ability to employ the principles of agility. Agile manufacturing is not flexible manufacturing, lean manufacturing or computer integrated manufacturing, rather it is a combination of such useful techniques, methods, and philosophies. An agile organization is one whose organizational structures and processes enable fast and fluid transitions of an initiative, to respond changes in customer enriching business activities. Agility is dynamic, context-specific; aggressively change embracing and growth oriented. Agility is about winning, about succeeding in emerging competitive arenas, and about winning profits, market share, and cus-

tomers in the very center of the competitive storms many companies are in [46].
Many scholars and authors cite Virtual Enterprises as a key enabler of Agility [46,52]. Among other enablers such as concurrent engineering, e-commerce, integrated product/ production information systems, VE is special interest because it places the greatest demands on a company to co-operate in achieving collaborative production. A VE is a temporary consortium formed by real autonomous companies on the basis of strong collaboration to respond temporary demands, which a single company with limited core competencies and production capacity, is unable to respond. VE may accomplish tasks which could not be done by consortium members by their alone. This is a team work and it will gain predefined targets if all the consortium members do their job perfectly. Beside collaboration and sharing competencies, there are other strategic reasons to use virtual organization model.

1 Reach critical mass and be a higher class competitor by sharing resources and facilities.

2 Share total cost and risk between VE stakeholders

3 Increase the chance to enter various industrial sectors

According to the customer order VE output volume could be arranged. VE may respond to one of a kind projects with high customization and also respond to batch size or mass productions with less customization requirements and high production volumes [41]. VE customization level, production volume is determined by type of incoming order and customer requirements [41].
VE must deal with instant customer orders quickly and exploit from arising opportunities. VEs also are formed up because, a single SME may not respond to an opportunity as fast as required by itself. In this situation a VE is formed up to make ordered product or provide requested service in the opened opportunity windows. Goldman suggests that taking the advantage of an opportunity in the first half of opportunity window is much profitable for companies than the second half of window. Therefore VEs must act agile to respond request for quotes from customers. This requires a highly flexible and agile structures for VEs. Many scholars and industry experts advocate Information and Communication Technologies (ICTs) and networked world via information highways, are the key enabler of reaching these objectives.

1.2 Literature Survey

Virtual Enterprise (VE) is a collaboration framework between multiple independent business partners to pursue a common business target by sharing their core competencies by means of Information and Communication Technologies (ICT). In order to develop a virtual enterprise platform a research was carried out by Integrated Manufacturing Technologies Research Group (IMTRG) group of the mechanical engineering department of Middle East Technical University (METU) on 2006 under the supervisory of Prof. Dr. S. Engin KILIC [106].

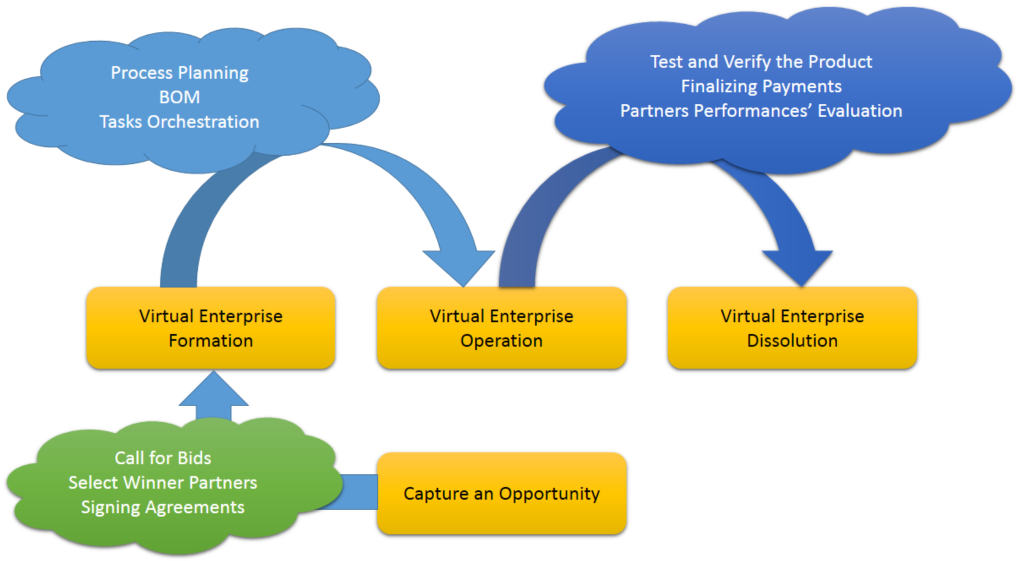


Figure 1.1: Virtual Enterprise Life Cycle in Literature [103]

Sari et al. proposed a new methodology for VE systems, and developed a new VE reference ICT system architecture. Developed VE system architecture ibased on ICT architecture of Microsoft's multi-layer client/server .NET technology Figure 1.1 [103].

In this research VE phases are separated into three main steps as follow;

1. Partner selection and VE formation phase
2. VE Operation phase
3. VE Dissolution phase

In order to form up a new VE project, very first step is to select most appropriate partners from members' pool which is called Virtual Breeding Environment (VBE). The partner selection process is carried out based on four main criteria; price, caution cost, completion probability and past performance. These four criteria are an Analytic Hierarchy Process (AHP) attributes. The overview of the partner selection process using AHP is given in Figure 1.2.

1.3 Literature Survey on VE Formation Phase

Task completion probability by the given project timetable and scheduling is calculated using Program Evaluation Review Technique (PERT) [49]. In order to evaluate partners' performances in the project multi-layer artificial neural network (ANN) was developed [51]. In order to fulfill VE requirements and increase flexibility and re-configurability in formation and operation phases of VE and working seamlessly, Lotfisadigh et al. proposed a three layered

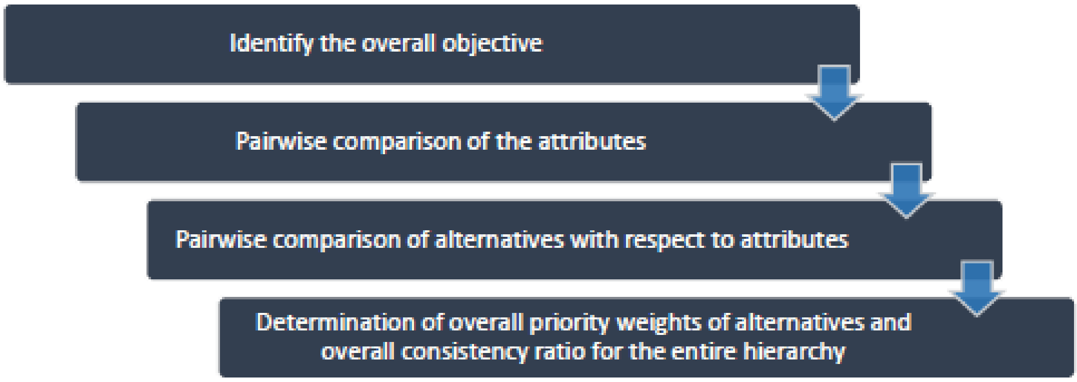


Figure 1.2: Overview of the selection process using AHP [106]

ICT architecture for conceptual operational virtual enterprise platform based on Multi-Agent Systems (MAS) architecture [103].

Most of the researches in VE field are concentrated on partner selection step of VE formation phase. In order to select the most efficient partners for forthcoming VE project various methods and theorems are applied. One of these methods which is highly recommended by researchers and is also effective in bidding procedures is agent based systems. Multi agent systems are widely used in virtual enterprise systems by different researchers and academia. Yang et al. also has proposed a multi-agent based partner selection platform in order to choose best possible members, decompose project tasks and distribute them effectively among partners in the most appropriate way for VE projects [130]. Based on this proposed model between agents bargaining during bidding procedure, and the best proposal wins the auction [130]. A three layered multi-agent based architecture model containing business processes properties, registration and management for dynamic virtual enterprise has been proposed by Feng et al. [35]. Another multi-agent based approach for virtual entrepreneurship modeling and business processes is proposed by Gou et al. In this study two main group of resources and action agents cooperate to form up VE consortium [48].

One of the prominent studies related to the formation phase of VE is PRODNET II project. In this project different tools and application to estimate resources, define enterprise profiles, configure VE structure, partner selection and evaluation, define management definitions and communication protocols and etc. were developed [65]. VIRTEC and ALFA COSME-VE are similar projects which were developed in Brazil by Bremer and in Mexico by Molina respectively [13, 80]. COWORK project developed by Alzaga also is in the same category [4]. In all these projects, in order to develop an enterprise pool for potential partners with similar capabilities and competitiveness special criteria and management system architectures are proposed. In the case of forming a new VE project, pool members search for best alliances in the same pool to accomplish their capabilities and enable them to capture the opportunity to take apart in the forthcoming project. These enterprises are rival cooperation but at the same time they try to eliminate their weaknesses by cooperating with other companies. VE members' pool provides a unique format and standard for saving enterprises data and information therefore it secures and facilitates partner selection processes. The unique forms keep different data and information regarding enterprises from their past performances to their financial performances and commitments. Consequently this system brings huge advantages in forming the next VE and especially in partner selection phase of VE formation [4,13,80,113].

Different Researchers attempt to develop fully automatic agent based platform for VE systems. Rocha, Daviddrajuh, Deng, Gou and others developed and defined broker and customer agents for different enterprises and designed a VE infrastructure. Unfortunately, due to lack of worldwide standards regarding agents messaging and collaboration, these systems encountered tough problems and they barely fulfilled their orders [30, 48, 92, 97]. In order to overcome to these problems, interests over hybrid systems increased. In hybrid systems program and human agents are collaborating to satisfy system requirements. One of these hybrid designed systems was introduced in MASSYVE project. MASSYVE project is benefiting from an agent-based approach for partner selection and generating intra and inter organizational scheduling. For all the enterprises in virtual breeding environment (VBE) a common agent framework and standard is defined. System broker agents evaluate the business opportunities and in order to reach an agreement and select the best enterprise according to the defined criteria for cooperation in VE, facilitator agents are defined. Facilitator agents are responsible for designing and planning of VE negotiation with consortium agents in VBE. Here agents could be both human and software [21, 30, 97]. As a pilot system this platform was successfully applied in a molding industry project in Brazil by Camarinha-Matos and Rabelo [42,97].

1.4 Literature Survey on VE Operation Phase

In order to manage and monitor business processes in virtual enterprise operation phase different concepts and approaches have been proposed by researchers [70, 92, 93]. Following a business process commencement in VE, it is divided to different sub processes. Different enterprises are responsible of these sub processes. This concept is called distributed business processes (DBP). The most challenging problem here is the coordination and monitoring of all these sub processes. VE coordinator is responsible for orchestrating the enterprises in charge of sub processes. Here the coordinator enterprise also could be responsible for one or more sub-processes at the same time. Similar approach is proposed in MASSYVE project and in this project Distributed Business Processes VE (DBP VE) concept also was introduced. Some sub processes are really time consuming and complicated and requires proficiencies beyond the capabilities of a single enterprise. In this case sub process coordinator (the enterprise in charge of sub process) could form up a temporary consortium called distributed business process virtual enterprise (DBP VE) [92].

To monitor and manage operation processes of partners a multi layered ICT architecture was proposed by Camarinha-Matos and Lima in PRODNET II project. PRODNET II project was a joint project between European and Brazilian enterprises to produce a bicycle from designing phase to final production [42]. In proposed architecture for operation phase of VE, all enterprises management and manufacturing tools (ex. ERP/PPC, PDM, CAD and etc.) communicate between each other in a layer responsible for communications among partners. The core concept of communication layer is based on distributed information and work flow management model [65] [55]. This system is supported by a service library. In this system CAD data communication is based on Standard for Exchange Product Model Data (STEP) model [42].

1.5 Literature Survey on VE Dissolution Phase

Virtual enterprise dissolution phase is the final step of VE. From partner selection to product delivery to the customer lots of information and data are produced. These knowledge and data must be kept and updated to be reused in the next projects by system management and Virtual Breeding Environment (VBE) members. Some of the researches relevant to VE have been concentrated on this phase of VE. Metes et al. developed product life-cycle management for VE, special tools to regulate and organize agreements between customers and partners and product guarantee and services [30]. Some special tools also were developed in PRODNET II project but still there are lots of works to do in dissolution phase of VE [65].

Ricardo Chalmeta and Reyes Grangel developed a Performance Measurement System (PMS) in order to evaluate the partners' performances according to the predefined criteria. This system measures partners present conditions, situations and compare them with the projects strategic targets and help VE management to make right decisions in next VE projects partner selection phase. The main data for performance evaluation of this system is gathered in the dissolution phase of VE project and all data are updated after the evaluation [21].

1.6 Literature Survey on VE Applications

Researches regarding VE platform and architecture development are along with one or more pilot demonstration(s). In order to prove and verify system performance these pilot applications were mostly concentrated on a specific sectors or projects. Based on VE research comprehensiveness, reusability of VE systems are questionable. Some projects were not responding new requirements and conditions for new sectors or projects, however some other are partly able to fulfill new projects or sectors requirements by minor reconfigurations. Actually the main challenging issue about designed and applied VE systems is their reusability and ability to be reconfigured for new platforms' necessities. New developments in information, communication and network technologies enable researchers to empower VE tools.

One of the VE applications was demonstrated in TECHMOULD project. This project was developed with concentrating on casting and molding industry. Rival SMEs in molding industry from Brazil came together to collaborate for capturing opportunities from market and make more powerful cooperation and increase their compatibility in the market. In order to respond customers quickly and act agile a decision support system (DSS) to select the most appropriate partner for VE was developed. In this way, a broker system was designed to collect bids from selected partners to respond customer quickly. Actually here DSS was developed based on MASSYVE project's DSS tool. Developed DSS in MASSYVE project has a multi-agent based architecture and established on bargaining between enterprise and coordinator agents. In order to be a partner for the forthcoming VE project it is not enough to being selected by DSS tool. Outcomes from DSS tool are sent to TECHMOULD management board and final approve comes from management board of the project. Here a hybrid system structure for partner selection has been applied [47] [92].

1.7 Literature Survey on Ontology Theory and Applications in Manufacturing Systems

Ontology in computer engineering and informatics is a knowledge representation method. It means that ontology is a way of modeling the domains, environments or widely the world around us as set of objects, their properties, rules and relationships. Ontology enables classification of data in domain specific extensions and link data to each other in an organizational structure [123] [50].

Knowledge and information transfer in distributed smart manufacturing system between agents is possible by defining clear terms and descriptions in the system. One of the first attempts to develop a web ontology language (OWL) for manufacturing management systems was done by Merdan [79]. Koppensteiner also proposed an ontology based architecture for management of assembly processes [67]. In this research different agents, pallets, products, conveyor and other parts are observed continuously and system management can control them [67]. In order to assign right tasks to the right work stations integration of ontology and agent based system is proposed by Candido and Barata. In this study two types of concept is defined; modules and skills [27]. Modules are considered as work station and skills are considered as the properties of these work stations and manufacturing elements. Therefore manufacturing resource agent by examining workstations conditions and their properties is enabled to assign the right task to the right manufacturing resource. In more complicated manufacturing projects, coalition leader agent using these information could be able to choose different types of stations to collaborate for producing more complex products [27]. This system has been tested and verified in UNINOVA smart robotic center in NovaFlex in Portugal [27].

In order to reconfigure the multi-agent based VE system an OWL based ontology developed by Al-Safi and Vyatkin [3]. Developed structure is very similar to the structure which has been developed by Candido and Barata but there are also some differences. For example, instead of using modules and skills here Material resources and operations are considered as the main concepts of the ontology model. Reconfiguration agent in this system according to the changes in the product line and shop flour requirements reconfigure the process planning and scheduling of the products and load of machine tools [3]. This research is mostly focused on shop floor and process planning and scheduling in a single enterprise however Vrba extended such system to the distributed manufacturing systems from accepting new orders to arranging transportation according to the work load of enterprises [122]. This system was integrated with the previous research outcomes later on [122].

From introduction of first VE platform till today different studies have been accomplished to develop most appropriate VE architecture and different types of tools and applications. According to the requirements of different sectors of industry, different types of VE infrastructures were proposed. However attempts for establishing a generic, multi purpose VE system got few or even no results. Insufficient computer and information technology developments could be mentioned as the main reason behind fruitless endeavor. By considering recent developments in information and communication technologies and computer sciences a new approach to redesign VE infrastructure and data management in VE systems is proposed in this research. In this dissertation in order to enhance VE systems flexibility and create a generic, multi purpose VE infrastructure a new ontology based VE domain model and an agent based infrastructure is developed. This proposed VE infrastructure enhance VE systems' reusability and reconfigurability.

In the following sections, first OMAVE system architecture is described. Then Developed OMAVE system's IDEF0 and UML diagrams are illustrated and discussed. Next chapters disclose OMAVE system's ontology model and multi agent based infrastructure of OMAVE system. In this thesis beside development of an infrastructure for VE systems different data and information management, partner selection and performance evaluation tools are also developed. Partner selection process and defined partner selection procedure and algorithms are explained in chapter 6. Chapter 7 demonstrates other developed tools and components in OMAVE system. Finally through a use case system performance and verification is evaluated and tested.

CHAPTER 2

OMAVE SYSTEM ARCHITECTURE

2.1 Proposed Architectures for VE systems in Literature

The main reasons for forming a new VE are to share the core competencies of multiple independent companies and to create an alliance in order to benefit from short or long term business opportunities. SMEs must react quickly in order to take the advantage of potential business opportunities identified through market analysis or based on incoming orders [106]. Limited capabilities may cause SMEs to lose local and temporary customers to large enterprises. Thus, in order to foster competitiveness and growth, it is essential for SMEs to collaborate with:

a) each other, in order to benefit from various vertical competencies and,
b) high tech start-ups in university techno-parks

To produce high value added final products [9, 17, 121]. By forming a VE consortium, SMEs increase their competitiveness by sharing risks and complementing their capabilities in the face of ambiguous market environments [25]. Many scholars agree that the VE life cycle has three distinct phases; formation, operation and dissolution 2.1 [121].

Each phase of VE life cycle includes several special activities with different procedures. Concurrently different level of integration should also take into the consideration. Several aspects of integration including technical, financial, organizational and legal points of views should be considered at the same time. Virtual enterprise systems are quite complex and consequently their installation, set up for different sectors, and configuration with companies -VE members-

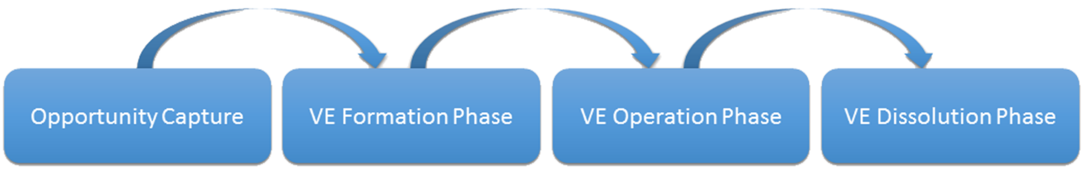


Figure 2.1: VE Lifecycle [121]

information and business infrastructures is quite time consuming and expensive [134]. Normally, success of virtual enterprise systems is highly dependent on partners' performances. Most of VE activities have low level of formalism thus, traceability of provisionally made decisions is low therefore repeatably of successful decisions decrease.

In order to cope with all these complications different architectural frameworks for VE systems have been proposed. The main object of these frameworks are to bring formalism to VE systems by proposing and establishing an information and knowledge hierarchy. In this way, all information and data in the system could be organized and knowledge repeatability is increased. System efficiency increases significantly because of enhancements in re-usability and traceability of previously taken successful decisions, used tools and methods.

VERAM (Virtual Enterprise Reference Architecture and Methodology) is one of the successful architectural frameworks for virtual enterprise systems which was proposed and created in Global Engineering and Manufacturing in Enterprise Networks (GLOBEMEN) project. VERAM is developed based on GERAM (Generalised Enterprise Reference Architecture and Methodology) framework which is a platform for integration of enterprises existing knowledge and information. By the same way GERAM itself is drived from other enterprise integration platforms like CIMOSA (Computer Integrated Manufacturing Open System Architecture) [34], GRAI/GIM(Graphs with Results and Actions Inter-related/ GRAI Integrated Methodology) [77] and PERA (Purdue Enterprise Reference Architecture) [11]. As it can be seen from the definition of the these reference models, they all are dealing with enterprises integration frameworks and not a VE framework. In virtual enterprise system, partner enterprise information or business systems are not integrated to each other or to a core enterprise [134]. VERAM framework is mainly focuses on formation and operation phases of VE life cycle. VERAM has three layered structure and each layer has its own architecture 2.2. These three layers are named as follows:

1) Virtual Enterprise Concept,
2) VERA - Virtual Enterprise Reference Architecture
3) VERAM Components

VERAM components has four sub-layers naming; contingency factors, modeling, applications and infrastructures and methodology. Contingency factors are including roles, legal aspects, business environments, standards and technologies. Modeling encapsulates different system modeling methodologies and tools. Enterprises applications, tools and every VE configuration tools are placed in applications and infrastructure sub layer. Methodologies sub layer include guidelines for VE implementations and any system implementation requirements. Guidelines actually depicts how all other sub layer components must be used in practice for that special implementation use case.

For modeling VE infrastructure, different modeling tools are used. VERAM has 5 types of models including entity relationship, Integration Definition for Function Modeling (IDEF0)

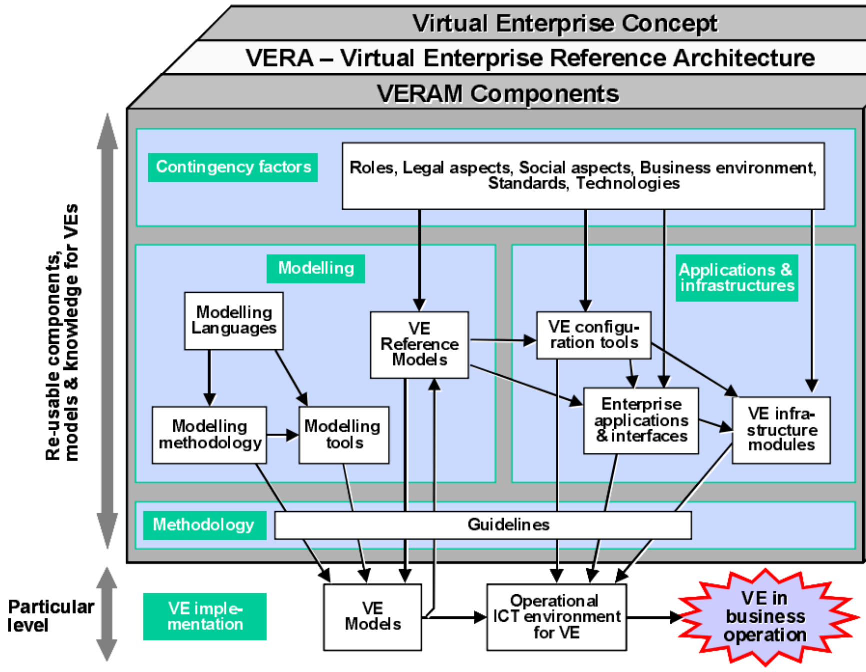


Figure 2.2: VERAM- Virtual Enterprise Reference Architecture and Methodology [134]

and Unified Modeling Languages (UML) models. These models are, process models (IDEF0 diagrams), use case, sequence diagrams, classes (UML models) and components models (entity relationship) and finally interface and application specifications.

2.2 OMAVE System Architecture

Like VERAM reference architecture, a new approach for VE reference architecture is being considered in this research. The main focus on this proposed architecture is to enhance data and knowledge re-usability and generalize VE architecture for responding more diverse industrial sectors also for producing high value added products by manufacturing SMEs their collaboration with design and research SMEs and institutions also is taking into account. In order to include innovative research results or redesign and optimize an existing product operationally and/or environmentally, a step prior to the operation phase is required. This phase, called the collaborative design and engineering phase in some VE projects, is supposedly included in the operations phase; however, it has not yielded the expected results [4, 113]. When collaborative design and engineering is part of the operations phase, often only minor changes are possible to be made, and those changes are initiated inside the organization or enterprise that originally produced the product. Such changes are not considered collaborative design, because there are very few with almost no contributions from other partners. The proposed VE architecture in this thesis separates the design phase from operations and proposes a completely distinct design phase inside VE life cycle2.3.

The approach to develop a VE framework here is different than VERAM. Here the concentration is over information and communication technologies which are going to be used and

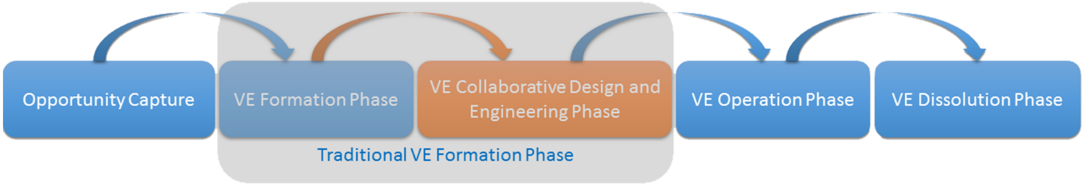


Figure 2.3: Proposed VE Lifecycle [104]

launched in VE architecture. In this research a three layered Information and Communication Technologies (ICT) hierarchy of VE framework is revealed (Figure 2.4).

1) Enterprises layer,
2) Shared Database and Applications Layer
3) Customer and Interfaces

The bottom level of this hierarchy involves data and information pertaining to system members and partners, such as SMEs, Original Equipment Manufacturers (OEMs), research institutes and R&D start ups in techno-parks. Incoming Information from these members information and business tools are gathered by system interfaces designed in this layer. These interfaces and tools which are responsible for obtaining required information and knowledge from members are agents. Different type of agents are assigned to capture various type of data from member enterprises manually or automatically. If member enterprise system is equipped with an automated enterprise management system like Enterprise Resource Planning (ERP), Manufacturing Execution Systems (MES) or similar tools these agents directly connect to those system databases and collect required information immediately. However, if the enterprise does not have so called automated systems agent could gather information by sending questionnaires to the assigned authorities by member enterprises.

Middle layer of this hierarchy embeds system shared data warehouse, system model and all integrated VE applications. This layer which is also could be called as administrative layer of VE, itself is divided into two separate layers. Lower layer is system shared database which is developed based on designed VE ontology model. The main difference of newly proposed VE framework in this research and other proposals in VE literature is in this point. Rather than using a object oriented or relative databases, this system makes good use of triple stores which is developed based on expanded VE ontology model. Detailed information about VE ontology model and culminated triple store is outlined in section 4.2.

The upper layer here encloses all applications and tools those are going to be used in distinctive phases of VE life cycle. For instance developed VBE tools for collecting enterprise pool members information, agent based decision support tools could be listed as tools which are used in VE formation phase. These tools and their functionalities are described in 5.3 and 6 sections in detail.

In this research, in order to manage product design and development activities and also han-

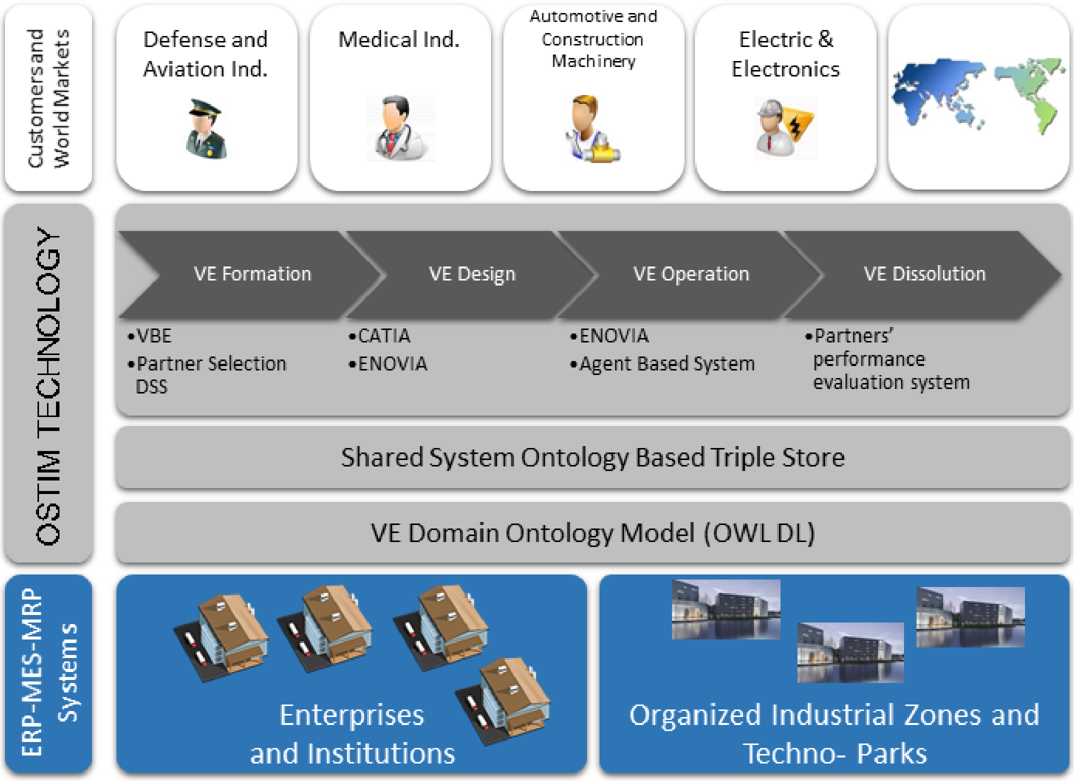


Figure 2.4: OSTIM VE Framework [102]

dle production processes and operation management recently introduced web based Dassault Systems CATIA/ENOVIA V6 tools are utilized. CATIA V6 is predecessor of CATIA V5 but its software structure is highly evolved and a new approach on concurrent and collaborative design and development is introduced by launching this software. In this approach, all design and development activities are carried out on a web based platform and all system users connect to system main frame which is running on a central web server. Clients connect to the main frame based on defined user/passwords and assigned rules by project managers and contribute on designing, analyzing and development of products. ENOVIA V6 is the backbone of this software package. Data and information generated during design and development stage is supported by ENOVIA. ENOVIA could be named as Product Life Cycle (PLM) management tool in this package which supervise and handle data and information flow during all stages of a product life cycle.

Disperse enterprises with different core competencies may contribute remotely during product development stage of product by benefiting from CATIA/ENOVIA tools and this approach fulfills this research ambitions for accomplishing a collaborative platform for SMEs to produce high value added high tech products with cooperation of R&D start ups, universities and institutes from all over the world. These applications in the upper administrative level of the VE architecture are served to system members based on the Software as a Service (SaaS) principals. All members are able to benefit from the applications, independent of user platforms [98].

In dissolution phase of VE, partner performance evaluation tool is running. As it is obvious from this tool name, it evaluate contributed partner enterprises' performances after project completion. In order to assess companies achievements and decide about their circumstances in the forthcoming projects, it is crucial to evaluate their past performances [106]. Finalizing every project partner enterprises activities and operations is precisely assessed and their information, past performance, quality and service values will is updated in the system.

The customer interface is in the upper layer of the VE architecture. In order to communicate and exchange information and data with customers, customer agents are designed and these agents transfer incoming data from customer and bring analyzed, information back to the customer through user friendly graphical interfaces.

2.3 Summary

In order to match with OMAVE system requirements a new approach to design VE system architecture is presented in this research. The main focus in development of this architecture is to focus on ontology model, and establish all system components based on system ontology model. The other main feature of this architecture is to connect to partners' systems and establish required data transactions without interfering in their internal system.

CHAPTER 3

OMAVE SYSTEM MODELING

3.1 System Modeling Theory

To study behavior and functionality of systems in business, engineering, IT developments and systems are conceptualized by means of different types of models. There are different methods to model system functionality and study their behaviors such as Functional Flow Block Diagrams (FFBD) and IDEF0 Diagrams. FFBDs can be developed in a series of levels. FFBDs show the same tasks identified through functional decomposition and display them in their logical, sequential relationship. These diagrams are used both to develop requirements and to identify profitable trade studies. The FFBD also incorporates alternate and contingency operations, which improve the probability of mission success. The flow diagram provides an understanding of total operation of the system, serves as a basis for development of operational and contingency procedures, and pinpoints areas where changes in operational procedures could simplify the overall system operation. In certain cases, alternate FFBDs may be used to represent various means of satisfying a particular function until data are acquired, which permits selection among the alternatives [1].

Likewise IDEF0 modeling method also is developed to model systems functionality based on graphical modeling language of Structured Analysis and Design Technique (SADT). Rather than FFBD which concentrates on functional flow of product, IDEF method is mainly focuses on data flow, system control and also functional flow of a product during its life cycle. IDEF method covers most of the enterprises operation fields and empower analyzers and system engineers to develop system models to any level of details with simple graphical representation tools. Functional system modeling of VE in this research is developed using IDEF modeling methods and more information about IDEF modeling methods are given in the following Section 3.2. [61].

An open consortium of companies called OMG (Object Management Group) developed and controls a family of graphical modeling notations backed by meta-models called UML (Unified Modeling Languages). OMG group is a consortium to create standards to support inter operable object oriented systems. CORBA (Common Object Request Broker Architecture) is one of the most well known standard which is developed by this group. UML was born by

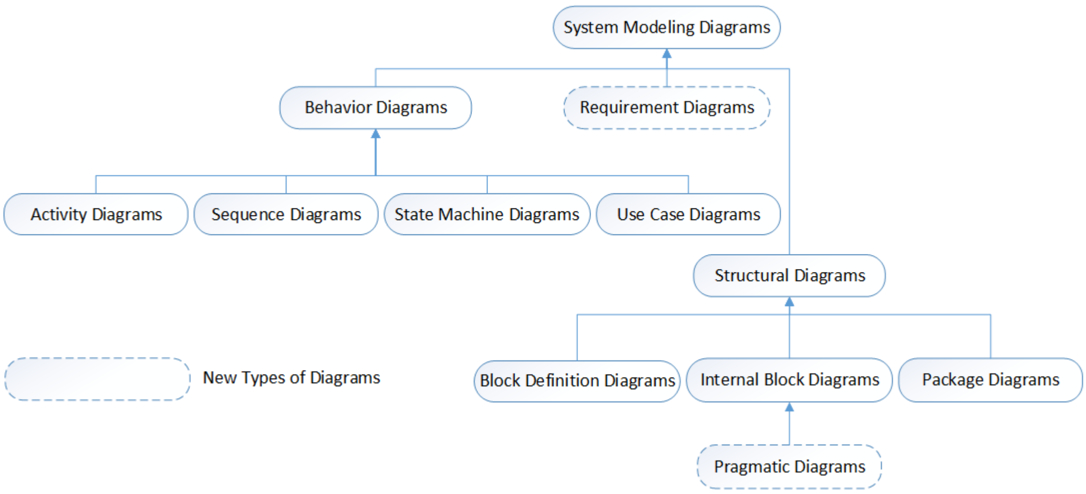


Figure 3.1: UML Diagrams Hierarchy [43]

unification of several graphical modeling languages [40]. UML has a graphical notation and a a supporting meta-model. Graphical notation is displaying method in diagrams and meta-model part deals with relations between features of a model. Different types of diagrams are available in UML modeling method. These diagrams are classified into three main categories. This classification is shown in Figure 3.1.

In this research, in order to depict the system functionality, IDEF0 diagrams are used. For illustrating system components, interactions and behavior different defied diagrams, UML models are implemented. In order to show the system structure, UML class diagrams are used. Meanwhile for modeling system behavior Activity, sequence and use case diagrams are implemented. Based on system or software designers and their requirements different configuration of diagrams could be used in modeling of systems or software.

3.2 IDEF Modeling Method

In the frame of technology modernization of US air forces, “US Air Forces Integrated Computer-Aided Manufacturing (USAF ICAM)” program was developed on 1976 by USAF Materials Laboratory in order to integrate manufacturing activities of firms [60, 109]. Following huge developments in ICAM program, requirement for a new standard for modeling and analyzing business and management processes in manufacturing systems led to establishment of ICAM Definitions (IDEF). Then the ICAM program was renamed to IDEF on 1999. IDEF is a family of modeling languages for various systems and software engineering which is divided into 16 methods from IDEF0 to IDEF14 as shown in Table 3.1 [60, 76].

Table 3.1: IDEF Standards [60]

IDEF Standard	Description	Development Status
IDEF0	Function Modelling	Fully developed
IDEF1	Information Modelling	Extensions stopped on 1985
IDEF1x	Data Modelling	Following merging IDEF1 and LDDT1 , Fully developed
IDEF2	Simulation Model Design	Fully developed
IDEF3	Process Description Capture	Fully developed
IDEF4	Object- Oriented Design	Fully developed
IDEF5	Ontology Description Capture	
IDEF6	Design Rationale Capture	
IDEF7	Information System Auditing	Have not developed further after Initial Definition
IDEF8	User Interface Modelling	
IDEF9	Business Constraint Discovery	
IDEF10	Implementation Architecture Modelling	Have not developed further after Initial Definition
IDEF11	Information Artefact Modelling	Have not developed further after Initial Definition
IDEF12	Organization Modelling	Have not developed further after Initial Definition
IDEF13	Three Schema Mapping Design	Have not developed further after Initial Definition
IDEF14	Network Design	

3.2.1 IDEF0

Integration Definition for Function Modeling (IDEF0) is the first method of IDEF Modeling language family in the category of Software and Modeling Techniques standards and published by Federal Information Processing Standard (FIPS). IDEF0 is especially designed and developed to model, system actions, behaviors and decisions. IDEF0 is originally developed by Douglass T. Ross and SofTech Inc. based on Structured Analysis and Design Technique (SADT) [61]. By developing IDEF0 model of systems it is possible to analyze systems functions, mechanisms and finally evaluate systems' performances. IDEF0 is a hierarchical set of diagrams, texts and glossaries interconnected to each other in order to model all types of systems [61]. IDEF0 diagrams are consist of two main concepts; boxes and arrows. Boxes indicates functions and arrows are representing data and object flow in diagrams [61].

3.2.2 IDEF Applications in VE

IDEF modeling languages are widely used in modeling systems in different industry fields. Researchers also benefited from IDEF modeling languages to model and develop VE system models. These attempts started mainly after 2000. One of the first studies that applied IDEF modeling language to model VE system was "fruit and vegetable business virtual enterprise (VITE)" [84]. The main target of VITE project was connect farmers, wholesalers and retailers in real time to a reliable network of fruit and vegetables and keep costs lower without dropping in quality [84]. The functional and Information Model of VITE are shown in Figure 3.2 and 3.3 below.

Another important application of IDEF in virtual enterprise system modeling is "Open Sys-

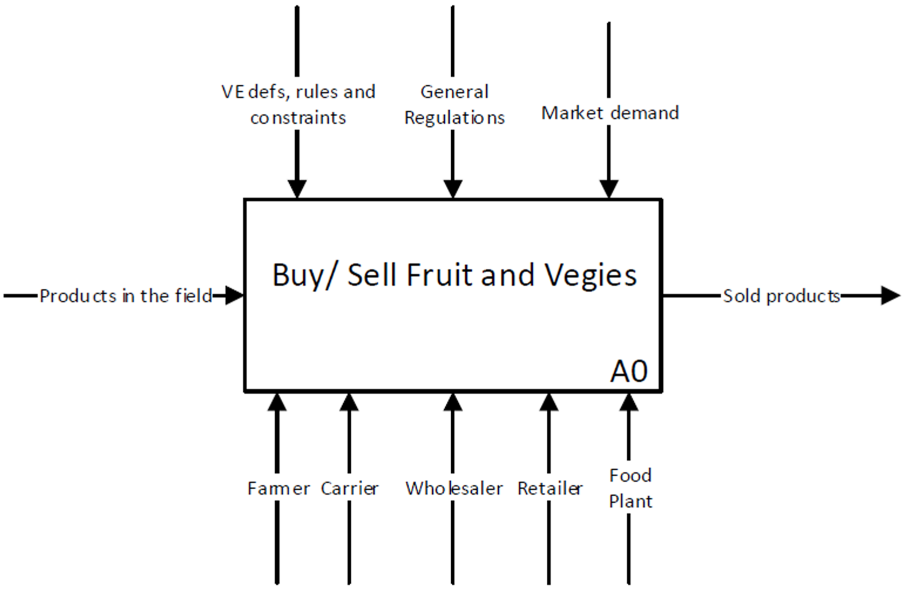


Figure 3.2: Functional model of VITE [84]

tems for inter-enterprise information Management in dynamic virtual Organizations (OSMOS)" project. OSMOS was developed in the frame of European RTD projects and this project targeted to enhance the capabilities of construction enterprises, including SMEs, to act and collaborate effectively on projects, [74]. The OSMOS working principle is project based. Project work packages requirements are recognized. Then partner enterprises business processes and information management practices are analyzed and compared with the project requirements. As shown in Figure 3.4, IDEF0 is used to define high level process activity models describing the business processes and information management practices taking place in the building process, within the OSMOS end-users companies, and also between partners on a construction project, i.e. at level of inter-companies communication [74]. Another IDEF modeling application was "Global Engineering and Manufacturing in Enterprise Networks (GLOBEMEN)" project. This project was developed under the Intelligent Manufacturing Systems (IMS) program [45]. GLOBEMEN aims to create an IT infrastructure, new VE methodology and required tools for enterprises with different information and communication technology (ICT) and different business processes to collaborate efficiently [45] (3.5).

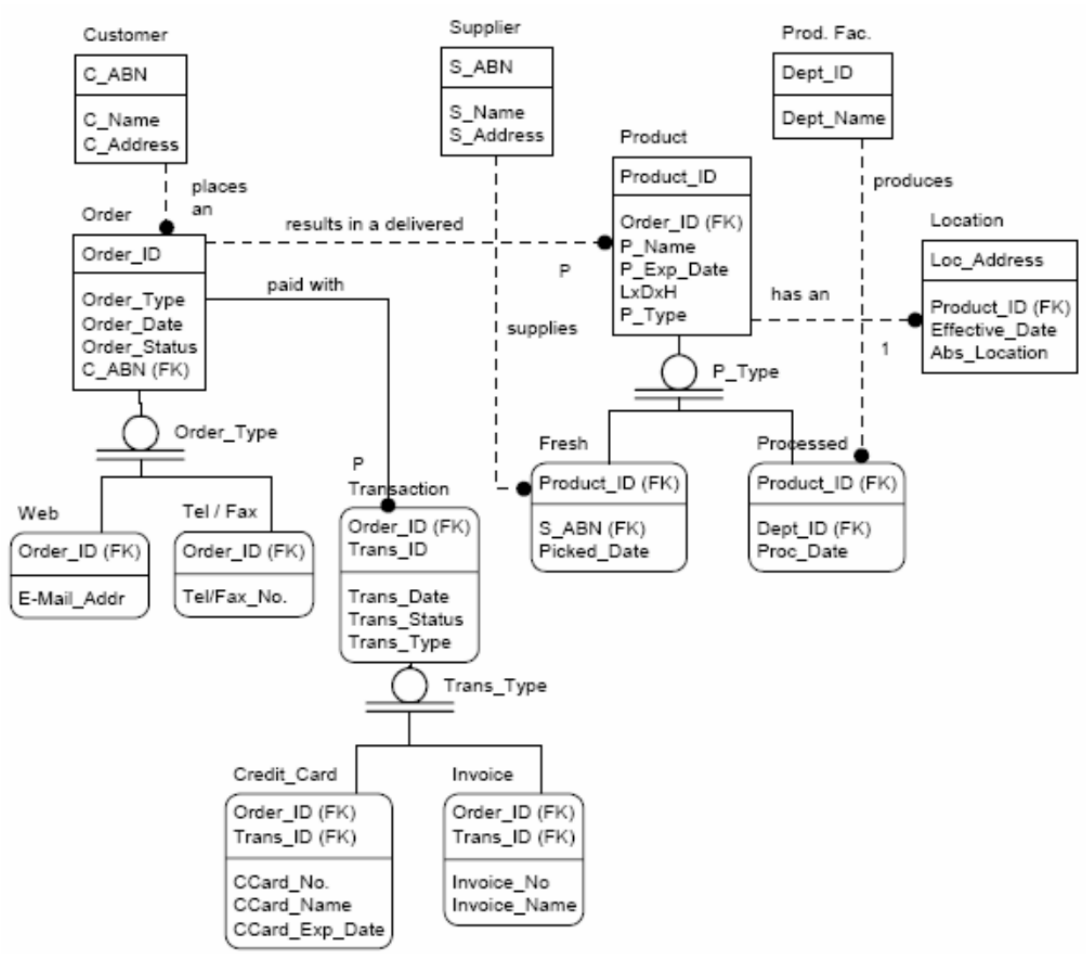


Figure 3.3: Information model of VITE [84]

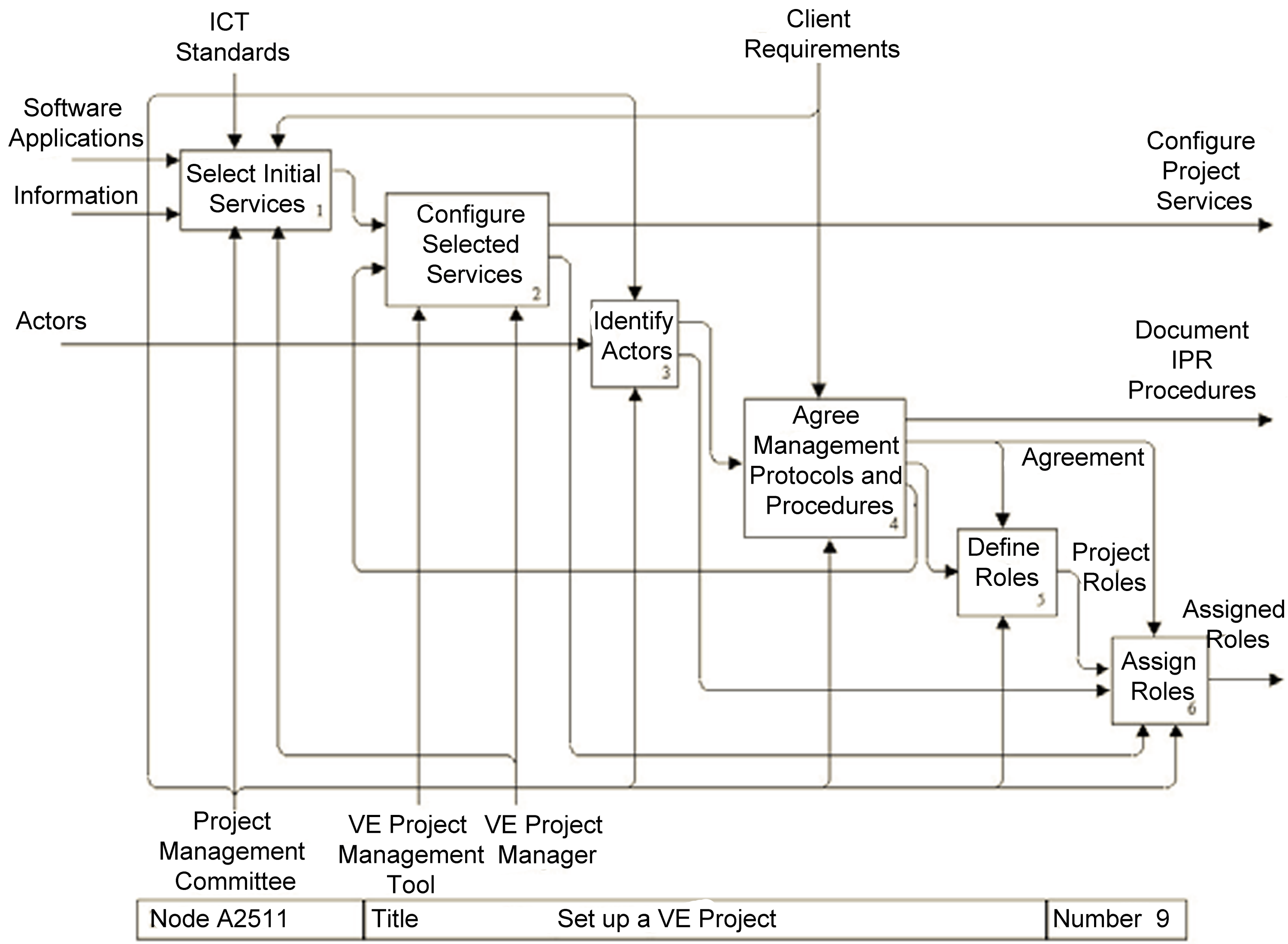


Figure 3.4: Integrated Generic VE Process Model [74]

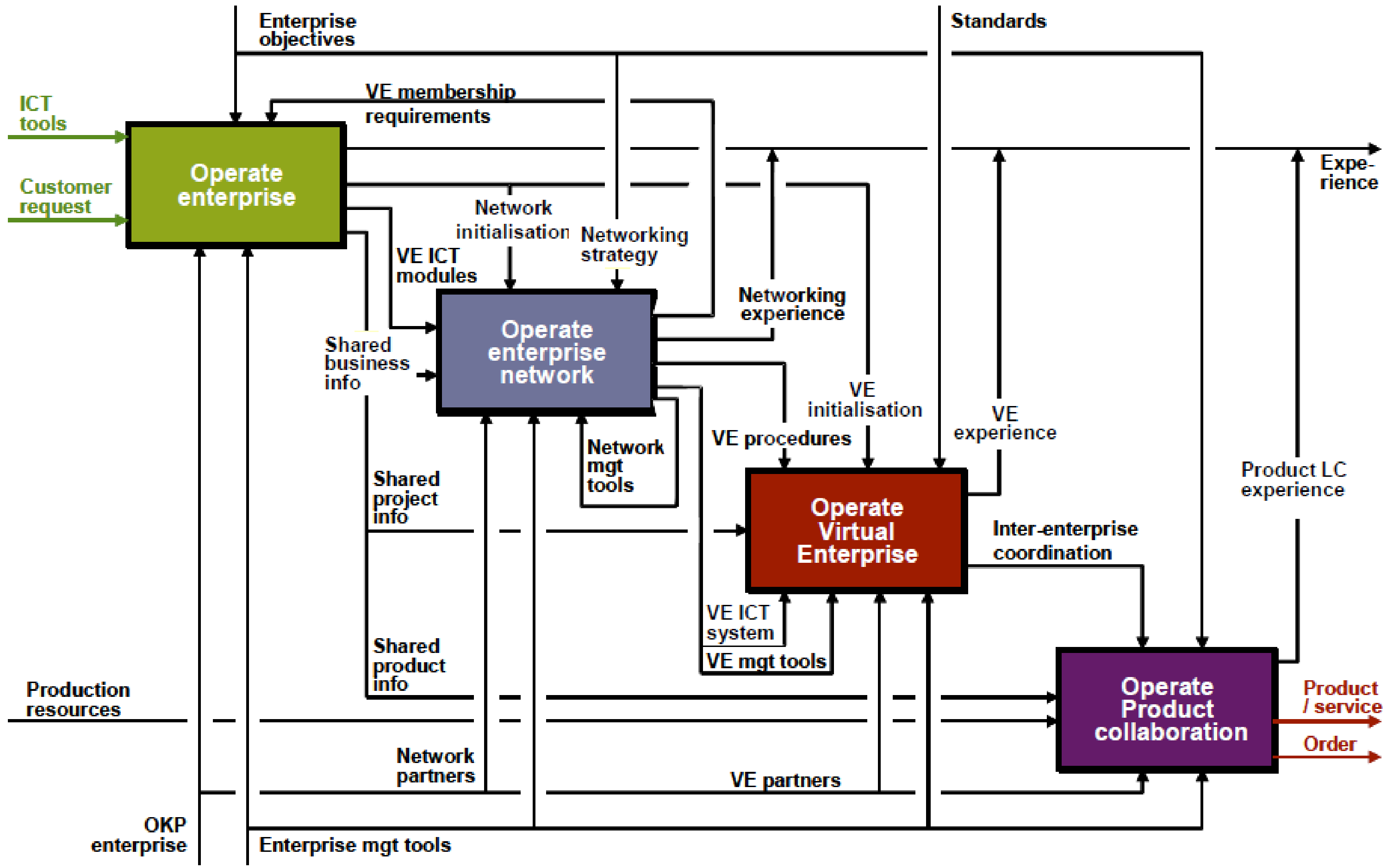


Figure 3.5: High level IDEF0 diagram, Core functions diagram of GLOBEMEN project [74]

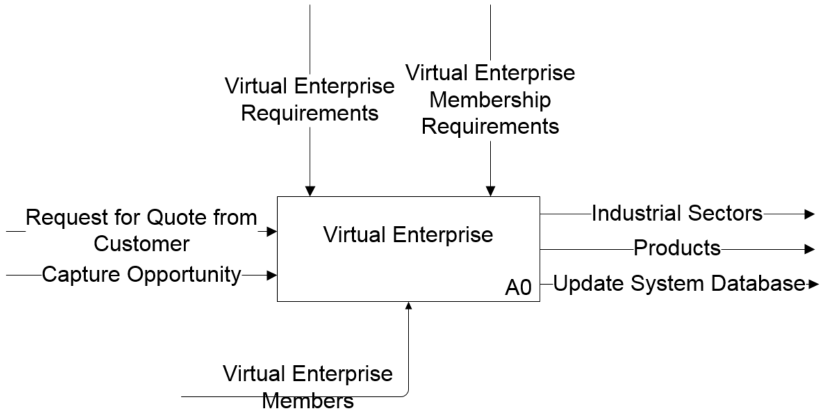


Figure 3.6: VE A-0 Diagram, IDEF0 Diagram of the VE

3.3 OMAVE IDEF0 Models

A graphic description of VE system for targeted purposes was developed. For this reason a set of multiple IDEF0 diagrams that depict the functions of VE system or subject area with graphics, text and glossary was created. The main parent diagram is A-0 Node diagram. Other diagrams are detailed diagrams to describe A-0 node in details. A-0 node is the top level functional diagram of VE system which depicts all system inputs, controls, outputs and mechanisms, along with statements of model purpose and viewpoint. Other detailed diagrams are child diagrams or node decomposition of node A-0.

3.3.1 OMAVE A-0 node

A-0 Node illustrates OMAVE system components, inputs, outputs, control mechanisms generally. Here OMAVE system is considered as a black box which encloses all system components and internal functions in a box called node A-0. The first node or A-0 node of OMAVE system shown in Figure 3.6.

3.3.1.1 OMAVE life cycle IDEF0 Diagram, node A-0 Decomposition

After decomposing the A-0 diagram OMAVE system steps and details could be realized. Figure 3.7 depicts the IDEF0 diagram of OMAVE system components which starts with VBE formation. VBE formation is the primary step of VE to form up a pool of potential members for forthcoming VE partners which can meet the VE requirements. This infrastructure facilitates VE administrator job to decide vigorously on selecting the most appropriate volunteers for the VE projects. The next phase after VBE formation is VE formation phase which is the first phase of VE life-cycle. Then, this procedure continues by VE Design, operation and dissolution and ends by VE management board consent in order to finish the VE project.

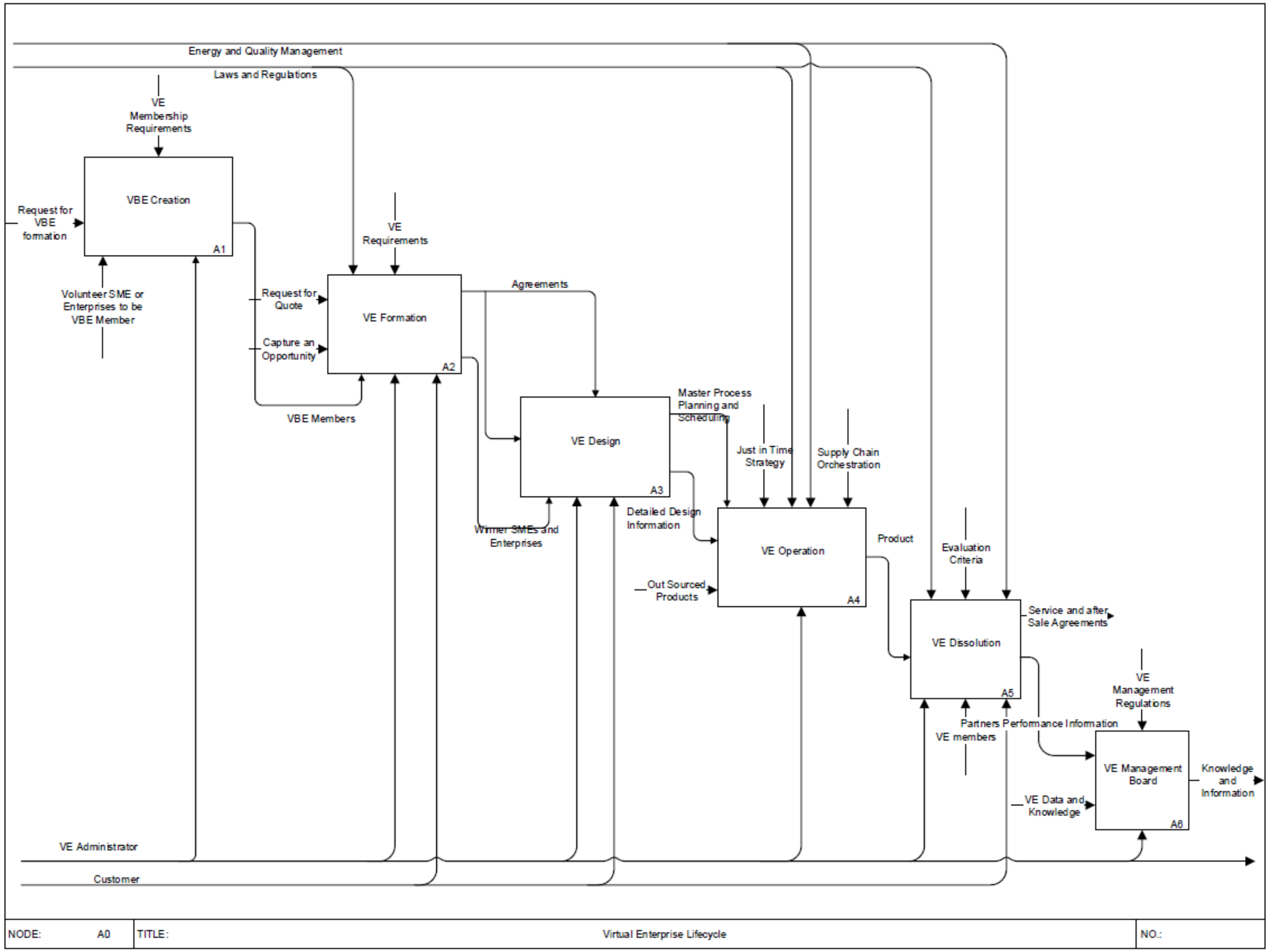


Figure 3.7: Decomposition of A-0 Node, Virtual Enterprise Lifecycle

3.3.2 OMAVE Virtual Breeding Environment (OMAVE- VBE)

In order to establish a virtual enterprise consortium to respond an order from a customer or market research, it is obviously needed to call volunteer enterprises to submit their bids, and compete with each other. VE administrator should have access to the enterprises directly for communication, send and acquire data and information to and from enterprises through a standardized protocols and network. Establishing communication through the internet just by searching capable companies is next to impossible because there are no homogeneous formats and communication ways between enterprises and VE administrator and also the chance of finding enterprises from the world wide web is not equal, besides enterprises without formal websites are totally unavailable through world wide web.

To design a robust and agile VE system, it is required to create an enterprise pool for selecting the most appropriate partners for the projects. This pool is actually a breeding environment for all the volunteer enterprises, eager to participate in VE projects. These enterprises are ought to register in this breeding environment and provide required data and information to OMAVE system. As this is a virtual pool of enterprises, it is called virtual breeding environment (VBE). First advantage of VBE is to gather volunteer enterprises requisite for the future VE projects partner selection stage. The second important advantage of establishing a VBE system, is to homogenize all information and data about the submitted enterprises. Therefore, VE administrator treat equally to all VBE members. Members with incomplete information and data probably lose the competition thus in order to increase the chance of winning in partner selection competition members have to complete the information requested by VE administrator.

All the information regarding VBE members are gathered and saved in standardized forms. General information part for all clusters are the same in the forms. However, there might be needed to have some come complementary information from enterprises regarding their industrial sectors. These sections also are standardized for different industrial clusters.

3.3.2.1 OMAVE-VBE IDEF0 Diagram, node A-1 Decomposition

As shown in Figure 3.8 VBE processes inside A-1 node are displayed in detail. The required processes for registering a new volunteer enterprise to the pool and registration approval process by VE management board are the main processes of A-1 node.

In order to participate in a VE project, being registered in VBE system is compulsory. Therefore, all the VE potential partners should register in VBE system first. After passing the partner selection process of VE, winning the negotiation procedure and get the management board approval, enterprises would be eligible to join VE consortium.

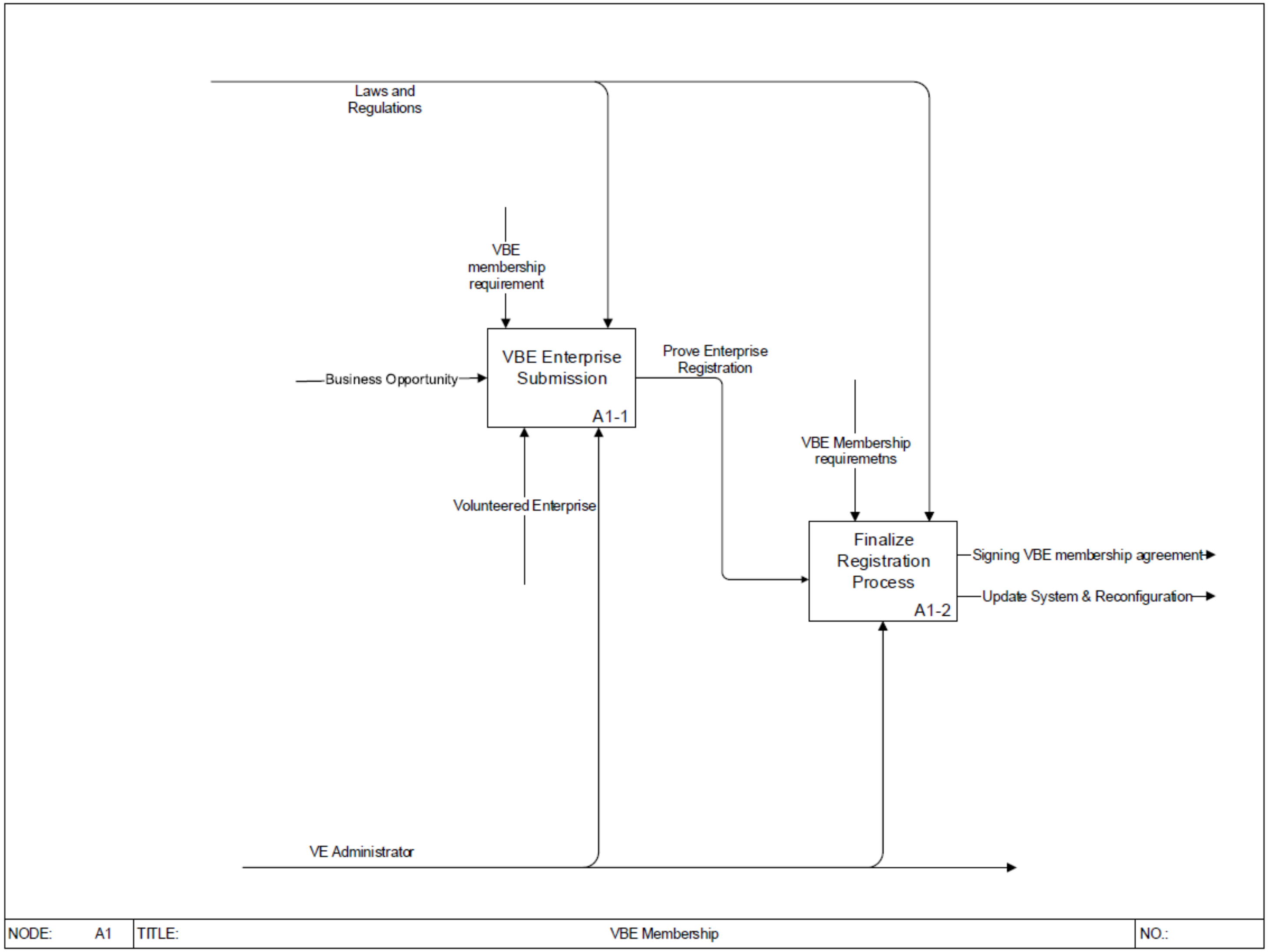


Figure 3.8: Decomposition of A-1 Node, Virtual Breeding Environment (VBE)

3.3.3 OMAVE Formation Phase- Request for Quote IDEF0 Diagram, node A-2 Decomposition

Before Giving any order to any company to produce any product, it is inevitable to discuss some technical and financial issues by customer and manufacturing company. It is obviously the same in VE. Here in VE organization, VE administrator is responsible to conduct customer and call enterprises to give their bids, and ask VBE members to discuss technical issues of the product. In order to start the product order submission, request for quote phase is started by customer. In this stage, as shown in Figure 3.9, customer submit its request. Customers' information is obtained and registered in VBE database. This registration process needs VE management board approval.

After this stage product technical requirements and expected operational specifications are submitted by customer. VE administrator defines project and potential tasks in the project. VBE manager provides a list of eligible enterprises according to their resources and qualifications upon the incoming request from VE administrator and project tasks. Call for bids are sent to the selected enterprises. Based on enterprises updated data in VBE database and incoming bids, enterprises are ranked. Note that this partner selection process is a multiple stepped hybrid procedure. After auctioning and discussions between potential VE partners and customer in the case of getting an agreement, ranked list of enterprises and winner enterprise name is sent to management board to finalize partner selection procedure.

Final step in VE formation phase is to sign the agreements between VE administrator, customer and also between VE administrator and VE partners. These agreements are signed separately. Definitely, the included items and substances are different. Officially, VE project starts after signing the agreements by all of the partners, customer(s) and VE administrator.

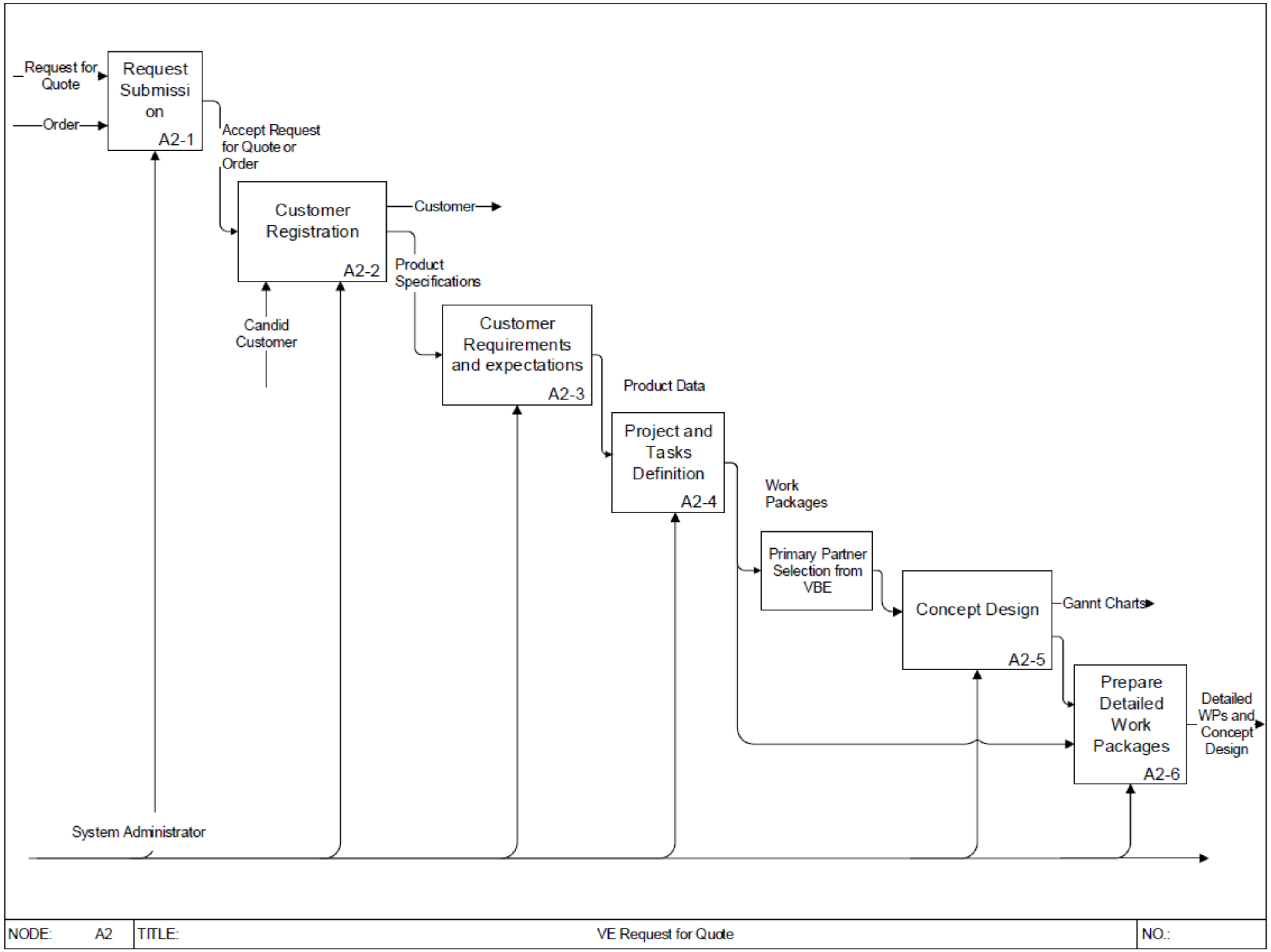


Figure 3.9: Decomposition of A-2-1 Node, Request for Quote Stage- VE Formation Phase

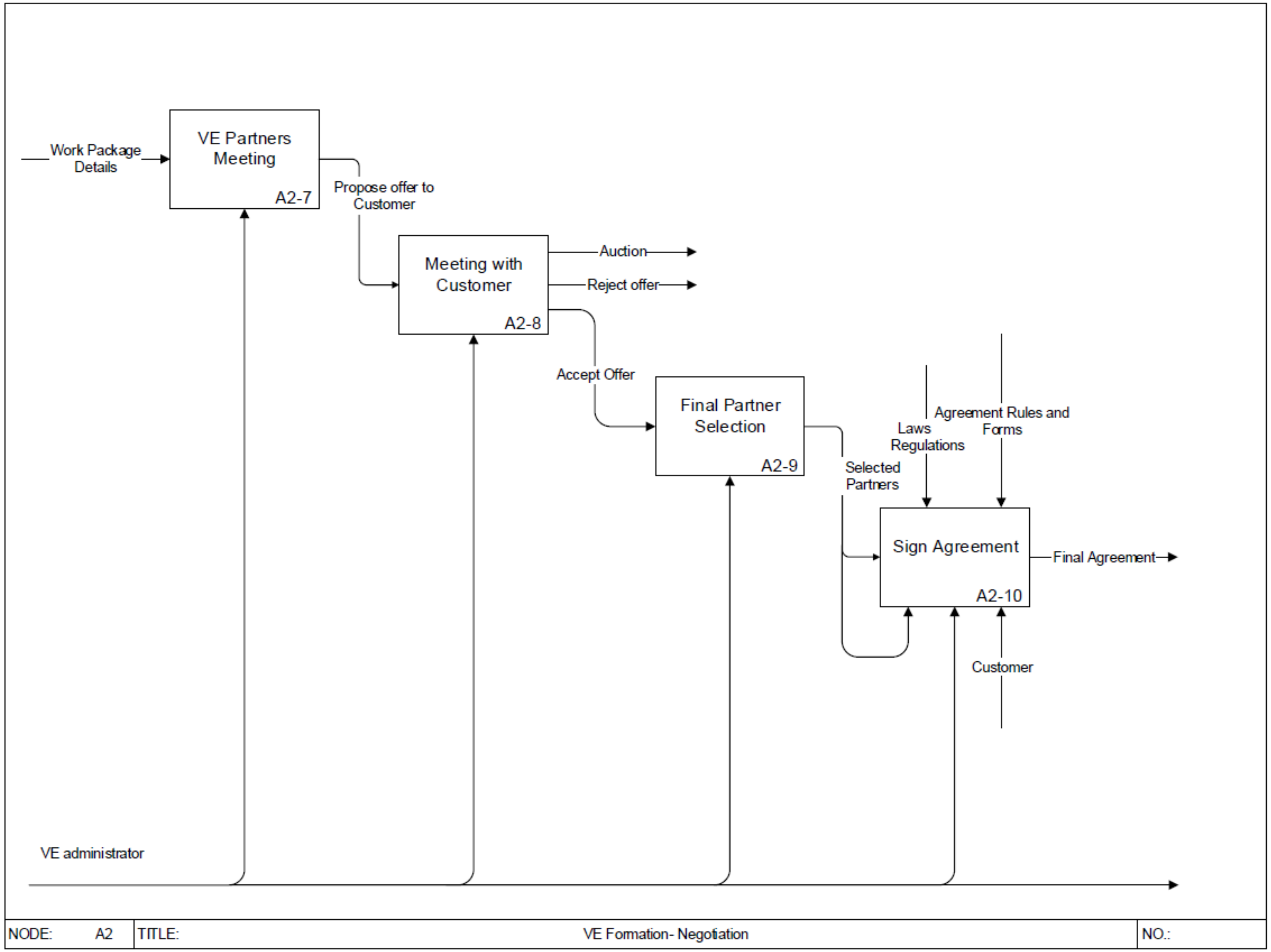


Figure 3.10: Decomposition of A-2-2 Node, Request for Quote Stage- VE Formation Phase

3.3.4 OMAVE Design Phase IDEF0 Diagram, Node A-3 Decomposition

After VE formation phase, next step is product design phase. Definitely in some of the cases, where customer provides product detailed design this phase, could be ignored. But in this research perspective, both scenarios are considered. In order to describe the VE design phase it is assumed that a product order without design is submitted by customer. Design phase IDEF0 diagram is depicted in Figure 3.11 below. Before continue to describe the design phase of VE it would be useful to concisely have a look at collaborative design tool explanation.

Collaborative design is a challenging part in VE architecture. Different studies with various approaches have been taken to develop a collaborative design platform or tool. The results were not satisfying and developed tools were not working properly most of the time. But along with enhancements in computer, information and communication technologies, different commercial collaborative design tools released and are available in the market today. Due to defense industry requirements and standards most of the enterprises in this sector are preferring to use Dassault Systemes new web based design, Product Life cycle Management (PLM) and simulation tools. As this study's outputs will be demonstrated in defense industry cluster of Ankara Industrial Zone (OSTIM) it is preferred to implement CATIA design and ENOVIA PLM tool. Developed OMAVE system will be implemented in OSTIM Organized industrial zone defense cluster by contributions of three active SMEs which are concentrating on metal cutting, sheet metal forming processes. Comprehensive details about the related project and pilot implementation is described in chapter 8.

Different types of accounts based on responsibilities of designers and product design requirements are assigned to designers from different enterprises. Some accounts are designed to only supervise the design procedures, some have limited access to view or edit, and some are restricted to view or edit special part(s) of design. Design accounts have different access levels which are arranged and controlled by project manager. Designing process starts after creating all accounts for designers and according to the design scheduling detailed product design is carried out. Detailed design is developed according to the customers' requirements, approved concept design and products predefined specifications. Definitely customer also could be included in design development activities if it is needed.

Before commencement of process planning and production scheduling, final approval of customer for the finalized product deign must be obtained. Then, master process planning will be developed. VE design phase outcomes are detailed design of all parts of the product, master process planning and scheduling of VE project. Note that, as here consortium is consisting of independent partners, system could not interfere in shop floor scheduling and production planning of enterprises. Master scheduling only provides data and information about project tasks order and requested time table of different work packages and it is expected from enterprises to handle their assigned work packages according to the project time table and take responsibilities regarding their owned tasks.

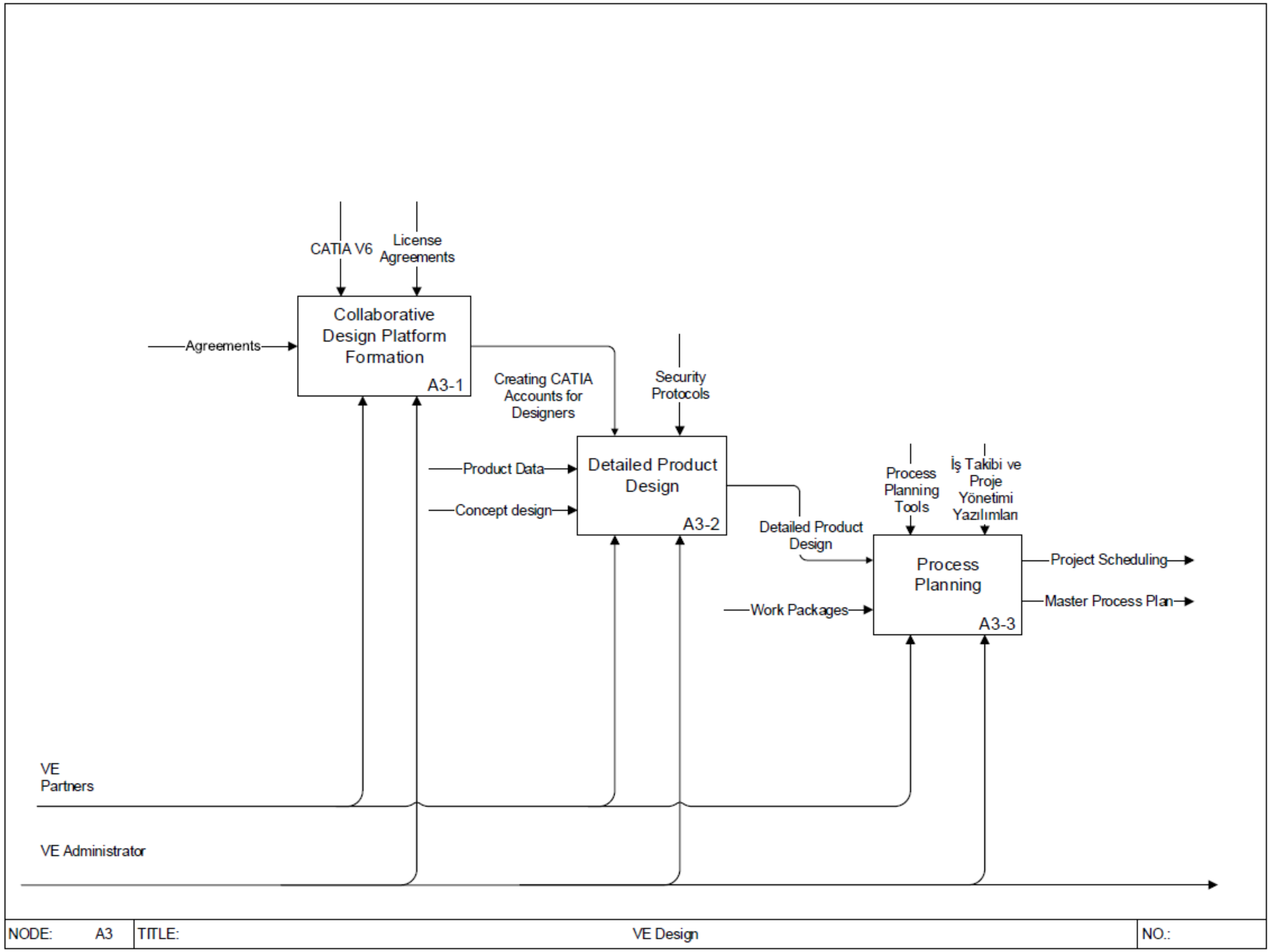


Figure 3.11: Decomposition of A-3 Node, VE Design Phase

3.3.5 OMAVE Operation Phase IDEF0 Diagram, Node A-4 Decomposition

Orchestrating the activities of enterprises and work flow management are the most challenging issue in virtual enterprise management. There are different PLM, manufacturing execution systems (MES), work flow management (WFM) systems, and so on which are dealing with management problems. However, most of these tools are designed to manage shop floor activities inside a single enterprise, or at most between different departments and sections of bigger multinational enterprises. In this situation all detailed shop floor and activity data are open to the system, and there is no security restrictions for management systems. But, working and managing conditions in VE platform is completely different. Here the main target is to avoid from interfering in partner enterprises shop floor activities and do not intervene with their internal policies and programs. In order to fulfill these requirements and manage VE work flow, a multi-agent based approach is proposed. IDEF0 diagram and model of VE operation phase is shown in Figure 3.12. Here according to the agreement items, work packages, product designs, process plans, scheduling and assigned task necessities, partners are requested to submit periodic reports about their activities. Time table to submit reports are clarified based on project task scheduling and master process plan. VE administrator monitors, follows submitted reports, evaluate them and tries to keep up with the master scheduling of the project task. Report evaluation process by VE administrator is discussed in details in 3.4.8 section.

3.3.6 OMAVE Dissolution Phase IDEF0 Diagram, Node A-5 Decomposition

VE dissolution is the final phase of VE life cycle. Manufactured and assembled product is ready to pass final quality control and tests. This phase's IDEF0 diagram and processes are shown in Figure 3.13 below. Product should meet all the required specifications, properties and fulfill are requirements which are mentioned in agreement. Product verification procedures pass under the customer supervisory and needed reconfigurations and setups undertaken accordingly. If it is the case the assembly and system configuration should be carried out in customer's predefined location. In order to finalize the project, it is compulsory to get the customer consent. Thus, product must meet expected characteristics and specifications, stated in the project agreement.

After getting the customer's satisfaction, final payments are done. Final reports, are released and VE administrator asks for permission to terminate the VE project. Termination permission is guaranteed by VE management board. Getting permission, VE administrator evaluate VE partners' performance based on the final project reports, and update VBE database accordingly. Project accounts, including design process (Dassault Systems accounts), financial bank accounts, and etc. are closed after project dissolution and final reports are archived in VE records.

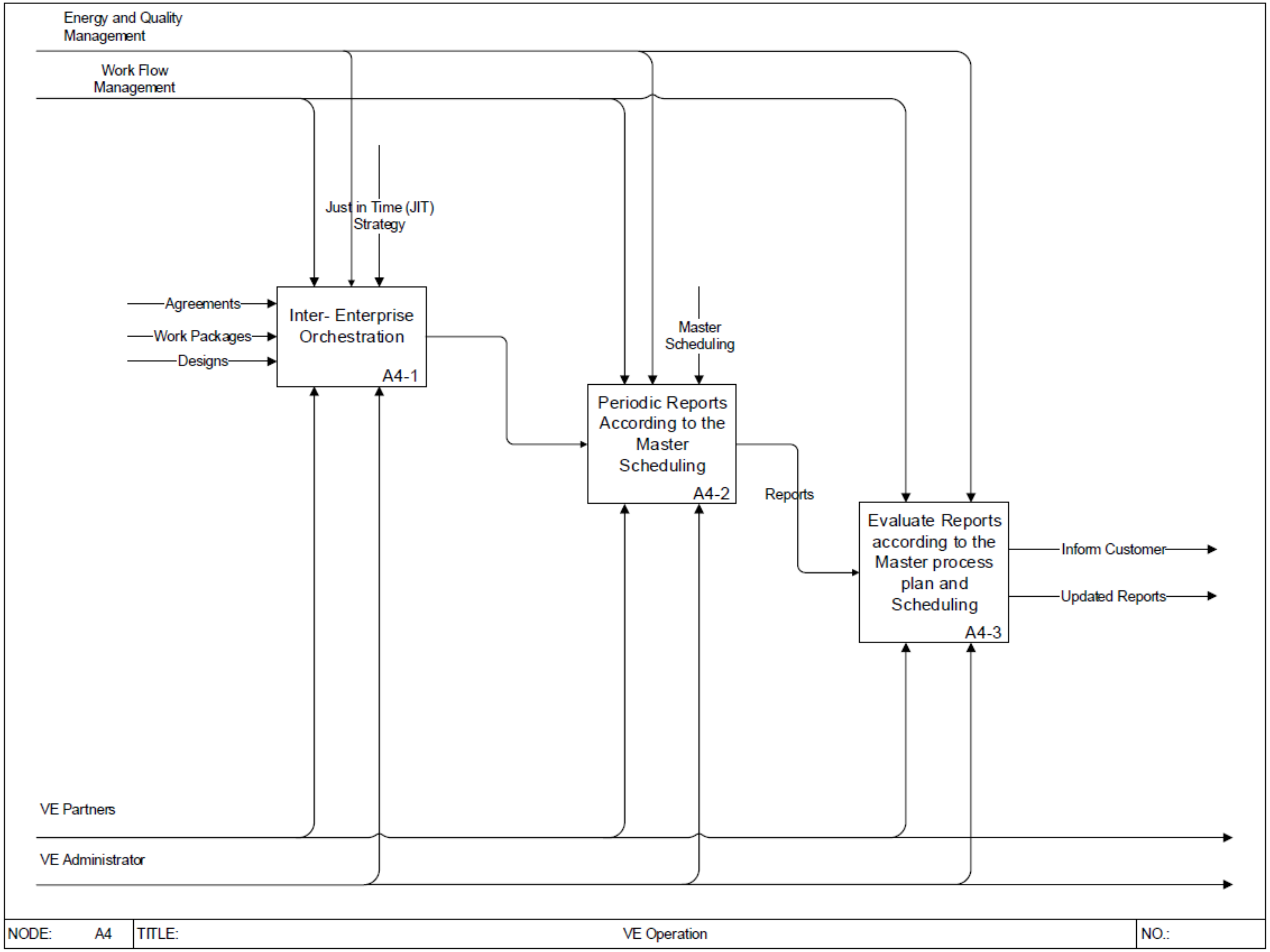


Figure 3.12: Decomposition of A-4 Node, VE Operation Phase

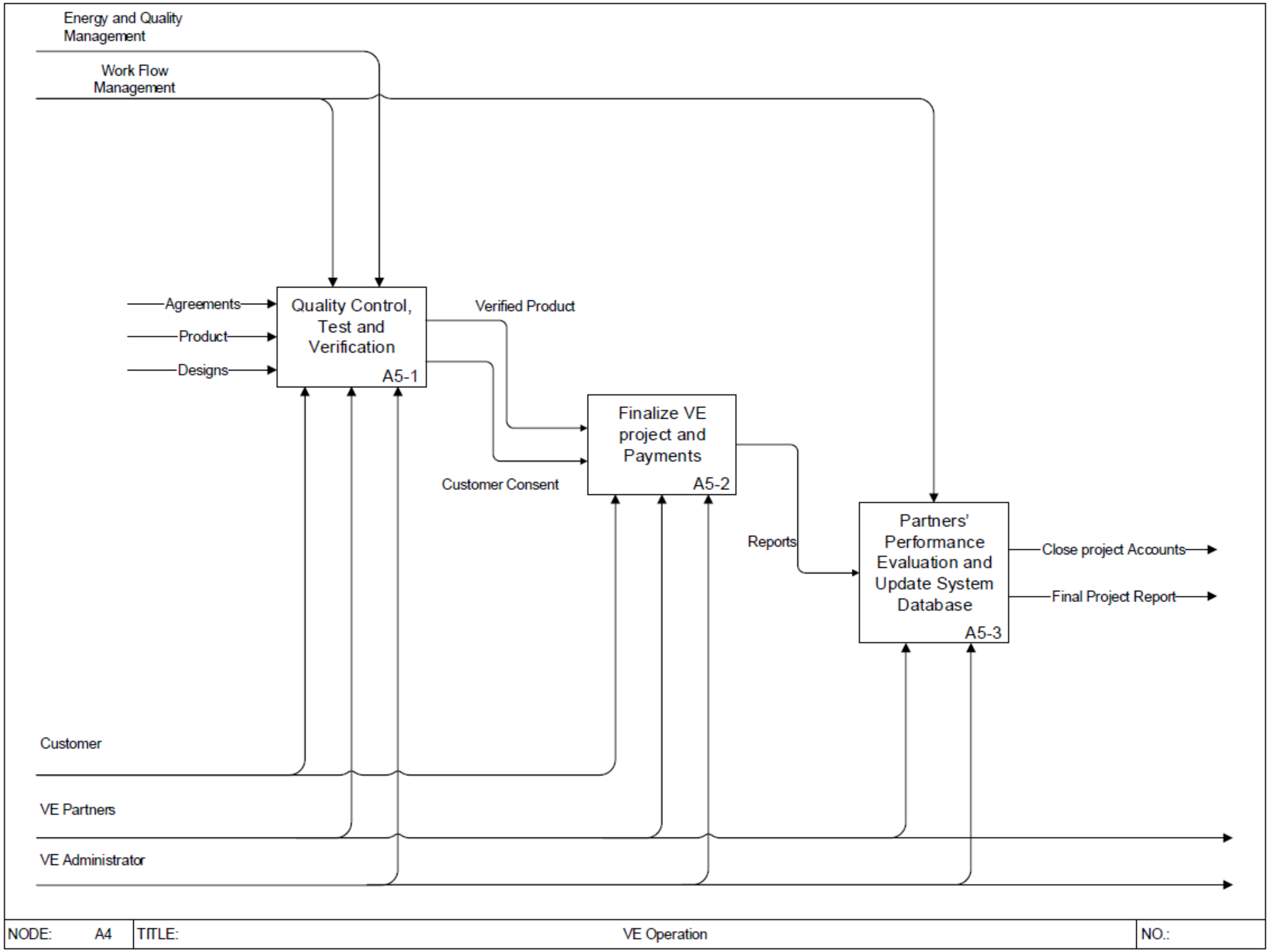


Figure 3.13: Decomposition of A-5 Node, VE Dissolution Phase

3.4 Unified Modeling Language (UML)

Unified Modeling Language (UML) is a visual modeling language for developing software. UML was developed and is maintained by Object Management Group (OMG) founded in 1989, which is an international, open membership, not-for-profit computer industry standard consortium. UML is standardized ISO/IEC 19501:2005 as general purpose modeling language. UML modeling language provides tools to specify, visualize and document models for different purposes. Actually, the starting point of UML development was to prepare a visual modeling tool for software development. However this modeling tool now can be used in various software and system modeling in different engineering areas. Taking advantage of UML tools, it is possible to model a system and analyze system applications and functions, find out the problems then provide required solutions. By end of 2004 UML 2.0 was released. Thirteen types of diagrams are defined in UML 2.0. These diagrams are divided into three main categories as follows;

1) Static Application or Structural Diagrams:
This group includes 6 different diagrams including: Class Diagrams, Object Diagrams, Component Diagrams, Composite Structure Diagram, Package Diagram and Development Diagram.

2) Behavior Diagrams:
This type of diagrams includes three diagrams as; Use Case Diagram (used by some methodologies during requirements gathering); Activity Diagram, and State Machine Diagram.

3) Interactions Diagrams:
Sequence Diagram, Communication Diagram, Timing Diagram, and Interaction Overview Diagram.

In order to make system more productive and understandable for UML proficient people eager to join the project it would be reasonable to use UML diagrams. In modeling systems or software, use case diagram, class diagram, sequence diagram, activity diagram, are mostly preferred.

3.4.1 UML Class Diagrams

Class diagrams are defined to show entities (data, things, people and etc.) and their relationship in a system. Entities in class diagrams are shown by rectangles which are divided into three separate sections. In upper section of rectangle class name, in middle section, class attributes, and in lower section class methods are represented. A sample class diagram is illustrated in Figure 3.14. Here 4 different classes called; student, enrollment, seminar and professor are clear. Each of these classes have their own properties like name, address, phone

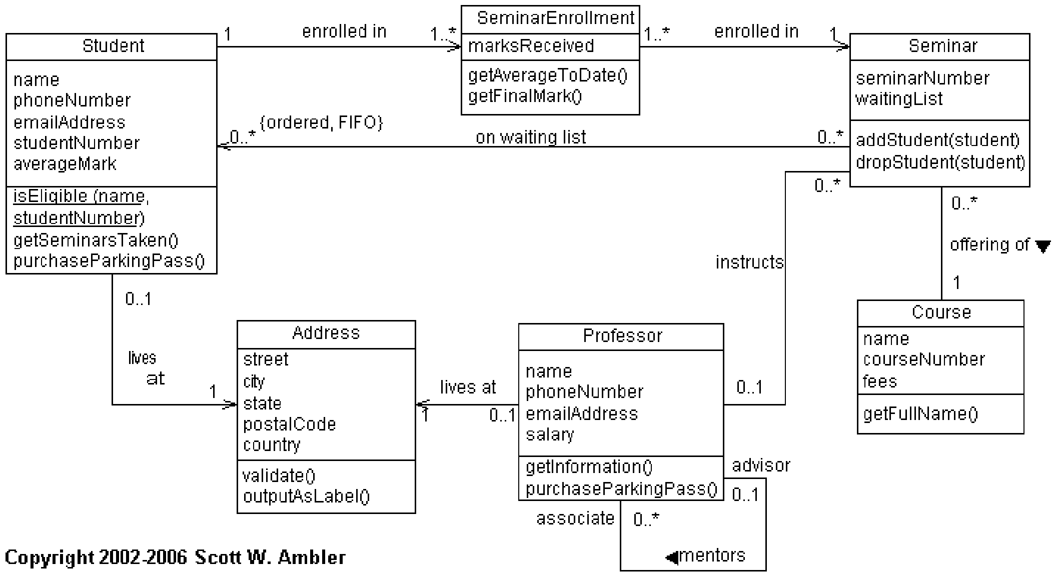


Figure 3.14: Sample Class Diagram [8]

number, marks received and etc. and also include methods separated for each class. For example in this Figure for professor class; name, phone number, email address and salary are defined attributes and "get Information", "purchase Parking Pass" are defined methods for professor class.

3.4.2 OMAVE model Class Diagrams

OMAVE class diagram is illustrated in Figure 3.15. In this model, VE entities are divided into different classes based on their properties and functionalities. The classification is based on the production of high value added product idea. According to this target two types of companies has been mentioned in the model; Manufacturing Companies and Research Institutes [8]. To form up a VE consortium, it is required to find out the required capabilities and capacities in VBE (Virtual Breeding Environment). In order to realize the capabilities of manufacturing companies their resources should be evaluated. Enterprises' resources like machine tools, software, human resources and etc. is evaluated [8] [101]. These resources enable enterprises to realize different manufacturing, research and testing processes. Making right decisions for selecting appropriate companies to participate in a special project needs exact information about the companies capabilities and capacities and this is only possible by inspecting their inventory and equipment. Based on this idea, resources class is divided into three main groups; Human resources (people), Software and Equipment. Each of these categories may acquire special properties which are illustrated as attributes. Furthermore, these equipment have several types of attributes and properties that distinguish them from each other. In most of the times these attributes are critical for special production or research processes. Therefore, it is obvious that, it is needed to acquire and save these attributes, their specifications and their units in the system. For this purpose, attributes class also is established in OMAVE model.

As it was mentioned before, one of the main objectives of this virtual enterprise platform is to empower the collaboration between research institutes and R&D companies activating in different techno-parks with manufacturing companies operating in industrial parks. The reason for increasing such collaboration is to transfer researches results to industry and enable producing high value added technological goods. Both sides (manufacturing companies and research institutes) will benefit from participating in this network. There will be two types of companies in VBE, and each type has its own requirements. In Figure 3.16, manufacturing companies, research institutes and their relations is presented.

In order to discover the most appropriate partners for the forthcoming projects and form up a reliable VE consortium it is necessary to have exact information about the members (VBE members) abilities and capabilities. As it was described above enterprises are related to their capabilities through their resources. Figure 3.17, demonstrates the relations between capabilities, manufacturing companies, resources and resources' attributes.

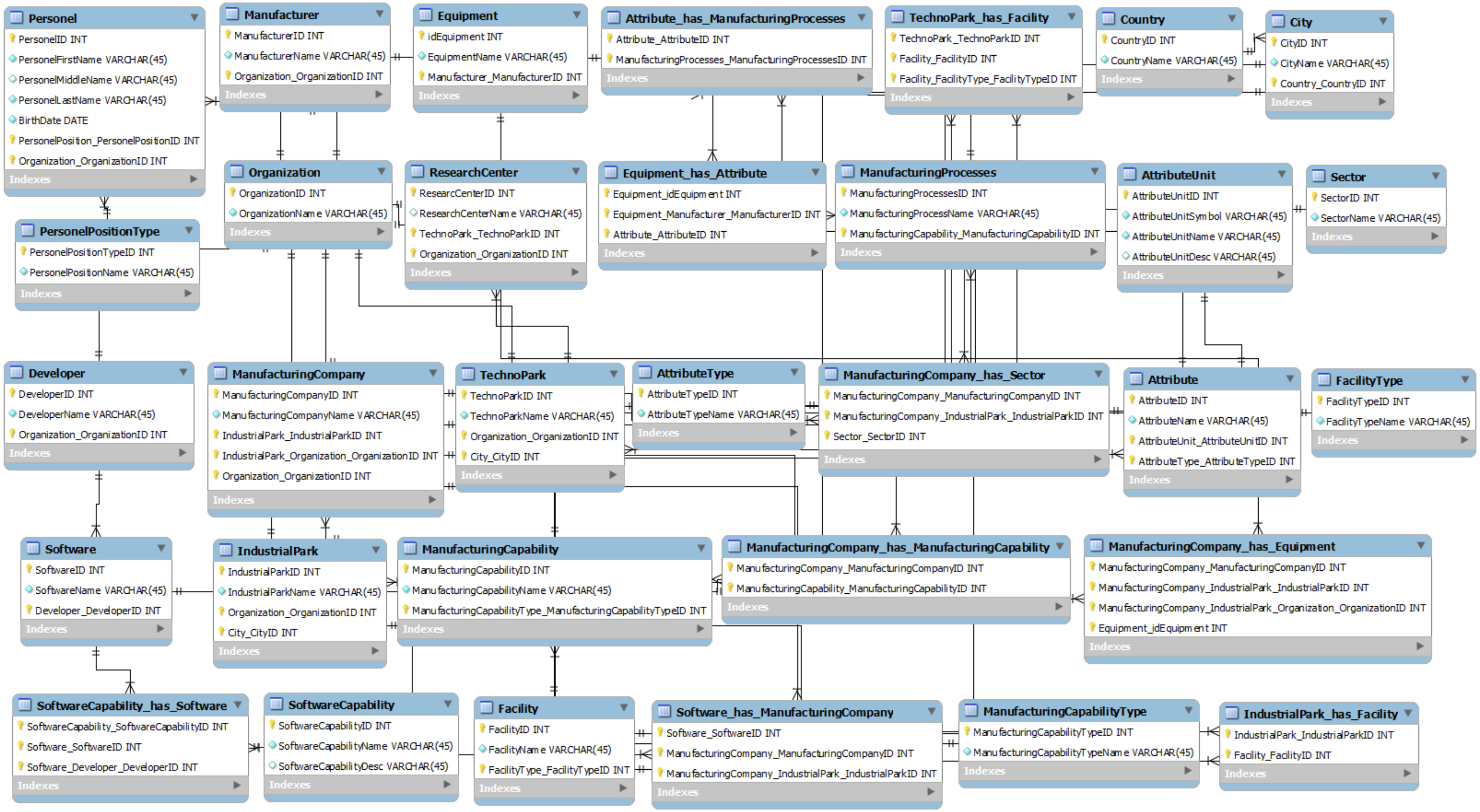


Figure 3.15: VE System Entity-RelationShip Diagram

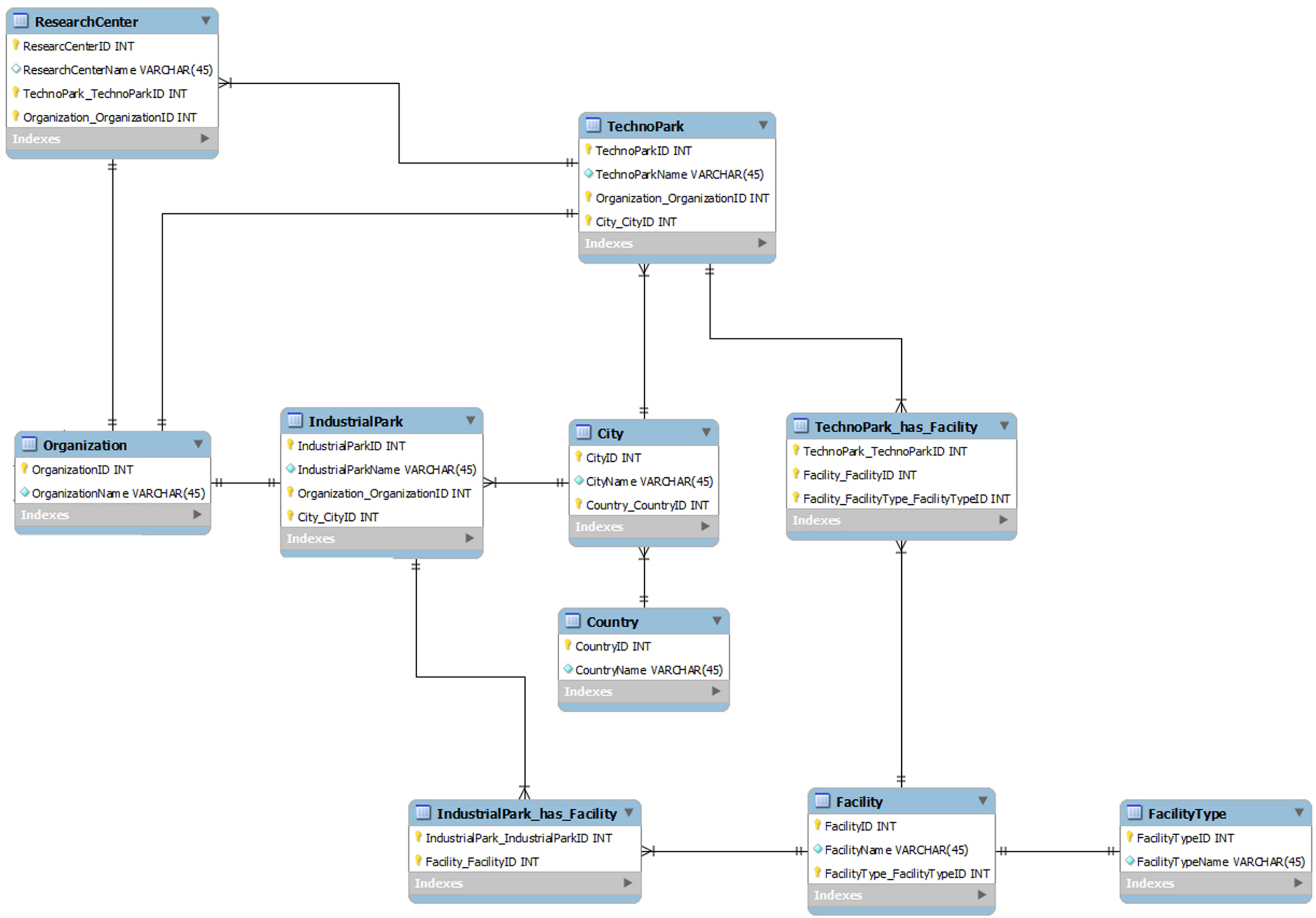


Figure 3.16: Entity-Relation Ship Diagram- Manufacturing and Research Company

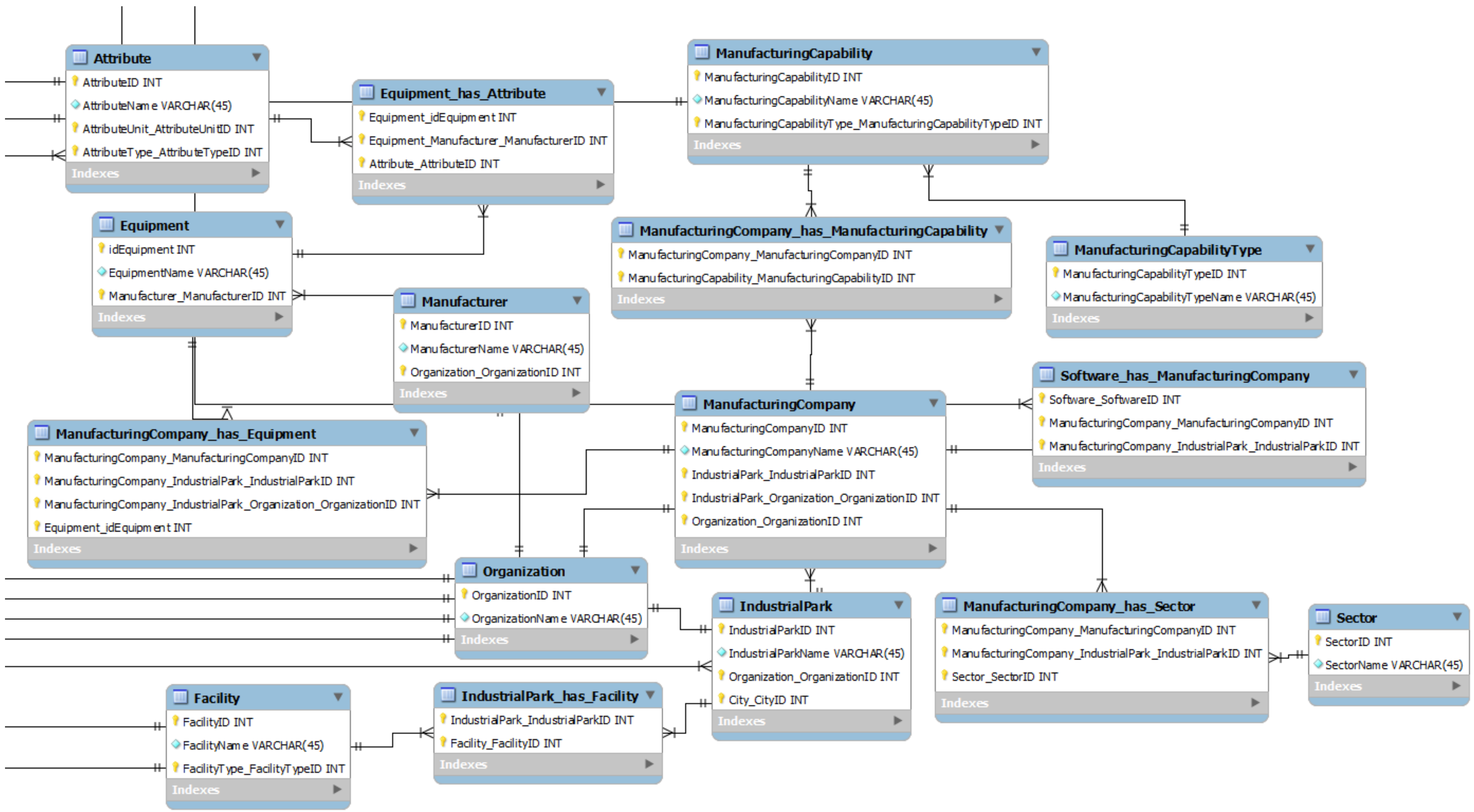


Figure 3.17: Relations between Manufacturing companies and their capabilities

3.4.3 UML Use Case Diagrams

A use-case illustrates a unit of functionality provided by the system. The main purpose of the use-case diagram is to help developer teams, visualize the functional requirements of a system, including the relationship between "actors" (human beings who will interact with the system) and essential processes, as well as the relationships between different use cases. Use case diagrams generally show groups of use cases — either all use cases for the complete system, or a breakout of a particular group of use cases with related functionality (e.g., all security administration-related use cases).

Here, in use case diagrams, domains are shown by drawing ovals as actions inside a rectangular frame. The name of domain is written in the corner of rectangle and the name of action is written inside or under the oval. In order to draw an actor as a system user, drawing a stick people is enough. These notations are placed in left or right side of the domain rectangle. A sample use case diagram is shown in the Figure 3.18. A use-case diagram is typically used to communicate the high-level functions of the system and the system's scope.

3.4.4 OMAVE Model Use Case Diagrams

Considering the characteristics of the use case diagrams and their importance in system modeling and analyzing system from actors (users) perspective, it is necessary to develop related use case diagrams for OMAVE system as well. To do so, use-case diagrams of dashboards, tools and applications for OMAVE system and OMAVE VBE are designed and developed. First, diagram VBE submission process diagram as it is depicted in Figure 3.19.

Next step is to design VE administrator related activities and their related use case diagrams. As it is quite clear from VE and VBE structure the most authorized, involved and key actor in virtual enterprise platform is VE administrator. Here, VE administrator activities use case diagram is illustrated in Figure 3.20.

Enterprises can act in four different areas in OMAVE system. VBE dashboard activities are open for registered enterprises in VBE, thus registered enterprises in VBE are able to get benefited from these applications and tools. Secondly, if enterprises are selected to a VE consortium they will be able to use both VBE and VE project dashboards. Third condition is independent from VE project selection. Companies may follow and monitor their environmental performances by refereeing to the green dashboard designed for VE platform. Electrical energy usage, water and solid wastes, their probable treatment facilities conditions, treated waste amount and also carbon footprint per product, income and etc. for every company is monitored and represented through VE green dashboard. The last use case alternative is independent from VE platform. Third parties provided special tools and applications are places on system interface to just to ease enterprise communication by only using a single interface (here VE platform). Use case diagram regarding enterprises is illustrated in Figure 3.21.

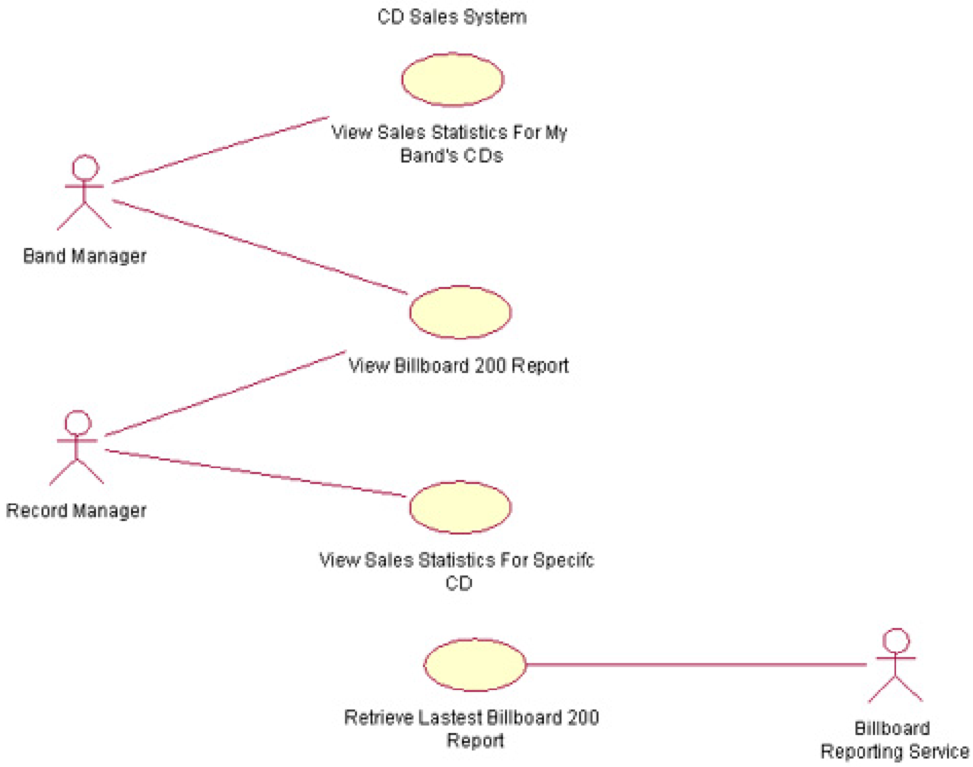


Figure 3.18: Use Case Diagram Sample [8]

3.4.5 UML Activity Diagrams

Activity diagrams show the procedural flow of control between objects during an activity process. In order to model higher level business processes or low level internal actions activity diagrams are used. Activity diagrams are best used to model higher-level processes, to assess a business process in an enterprise. These diagrams are less technical and business minded people may understand these diagrams more quickly. Activity diagram starts with a solid circle connected to the initial activity. The activity is modeled by drawing a rectangle with rounded edges, enclosing the activity's name. Activities can be connected to other activities through transition lines, or to decision points that connect to different activities guarded by conditions of the decision point. Activities that terminate the modeled process are connected to a termination point (just as in a state chart diagram). Optionally, the activities can be grouped into swim lanes, which are used to indicate the object that actually performs the activity, as shown in Figure 3.22.

3.4.6 OMAVE Model Activity Diagrams

There are different simple and complex activities in VE system, from VBE submission activities to the different operational procedures. For all these activities in VE, it is required to design and develop related activity diagrams. In this section virtual enterprises activity diagrams are described. As it was mentioned before, in order to enter a VE project the first

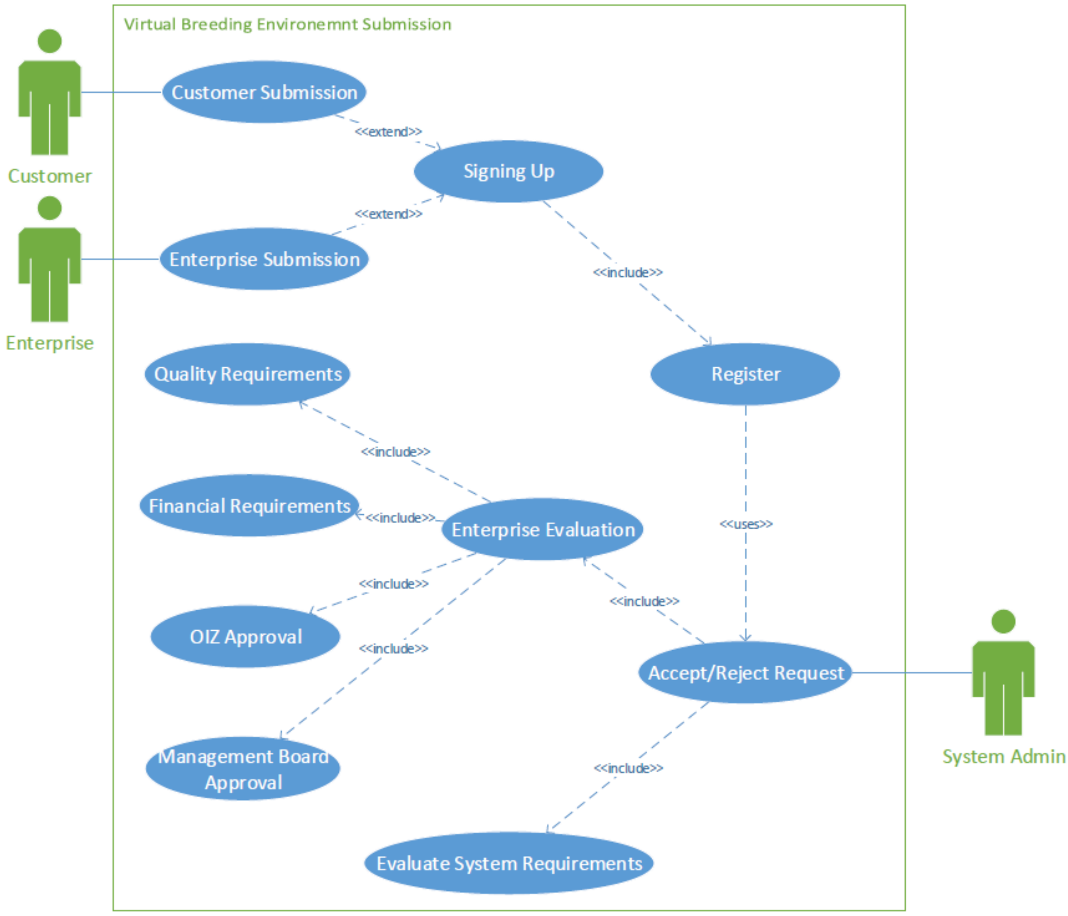


Figure 3.19: Virtual Breeding Environment Submission Use Case Diagram

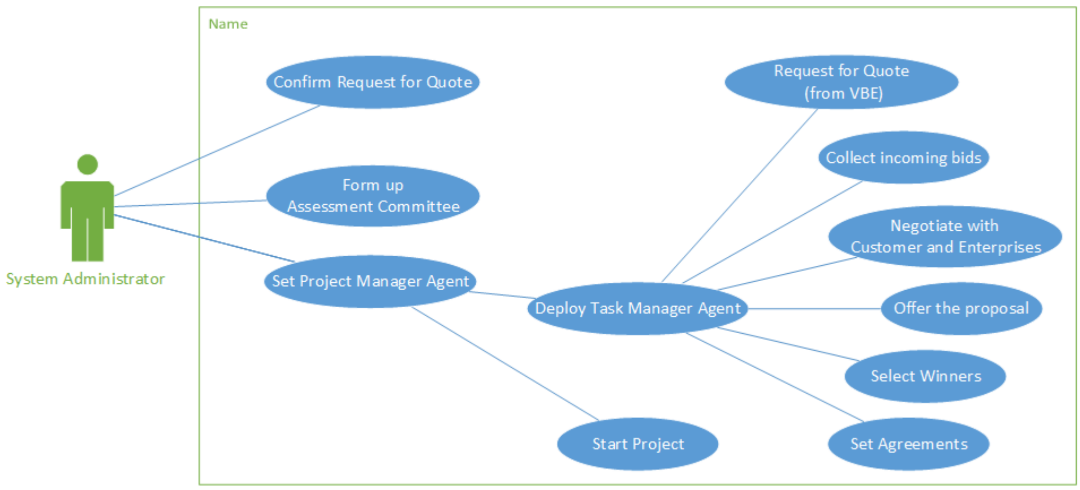


Figure 3.20: VE administrator Use Case Diagram

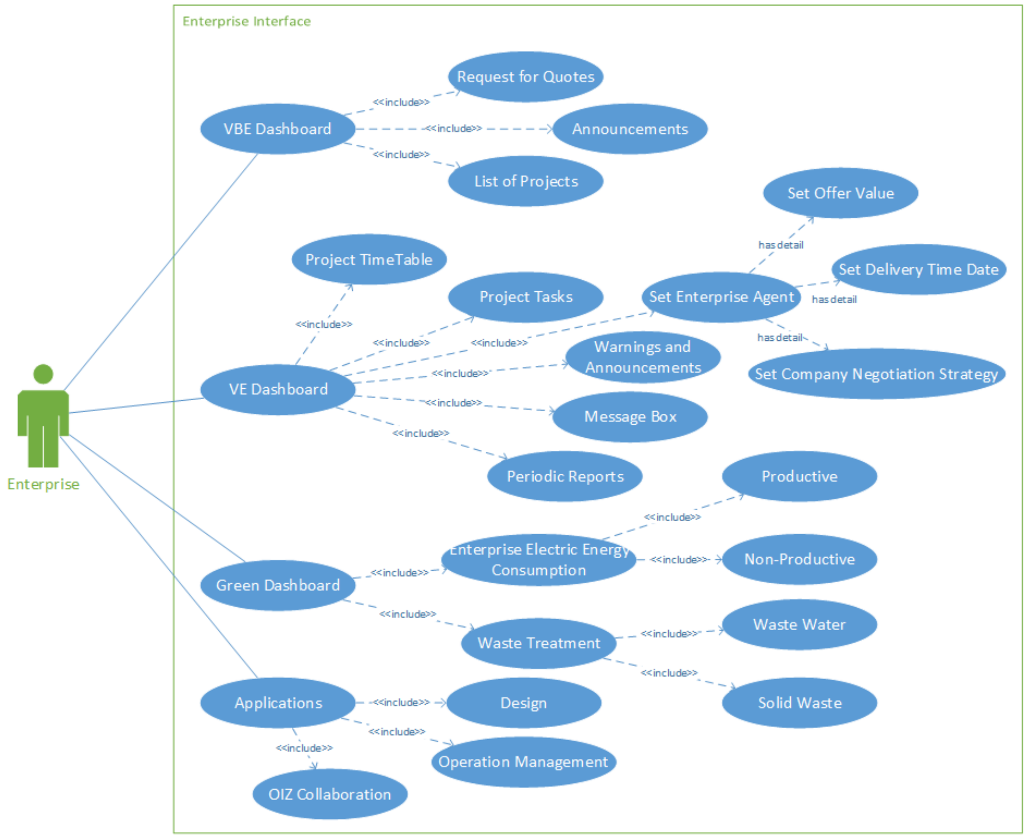


Figure 3.21: VE enterprises Use Case Diagram

necessity is VBE membership. Therefore, all volunteer enterprises and customers have to be registered in VBE pool. This is considered as the first activity in VE platform for all enterprises and customers. In Figure 3.23, the VBE submission activity diagram is illustrated.

This diagram is consisting of three parallel vertical columns which are representing the activity domains of the related system user(s) or administrator. The domain name is written on top of the column. Here, the left domain is customer domain. Right hand side domain is enterprise domain while the middle one is VE administrator domain. This activity is triggered by customer or enterprise. First of all, a request for registration is submitted by enterprise or customer. This request is evaluated by VE administrator, complementary documents and certificates are asked to complete the registration evaluation procedure. The registration request along with the necessary documents are sent to the VBE management board. Registration request is evaluated by management board and rejection/approval message is sent back to the VE administrator and customer/ enterprise regarding their request. This activity ends after sending the reject or approval message by VBE management board. This is a very simple activity diagram in OMAVE platform. A virtual enterprise project is started by customers request for quote. Request for quote is followed by a proposal by VE administrator to the customer and submission of an order by customer. All these processes and activities are shown in detail in the related diagram which is called project submission VE diagram (Figure 3.24).

As it is clear from Figure 3.24, after a quote submission by customer, VE administrator,

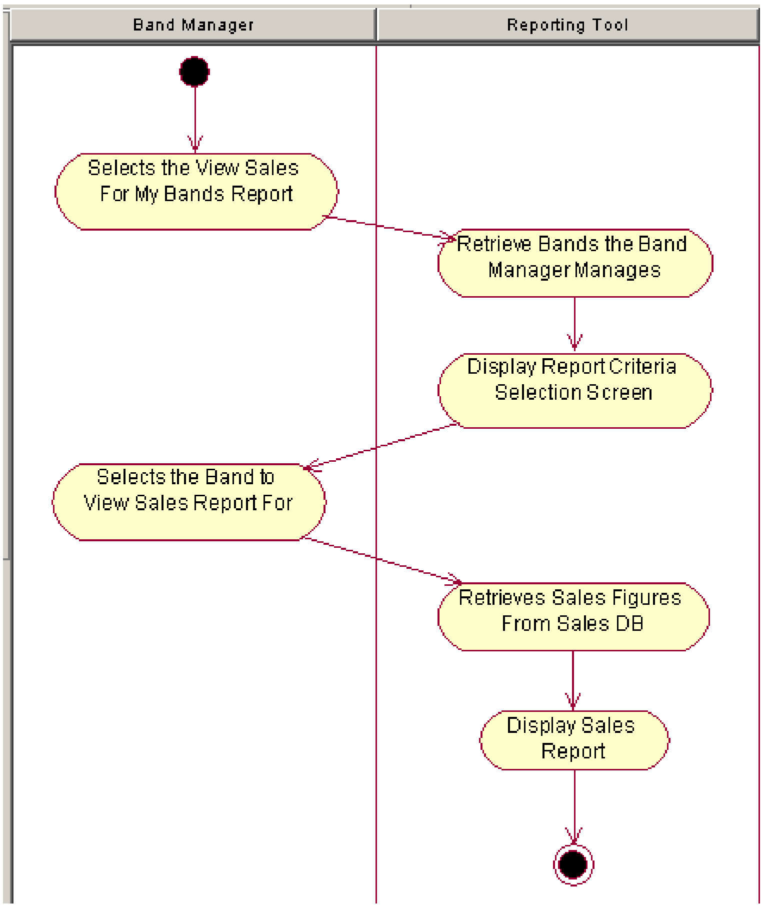


Figure 3.22: Sample Activity Diagram [8]

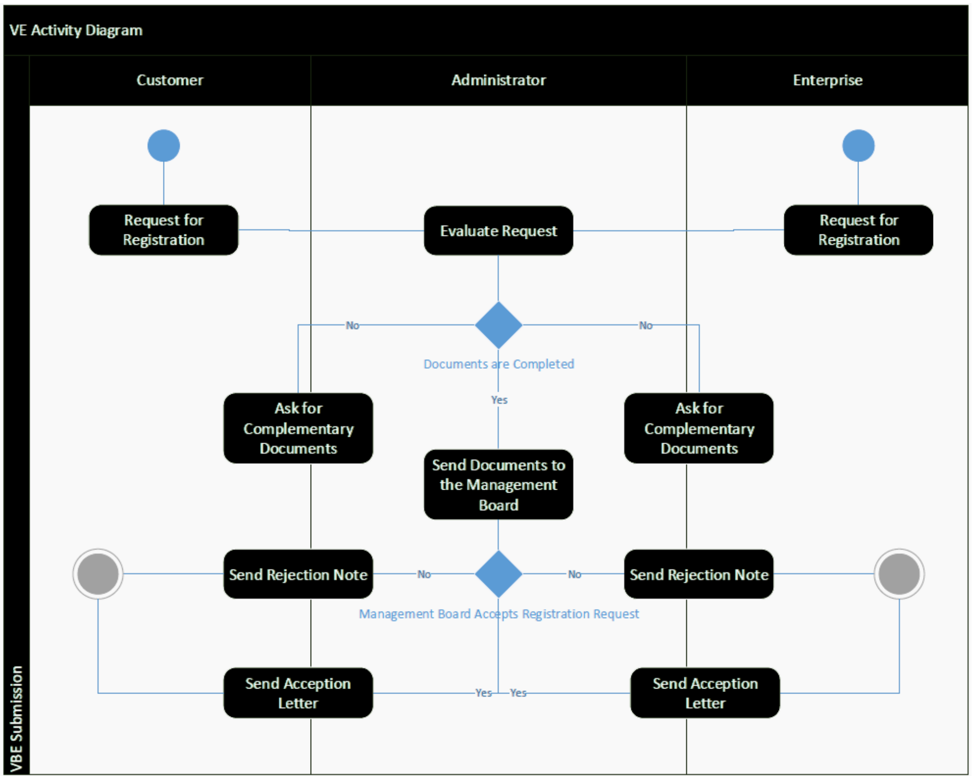


Figure 3.23: VBE Submission Activity Diagram

relevant sector leaders, enterprises and VE management board evaluate the incoming request and accept or reject the request. In the case of accepting the request, new bids are requested from VBE. Customer is given a proposal by VE administrator which is the collection of the coming bids from enterprises. In order to reach the final agreement, a negotiation process is predicted for this step. In order to finalize agreement preparation steps, and also create a product perspective, enterprises, customer and VE administrator have to start to develop a product concept design and get the customer approval over the concept design. Finalizing this step, it is time to call for bids to VBE to form up the final VE project group. After selecting the winners from the last call, final agreement between VE partners (bidding process winners), VE administrator, also between customer and VE administrator is signed.

Agreement preparation activity is the following activity in VE processes hierarchy. In order to arrange the final agreement, VE administrator asks customer and enterprises views. The agreement draft is prepared by VE administrator. A judiciary committee also supervises the agreement preparation processes and evaluates the agreement terms with the regulations and laws, and interferes in agreement preparation processes if necessary.

Final confirmation from all sides is needed before preparing the final version of agreement. After confirming the agreement terms by enterprises, customer, VE administrator and Judiciary committee, final agreement is signed by them and is submitted officially. In this research, it is assumed that all the collaborative design phase is based on Dassault Systems collaborative design and PLM environment called CATIA V6 (ENOVIA V6 PLM tool is working seamlessly in background). Therefore, no activity diagram is developed for this phase intentionally and directly it is skipped to the operation phase of VE.

Prior to the operational phase of VE (as this system is designed to be multi-agent based system) it is necessary to assign agents for managing different activities in OMAVE system. Agent registration process activity diagram is depicted in Figure 3.26. Here, two types of agents; work flow manager agent (WFMA) and penalty and rewards manager agent (PRMA) are illustrated. In this part, OMAVE administrator agent is calling work flow manager agent for assigning a new project. Required documents in standardized VE format is addressed to the WFMA and PRMA then these agents arrange new settings and configurations according to the project documents to manage the new project. Considering new settings, project program and scheduling is sent to the partner enterprises and their confirmation are inquired. In order to monitor partners' performances, project work flow manager agent gathers information about project tasks from partners and control them with project master scheduling. Collecting inforamtion from enterprises is realized by communication between work flow manager agent and enterprise operation agent. Reporting action is done periodically. Related reporting activity diagram is shown in two separated activity diagram (3.27, 3.29).

Project task is monitored and managed by work flow manager agent. After getting permission to start the project, work flow manager agent assigns tasks to partner enterprises and sends them, VE formatted project documents including designs, process planning, and scheduling. According to the project scheduling and predefined reporting dates, work flow manager agent

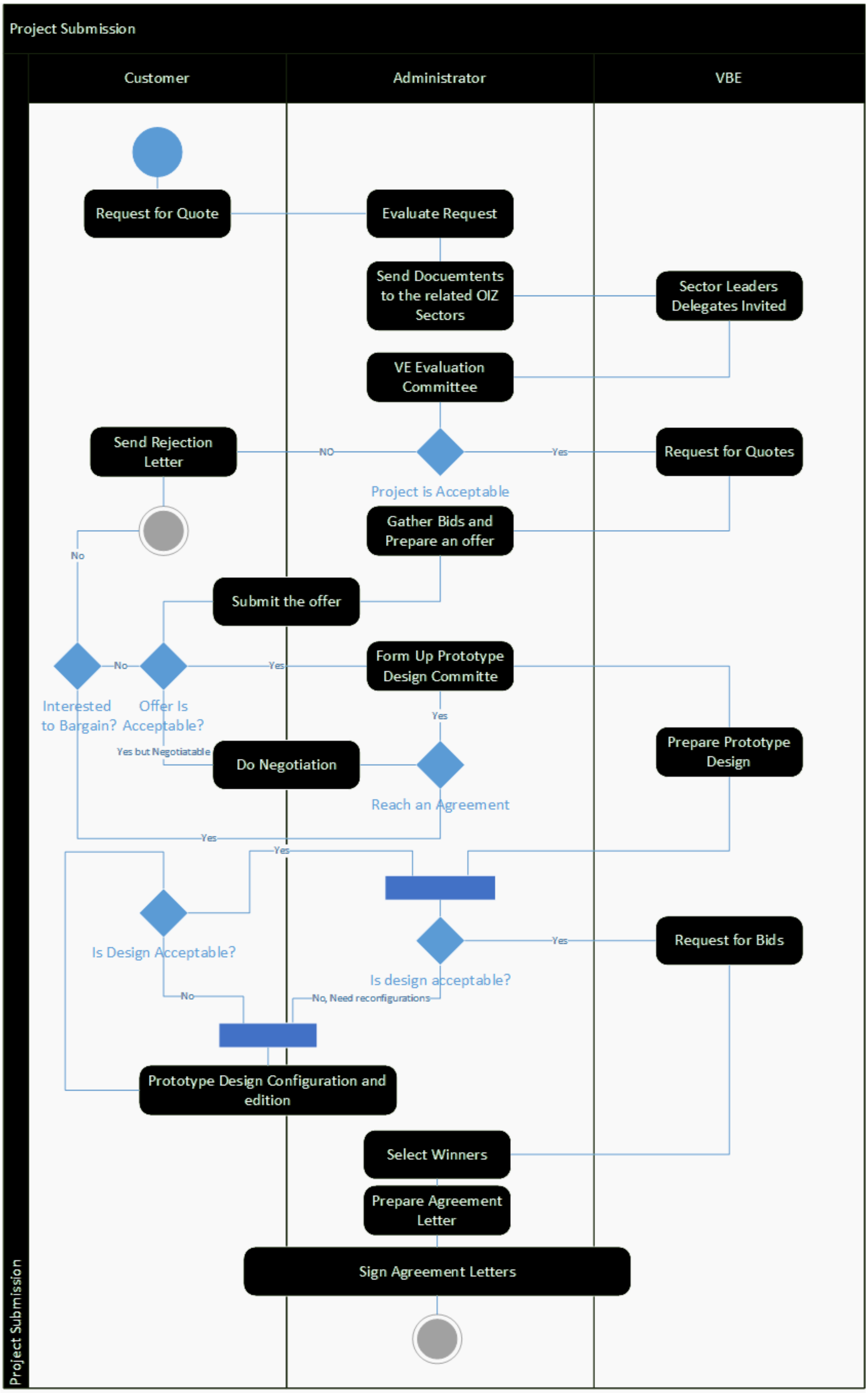


Figure 3.24: Project submission activity diagram

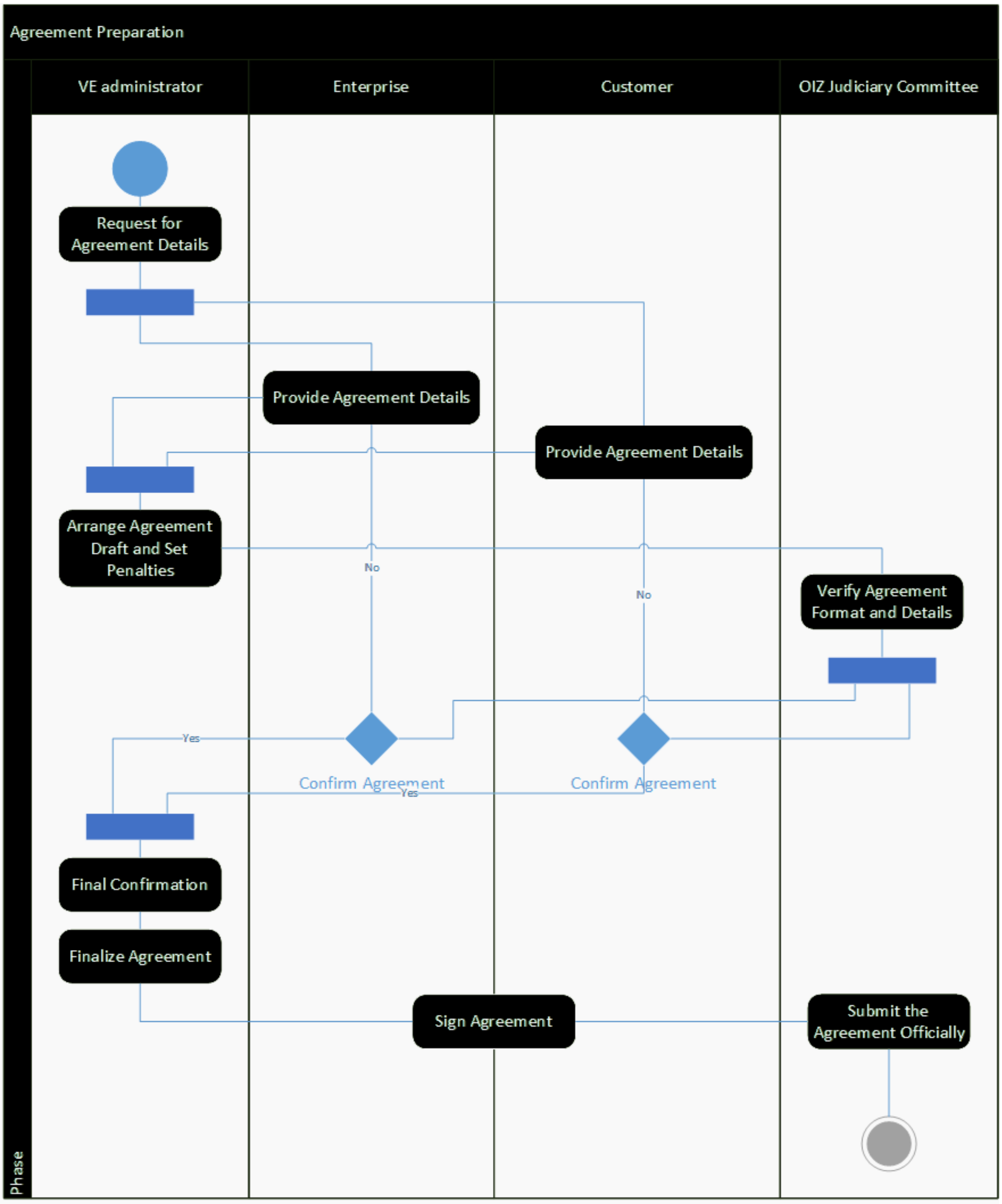


Figure 3.25: Agreement preparation activity diagram

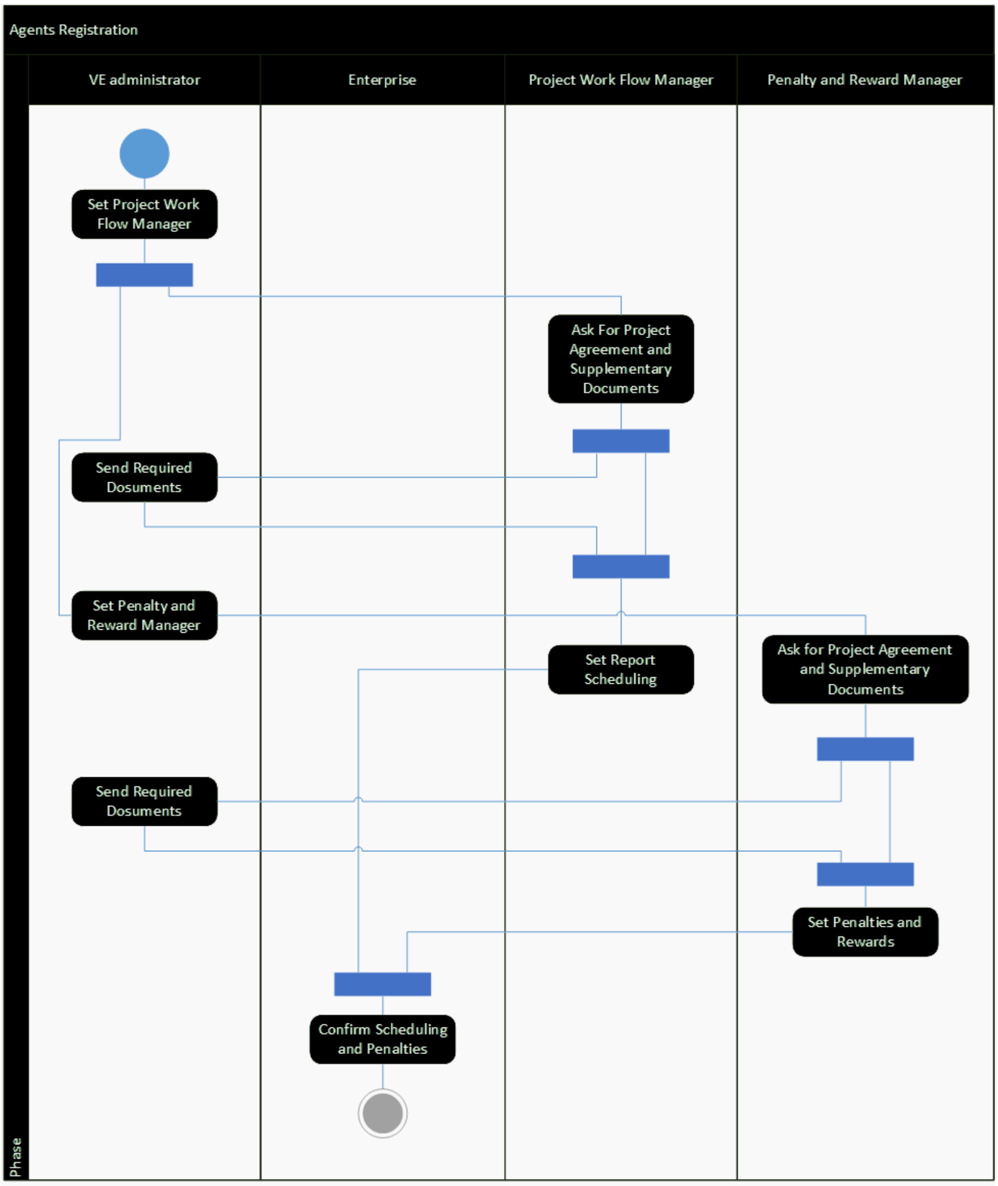


Figure 3.26: Agent registration activity diagram

periodically asks for report submission. Enterprises are ought to submit the report in the requested dates. Enterprises may encounter various problems and difficulties to perform their jobs. All types of problems and difficulties should be reported by enterprises. Project manager and VE administrator should be aware of all problems in project progression. Enterprise submits the report, this report is evaluated and archived by work flow manager. Some little corrections regarding report format or report details could be asked by work flow manager but after modifying report by responsible enterprise report would be recorded by work flow manager.

The most important and crucial part of reporting is request for extension by enterprise. Request for extension is assessed by work flow manager and OMAVE administrator. This request is replied based on project timetable, master project scheduling, responsible enterprise work load and conditions, performance of other partners in the project and the effect of this extension on other activities and project overall performance. Based on these criteria different possibilities and situations are predictable. In order to make the best possible decision for continuation of the project, the following algorithm is developed. This algorithm is illustrated in Figure 3.27. From accepting extension request without giving any warning, to excluding enterprise from VE project and penalizing the company and redistribute task to other potential enterprises in VBE, there are different choices and possibilities in front of work flow manager and OMAVE administrator. As shown in the Figure 3.33, penalizing procedure is under the responsibility of penalty and reward agent, and this agent penalizes the faulty enterprise according to the project agreement and declares penalty to the enterprise and OMAVE administrator. Figures 3.27 and 3.28 are to be enlarged for better understanding of the activities during the report submission process. Figures 3.29, 3.30, 3.31, 3.33 and 3.34 are detailed versions of Figure 3.28. Enterprise exclusion, call for potential alternate enterprises and task project accomplishment are too complicated tasks to be shown in just one diagram therefore in order to visualize better the activity diagram of these procedures it is necessary to show different activities in separated diagrams.

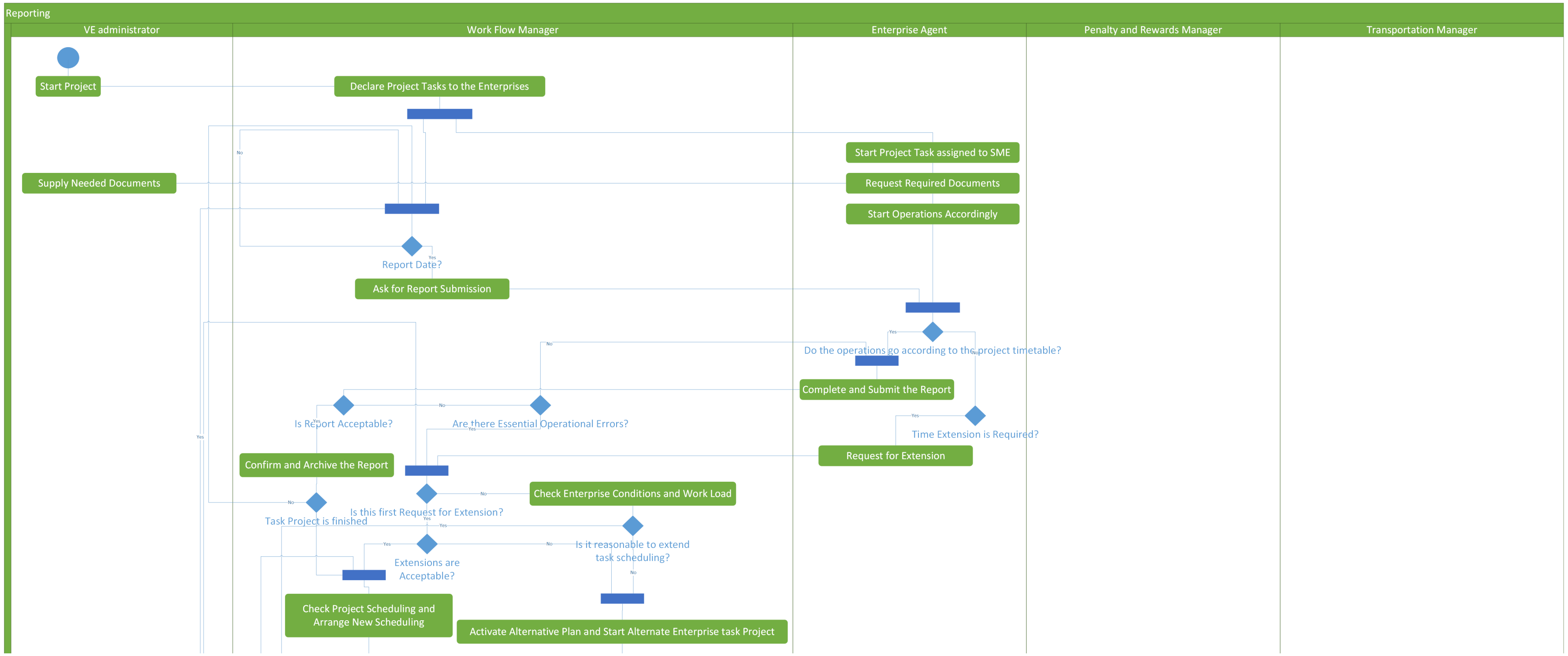


Figure 3.27: Reporting activity diagram- Expanded

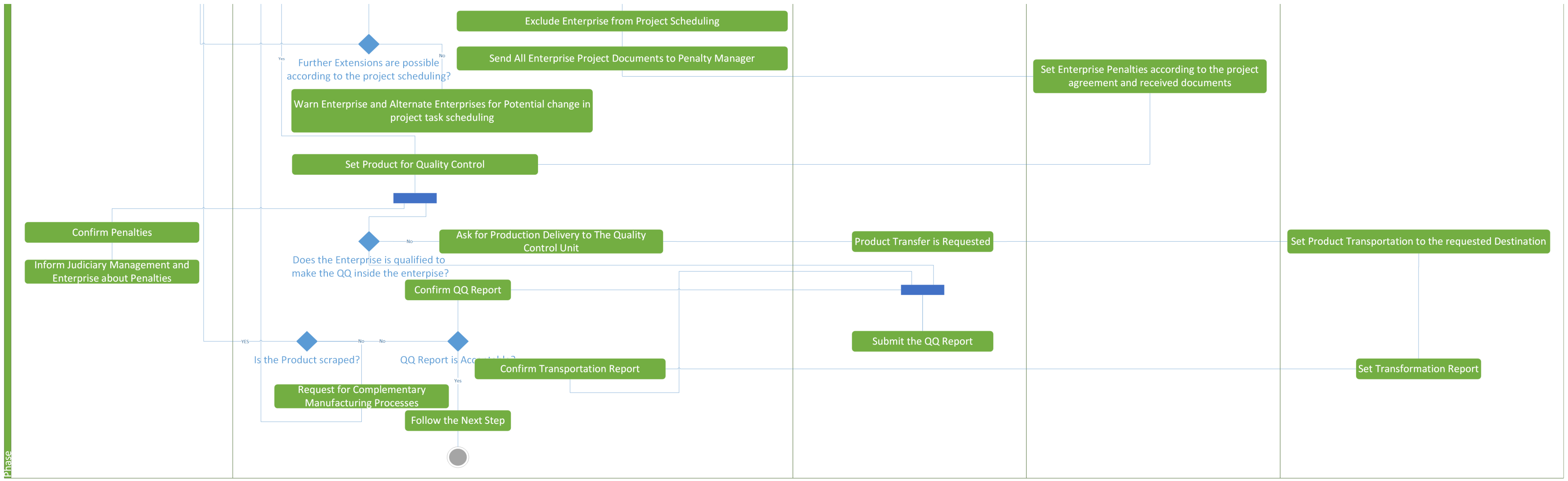


Figure 3.28: Reporting activity diagram- Expanded (Cont'd)

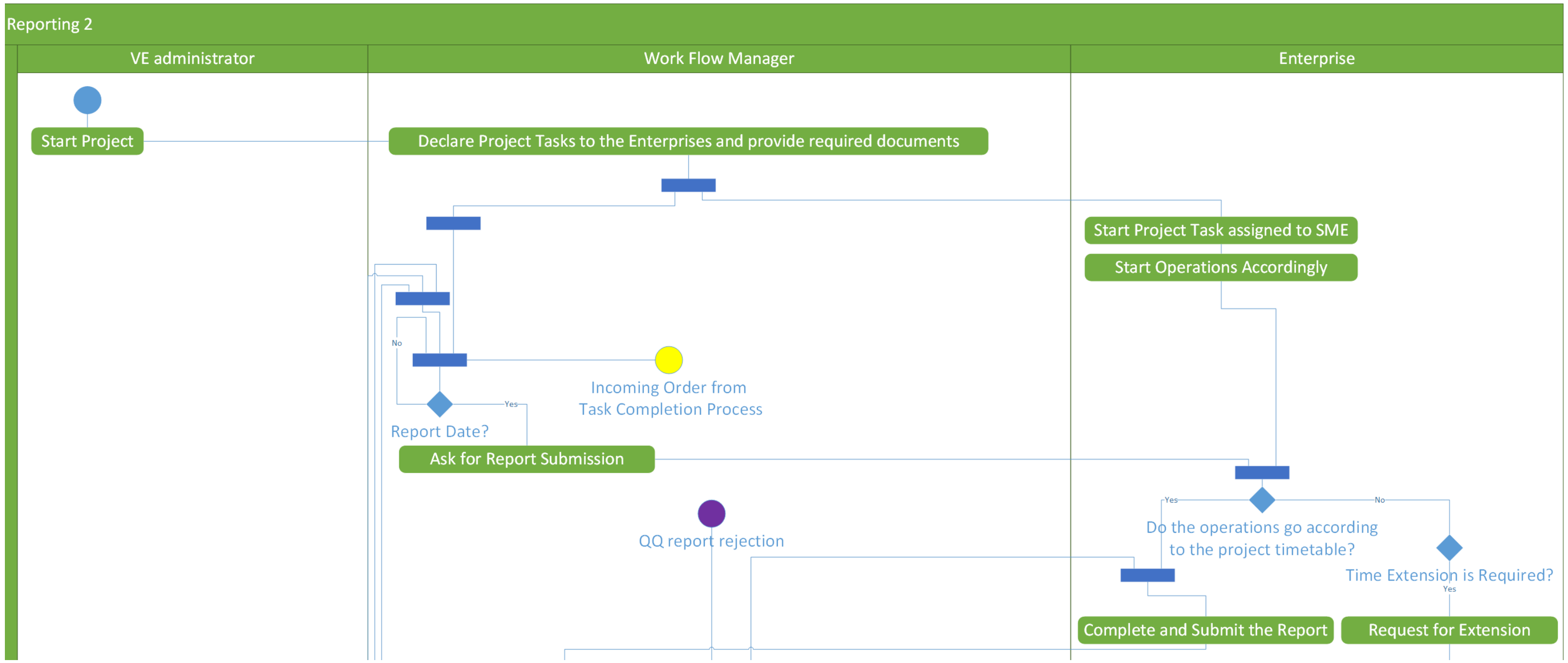


Figure 3.29: Reporting activity diagram- Enclosed

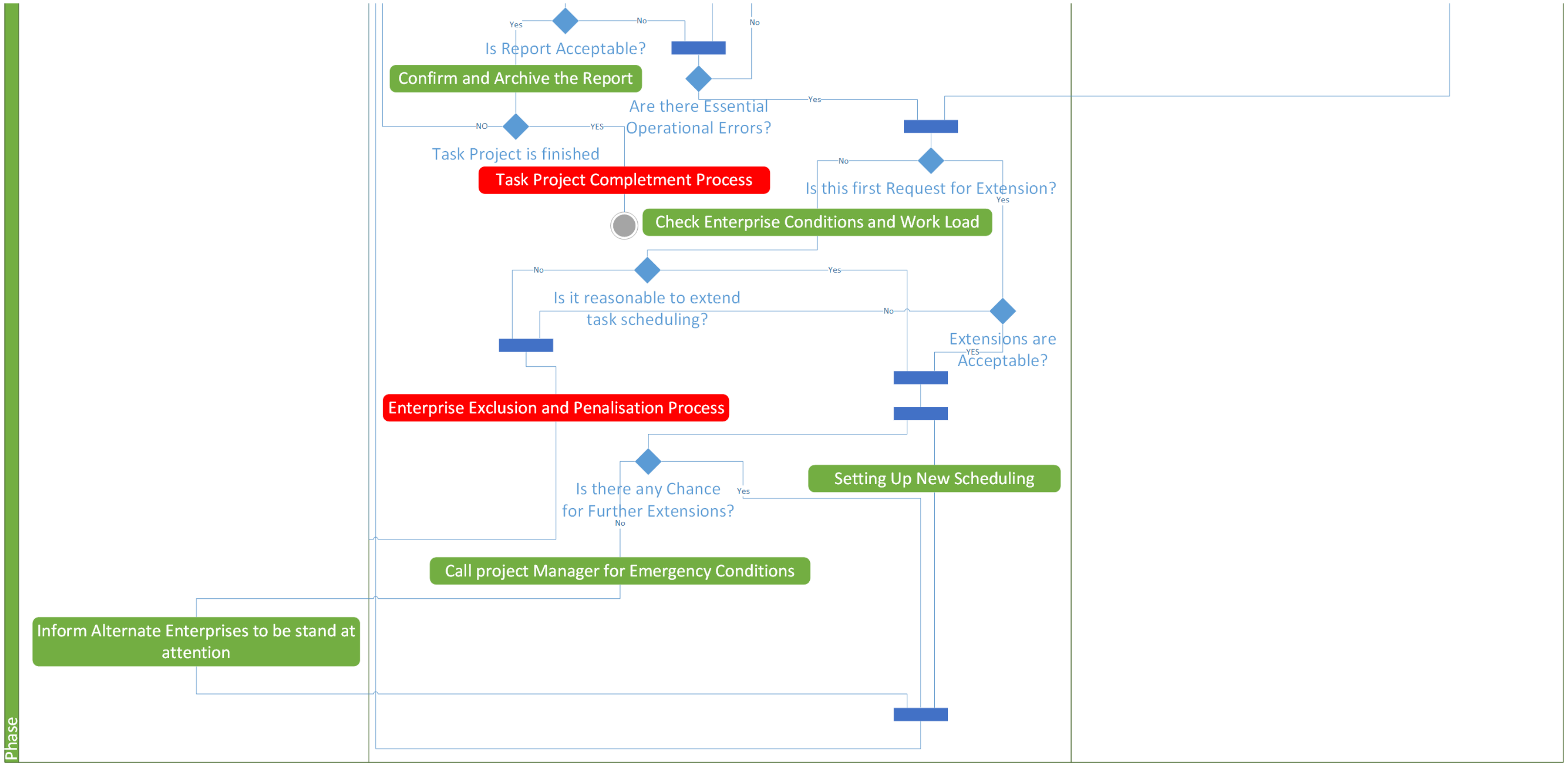


Figure 3.30: Reporting activity diagram- Enclosed (Cont'd)

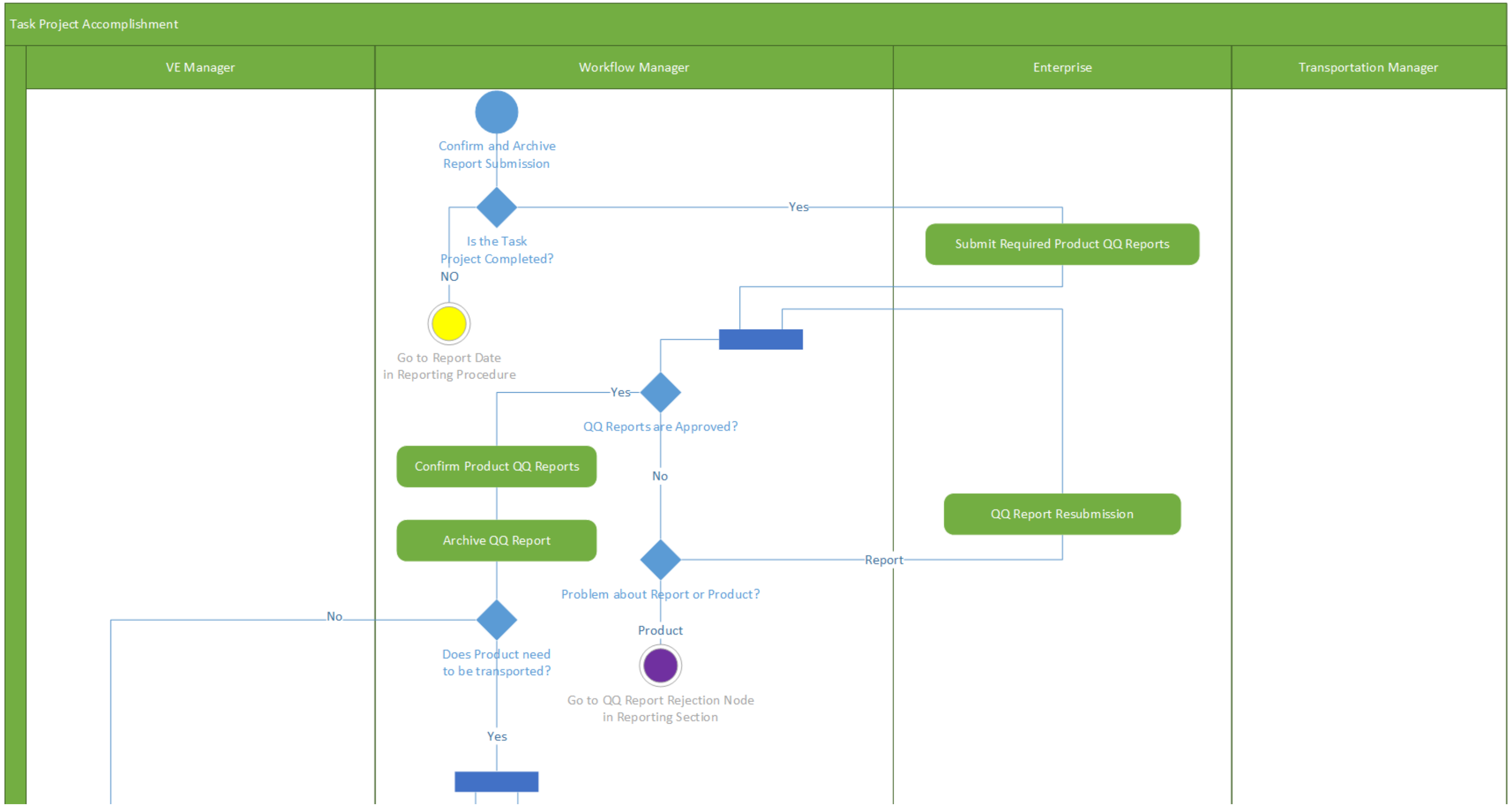


Figure 3.31: Task accomplishment activity diagram

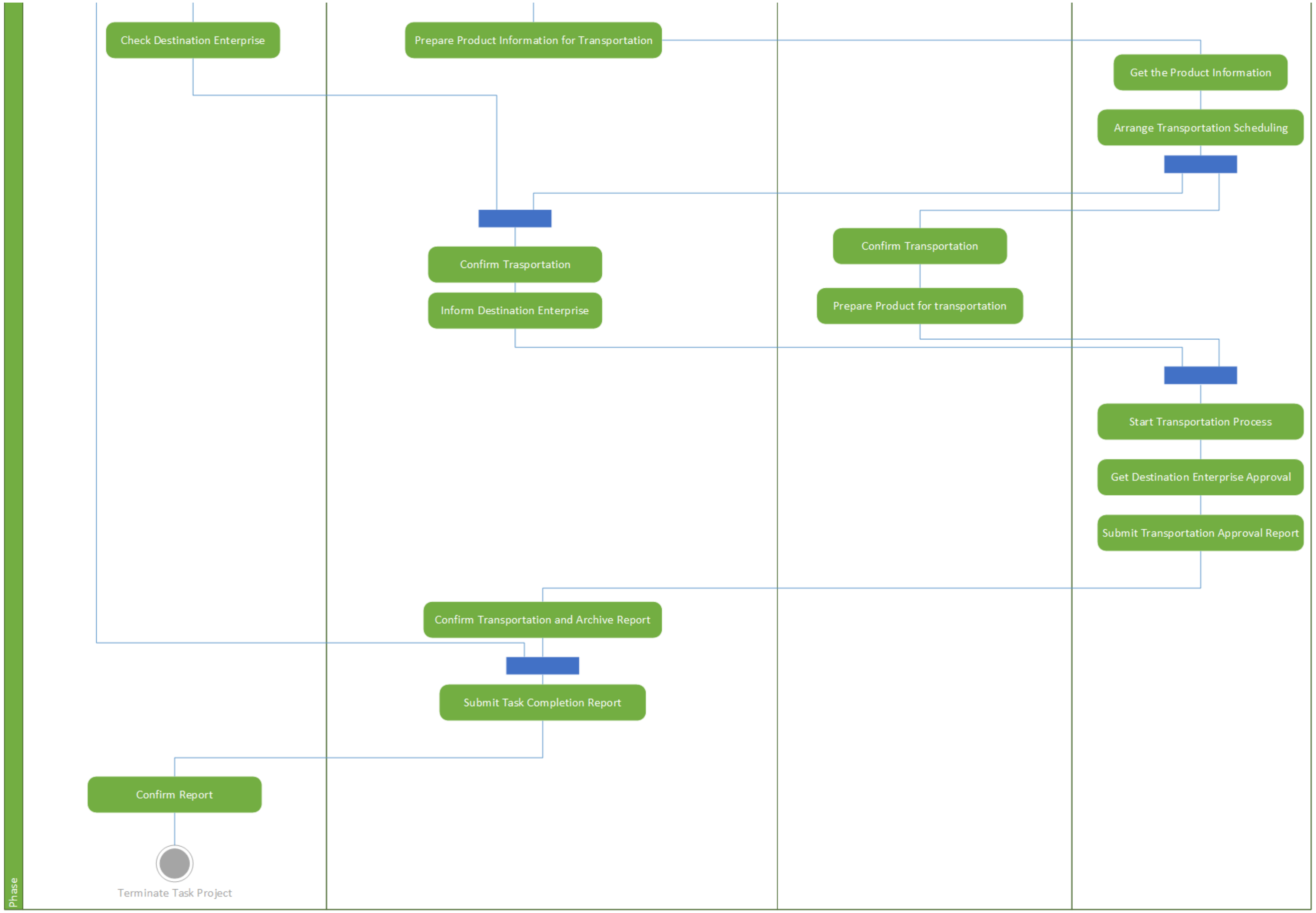


Figure 3.32: Task accomplishment activity diagram (Cont'd)

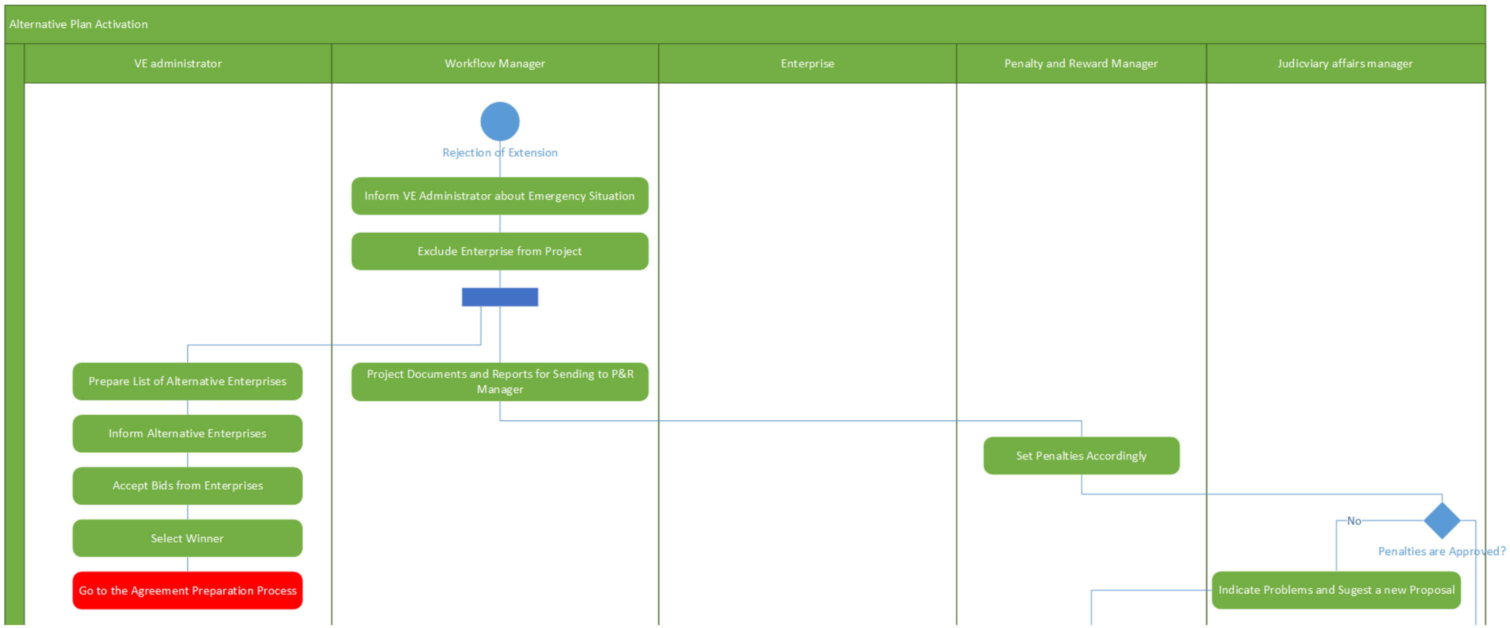


Figure 3.33: Enterprise exclusion and penalizing processes activity diagram

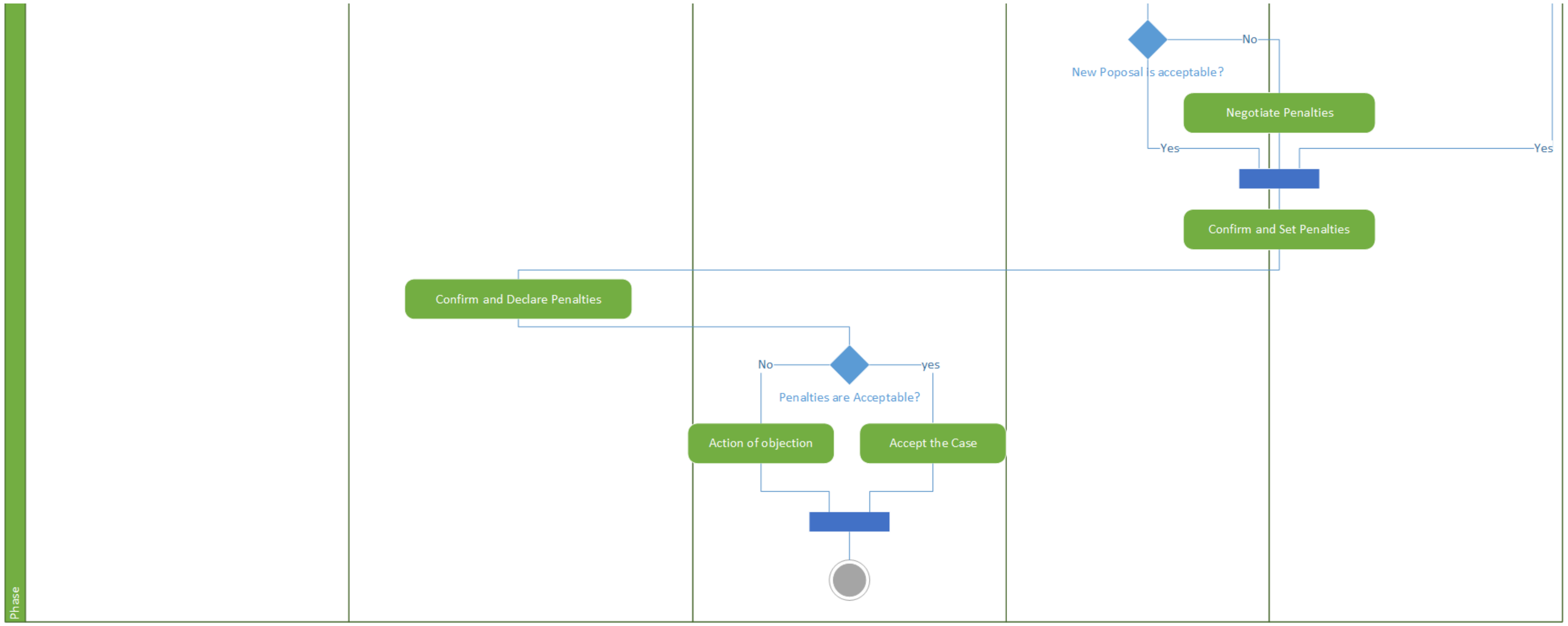


Figure 3.34: Enterprise exclusion and penalizing processes activity diagram (Cont’d)

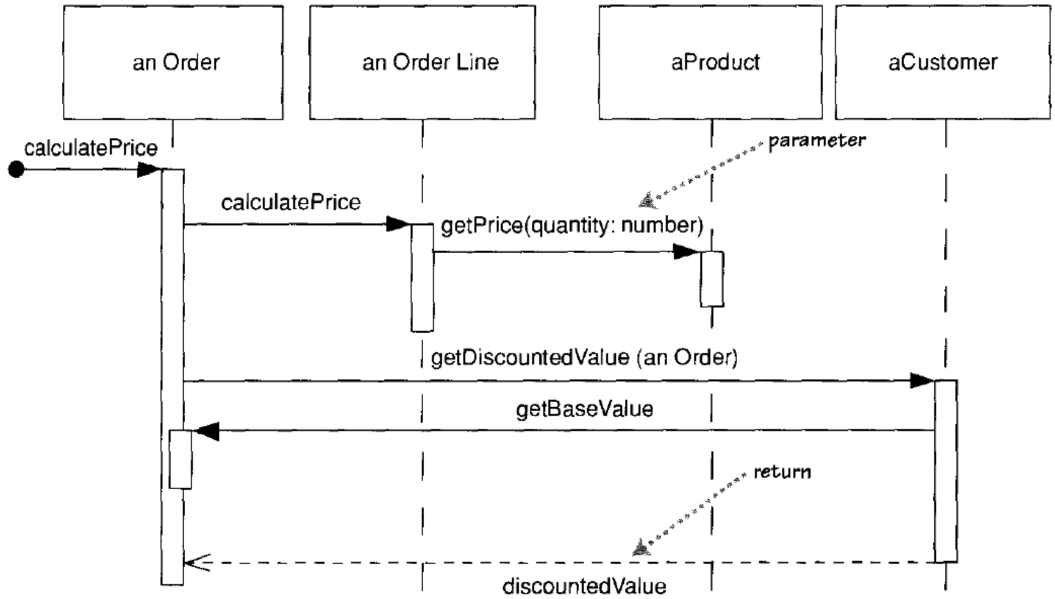


Figure 3.35: Sample Sequence Diagram

3.4.7 UML Sequence Diagrams

Sequence diagrams, also called event, timing diagrams or event scenarios do show object interactions chronically for a specific use case. These diagrams show the details of operations, actions and calls between different objects in their sequence. There are two main streams in sequence diagrams. One of these streams, which is shown in vertical dimension is the sequence of message and actions. Second stream is messages object instances. In other words, a sequence diagram shows, parallel vertical lines (lifelines) of different processes or objects that live simultaneously, and, horizontal arrows, symbolize the messages which are exchanged between these objects, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

The sequence diagram is used primarily to show the interactions among objects in the sequential order that those interactions occur. From different point of views sequence diagrams may have different advantages and contributions for system developers. Sequence diagrams may enlighten an organization's business staff about business processes order and interactions between objects. Besides documenting an organization's current business processes, a business-level sequence diagram can be used as a requirements document to communicate requirements for a future system implementation. During the analysis phase of a project, analysts can take use cases to the next level by providing a more formal level of refinement. When that occurs, use cases are often refined into one or more sequence diagrams [8].

An organization's technical staff may find sequence diagrams useful in documenting how a future system should behave. During the design phase, architects and developers can use sequence diagrams to force out system entities interactions, and enhance system design details

[8]. Sequence diagrams play an important role in the transition from requirements expressed as use cases to the next and more formal level of refinement. Use cases are often refined into one or more sequence diagrams. In addition to their use in designing new systems, sequence diagrams can be used to document how objects in an existing (call it "legacy") system currently interact. This documentation is very useful when transitioning a system to another person or organization [5, 8].

3.4.8 OMAVE Model Sequence Diagrams

In order to develop VE and VBE systems, related sequence diagrams created. One of these sequence diagrams deals with customer's request for quote submission procedure. This diagram shows how a request for quote is submitted by customer and what procedures are passed to evaluate the request and give an appropriate response to customer. If the evaluation result is positive, an offer is prepared and proposed by OMAVE administrator to the customer. Next diagram from this section is VBE registration sequence diagram. As was mentioned before, taking any action in VE or VBE is prohibited without registration in VBE. All volunteer enterprises and customers are required to be registered in VBE for joining the negotiation processes to enter VE projects. Following diagram depicts enterprises' VBE registration procedure. If incoming proposal from OMAVE administrator is acceptable, and customer is eager to submit a product order, order submission process starts. Figure 3.38 Illustrates new product order submission process, which includes the negotiation process between customer, VE administrator, VE management board, clusters management and potential VE partners.

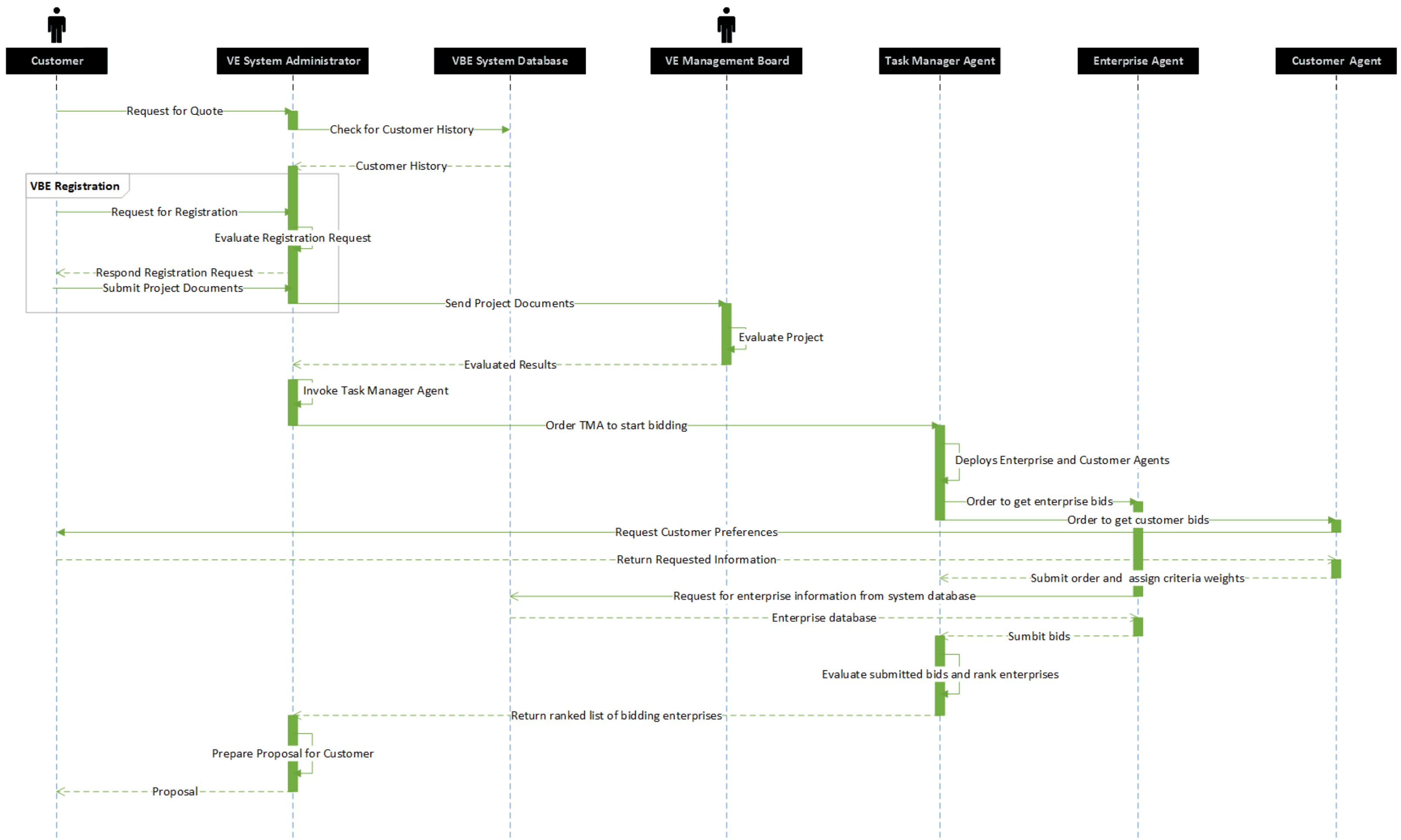


Figure 3.36: Request for quote coming form customer sequence diagram

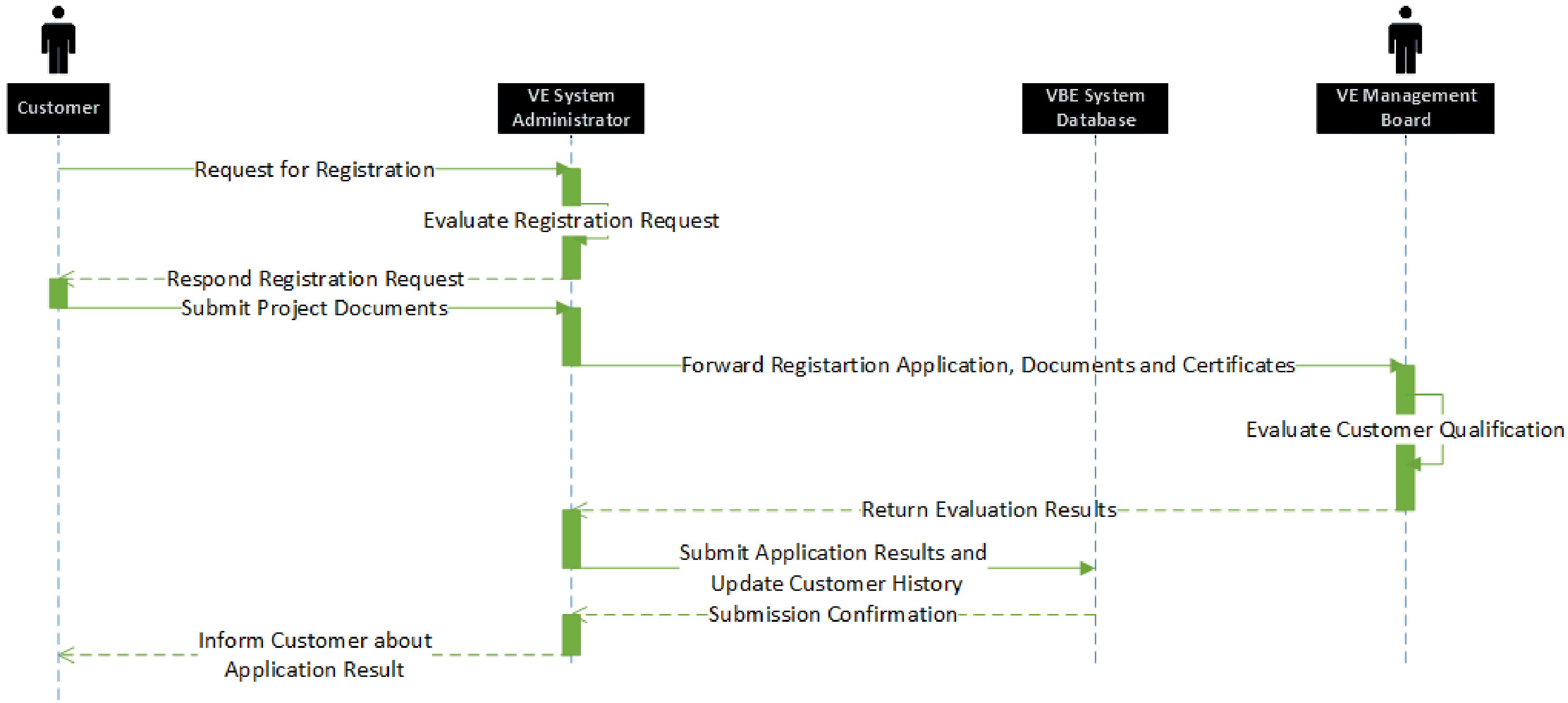


Figure 3.37: Sequence diagram of enterprises registration to VBE

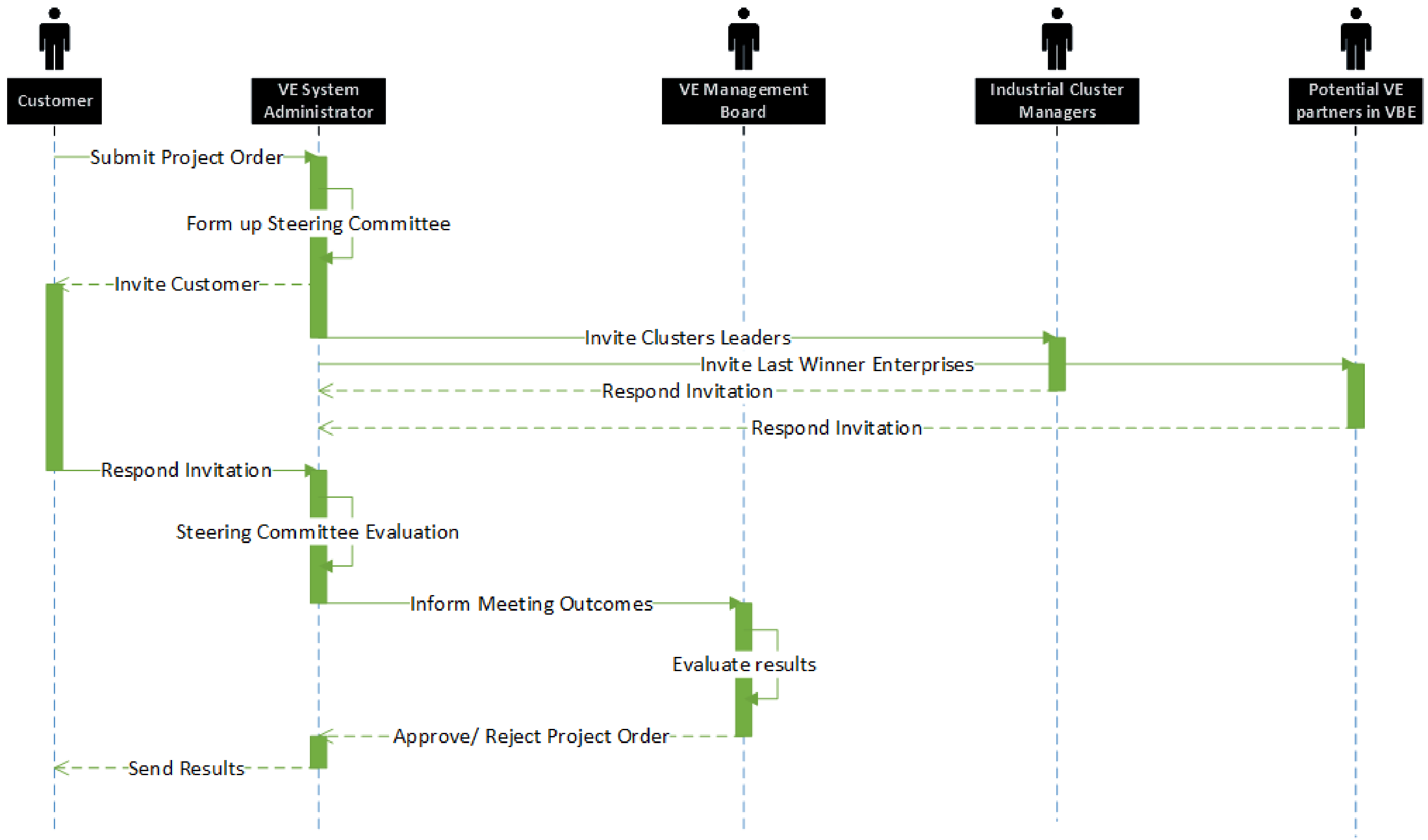


Figure 3.38: Incoming order from customer submission process sequence diagram

Up to here all the sequence diagrams were related to the preliminary stages of OMAVE formation. The most important stage in OMAVE formation phase undoubtedly is partner selection stage. After announcing call for bids for different project tasks from OMAVE administrator, potential partners from VBE put their bids. System administrator based on recorded enterprises data, their conditions and incoming bids, rank and select the most suitable enterprises for different work packages and form up VE project enterprises consortium to collaborate for fulfilling new submitted product order. This partner selection procedure is illustrated in Figure 3.39.

Partner selection stage is finalized by preparing project contract by OMAVE administrator under the supervisory of OMAVE judiciary committee, and customer confirmation. Final step is signing the prepared contract by customer, OMAVE administrator and consortium partners. Note that, here two distinct agreements are prepared. One is between customer and OMAVE administrator and the other is between OMAVE administrator and consortium members (winner enterprises). Actually, here customer outsources the product to a virtual manufacturing company which is OMAVE administrator. Customer does not take care of remaining manufacturing problems and production procedures, OMAVE administrator gets the responsibility of orchestrating and managing of production processes. Second phase in virtual enterprise in the most of the literature is entitled as operation phase. Designing activities are also included in this phase. Collaborative design process and preparing master process planning and scheduling for a new product is the most important phase of product life cycle and it shouldn't be condensed. Therefore in this research, the operation and design phases are completely decoupled from each other. In order to enable the project members to design product remotely from their own location simultaneously, discuss and monitor running activities is a challenging research issue. Also, there are some commercial collaborative design tools developed by different companies. As development of such collaborative design platform is out of range of this study it is preferred to use one of these commercial tools (CATIA V6) in this research. Post designing procedure is development of master process planning and master scheduling for designed product. Diagram 3.40 is illustrating the management and work flow of product design, master process planning and scheduling.

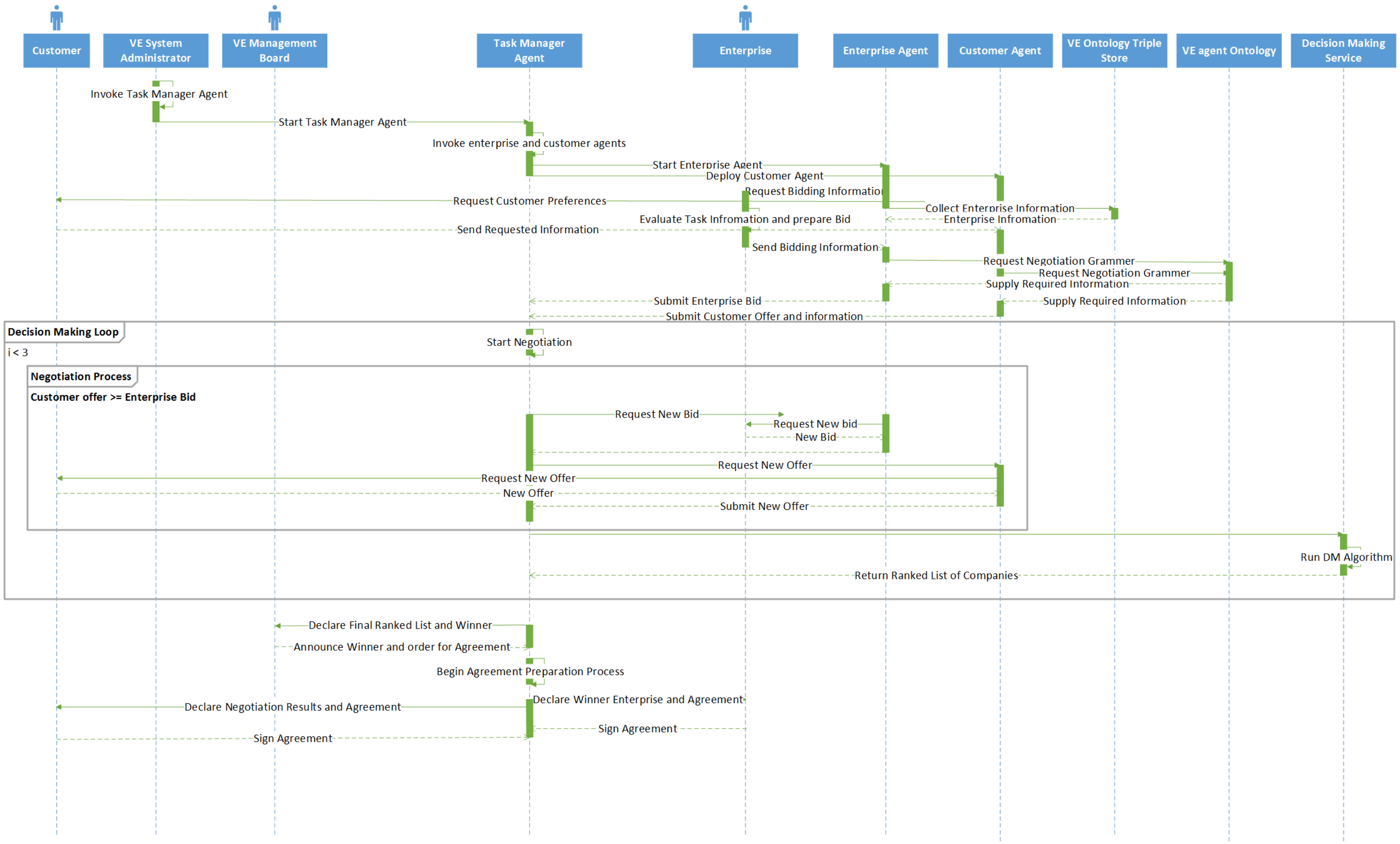


Figure 3.39: Partner Selection Process Sequence Diagram

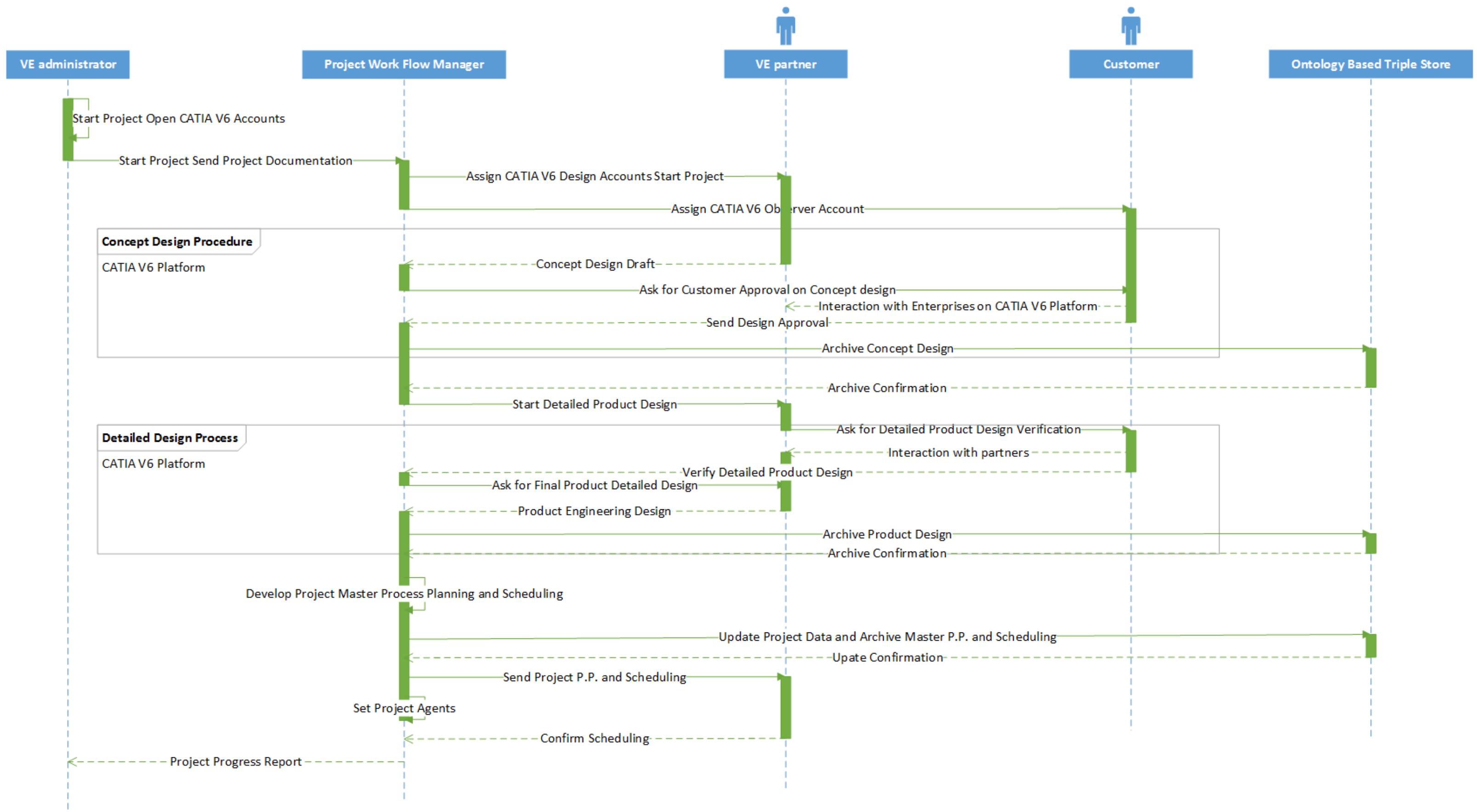


Figure 3.40: Design, process planning and scheduling sequence diagram

3.5 Summary

Developed IDEF and UML models give a clear view of OMAVE system requirements. OMAVE system infrastructure, applications and tools should be able to satisfy all mentioned system operations and necessities. As described earlier, to respond system needs and prerequisites a flexible, and highly reconfigurable structure for OMAVE system should be developed. Traditional tools and system architectures are not able to accomplish all system requirements. Therefore, a new approach to develop a semantic infrastructure for VE system is proposed in this study. To develop a semantic infrastructure for VE system, it is needed to develop an ontology of VE domain. VE data base, applications and tools will be defined based on developed VE ontology model.

CHAPTER 4

ONTOLOGY BASED DOMAIN MODELING

4.1 Literature Review on Implementation of Information Models on Virtual Enterprise Systems

Virtual enterprise concept was first introduced by Byrne on 1993 as "temporary nature of interactions between independent enterprises using Information and Communication Technologies (ICT)" [16]. In order to establish a suitable collaboration platform for VE and create a robust and reliable information management system, several studies were accomplished. Several researchers emphasized over using information and communication technologies (ICT) in temporary strategic business alliances. Further, numerous information management systems and data models were proposed by various scholars.

An information system, which only supports formal part of the inter-enterprise communication was developed by BIBA institute [54]. A specified information management system for process planning and control activities in X-CITTIC semiconductor manufacturing project, was introduced by Zhou et al. [31]. One of the earliest researches about development of internet based information systems for virtual enterprises is development of an innovative network centric information system for VE which proposed by Park and Favrel [88]. Implementations of ontologies in decision support system of virtual enterprises was introduced by Soares et al. for requirements analysis of semiconductor manufacturing sector virtual enterprises [114].

Supported by European Union, PRODNET project, was one of the cornerstone implementations of VE systems. In this project in order to manage complicated requisites of VE information systems, a federated information access mechanism proposed. The distributed information management system (DIMS) for different user levels, different access rules defined and used in the project. During implementation, DIMS developed using ORACLE database, and federated schema management modules developed in C++ [27, 42, 65, 93].

Persistent distributed data store (preDiS) based on distributed shared memory for VE software infrastructure was developed by Sandakli et al. [105]. A persistent software infrastructure, with concurrent access, coherent and secure distributed data store based on the distributed shared-memory paradigm was proposed. Another research related to Distributed information system architecture which implements CORBA and STEP standards for work flow management and information exchange was developed by Zhou and Nagi. Researchers in this study tried to develop a transparent communication channel and uniform data model format, to deal with heterogeneous data and knowledge in work flow management of virtual enterprise [133].

Vanderhaeghen and Loos developed a distributed model management platform to describe virtual enterprises complex business processes globally and locally to support virtual enterprise network design and implementation sufficiently [32]. Vifrebras was VBE environment which concentrates and contains enterprises from die and mould sector. A VE framework called AmbianCE was developed by Vallejos et al. relying on Vifrebras VBE to enhance the competitiveness of enterprises in VE [96].

In order to support a Service Oriented Architecture (SOA) based interoperable and distributed platform for automated agents negotiation of multi agent based VE negotiation process individual ontologies for VE initiator and potential partners was developed by Wang et al. [128]. A goal oriented trust model approach containing project constraints and strategies for partner selection process of virtual enterprises was developed by Mun et al. [81]. To facilitate and develop a context aware work flow management system based on multi agent architecture a design methodology was introduced by Hsieh and Lin. To describe work flows and resource activities in VE Petri net models are used for the coordination scheme of agents [58]. Initial Framework for Inter Sensing Enterprise (FISEA) is an inter-enterprise architecture with the support of sensing behavior technology, to allow showing collaboration components and real business relations applicable in virtual enterprises was introduced by Vargas et al. [120].

The summary of literature survey is given in Table 4.1. As it can be seen from this table, there is a lack of investigation over development of ontology based VE model, and implementation of unstructured data warehouses and web services to facilitate the communication of agents in VE systems. In this paper, the results of ontology based VE model and implementation of related unstructured triple store as a flexible data warehouse for OMAVE system is revealed.

Table 4.1: History of Information and Data Management Systems Development for VE Platforms.

Authors	Title	Focus Area in VE	Novelty	Sector/ Cluster Focus
Byrne, Brandt and Port (1993)	VE platform	VE System	For the first time the term VE introduced in literature	General
Hirsch, Bernd E., Thorsten kuhlmann, Zbigniew K. Marciniak, and Christian Mabow (1995)	Information system concept for the management of distributed production	Formal part of the inter enterprise communication	A VE information system introduced	General
Camrinha-Matos, L. M., H. Afsarmanesh, C. Garita, and C. Lima (1998)	Towards an architecture for virtual enterprises	VE Architecture and Framework	VE lifecycle and system architecture have been revealed	General
Zhou, Q., P. Souben, and C. B. Besant (1998)	An information management system for production planning and control in virtual enterprises	Production Planning and Control	An information system for production planning in VE was developed	Semiconductors Manufacturing
Park, Kyung Hye, and Joel Favrel (1999)	Virtual enterprise- information system and networking solution	VE Information Systems- Networking	Networking solutions for VE were presented	General
Soares, Antonio Lucas, Americo Lopes Azevedo, and Jorge Pinho De Sousa (2000)	Distributed planning and control systems for the virtual enterprise: organizational requirements and development lifecycle.	VE Information Systems- Distributed Planning and control	A new innovative planning and control systems were developed for VE	General
Garita, C., Hamideh Afsarmanesh, and L. O. Hertzberger (2001)	The PRODNET cooperative information management for industrial virtual enterprises	VE information Systems	Cooperative information management system for VE has been developed	PRODNET
Klen, Alexandra, Augusta Pereira, Ricardo J. Rabelo, Aureo Campos Ferreira, and Luiz Marcio Spinosa (2001)	Managing distributed business processes in the virtual enterprises	VE Business Process	Distributed business processes management system was revealed	General
Osorio, A. Luis, and Manuel M. Barata. (2001)	Reliable and secure communications infrastructure for virtual enterprises	VE Information System and Data Transaction	VE platform was equipped with a communication infrastructure	General
Sandakli, Fadi, Joao Garcia, Paulo Ferreira, and Patrice Poyet (2001)	Distributed shared memory infrastructure for virtual enterprise in building and construction	Data Share and Information System	Shared memory infrastructure was applied in VE	Building and Construction
Zhou, Liangyu, and Rakesh Nagi (2002)	Design of distributed information systems for agile manufacturing virtual enterprises using CORBA and STEP standards	Distributed Information System	CORBA and STEP standards were applied in VE	General
Vanderhaeghen, Dominik, and Peter Loos (2007)	Distributed model management platform for cross-enterprise business process management in virtual enterprise networks	Modeling and Management in VE	Distributed model and business process management application in VE were introduced	General
Vallejos, R. V., C. P. Lima, and G. Varvakis (2007)	Towards the development of a framework to create a virtual organization breeding environment in the mould and die sector	VBE	VBE framework introduced	Dies and Mould

Table 4.1: History of Information and Data Management Systems Development for VE Platforms (cont'd)

Authors	Title	Focus Area in VE	Novelty	Sector/ Cluster Focus
Wang, Xiaohuan, T. N. Wong, and Gong Wang (2010)	Service-oriented architecture for ontologies supporting multi-agent system negotiations in virtual enterprises	VE architecture and Negotiation Process	SOA, ontology and MAS technologies were implemented in VE	General
Mun, Jungtae, Moonsoo Shin, and Mooyoung Jung (2011)	A goal-oriented trust model for virtual organization creation	VE modeling	Goal oriented trust model implemented in VE	General
Moisescu, M. A., and I. S. Sacala (2014)	Towards the development of inter-operable sensing systems for the future enterprise	VE Communication Infrastructure	Sensing systems implemented and tested in VE	General
Hsieh, Fu- Shiung, and Jim- Bon Lin (2014)	Context-aware work flow management for virtual enterprises based on coordination of agents	Work flow Management	Implementation of Petri net models in Agent coordination scheme in MAS based VE work flow management systems	General
Vargas, Alix, Lianos Cuenca, Andres Boza, Ioan Sacala, and Mihnea Moisescu (2014)	Towards the development of the framework for inter sensing enterprise architecture	VE Framework	Inter- sensing enterprise architecture was developed	General

4.1.0.1 Ontology Domain Modeling and Virtual Enterprises

In order to make appropriate decisions for VE (e.g., partner selection, VBE bench marking, risk management, etc.), all concepts, relationships and constraints must be defined in a comprehensive domain knowledgebase. VE specific data should be stored in a logical and reliable manner [59, 131]. Ontologies and the corresponding knowledge bases provide the best tools for modeling such complex domain knowledge and highly dynamic data requirements. Ontologies not only help model and capture complex domain knowledge, but also improve sharing and re-usability of data and knowledge and provide a suitable environment for agent and human communications [22, 51].

Ontologies represent including knowledge formally agreed upon concepts and their relationships for an application domain. Therefore, unlike task-specific and implementation-oriented data schema, ontologies should be as generic and task-independent as possible [115]. Ontologies play an important role in defining the terminology used by agents in the exchange of knowledge-level messages; therefore, the choice of an ontology representation language is a significant issue when designing a multi-agent system [29].

Considering these properties of ontology knowledge representation, ontology based domain models are developed to increase flexibility and meaningful communication between VE agents and entities, and to enhance the integration of various business management process systems (e.g., MRP or MES) used by VE partners [18, 132]. The main reasons to develop an ontology based domain model for agent based VEs are [51, 86];

- Create a common vocabulary for software agents to understand each other
- Enable agents to reuse and analyze domain knowledge
- Separates domain knowledge from operational knowledge

Virtual enterprise applications are designed based on modular software architecture, and these applications are connected to a central data warehouse. Information from heterogeneous data sources transfer continuously; therefore, a VE system should be able to handle a huge amount of dynamic data. All applications in this system are served to partners based on the SaaS concept, and web services connect databases to the VE system data warehouse [6]. These applications require an up-to-date, flexible, dynamic and reliable system database structure to handle heterogeneous data from diverse sources with different formats. By developing an ontology based database structure, it is possible to manage a large amount of heterogeneous, dynamic data and knowledge [6].

As can be seen from literature review which gathered from resources concentrating on VE modeling issues very rare studies and research could be found which use ontology as a tool for modeling virtual enterprise systems. One of the eminent works in this area has been done by Wang et al. in which ontology supports agents negotiation process in virtual enterprise. In

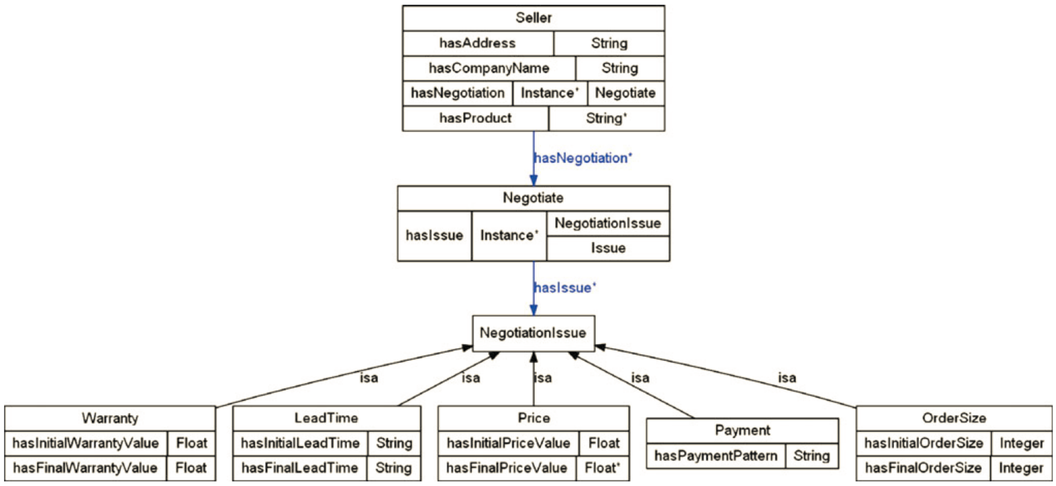


Figure 4.1: Ontology model for the Seller designed by Wang et al. [128]

this work ontology model is proposed (Figure 4.1) for agents to understand each other during the bidding procedure.

In order to harness all the potential in an industrial organization park, all resources available in different companies and sectors must be considered. When creating a VE, it is necessary to understand each company's resources and capabilities and whether a company's qualifications meet customer requirements. Thus, the VE system must be able to demonstrate what types of resources exist, where they are located, and how much resource capacity is available to be used by consortium. At the same time, private corporate information must be protected and guaranteed to remain secret, inaccessible and coded. Aiming to reach these goals this research could be assumed as one of the state of art study in the case of developing a comprehensive VE domain model. In the forthcoming sections (section 4.3 and 4.2.3) detailed information about proposed VE ontology domain model and its application are given.

4.2 Ontology Based Domain Modeling

In different science fields ontology has different meanings. The word ontology is the combination of ancient Greek words $(o\nu\tau o\varsigma)$ and $(\varepsilon\sigma\mu i)$ which means *being which is*. Originally philosophically studying of beings, nature, existence and realities and finding their relations is called ontology. Ontology deals with the hierarchical structure of beings in the reality and their classification in different existing being groups and study their properties and relations. This issue is actually to define human beings understanding from life, reality and shapes human's mind regarding surrounding world and helps him/her to study and interpret logically entities, their relations and rules in nature.

In Information Technology (IT) and computer sciences the same approach is also considered with respect to ontology. In order to enable intelligent systems to interpret and enhance their reasoning capabilities it is inevitable to capture and organize knowledge and establish computational models for automated reasoning. On 1990s Tom Gruber in his famous article *"Toward Principles for the Design of Ontologies Used for Knowledge Sharing"* used ontology term as a technical term in in computer and information sciences with the definition of [50]:

Tom Gruber, 1993

> Ontologies are often equated with taxonomic hierarchies of classes, class definitions, and the subsumption relation, but ontologies need not be limited to these forms. Ontologies are also not limited to conservative definitions — that is, definitions in the traditional logic sense that only introduce terminology and do not add any knowledge about the world. To specify a conceptualization, one needs to state axioms that do constrain the possible interpretations for the defined terms. [50]

It is obvious that all surrounding world around us could be modeled and classified in this way. Indeed the main aim which looks like a dream for human-being is to model all the world around us to empower machines and artificial intelligent systems understand reality as we do and launch a tremendous semantic network of information and knowledge interpretable by machines. However modeling all these information and data comprehensively is next to impossible. Therefore, domain based ontologies is generated. Based on problem definition and system requirements, a domain ontology could be designed. A domain ontology represents knowledge and information about entities, objects, functions, events, other system building blocks also their properties, variants, and their relations. Meanwhile, special concepts and terms for using in that domain by the system also could be provided.

4.2.1 Ontology Based Domain Models

An ontology model is formal way of representing shared knowledge in a specified target domain precisely and explicitly. All types of concepts used in the domain are defined and detailed comprehensively and clearly by ontology domain model. As all these information and knowledge are machine understandable therefore we call this formal representation of knowledge. Ontology model provides universal knowledge which is acceptable and approved by domain entities and share the common concepts served by ontology model. As was mentioned before, ontology domain models present knowledge in a machine understandable way. Obviously this type of language should be standardized and universally accepted by organizations [66].

Open Knowledge Base Connectivity (OKBC) ontology models have been developed earlier and are assumed as more mature ontology models which are called Frame based ontology models. These type of models encapsulate frames, slots, facets, classes, individuals and knowledge base [75]. In order to enhance semantic web infrastructure, World Wide Web Consortium (W3C) developed web ontology language (OWL) standard which include entities like classes, properties, and individuals. Properties are also divided into object and data properties as well. There are different standard languages for ontology models including layered W3C standards which are as follows;

XML/S
: Extended Markup Language / Schemas (XML/S), is a developed by W3C to handle large scale data exchange mainly over the web.

RDF/S
: Resource Descriptive Language / Schema (RDF/S), standard language is also developed by W3C to represent information regarding resources on the web. RDF metadata model is working based on triple concept. Triples are consisting of subject, predicates and object concepts. Subjects are actually the resources or the target which is going to be described. Predicate is the property or relation about describing resource and the object is the value of that specific subject. In order to add a semantic extension for RDF languages, RDF/S schemas are added. RDF Schema are actually defining the representation way of concepts and entities in RDF language and can be considered as semantic extension of RDF languages.

OWL
: Web Ontology Language (OWL), in fact is vocabulary extension of RDF to extend semantic web project. Some shortcomings of RDF/RDFS language has been eliminated by developing OWL standard. some of these drawbacks could be named as scope of properties, class disjoints, classes boolean combination and criminality restrictions. OWL itself has three sub languages as; OWL Lite, OWL DL and OWL Full.

OWL Lite
: OWL lite is used for simple creation of classification hierarchy and constraints with lower formal complexity than OWL DL and OWL Full.

OWL DL
: It provides maximum expressiveness and computational completeness. It means that all computations are guaranteed. language constructs are available but there are some certain restrictions. (for example a class can only be subclass of many classes however it can not be a instance of a class).

OWL Full
: It offers maximum expressiveness and syntactic freedom but no computational guarantee. On oppose to OWL DL a class here could at the same time a collection of individuals and also be an individual itself. Here predefined RDF or OWL vocabulary meaning could be augmented but not reconfigured.

Here is the summary of ontology model language standards and their advantages and disadvantages [53]:

XML A syntax surface structure without any semantic meanings for documents is provided.

XMLS Restricts XML language but at the same time extends XML standard by adding *"datatypes"* to XML

RDF A data model to describe *resources* as *objects* and define their properties. Provides simple semantic for data model. It also can be represented in XML syntax.

RDFS It adds a vocabulary for RDF language to describe RDF concepts like classes and properties.

OWL Even more extensive describing vocabulary for classes, properties, relations, cardinality and enumerated classes.

In this research, OWL DL language is selected for development of VE domain model. This selection was done by considering requirements for semantics and computational logic, augmentation in OWL concepts. Requirements for semantics are due to using intelligent agents in different phases of VE hierarchy. These agents need to interact each other and system administrator continuously. Therefore, a common communication language with definite standard is required. Proposed VE system poses intelligence in different stages of decision making or operation. These decision making, operation management or partner selection procedures require sophisticated computational or conjectural calculations which are also supported by OWL DL structure. The other necessity for using OWL DL standard for VE domain modeling is availability of augmentation in standard concepts. VE structure requires more flexible infrastructure for serving different industrial sectors. Thus in order to fulfill various types of requirements from heterogeneous resources it is needed to make the definitions very precisely. In order to reach this goal OWL DL allows to enhance concept definitions in OWL standard to feet our requirements in domain modeling.

Formal semantics of ontology modeling enables logic reasoning over principles, concepts and properties. For example: In OMAVE model it is assumed that Medium Size Enterprise employee number is between 50-200 [28]. *"Enterprise A"* is a Medium Size Enterprise. Therefore logically it can be concluded that *"Enterprise A"* has employee number between 50-200.

$$(\forall x)\,(Mx) \rightarrow E\,(x) \tag{4.1}$$

Here in this syntax (4.1) consider; x is a Medium Size Enterprise (*"x = Enterprise A"*), M stands for relation or predicate *"M = is a Medium Size Enterprise"*, E stands for Employee Number between *E = [50-200].* Therefore if 'Enterprise A' is an instance for Medium Size Enterprises then, it could be reasoned that "Enterprise A" has employee number between 50-200. Based on given knowledge and assigned rules to the ontology model, logically system deduce an inferred model. Here another example can be illustrated;

All professors are faculty members and all faculty members are considered as faculty staff. *"P"* is a professor. Logically system automatically reasons that as *"P"* is a professor therefore he is a faculty member, thus her is also a faculty staff.

(Prof(x)) $\rightarrow$ (Faculty Member(x))
(Faculty Member(x)) $\rightarrow$ (Faculty Staff(x))
Professor(P) $\rightarrow$ Faculty Member (P) $\rightarrow$ Faculty Staff (P)

Description Logic (DL) as one of abilities of OWL DL language which mentioned before, and in this research several types of rule based reasoning are applied. These rules and their results are discussed comprehensively in section 4.2.3.

4.2.2 Protege Ontology Editor

There are several available ontology editors. In this study, Protege (a free, open-source ontology editor and framework for building intelligent systems) has been utilized. This program has been developed by Stanford Center for Bio medical Informatics Research group and is widely used in academia to develop knowledge based solutions for bio-medicine (mostly) and modeling organizations and e-commerce sectors [38]. This software support W3C developed standards and languages for semantic web which is this research priority in selecting a suitable platform for development of VE ontology model.

4.2.3 Declaring Concepts, Rules and Properties

As it mentioned before, an owl ontology model is created based on triples (objects, predicates and subjects). Here, objects and subjects are defined concepts in the model and predicates covers model properties. Concepts in ontology models are classified into classes and individuals. Classes (owl:Class) are main building blocks of class axiom. Classes are defined to classify a set of entities or objects which are sharing a similarity. The other concept in ontology models are individuals. Individuals are members of classes those represents unique object or entity in the system. Two types of properties are in the models. One type of properties declare relations between these concepts which are called object properties and others give more detailed information about those concepts and link them to data values. This type of properties are called data type properties. These property types by themselves are also instances of two distinguished classes: *owl:ObjectProperty* and textitowl:DatatypeProperty.

Semantic Web Reasoning Language (SWRL) rules were added to the developed ontology to add formalism and reasoning capabilities to the developed VE ontology model. OWL document without rule axiom is an abstract syntax containing a sequence of concepts and facts like subclass or individual axioms. A rule axiom is consisting of antecedent and consequent parts. Each of these parts consist of a set of atoms. Each rule is referenced by a unique Uniform Resource Identifier (URI) in OWL document. In a rule if antecedent holds *"true"* value, the consequent function or value also must be hold. Empty antecedent value is presumed as true however empty consequent contains false value.

rule::= 'Implies ('[URIreference] $\{annotation\}$ antecedent consequent')'
antecedent ::= 'Antecedent (' $\{atom\}$ ')'

consequent ::= 'Consequent (' $\{atom\}$ ')'

Atoms are presented in the form of $C(x)$ as it was illustrated in relation 4.1 or in the form of $P(x, y)$. Examples for $P(x, y)$ could be *sameAs(x,y)*, *differentFrom(x,y)* or *builtin(r,x,y,..)*. The atoms with one arguments are owl descriptions or data-values. But atoms with multiple arguments are owl properties. These properties or description could be owl predefined or designer defined concepts. Atoms could refer to different concepts' URI like individuals, data properties, variables and etc [57].

atom ::= description '('i-object')'
| dataRange '('d-object')'
| individualvaluedPropertyID '('i-object i-object')'
| datavaluedPropertyID '('i-object d-object')'
| sameAs '('i-object i-object')'
| differentFrom '('i-object i-object')'
| builtIn '('builtinID'(d-object)')'
builtinID ::= URIreference

The abstract syntax is very similar to formal XML format (Extended Backus-Naur Form (EBNF)) notation [12]. This notation is difficult for human to read and understand. Therefore, human readable standards and formats are also developed based on different platforms. In this research, two types of notations have been used. One is SWRL notation which is described here and the other is Jena notation. Jena is a Java framework for building semantic applications which is developed by researchers in HP labs. Based on W3C recommendations, Jena provides java libraries to enable developers to develop codes to handle RDF, RDFS, OWL and SPARQL Protocol and RDF Query Language (SPARQL) formats inline. Jena also has a inference engine to make reasoning based on OWL or RDFS ontologies through created triple stores. In order to use Jena reasoning abilities in this research, Jena notation also is applied [57].

Human readable rule syntax is quite simple and understandable;

antecedent ⇒ consequent

As an example, for this type of notation a composition of parent and brother relation with is resulted in an uncle relation is as follows;

parent(?x , ?y) ∧ brother (?y, ?z) ⇒ uncle(?x , ?z)

In this notation, the antecedent side or rule express that y is a parent of x. Second atom is declaring that y has brother relation with z. Consequently, z will have an uncle relationship with x.

4.3 OMAVE Ontology Based Model

This research is the very first step of a multi-step project to establish a flexible, reconfigurable and operational VE system applicable for most of the industrial sectors. In this step, machining and metal forming sectors are targeted as a pilot test bed for this system. However, a comprehensive VE domain model is derived and designed. But special segments related to mentioned sectors are developed in details, and the other segments will be developed step by step in the next phases of project.

4.3.1 OMAVE Ontology Model Class Axiom

The first stage of establishing ontology model is defining concepts, model classes and subclasses. Model concepts are classified into 9 main categories (Classes). Each of these classes and sub classes has their own specific instances, properties and relations. All assigned properties for the upper level classes is inherited by lower level classes and their members (Individuals).

Organization
All the organizations, like SMEs, original equipment makers, universities, Institutes, R&D companies, Multi-National Enterprises and etc. is going under this category.

Resources
Class resources itself is categorized into three main sub classes. These sub classes are named as;

Human Resources As it is clear from its name, enclose all types of human resources registered in the system such as; engineers, technicians, office employees, workers, managers and etc.

Physical Resources Physical resources class is containing all manufacturing, research, test and quality control equipment and machinery. This is one of the main pillars in this system. As all manufacturing processes require different types of properties and manufacturing restrictions. Therefore they need special enablers as machinery. Based on these requirements, the most appropriate machinery and equipment for that special product could be searched from the system. Here, for each class of machinery, very specific properties also assigned.

Software Nowadays from prototype step to after sale services vast variety of software are used by various resources, software compatibility is a very serious problem. VE platform is embracing different types of companies with different strategies and inventory. Therefore to procure compatibility, keeping information about diversified software resources seems quite important to this system.

Manufacturing Processes
Manufacturing Processes class perhaps is the most important part of this research. All the reasoning, selection, evaluation processes begin with evaluating and requirement analyze of a manufacturing process. Based on these process necessities, suitable machine tools, machine tool owners and others are listed and partner selection procedure

starts. Manufacturing processes class is including 10 different sub-manufacturing process sub classes. The target was to cover all existing manufacturing processes as possible as it is. These 10 sub classes are as follows; Casting Processes, Coating Processes, Forming Processes, Heat Treatment Processes, Joining Processes, Machining Processes, Molding Processes, Painting Processes, Rapid Manufacturing Processes.

Facility

There are some reasons that this section is added to VE platform. Since this system is a web based program and operation management part of this project which is Dassault systems tools are SaaS based software, it is crucial to have information about internet and LAN infrastructure of contributing organizations. Hence, there is a need to keep information about organizations energy, environmental and waste treating performances, in later phases of project ecological and environmental issues will be integrated to this system. Another reason arises from special delivery requirements for special products. In some cases, special transportation requirements for part may be needed by customer like rail transportation etc. Considering all these motivations this class is added to VE model.

Project

Project is the building block of a VE platform. Based on a project a consortium is formed up and remaining processes starts. Completed, ongoing, future projects are going to be included under the project class.

Product

Every project is targeting to produce one or more product(s). Each product(s)'s information, manufacturing processes progress, and other are kept under this class.

Part

Likewise products also are a combination of part(s). Like product related instances are categorized under this class.

Task

In order to produce a part, different tasks should be accomplished (For example, turning task, milling task, grinding task and etc.). These tasks are combination of several consecutive manufacturing processes which are packaged as a single task for outsourcing.

Work Piece Material

For deciding about machine tools capability for machining or casting or any other manufacturing processes, products material is playing an important role. Therefore, a material library is absolutely needed for decision makings.

Figure 4.2 depict all mentioned system classes and sub-classes described above.

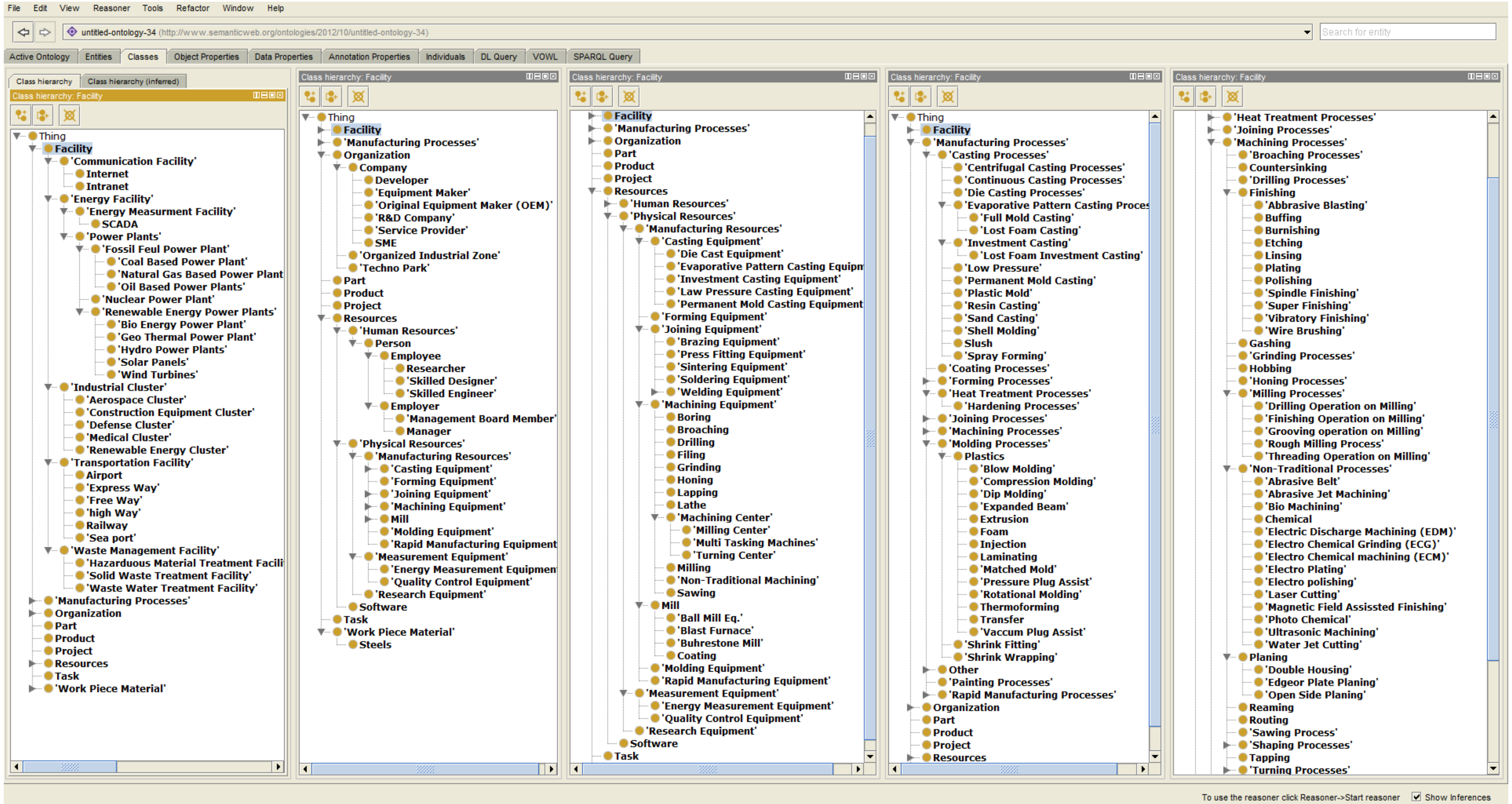


Figure 4.2: Ontology Model Classes- Protege 4.3

4.3.2 OMAVE Ontology Model Individual Axiom

VE system instances as members of classes are represented as Individuals in OMAVE model. Individuals may be member of one or many classes. Here, classes could not behave as individuals. It means that a class cannot be at the same time a class and an individual of another class. Each individual inherits its belonging classes properties. By means of properties both type of properties are considered; owl:dataTypeProperty and owl:objectPropertiy (predicate). As shown in 4.3 Mazak Variaix II as a machining center tool, is placed under Resources/Physical Resources/Manufacturing Resources/Machining Equipment/Machining Center/ Milling Center category. It is also a member of all upper levels and as it is mentioned before it may inherit all those upper level properties and relations as well.

It can also be observed from Figure (4.3) that all the individuals are in the same pool but they are categorized by their type which clarifies individuals' belonging class, their inherited properties and relations.

This platform is a bilingual platform and has been designed for both English and Turkish users. This language change issue is supported by assigning labels for all the entities, concepts and properties in the system. As a sample in upper right side of Figure 4.3, both labels are visible. As Mazak Variaxis II is a special name here for both languages, it gains the same value however (for example for an SME type company, English label set as *'SME'* but in Turkish it has been labeled as *'KOBİ'*).

Individuals are related to each other based on defined predicates (owl:objectProperties) or relations. Detailed information about object properties will be given in the section 4.3.3.2. As an example from Figure 4.3, Mazak Variaxis II individual has a relation *'ia an asset of'* with individual EMGE.

Literal values are assigned to individual indirectly through data type properties (owl:dataTypeProperty). Data type properties originally are assigned to a class and consequently member individual of that class inherits that property. Detailed information about owl:dataTypeproperty is described in section 4.3.3.1. For instance, here max Spindle Speed for this machine tool is $12000 rpm$, or Linear Rapid Traverse Rate in X direction is $60000\left(\frac{mm}{min}\right)$.

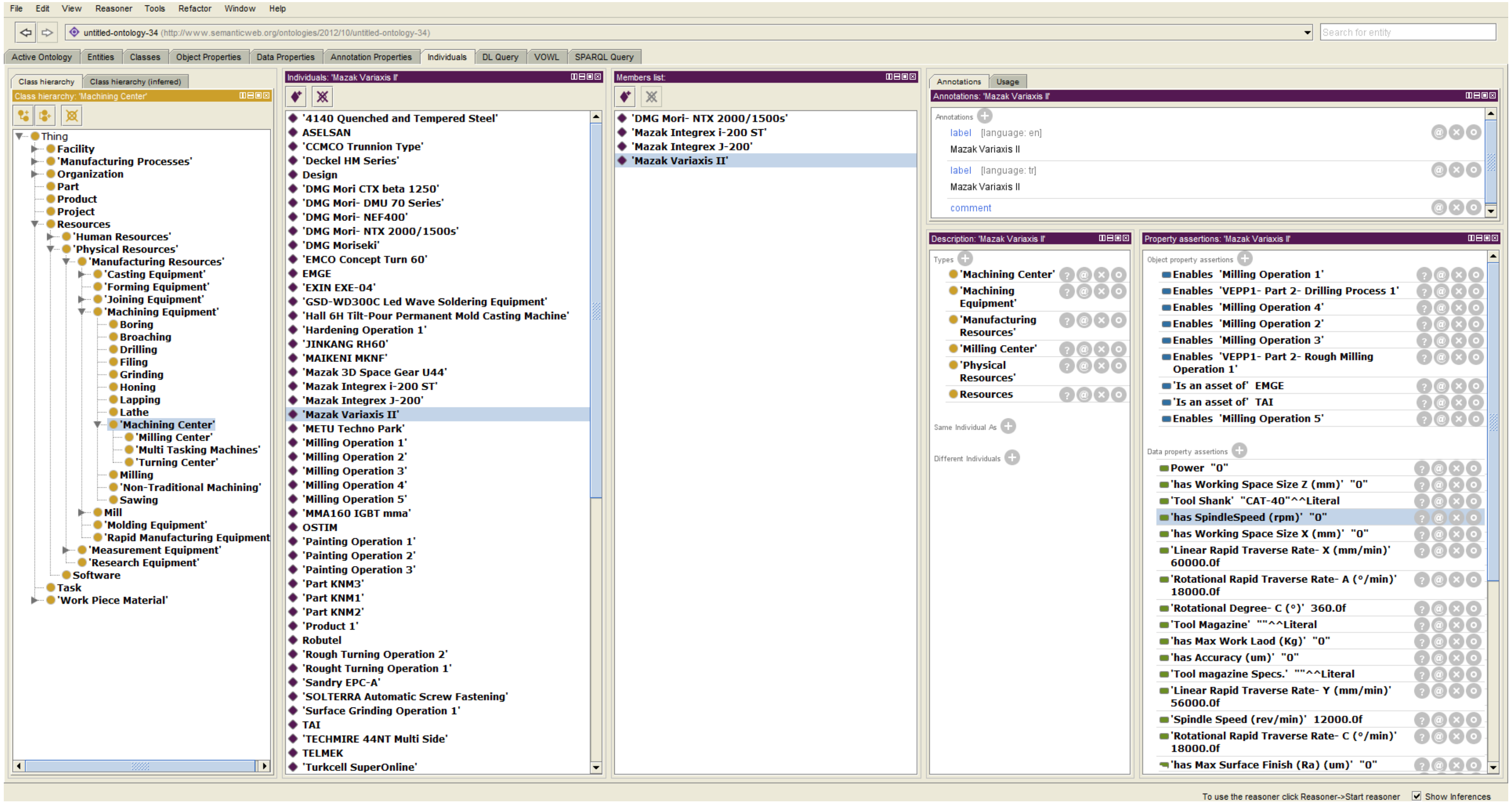


Figure 4.3: Sample Individual (Mazak Variaxis II) and belonging class and sub classes- Protege 4.3

In the right side of Figure 4.3 there are several *'Enables'* relation that connects *'Mazak Variaxis II'* individual to different task or processes. These relations are resolved automatically by model. Actually this Figure is obtained from inferred ontology model after automated reasoning from originally created OMAVE ontology model. Different reasoners could be applied in protege and based on designed rules and selected reasoner engines the inferred model could be obtained. For more information about rules and protege reasoners, please refer to section 4.3.4.

4.3.3 OMAVE Ontology Model Properties Axiom

In OWL language two main categories of properties are available.

owl:dataTypeProperty
: This type of property link individuals to data values

owl:objectProperty
: This type of property connect individuals to other individuals, It is also called predicate

Both of these properties are categorized under class of rdf:Property. In OWL full format these two types of properties are not separated and they are acted as the same. This is because of the OWL Full definition about data values which are treated as individuals. In OWL DL format two other type of properties are also used for semantic reasoning;

- owl:AnnotationProperty
- owl:OntologyProperty

Given property values, properties functionality and other restrictions also could be managed in OWL standard. A property could be a sub property of another property. This characteristic *(rdfs:subPropertyOf)* and property's domain and range construction *(rdfs:domain and rdfs:range)* comes from RDF schema which also is a upper level of OWL format and OWL also inherit this characteristic as well.

Predicates may also have inverse and equivalent relations *(owl:equivalentProperty)* and *(owl:inverseOf)*. For example *'is an asset of'* and *owns* relations are inverse relations in OMAVE ontology model. Assigning cardinality (owl:FunctionalProperty) and *(owl:InverseFunctionalProperty)* to the relations is one of the other characteristics of OWL DL format which is discussed before.

Properties may also catch transitive or symmetric characteristics *(owl:SymmetricProperty)* and *(owl:TransitiveProperty)*.

Here Symmetric property means that if a pair of (x, y) is an instance of Property P then (y, x) also is an instance of property $'P'$.

Transitive property means that if pairs of (x, y) and (y, z) are instances of Property then (x, z) also is an instance of property $'P'$.

Inverse functional property means that there can not be two distinctive $x1$ and $x2$ where both $(x1, y)$ and $(x2, y)$ are instances of property $'P'$. or in another words, if we inverse property $(x1, y)$ and $(x2, y)$ we will get $(y, x1)$ and $(y, x2)$ and as first argument (y) is the same, we should also get the same second argument $(x1 = x2)$.

In OWL DL format annotation, properties also allowed. Five types of annotation properties are available:

1) owl:versionInfo
2) rdfs:label
3) rdfs:comment
4) rdfs:seeAlso
5) rdfs:isDefinedBy

In OMAVE ontology *rdfs:label* and *rdfs:comment* properties are used. For bilingual support of OWL DL, *rdfs:label* annotation property is applied. Two types of labels $'en'$ for English and $'tr'$ for Turkish language.

rdfs:comment annotation property also is used to link individuals to their belonging web URIs. For example for 'Enterprise A' individual a comment annotation property with the value of $http : //www.enterpriseA.com$ as a web address has been assigned.

4.3.3.1 VE Ontology Model owl:dataTypeProperty Axiom

As was mentioned in the last section, data type properties relate individuals to the data values indirectly through their super classes. In developed OMAVE ontology model, different type of data type properties for various classes is assigned. In this model nearly,405 different data type property are designated. Most of these properties are manufacturing process properties and machine tools properties. Despite focusing on machining, metal forming processes, machine tools and other manufacturing processes properties are also defined in the model. Likewise sample machine tools for several manufacturing processes are exemplified.

4.3.3.2 VE Ontology Model owl:objectProperty Axiom

Nearly 50 different relations have been named in OMAVE ontology model to relate different individuals and classes to each other. Each of these relations have their own special restrictions (and properties like transition, reverse and etc.) as described before in section 4.3.2. General list of properties in VE ontology model is illustrated in Figure 4.4.

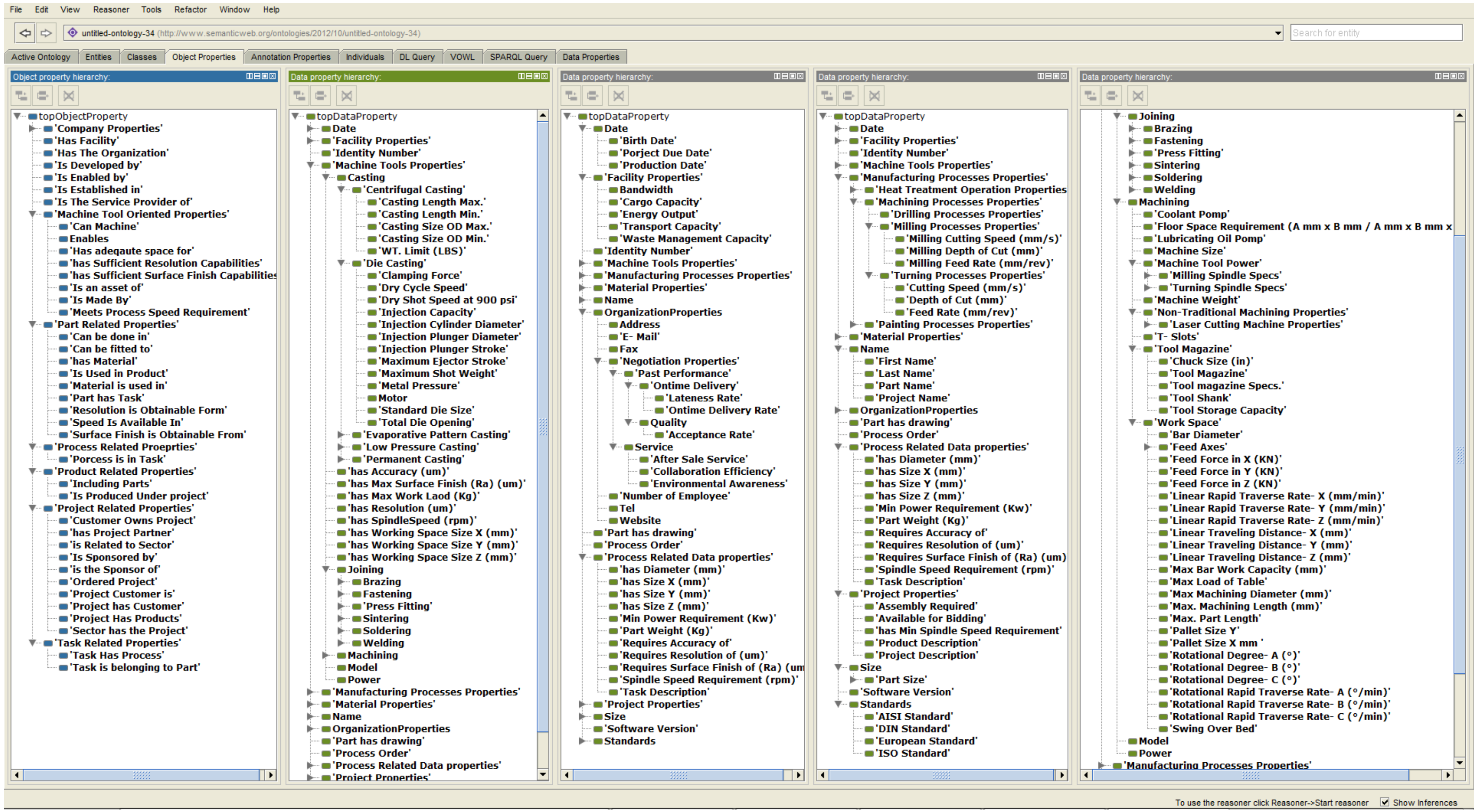


Figure 4.4: List of object and dataType properties of VE model- Protege 4.3

4.3.4 OMAVE Ontology Model Rules Axiom

A reasoner or inference engine actually deduces new knowledge and information from provided knowledge and information in the OWL model. Some reasoner engines in OWL models are undertaking this reasoning action for developers. OWL model are containing statements (called triples). Inference engines use these statements to acquire new statements and shape inference models based on developed model. There are four different reasoner engines available in Protege 4.3. [38].

FaCT (Fast Classification of Terminologies)
Is a Descriptive Logic classifier and modal logic satisfaction testing tool. [56]

FaCT++ Is a new version of FaCT OWL DL reasoner tool. FaCT++ uses the same algorithms as FaCT but with different architecture. Also FaCT++ is developed on C++ language to enhance software tool efficiency and portability. [56]

Pellet Pellet is a java based OWL DL reasoner with use of both Jena and OWL DL libraries with DIG (Description Logic Reasoner) interface. Pellet is developed based on tableaux algorithms for expressive DLs [90].

Hermit Hermit is for determination of an OWL format to check the file consistency and verify relations and concept of model. This reasoner is based on hyper tableau calculus to reach a better efficiency in reasoning [44].

Jena The Jena inference subsystem is designed to allow a range of inference engines or reasoners to be plugged into Jena platform. [78]

Pellet supports SWRL rules. If model contains SWRL rules and run Pellet reasoner engine to reason over the model, it takes those SWRL rules into consideration [78]. However, though both Pellet and Jena support a notion of rules, the intended domains of SWRL rules and Jena rules are very different. SWRL rules are OWL-level constructs; the binary predicates in a SWRL rule are class expressions, or the binary predicates are object and data properties. Additionally, SWRL rules only match named individuals; they don't match individuals whose existence is only inferred. Jena rules, on the other hand, are RDF-level, and designed to work on RDF-graphs. While RDF and OWL are often used together, (e.g., OWL data is serialized in RDF), the two are conceptually different. An OWL reasoner could be implemented that makes no use of RDF, and a SWRL engine could be built that make no use of RDF graphs [78].As OMAVE ontology model APIs are developed on Jena platform. The final model system rules are designed on RDF-level, it is preferred to use Jena rules in this research. In table 4.2, developed Jena rules are illustrated. The equivalent SWRL rules of the ontology model, Jena rules also were developed and tested. But as they were not used in the model due to using Jena platform, only the equivalent SWRL rules for part size check and part tolerance check Jena rules are illustrated below. Just to mention the difference between Jena rules notation, and SWRL rules notation following examples for SWEL rules are given. They could be compared with their equivalent rules in Jena in table 4.2:

Part Size Check SWRL Rule:

hasSizeX(?part, ?partXSize)^
hasSizeY(?part, ?partYSize)^
hasSizeZ (?part, ?partZSize)^
hasWorkingSpaceSizeX(?machineTool, ?mTXSize)^
hasWorkingSpaceSizeY(?machineTool, ?mTYSize)^
hasWorkingSpaceSizeZ(?machineTool, ?mTZSize)^
swrlb:greaterThan (?partXSize, ?mTXSize)^
swrlb:greaterThan (?partYSize, ?mTYSize)^
swrlb:greaterThan (?partZSize, ?mTZSize)
→
canBeFittedTo (?part, ?machineTool)^hasAdequateSpacefor(?machineTool, ?part)

Part tolerance Check SWRL Rule:

hasMaxTolerance (?part, ?partMaxTolerance)^
hasMaxPrecision (?machineTool, ?mtPrecision)^
swrlb:greaterThan (?mtPrecision, ?partMaxTolerance)
→
canBeMachined(?part, ?machineTool)^
hasEnoughPrecisiontoProduce(?machineTool, ?part)

In Table 4.2 these prefixes are used:

PREFIX rdf: http://www.w3.org./1999/02/22-rdf-syntax-ns#
PREFIX owl: http://www.w3.org./2002/07/owl#
PREFIX xsd: http://www.w3.org/2001/XMLSchema#
PREFIX rdfs: http://www.w3.org./2000/01/rdf-schema#
PREFIX veonto: http://www.ostim.org.tr/omave#

Table 4.2: List of assigned rules for reasoning in OMAVE ontology model.

Rule Name	Rule Description	Sample Jena notation of rule
Machine tool selection	Search System based on process definition and list enablers machine tools for this process and assign listed machines to the named process. This rule is activated after every process or machine tool submission automatically. This rule also assigns reverse relation between submitted entities, which is called "is enabled by" relation.	(?mtm rdf:type veonto:Multi Tasking Machines) (?mtm rdf:type veonto:Turning Processes) (?mtm rdf:type veonto:Milling Processes) (?mtm rdf:type veonto:Drilling Processes) → (?mtm veonto:enables ?y) (?mtm veonto:enables ?z) (?mtm veonto:enables ?w)
Dimensional verification	Based on selected part's physical size and volume, suitable machine tools are filtered out.	(?part veonto:hasDiameter ?a) (?part veonto:hasSizeX ?b) (?machineTool veonto:MaxMachiningDiameter ?x) (?machineTool veonto:MaxMachiningLength ?y) greaterThan(?x ?a) greaterThan(?y ?b) → (?part veonto:canBeFittedTo> ?machineTool) (?machineTool veonto:hasAdequateSpaceFor ?part)

Table 4.2: List of assigned rules for reasoning in VE ontology model (cont'd)

Rule Name	Rule Description	Sample Jena notation of rule
Power verification	Minimum required power for the selected process is compared to the listed machine tool power and machine tools with less than needed power are eliminated.	(?process veonto:hasMinPowerRequirement ?a) (?machineTool veonto:MotorOutputPower ?x) greaterThan(?x ?a) → (?process veonto:canBeDoneIn ?machineTool) (?machineTool veonto:canMachine ?process)
Max spindle speed verification	Checks for process's spindle speed requirements in machine tools and available machines' list is populated	(?process veonto:hasMinSpindleSpeedRequirement ?a) (?machineTool veonto:SpindleSpeed ?x) greaterThan(?x ?a) → (?process veonto:speedIsAvailableIn ?machineTool) (?machineTool veonto:meetsProcessSpeedRequirement ?process)
Surface roughness verification	Process's minimum necessity for surface finish is considered and machine tools capable of machining with the required design specifications are selected from potential machine tools list.	(?process veonto:hasReqSurfaceFinish ?a) (?machineTool veonto:hasMaxSurfaceFinish ?x) greaterThan(?x ?a) → (?process veonto:surfaceFinishisObtainableFrom ?machineTool) (?machineTool veonto:hasSufficientSurfaceFinishCapabilities ?process)

Table 4.2: List of assigned rules for reasoning in VE ontology model (cont'd)

Rule Name	Rule Description	Sample Jena notation of rule
Controller resolution verification	Process's minimum necessity for machine tool resolution is considered and machine tools capable of machining with the required design specifications are selected from potential machine tools list.	(?process veonto:hasReqResolution ?a) (?machineTool veonto:hasMaxResolution ?x) greaterThan(?x ?a) $\rightarrow$ (?process veonto:resolutionObtainableFrom ?machineTool) (?machineTool veonto:hasResolutionCapabilities ?process)
Part weight verification	Checks part weight requirements of the machine tools. According to the machine tool structure and capabilities, work pieces weight is restricted. Therefore over weighted parts will be prevented from assigning to the unsuitable machine tools.	(?part veonto:hasWeight ?a) (?machineTool veonto:hasMaxPartWeightLimit ?x) greaterThan(?x ?a) $\rightarrow$ (?part veonto:lightEnoughtobeMachinedin ?machineTool) (?machineTool veonto:rigidEnoughtoMachine ?process)
Accuracy verification	Process's minimum necessity for accuracy is considered and machine tools capability to satisfy required design specifications are checked and appropriate machine tools are selected.	(?part veonto:hasAccuracy ?a) (?machineTool veonto:hasMaxAccuracy ?x) greaterThan(?x ?a) $\rightarrow$ (?part veonto:accuratelyMachinable ?machineTool) (?machineTool veonto:isAccurateEnoughtoMachine ?process)

4.4 Summary

In this chapter (4) developed ontology model for OMAVE system was discussed. This is one of the state of the art researches to develop an ontology model for VE systems. In developed model, all VE system components, their properties, relations and all semantic rules were considered and included. All system components, tools, applications and data store were established based on developed OMAVE ontology model. To enable system's reasoning engines to deduce new knowledge and information based on provided information and data, semantic rules were developed. Development of ontology model is the foundation of this research and system architecture, and performance is highly dependent on developed model efficiency and structure.

CHAPTER 5

MULTI AGENT BASED SYSTEMS

5.1 Literature Review of Multi Agent Based Systems Application in VE

Surveying literature for agent based approaches and their applications in virtual enterprise systems indicates that several studies accomplished to apply agent based systems in different phases of VE systems. Most of these studies have focused on partner selection and operation management steps of VE. A few of them also attempt to apply this infrastructure for collaborative design part of VE.

In a multi agent system, autonomous and self aware agents act independently. Each agent has its own specific target, follows its environment's conditions, and act accordingly to fulfill it's targets. Multi agent systems are very suitable to handle complex, and dynamic system environments like virtual enterprise platform. Agent interaction with each other and their environment go through an accepted language and protocols (like a grammar and vocabulary for their communication). The Foundation for Intelligent Physical Agents (FIPA)'s Agent Communication Language (ACL) and Knowledge Query Manipulation Language (KQML) are two types of these languages. In this section accomplished works in the field of multi agent systems in virtual enterprise systems are discussed. Implementation of agent based approaches in virtual enterprise systems can be roughly divided into 4 distinct categories:

5.1.1 Dynamic Scheduling and Agile Systems in VE

A mobile agent based architecture to support dynamic VEs introduced and its feasibility on required IT infrastructure including general services for trading, scheduling and ordering, of agents and of local integration components, called docks is studied. [2]

A holonic multi agent approach to improve scheduling flexibility in enterprises' shop floor management and its extension on the upper level between enterprises and holons is proposed in MASSYVE project and is called HOLOS multi agent scheduling system. In this method, multiple agents with different characteristics and targets are negotiating collaborating to find a solution for an enterprise's shop floor, inter holons or inter enterprises' scheduling problem.

Here, the multi agent system is composed of multiple autonomous, capable of interacting and independent processors which each of them are considered as an agent. HOLOS contains four type of agent classes [94] [95]:

1 Scheduling supervisor agent (SS)

2 Enterprise activity agent (EAA)

3 Local distribution centers agent(LDC)

4 Consortium agent (C)

By benefiting from Agent-cities agent platform inter operation platform and nested contract net protocol another holonic agent systems in order to integrate manufacturing and logistics service planning in dynamically changing configurations of virtual enterprises was developed by Karageorgos et al. [64].

5.1.2 Collaboration platform development: Inter enterprise knowledge sharing and system integration

Integration of business operations is one of the important and challenging issues in virtual enterprise concept. A multi agent based approach for business operations integration in VE is introduced by Gou et al. To integrate business operations, an agent based virtual enterprise model including two multi agent systems developed. Four type of agents (member enterprise agent, activity agents, role agents and resource agents) designed for this purpose. Based on Knowledge Interchange Format (KIF), which is a kind of knowledge representation language, distributed business processes model of VE and agent based VE Model integrated. Consequently agent based distributed business processes management system for VE developed [48].

To support inter enterprise resources and provide a platform for enterprises' collaboration through network a multi agent based system developed. In this approach, a hybrid agent architecture including four different types of agents; (Attribute Unit Agent (AU-Agent); Design Interface Module Agent (DIM-Agent); Service Wrapper Agent (SW-Agent); and Design Module Agent (DM-Agent)), to form a team oriented collaborating environment has been designed [83].

With the aim of creating inter and intra-enterprise work flow management system to be dynamic at runtime, an agent based web service work flow model developed by Wang et al. [126]. Web service work flow ontology evolves dynamically based on the changing situations and conditions. Developed model can be integrated will with heterogeneous enterprises' software and hardware systems [126].

To enhance inter organizational business processes, an agent based service oriented integration architecture (by targeting enterprises equipped with intelligent manufacturing infrastructures) developed by Shen et al. Researchers concentrated on implementing a service oriented computing paradigm on a developed unified framework to integrate software agents and web-services for inter enterprise resource sharing. Different agent types and system components were used in this platform. Agents communicate based on FIPA's ACL communication protocol and agent ontology provides semantic integration services for agents for service querying and reasoning [112].

XMLAYMOD is the name of an object oriented modular architecture empowered by STEP standard to enhance the collaboration between manufacturing agents. This platform, which is developed based on cloud computing paradigm, is introduced to support globe wide spread manufacturing agents collaboration and maintain CAx system data in XML data format [119].

A peer to peer service discovery framework called VPeers on top of smart agents has been developed by Xiang et al. [129]. This system designed in a decentralized way and all members could share their services or find required services using smart agents. Service documents are developed based on XML formats and offers service qualities, service query engine ranks found services for the customer [129].

An ontology based model of a manufacturing execution system (MES) with semantic web technologies is developed to enhance inter enterprise knowledge sharing of semantic information. This platform is a mediate ontology model to interpret and translate different engineering information terminologies between enterprises MES ontologies [72].

5.1.3 Partner Selection and VE formation process problems

A game theoretic multi agent approach for effective management of VE formation negotiation process is offered by Kaihara et al. [64]. In proposed system each potential partner enterprise is delegated by a software agent with multi utilities. The negotiation process in this method is programmed through game theory and contract net protocol (CNP) is applied as agents' coordination and negotiation mechanism. Negotiation process is undertaking looking to three main factors: cost; lead time and quality. Target is to minimize cost and lead time and maximize quality. 5 different methods were applied and authors claim that the game theory approach increases VE negotiation process stability [64].

Three types of heterogeneous agents (called; producer, intermediate and customer agents) interacts with each other to set proposing prices according to the complex virtual market provided for the system. It is observed that the micro behavior of agents emerges out a macro order of modeled virtual market and the results of this analysis are extended to the resource allocation problem of VEs [64].

To enable SMEs to respond quickly to customers product requests, a virtual CIM architecture is developed. In proposed architecture [124] three type of agents (namely, Customer, facilitator and resource agents) negotiate each other remotely. Based on information given from facilitator and resource agents according to the SMEs' capabilities, capacities, and customer agent's requirements best selections and task scheduling take place. Different algorithms and methods are used in different type of agents to reach the optimum result in task allocations and negotiation processes [124].

In order to assign tasks automatically between partner enterprises in a VE consortium, a multi agent based task assignment system was developed. Proposed system uses several agents (including product agent, task agent and resource agents). Project tasks are assigned based on enterprises active product processes in the assignment procedure [26].

In order to form up an ecological VE a multi agent system, based on ontology theory and intelligent agents has been established. Three types of intelligent agents (called; knowledge manager agent, manufacturer agent and supplier agent) are considered for this platform [127].

To improve the agility of manufacturing systems and to adopt with market changes and respond customer instantly it is required to plan and schedule for new production order as soon as possible. Hence, a currency based iterative agent bidding mechanism to reach a cost effectively optimized process plan and schedule was developed by Lim and Zhang [71]. To simplify bidding mechanism genetic algorithm (GA) techniques implemented to adjust currency values. It was shown in this study that utilizing this method the production cost of producing the components reduced gradually.

Multi-agent based architecture to support mobile and stationary agents' negotiation process introduced by Wang et al. [125] in this research a hybrid multi agent negotiation protocol to set up an agent based negotiation scheme, has been established. This ontology backed platform supports multilateral agent interactions for both mobile and stationary agents. In order to ease the agents' interaction an ontology based knowledge representation model has been developed.

One other research in VE formation stage is CONCOISE project [85]. In this project researcher apply agent based platform to form up a virtual organization insuring that contributing enterprises are agile (which means to be able to adapt themselves nimbly to the changing environment) and resilient (means to be able to manage to reach their goals in turbulent and uncertain environments). This system is working based on several types of agents and information resources and supervising enterprises activities and performances.

5.1.4 Operation management problems in VE

Agent based manufacturing control and coordination (AMCC) is an agent based framework which targets to monitor and control dynamic production processes using ontology and RFID technology. In this case, a reactive agent architecture (with following components; communication channel; agent memory; Message Parser; and Event Processor) eight types of agents; (order management agent(OMA); Production Monitoring agent (PMA); Warehouse

Management Agent (WMA); Kitting Management Agent (KMA); Product Agent (PA); RFID Event processing agent (REA); ERP Interface Agent (ERP-IA); and SCM Interface Agent (SCM-IA)) are collaborating and interacting with each other using block like representation of interactive components (BRIC) formalism. Agents of this model are developed based on JADE platform. For knowledge sharing and effective communication between agents an agent ontology model proposed in this method [24].

Concentrating on high volume gun drilling process an intelligent agent architecture capable of learning for process optimization has been introduced. In this architecture agents using machine learning algorithms and by using regression models, analyze and select manufacturing parameters to reach the optimal trade off between economic and technique factors [68].

It seems that integrating semantic and machine interpretable architecture by developing ontologies for agents, development of intelligent multi agent based negotiation process for partner selection stage of VE are missing researches in VE systems. The only integration between multi agent system and ontology model is accomplished in [112] in order to enhance service sharing and discovery in VE systems.

In this dissertation a multi agent based multi criteria decision making system is developed for partner selection stage of OMAVE system. In section (5.3) proposed agent based structure of OMAVE system will be described in details.

5.2 Agents Negotiation

5.2.1 Auction Types

The process of buying or selling services, goods to the highest or lowest bids coming from bidders is called auction process. Different types of auctions are available in literature but these types are classified into four main categories by explanation follows; [69].

English Auction

This is an open auction process for ascending price during the auction. It means that bidder offer higher bid in the each round of auction. This type of auctioning has two different variant. In one type, bidders start for bidding and starting price comes from bidders. In other version auctioneer conduct an opening price and bidders increase bids later. Bidding operation continues until only one bidder remains.

Dutch Auction

Dutch auction can be seen as the opposite type of the English auction. Dutch auction is price descending auction where bidders try to decrease their bids in each iteration of bidding operation. In Dutch auction, auctioneer starts bidding process by calling out a starting price and bidders try to offer lower prices. Bidding ends if only one bidder remains.

The Sealed-Bid First-Price Auction

In this auctioning procedure, bidders submit their bids once in sealed envelopes. Auctioneer opens bids and calls the highest bid and winner. The winner paying the proposed amount and buys the bidding stuff.

The Sealed-Bid Second-Price Auction

Like the Sealed-Bid First-Price, all bidders put their bids sealed in envelopes. Auctioneer open envelopes and call winner which is the highest bid owner. Up to here everything was like sealed-bid first-price auction, but here winner goes to pay the second highest bid offered in the auctioning to buy the stuff or service they bid for. The other name of this auction type is Vickrey auction.

Reverse Auction

In a reverse auction the place of buyers and sellers in reversed. Here, sellers as suppliers put their bids to get buyers interests. Buyers have opportunity to select the most appropriate offer from suppliers and buy the service or stuff for a suitable lower price [110].

These are different auction types based on price specification. However, another important criteria in auctions is time. The time duration required for putting bids or auction period. Every auction has its own duration and time specifications. For example, in Sealed-Bid First or Second Price auction, bidders prepare their bids after calling for bids. Auctioneer opens the bids and announce the winner and finalize the process [69]. On the other hand in English, Dutch or reverse auctions, auctioneer begins the negotiation procedure then bidders put their bids in an extended period of time. In the second format more time is required obviously but the final price could reflect more realistic market value of the auctioning service or stuff. However, in the first format time is saved but price could not be near the market value [111].

Based on given definitions about auctioning types and their specifications, in OMAVE partner selection auction process, a hybrid system of *'Reverse Auction'* and *'English Auction'* are used. In this research, to get the customer offer, an English auction concept is applied. The aim is to get maximum possible price from customer. English auctions complete conditions and predefined concepts could not be implemented here, because only one customer is participating in this negotiation and this customer price increment is a function of customer negotiation strategy, defined price limitations and definitely price changes in enterprise side. But in enterprise side a typical reverse auction conditions is established, supplier (here enterprises), are competing to offer best minimum price for the auctioning stuff (product or service) to get the job. Auctioneer here is task manager agent which manages both auctions simultaneously and continuously compares both sides offered bids and it ends negotiation if;

- Enterprise prices' matches customer offered price or
- There be no more remaining enterprise agent for negotiation.

In the first case the winner enterprise is called out by task manager, but in second case the auction is ended without any results.

5.2.2 Communication Language and Protocols

The Foundation for Intelligent Physical Agents (FIPA) established on 1996 and is a non-profit association to develop a collection of standards relating to software agent technology. The core principals of FIPA are as follows [9];

1) Provide agent technologies to solve problems
2) Maturate agent technologies
3) Standardize agent technologies
4) Standardization of generic technologies by cooperation with other
5) Standardization of agent communication languages, infrastructure required for open inter operation

During the time, FIPA evolved. Several new technologies and infrastructures have been also introduced. Some of these technologies maturated and reached standardization levels but the others development remained incomplete. Main achievements of FIPA is in the fields of agent communication, management and architecture [9].

5.2.2.1 Agent Communication Language (ACL)

FIPA- Agent Communication Language (FIPA-ACL) is FIPAs standard for agents communicative acts, speech acts or performatives [9]. ACL containing 22 communicative acts based on ARCOL primarily proposed by France Telecom. These 22 communicative acts describe an action in a reporting form with formal semantics. The most commonly used communicative acts are inform, request, agree, refuse, or not understood. For different auction types FIPA defined several interaction protocols and communicative acts like contract net for establishing agreements [9].

5.2.2.2 FIPA Architecture

FIPA communication stack has a multi layer structure within application layer of classical Open Systems Interconnection Model (OSI) or TCI/IP stack. These layers are as follow:

Sub-layer 1 Transport

FIPA's Defined message transport protocols for IIOP[1] , WAP[2],and HTTP[3] are in this layer.

[1] GIOP is a protocol for Object Request Broker (ORBs) communication developed and maintained by OMG group (UML diagrams developer and maintainer). IIOP is a platform independent Nutshell that implements GIOP to use over the internet. It maps GIOP messages to TCP/IP layer and vice verse. [36]

[2] Wireless Application Protocol is a communication protocol suite for inter operability of Wireless Access point devices, and software with different network technologies [89]

[3] HyperText Transfer protocol is communication protocol developed to enable and improve communication between clients and servers [37]

Sub-layer 2 Encoding

FIPA message does not have simple bit encoded message contents. FIPA uses high level data structures for encoding messages like XML, String and Bit-Efficient encoding standards. Bit-Efficient data structure is specially for low bandwidth connections.

Sub-layer 3 Messaging

In order to enhance the flexibility in sending and receiving messages, FIPA does not have special restricted message template and formats. However, it needs some key parameters to enable machine to interpret message content. These parameters are like payload for exchange content, sender, receiver for recognizing sender and receiver, message type, time outs for replying the message and etc.

Sub-layer 4 Ontology

To make message payload content machine readable, a common grammar and vocabulary is required to be shared through FIPA hierarchy. FIPA mostly allows using domain specific ontologies with any representation format to be implemented.

Sub-layer 5 Content expression

To embed logical formulas, predicates or algebraic content in FIPA messages, there are defined instructions. The language, which is mostly used in this case, is FIPA-SL. Some of FIPA-SL logic formula is like: *or, not, implies, equiv, ...* and for algebraic examples are *any, all and etc.*.

Sub-layer 6 Communicative act

Some of the communicative acts that distinguish the type of act which is (actions or preformatives) are; *inform, request, agree*.

Sub-layer 7 Interaction protocol or IP

Interaction between agents and platform and agents is done based on some defined interaction protocols. These protocols explain message exchange sequences and their contents type. For example, a communicative act request message meaning that one party requests other party and in turn the party should agree, refuse to acquiesce.

5.2.2.3 Agent Management

Agent management is a framework to handle agents; existence, operation and management. This framework establishes a reference model for agents' creation, registration, location, communication, migration and operation. This system has several components which is shown in Figure 5.1

In designing a new multi agent system (according to the FIPA standards) following components must be included in the system. However, remaining parts and specifications completely depend on designer of the system. The components of this reference model are as follows;

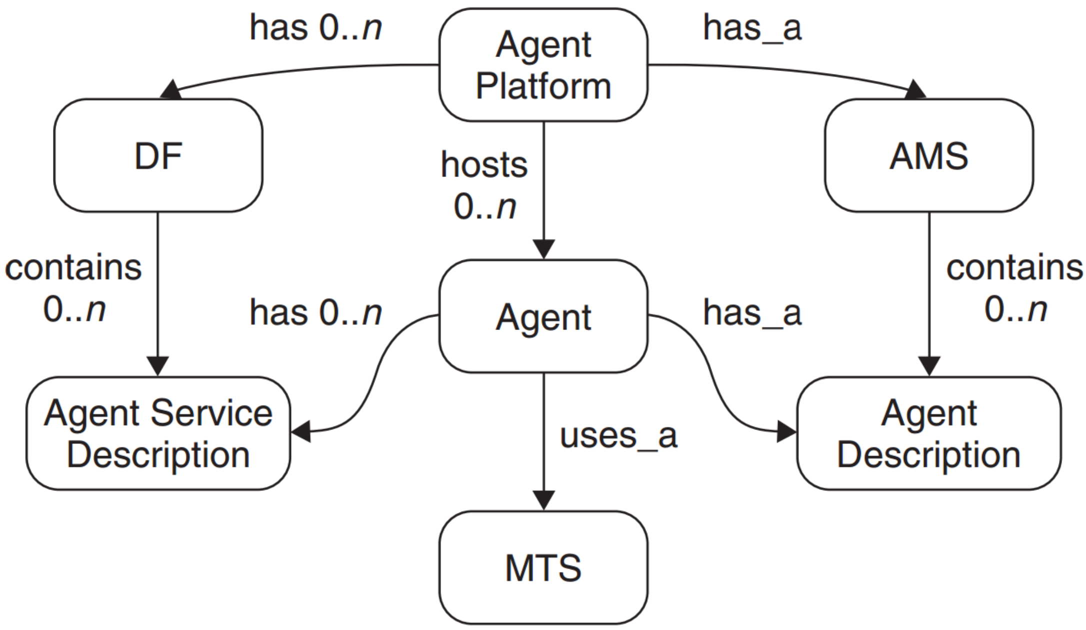


Figure 5.1: Agent Management Ontology [9]

Agent Platform (AP)

This platform is actually the base of the FIPA agent management platform and all other system components are deployed on this platform. AP consists of machines, operation systems, FIPA agent management components, agents, and any other designer specified tools and entities.

Agent

An agent is program which offers special services. FIPA does not care about agents design or what they are designed for. The only thing about agents which relates to FIPA, is the agents interaction and communication with their surrounding environment. FIPA labels an agent with an Agent Identifier (AID) to register agent on the platform and establish contact with agent if it is required.

Directory Facilitator (DF)

All agents require to introduce themselves and their publishing services to other agents to be discovered. Keeping registered agents IDs and their description regarding their services and other information is the duty of DF. At any time agents may modify their description, unregister or register themselves by requesting DF. DF publishes an updated list of agents and their descriptions that is called yellow pages. An Agent also may request DF to search for an agent with specific ID or services.

Agent Management System (AMS)

Creation, deleting, overseeing the agents migration from platforms to platforms and managing agents actions is the responsibility of the AMS component. All the agents have to register with an AMS to get a valid AID. The AMS can request an agent to

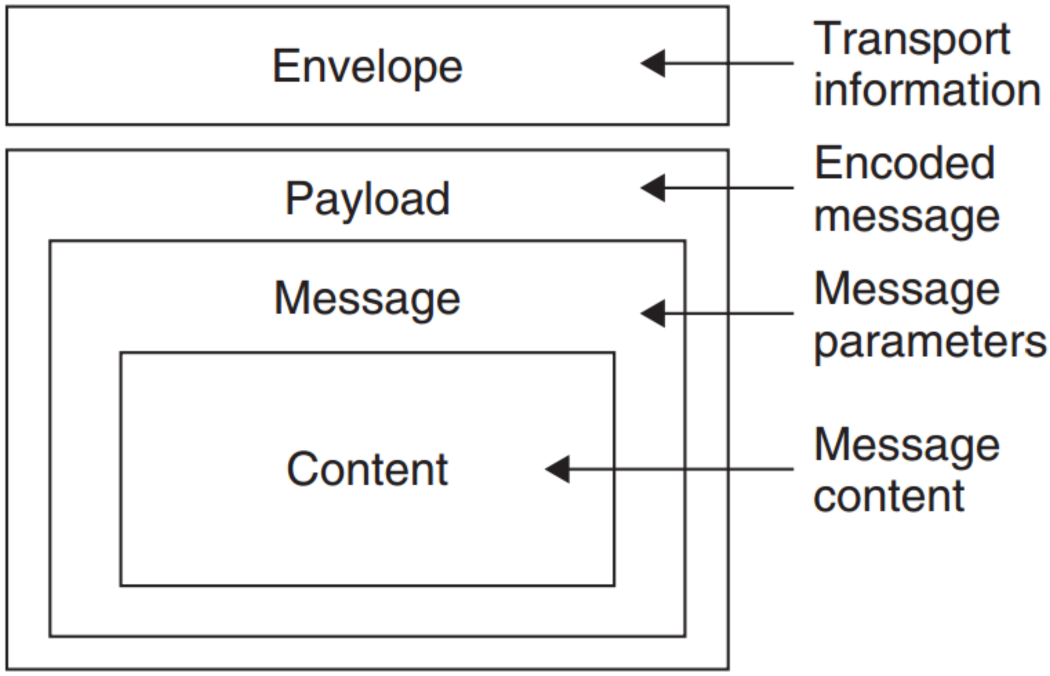


Figure 5.2: Agent Management Ontology [9]

perform a specific management function, such as to terminate its execution. It has the authority to enforce the operation if the request is ignored.

Message Transport Service (MTS)

MTS is provided by AP to transfer messages between agents. This is exactly the postal service companies do in our daily life. Agent message is to put in an envelope and the message is packed with special required parameters by MTS. These parameters are like, to whom and from whom, this message is sent, what is the payload and title of the message (Figure 5.2).

5.2.3 JADE Platform

JADE (Java Agent Development Platform) is a middleware to develop and run agents simply, with by a set of user friendly GUIs according to the FIPA specifications. JADE also provides platform independent inter machine agent distributions and interactions. JADE is completely developed based on Java language and the only requirement to run and apply this software is the utilization of java run time environment (JDK) version 5 or later [9].

JADE is a freeware and is developed and distributed by Italian telecommunication *Telecom Italia* which is the copyright holder, in open source under the terms and conditions of the LGPL (Lesser General Public License Version 2) license.

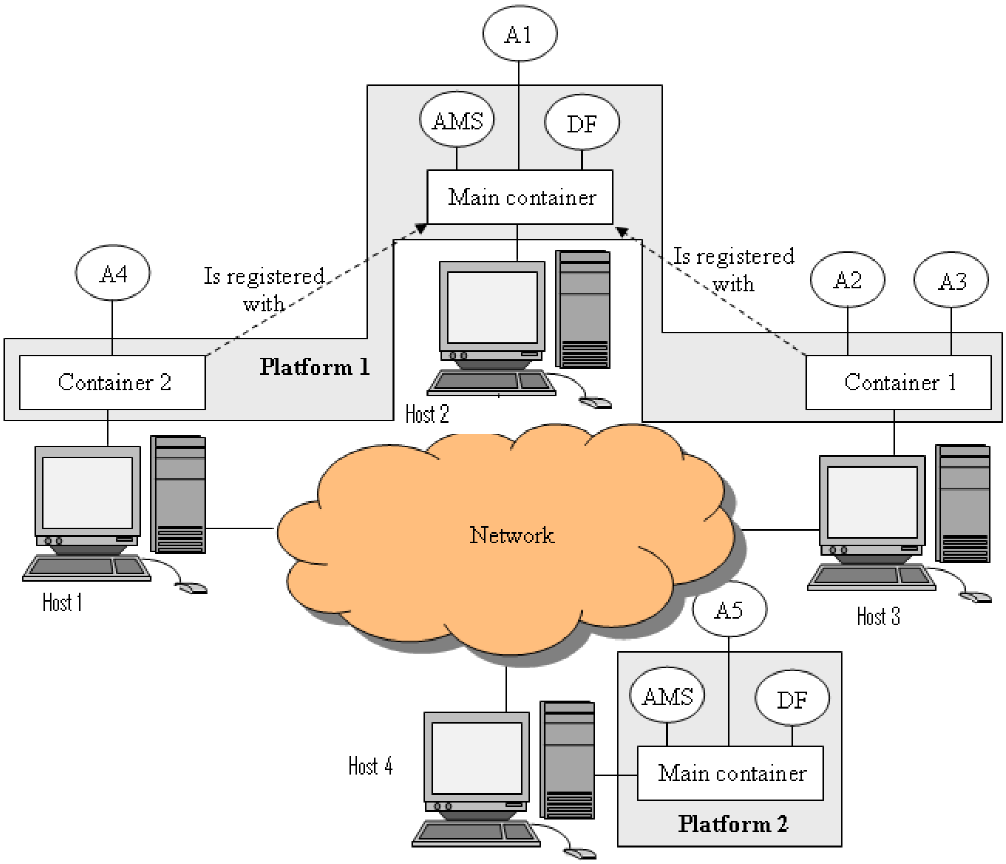


Figure 5.3: JADE Architecture [9]

Agents on JADE platform have message-based asynchronous communication. In order to start communicate agents should send a message to a or a set of destinations. There is no need to have a physical dependency between sender and receiver agents even receiver agents may not be available at time message is sent. Sender agent may even not recognize the identification of receiver agent but the message transport system send message and deliver it to the receiver agent.

In JADE architecture (Figure 5.3) agents live in a component called agent container. These containers are distributed on the network. Between containers there is an special container named *main container*. This is the starting point of JADE platform. All other containers deploy after main container and join it by registering themselves on the main container. These containers are registered by names like *'Container-1'* or *'Container-2'* [9].

Main container manages;

1 Registry of the object references and transport addresses of all container nodes containers which is called Container Table(CT).

2 Registry of all agents present in the platform and their information named Global Agent

Description Table (GADT)

3 AMS and DF are hosted on main container. These two special element are managing white and yellow page services.

Firing Main container JADE instantly initiates AMS and DF agents. AMS manages white pages in the system and in fact is the contact point of all agents. Agents life cycle is managed by AMS agent.

DF agent manages yellow pages services which is used by agent wishing to register their services or search for other services provided by other agents. It also notify agents if a new service compatible with their required needs is submitted to the system. If there several domains multiple DFs could be deployed and work concurrently. These DFs can be federated, if required, by establishing cross-registrations with one another which allow the propagation of agent requests across the entire federation [9].

JADE platform is also providing graphical user interfaces for developers however as in this research this specification of JADE is not used detailed information about user graphics are not given. For more information about JADE platform and FIPA Standards reader may refer to [9] Developing Multi-Agent Systems with JADE written by Fabio Bellifemine, Giovanni Caire and Dominic Greenwood.

5.3 OMAVE Agent Based Platform

Agents in developed OMAVE platform are commissioned to shoulder the negotiation procedure of partner selection stage. Here different types of agents backed by agent ontology and VE general domain model ontology, collaborate and compete with each other to reach an agreement. Here in this part of dissertation bidding procedure, agents role in this negotiation and also agents running algorithms are explained.

Following a new project submission, project is separated into assembly products containing parts. At this stage skilled engineers submit required manufacturing processes for each part and enter these information manually to the system. All these information are submitted into ontology model as triples. According to the created triples provided rules for the system ontology model, separate manufacturing processes into task groups. Similar manufacturing processes in series with same class axioms will be put into a single task group. In other words similar consecutive manufacturing processes placed in one task group. For instance, if two successive operation are turning operations and their required machine tool capabilities are similar, these tow processes form up a single task group. But if there is a milling operation between them these processes are separated into two different turning task groups.

After establishing task groups for a part, negotiation procedure may start for these task groups. Each task has its own specifications and requirements. Based on these prerequisites, list of enabler machine tools is created. These machine tool owners are related through *'is an asset* *of'* relation or reverse relation of *'owns*. According to these relations machine tools owner list also is established. For each task group, a list of enabler machine tools and their belonging list of owner enterprises are available. These process all are done automatically based on developed owl model's DL reasoning abilities and designed reasoning rules.

At this point system administrator may begin negotiation procedure by sending invitation for all selected potential partner enterprises from VBE and customer which is project owner. First of all, system automatically deploys Task Manager Agent which is responsible of supposed task negotiation process management and selecting winner enterprise for that task. In this stage, system automatically creates email templates according to the provided technical and legal agreement documents regarding entering bidding procedure. Responding these emails by clicking over provided hyperlinks in emails enterprises actually order to deploy their enterprise agent to enter negotiation process. Customer also follows the same procedure but at the end instead of enterprise agent a customer type agent with different hierarchy and characteristics is deployed.

Deployed enterprise and customer agents collect enterprise and customer data and information from VE triple store. VE triple store, in fact, is system shared database. Afterwards, agents ask for bidding information and data from their assigned enterprise. Detailed information about questionnaire and what kind of parameters are requested in these questionnaires are given in Section 6.

After obtaining required information, task manager agent check for the number of volunteer enterprises for entering the bidding procedure. If the number of bidder enterprises passes the threshold, task manager agent orders for commencement of negotiation. Enterprise agents gather required information from company authorities like; bid opening price, maximum bidding price and company strategy during the negotiation process. Based on the information, enterprise agent enters the negotiation procedure.

5.3.1 OMAVE Enterprise Agents

In negotiation procedure, enterprise agents are competing against each other to give the most competitive price to the task manager agent. In the negotiation process, all bids are sealed and enterprise agents are only informed about the best offer in the end of each iteration. According to enterprise agents information and incoming best bid of each iteration, agent recalculates the next iterations bid. The next bid from enterprise in OMAVE partner selection negotiation process, is calculated as(5.1);

$$a_i = \left(\frac{b_{i-1} + f(\alpha)}{2}\right) - E_{pp}.C_p.\left(\frac{b_{i-1} - f(\alpha)}{2}\right) \tag{5.1}$$

Where b_{i-1} is the best offered price in the last iteration, a_i is the next price in the bidding procedure (next iteration price of enterprise), and E_{pp} is the enterprise's past performance, C_p

Figure 5.4: Enterprise strategy point

symbolize how severe is the negotiation process for the company. $f(\alpha)$ is the price estimation formula for each company for the step. In this formula E_{pp} is obtained from system database and C_p is calculated as;

$$C_p = \frac{b_{i-1} - a_{min}}{a_{i-1} - a_{min}} \tag{5.2}$$

In equation (5.2) a_{min} is the minimum price of the company in the negotiation process. a_{i-1} is the last bidding price of the company in the last iteration. α and β are fixed factors for enterprise strategy. Enterprise strategy determines the policy and the related agent in the negotiation process. As shown in Figure 5.4 enterprise clarifies its strategy in negotiation. According to the enterprise selection α and β factors are determined in a way that $\alpha + \beta = 10$. As α increases more, enterprise's negotiation strategy becomes more aggressive.

In order to avoid radical bidding policies and agents' shocking reactions in VE negotiation process, which may lead to collapse negotiation in the very first steps, some prevention and stoppages are designed in the bidding procedure. At each step companies are allowed to bid in secure bidding range which is between a_{imin}, a_{imax}. a_{imin} is the minimum value that agents can bid for the next iteration likewise a_{imax} is the maximum value for bidding for agents. These values are calculated by equations 5.3 and 5.4;

$$a_{imin} = \frac{a_{min} + b_{i-1}}{2} \tag{5.3}$$

$$a_{imax} = \frac{3}{4} a_{min} \left(D_r + 2 \right) + b_{i-1} \frac{1 - 3D_r}{2} \tag{5.4}$$

Secure bidding range gap G could be calculated by;

$$G = a_{imax} - a_{imin} \tag{5.5}$$

Based on these relations, the formula for next bidding price of the enterprise could be driven. A third degree equation assumed for price estimation.

$$f(\alpha) = a\alpha^3 + b\alpha^2 + c\alpha + d \tag{5.6}$$

Considering boundary conditions;

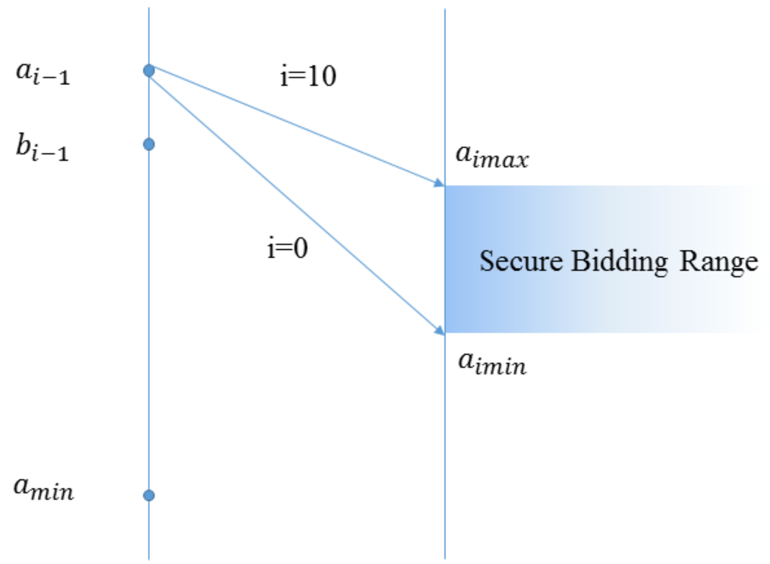


Figure 5.5: Possible Bidding band of enterprise agent for next bidding round

$$f(\alpha) = a\alpha^3 - 15a\alpha^2 + 75a\alpha + a_{imin} \tag{5.7}$$

Considering a_{imin} and a_{imax}, the possible bidding gap for agent for the next bidding round could be depicted like Figure 5.5

Now minimum and maximum values are replaced to find the value of *'a'*. If it is assumed that $a_{imnax} - a_{imin} = \gamma$ where γ, actually is the gap between a_{imax} and a_{imin} then;

$$a = \frac{a_{imax} - a_{imin}}{250} = \frac{\gamma}{250} = 0.004\gamma \tag{5.8}$$

Therefore, final form of the new pricing formula will be like the equation 5.9;

$$f(\alpha) = 0.004\gamma\alpha^3 - 0.06\gamma\alpha^2 + 0.3\gamma\alpha + a_{imin} \tag{5.9}$$

For each iteration for a_{imin} the relation is;

$$a_{imin} = \frac{a_{min} + b_{i-1}}{2} \tag{5.10}$$

Secure bidding range idea is proposed to prevent enterprises to bid aggressively and avoid them to put their minimum price in the very first stages of negotiation and broke system. Therefore, the minimum band of secure bidding range is designed to be the average of minimum price of enterprise and best price of the last iteration. From other hand enterprises have desires to hold prices as high as possible, then they will try to keep the price close to last iteration's best price. However, there should be a range to allow prices drop below the best price. In this case, a D_r constant is introduced to the system which is equal to a constant percentage of best price. This will be the minimum price decline ration that an enterprise should offer below the last iterations best price. For calculating a_{imax} the relation is;

$$a_{imax} = a_{min} + D_r \left(b_{i-1} - a_{imin} \right) \tag{5.11}$$

Therefore, γ becomes;

$$\gamma = \frac{1 - D_r}{2} \left(a_{min} - b_{i-1} \right) \tag{5.12}$$

Replacing these amounts in the main formula yields;

$$f(\alpha) = \left[\frac{1 - D_r}{2} \left(a_{min} - b_{i-1} \right) \right] \left[0.004\alpha^3 - 0.06\alpha^2 + 0.3\alpha \right] + \frac{a_{min} + b_{i-1}}{2} \tag{5.13}$$

Finally, in order to include E_{pp} and C_p factors to the final enterprise bid and calculate the bidding price of enterprise the following function is utilized;

$$a_i = \left(\frac{b_{i-1} + f(\alpha)}{2} \right) - E_{pp}.C_p. \left(\frac{b_{i-1} - f(\alpha)}{2} \right) \tag{5.14}$$

$$a_i = b_{i-1}. \left[\frac{1 - E_{pp}.C_p}{2} \right\} + f(\alpha). \left\{ \frac{1 + E_{pp}.C_p}{2} \right\} \tag{5.15}$$

Combining equations 5.13 and 5.14 gives;

$$\begin{aligned} f(\alpha) = b_{i-1}. &\left\{ \frac{1 - E_{pp}.C_p}{2} \right\} + \\ &+ \left\{ \left\{ \frac{1 - D_r}{2} \left(a_{min} - b_{i-1} \right) \right\} \left[0.004\alpha^3 - 0.06\alpha^2 + 0.3\alpha \right] + \frac{a_{min} + b_{i-1}}{2} \right\}. \\ &\qquad . \left\{ \frac{1 + E_{pp}.C_p}{2} \right\} \end{aligned}$$

In next step, bidding proposals, quality and past performance scores are sent to task manager agent. These scores will be used to evaluate the candidates by applying a logical partner selection algorithm which will be described technically in chapter 6. The output of this algorithm is the ranked list of candidates. In the case of any incoming new bid from agents the negotiation procedure will continue, otherwise it stops and final ranking is declared by task manager agent and winner is invited to sign the agreement.

5.3.2 OMAVE Customer Agent

Customer agent bidding policy is based on the average of last enterprise agents' bids. The new proposed price by customer agent is related to average of enterprises bids. As all agents prices are sealed and other agents are not able to see the information of other agents, customer agent behavior regarding enterprise agents faces a dead end. In order to give required information to customer agent, at each iteration, the average of incoming bids from enterprise agents is revealed to the customer agent by task manager agent. Customer agent, according to new proposed average price, sets its new offer for the next iteration. Setting a new customer default price is important because, both sides should sacrifice from their benefits to reach an agreement between customer and bidding enterprise agents. If one of bidder enterprise agents catch the customer offered new price, the negotiation will be stopped and the winner enterprise, which is the winner agent's corresponding enterprise, will be announced by task manager agent. A mechanism should be considered to encourage customer agent to increase proposal and prevent a dead end negotiation. In this way, like enterprise agents a new constant is defined for customer agents. But this time instead of α, customer agent's eagerness to change the price and to set customer agent strategy in negotiation process is represented by constant δ. For this purpose;

if δ = 0 then Customer agent price change rate is set on 0%
if δ = 10 then Customer agent price change rate is set on 100%

The logic behind the actions of customer enterprise to alter its proposed price is as follows: First in equation 5.16 enterprise agents' bids overall for each iteration is calculated;

$$\forall i : T_b^i = \sum_{j=1}^{m} E_{aj}^{\ i} = E_{a1}^{\ i} + E_{a2}^{\ i} + E_{a3}^{\ i} + \cdots + E_{am}^{\ i} \tag{5.16}$$

Here for $i^t h$ iteration the total of bids coming from m enterprises is calculated. Then the average of bids for this iteration is considered (5.17);

$$\forall i : A_b^i = \frac{1}{m} T_b^i = \frac{1}{m} \sum_{j=1}^{m} E_{aj}^{\ i} \tag{5.17}$$

For two consequent iterations the average price change rate can be found from equation 5.18;

$$\theta = \frac{A_b^{i-1} - A_b^i}{A_b^{i-1}} \tag{5.18}$$

As the rate of enterprise agents' price change increases, customer tries to decrease the change of rate in price and vice versa. Therefore customer agent price change rate should be proportional to the reverse of enterprise average price changes;

$$\bar{\theta} = 100 - \theta \tag{5.19}$$

From equation 5.19 the next price proposed by customer agent can be obtained from;

$$C^i = \bar{\theta}.\delta.\left(C_{max} - C^{i-1}\right) + C^{i-1} \tag{5.20}$$

5.3.3 OMAVE Task Manager Agent

Customer agent behavior is highly dependent on incoming bids from enterprise agents. In other words, customer agent inspects enterprise agents' acts and behaviors, then take appropriate actions accordingly. But it should be considered that customer agent can not see incoming bids from enterprise agents. Then, the question to be raised is, how it may act accordingly? The answer for this question is *Task Manager Agent*. Task manager agent is a bridge between customer agent and enterprise agents. It manage the negotiation procedure. Task manager agent collect all bids from enterprise agents and it also get the customer agent offer. Later on it first finds the minimum price offered by the enterprise agents and compare this price with customer offered maximum price. If Best Enterprise Bid is less than or equal to Customer Offer then task manager agent stop the bidding procedure, declare winning enterprise agent, rank all bids and offers enterprise bids to the management board. If this is not the case and Best Enterprise Bid is greater than Customer Offer then negotiation continues and task manager agent announce best bid of the last iteration to all enterprise agents in the bidding and ask for new bids. Besides it also calculates average bid of last iteration and announces this average to the customer agent to decide about its next round offer.

5.4 Summary

In Chapter 5, developed multi agent based system for OMAVE system was comprehensively discussed. As a summary following issues could be pointed out as main objectives of this chapter. First, to enhance the efficiency of partner selection phase of OMAVE system, a hybrid multi agent based auctioning process is proposed. The main target to propose such a system was to develop a neutral, reliable negotiation platform for OMAVE partner selection phase. To reach this aim, different types of agents; enterprise, customer, task manager and project manager agents were developed. Another important approach in this system is development of agent ontology to provide required concepts, rules and vocabulary for agents. This ontology is different than developed main OMAVE ontology model. Agents' interactions, behaviors and their understanding language is supported by developed agent ontology.

CHAPTER 6

OMAVE PARTNER SELECTION METHOD

Selecting the most appropriate partners to build a successful VE consortium is the most important step of VE formation phase. Since success of VE highly depends on the performance of its partners, selecting the best partners is essential. Partner selection problem is not a simple optimization problem due to the fact that generally it cannot be modeled with a single objective since the objectives of the problem are conflicting with each other. For instance, the high quality product usually offered for sale at high price. There is an inevitable trade-off between criteria and decision maker may need to sacrifice a criterion to gain some margin for the other criterion. This is why the partner selection problem is classified as a multi Criteria Decision Making (MCDM) problem. Based on the definition, MCDM approaches are the decision supporting tools allowing the decision maker to construct and solve the problems involving multiple conflicting criteria. Analytic Hierarchy process (AHP), Analytic Network process (ANP), Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) are among the most frequently used MCDM methods.
In MCDM problems, preferences of the decision maker should be specified clearly and precisely. Candidates are to be evaluated based on these preferences and their weights. Here, the problem is that human judgments are usually uncertain and the decision makers may not be sure about their preferences or interpreting the subjective expressions. Therefore, uncertainty is inescapable part of partner selection which should not be neglected.

The other important issue concerning the trustworthy of partner selection is, the evaluation criteria. In order to satisfy targets and get better results; the evaluation criteria should be selected properly. There are variety of factors affecting the performance score of companies. Neglecting a crucial criterion in evaluation phase may result in irrational outcome while including too many criteria leads the problem grow in size and complexity.

Including quantitative and qualitative criteria in decision making procedure is a challenging problem in evaluation process. Criteria such as price and delivery time, which can be represented with numbers, are quantitative criteria. On the other hand, there exist some intangible criteria such as communication skills and level of commitment, which are subjective in nature, called qualitative criteria. The definite value of qualitative criteria cannot easily expressed in numbers so including the qualitative criteria in mathematical models are not straightforward and needs extra effort.

6.1 OMAVE Partner Selection Algorithm

Among tens of evaluation parameters, the following list of criteria are selected to be included in partner selection algorithm. Enterprises are assessed based on following criteria:

- Company's efficiency
- Bidding price proposal
- Bidding delivery time proposal
- Company's background in terms of product quality
- Company's background in terms of delivering product on time
- Company's background in terms of communication skills and responsiveness
- Company's after sale service
- Environmental friendliness

Reliability of the model reduces because of increase in model complexity due to handling all the named criteria at the same time. Therefore, multi-stage algorithm is proposed for identifying, evaluating and ranking the enterprises. Proposed partner selection algorithm is schematically shown in Figure6.1.

In order to select the most appropriate partners for VE consortium and enhance chance of VE project success following steps are designed for selection process;

Manufacturing Requirement Elimination
The first stage is to identify the enterprises which are technically capable of fulfilling the task. To do this, qualified enterprises are selected from the pool of enterprises via applying SWRL rules.

Customer Prerequisite Elimination
At the second stage, specific prerequisite of customer, if any, is taken into account. For instance, a customer may look for a company with ISO9001 certificate, This stage eliminates the enterprises which do not obtained this certificate.

Data Envelopment Analyses (DEA) Elimination
To increase the system performance if number of potential partners is too high, one extra step to eliminate inefficient enterprises is added. In this case, enterprises with high efficiency score is permitted to participate in bidding. Efficiency check is conducted thru adopting DEA method.

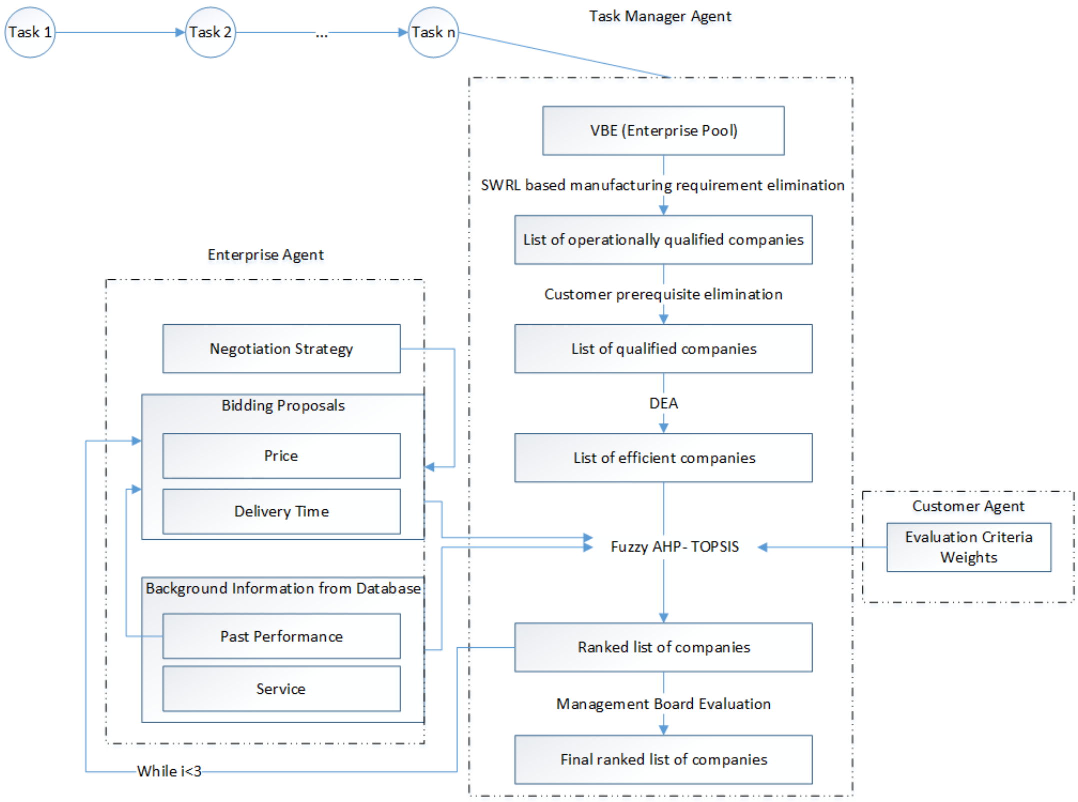


Figure 6.1: Proposed partner selection algorithm and agents interaction

Fuzzy Analytical Hierarchy Process(FAHP)-TOPSIS

Efficient enterprises are invited to propose for bid. Volunteer enterprises are evaluated considering their proposals and background performance. The evaluation process is based on customer preferences. This stage is the main decision making stage of the algorithm, several multi Criteria Decision making methods such as Fuzzy-AHP-TOPSIS and fuzzy logic is implemented and tested to validate the accuracy of the model.

Ranking

Evaluated enterprises, are ranked based on their gained points .

Management Board Final Approval

The list is presented to the management board, to get the final confirmation. Last, the winner of each task is announced.

6.2 OMAVE Partner Selection - Preliminary Eliminations

Preliminary manufacturing and customer requirement elimination steps are done using SWRL rules of OWL-DL reasoning engines. Only queries about required features of enterprises are considered and if the enterprise meets the requirements passes the filter. If not, it will be eliminated. A simple instance for a standard check rule is as follows;

(?x <http://www.ostim.org.tr/omave#hasStandard> ?y) ,
(?z <http://www.ostim.org.tr/omave#requireStandard> ?y)
$\rightarrow$
(?x <http://www.ostim.org.tr/omave#meetsRequirement> ?z)
'http:/www.ostim.org.tr/omave' is the ontology model URI and all model entities URI is added to the end of this URI. For example, for the relation *'hasStandard'* which is a object property, related URI is <http://www.ostim.org.tr/omave# hasStandard>. In this relation, *'x'* could be an enterprise and *'y'* could be a standard like ISO9000, and z is a project which requires this standard. If enterprise x fulfill this requirement, it may pass this filter.

6.3 OMAVE Partner Selection - Data Envelopment Analysis (DEA)

DEA is first proposed by Charnes and Cooper [23]. This method is used to empirically measure the efficiency of alternatives by deducing the efficiency frontier. DEA compares alternatives, based on their inputs and outputs, via an operation research method. The inputs of the units are all resources used and the outputs are services provided. An alternative which serves more output using less input is considered as the most efficient alternative. Applying DEA to partner selection algorithm will be beneficial in several aspects:

- Among list of criteria, some are chosen to be considered in DEA. Meanwhile, the number of criteria left for next stages of decision making become less and detailed evaluation of enterprises would be easier to handle.
- If there are too many candidates, DEA detects the inefficient enterprises and exclude them from the list so they lose the chance to participate in negotiation process. Referring to the literature most of the partner selection techniques are stuck when the number of candidates is large.
- Including just efficient enterprises in forming the VE increases the chance of satisfactory performance during the operation phase.

Adapting the bench marking idea in DEA, efficiency scores of each enterprise is obtained by comparing that enterprise with other alternatives. Therefore, the efficiency score of each alternative is sensitive not only to the performance of enterprise itself but also to the performance of its competitors. The reliability of the model highly depends on the accuracy of data. Even single inaccurate information may result in untrustworthy outcome.

As mentioned earlier, efficiency score of each candidate is calculated regarding its inputs and outputs. Among variety of assets of each enterprise, five main input are selected to be included in this model;

- Total Energy Consumption

- Total Machinery Value
- Total Area
- Total Human Resources
- Total Working Hours

Any company employs different types of inputs to acquire the output. The following two outputs are selected as representative of companies' outcomes.

- Total Sales Value
- Working Capital

The mathematical model of data envelopment analysis is solved via linear programming techniques. The objective of the model is to find the set of coefficients that gives the highest efficiency ratio for each alternative. Efficiency score is defined as the weighted sum of outputs over weighted sum of inputs. The primal mathematical model of the problem is nonlinear, though the dual can be modeled linearly as follows:

$$Max\ E = \sum_{k=1}^{K} Y_{ok}\nu_k \tag{6.1}$$

subject to:

$$\sum_{k=1}^{K} Y_{ok}\nu_k - \sum_{j=1}^{J} X_{ij}u_j \leq 0 \tag{6.2}$$

$$\sum_{j=1}^{J} X_{oj}u_j = 1 \tag{6.3}$$

$$\nu_k, u_j \geq 0 \tag{6.4}$$

I : Number of alternatives(enterprises)
J : Number of inputs (assumed to be 5)
K : Number of outputs (assumed to be 2)
E : Efficiency ratio of the alternative
X_{ij} : Amount of input j,used by alternative i
Y_{ik} : Amount of output k,generated by alternative i
u_j : Coefficient assigned by DEA to input j

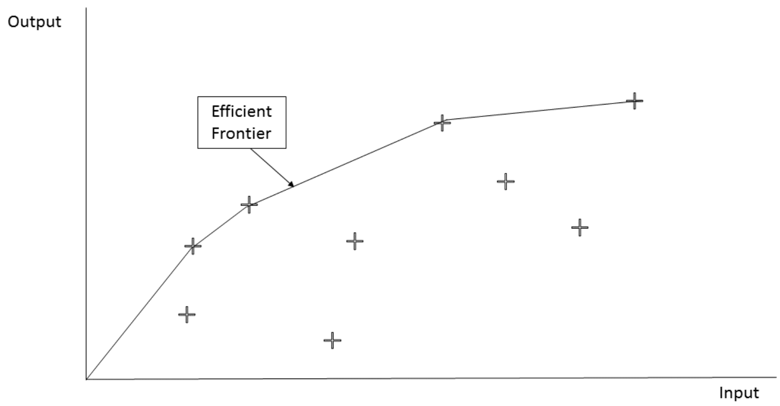


Figure 6.2: Efficient and inefficient enterprises

v_k : Coefficient assigned by DEA to output k

By implementing the mathematical model, efficient frontier alternatives are obtained and inefficient enterprises are detected and excluded from the enterprises list. Figure 6.2 shows the function of DEA method.

6.4 OMAVE Partner Selection - Fuzzy-AHP TOPSIS

The efficient enterprises are invited to participate in negotiation process. Considering the task properties and necessities, volunteer enterprises propose for bid. Bidding proposals are Price and delivery time. Beside these parameters, enterprise background is called from system database to perform an objective selection process. Since there are variety of both qualitative and quantitative criteria involved implementing a multi criteria decision making algorithm seems reasonable.

As mentioned earlier, due to the dynamic nature of VE, different customer attitudes should also be considered. Using Analytic Hierarchy Process (AHP) method would be beneficial to find out the customer preferences. AHP proposed first by Saaty et al. as a MCDM technique based on pairwise comparisons [118]. Evaluation criteria are arranged in hierarchy structure as 6.3

To find the relative importance of criteria comparisons are inquired from customer using questionnaire forms. Customer is only responsible for answering the questions containing pairwise comparison of importance of four main criteria; Price, delivery time, past performance, Service. Conventional AHP method uses crisp values to illustrate the preferences among criteria, neglecting the vagueness of data obtained. To overcome this issue, Fuzzy-AHP technique

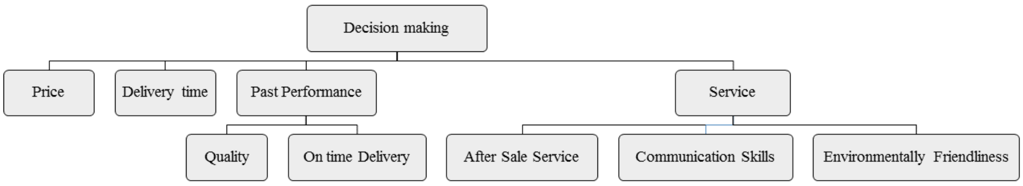


Figure 6.3: Criteria Hierarchy in OMAVE

Table 6.1: Pairwise comparisons of linguistic variables using fuzzy numbers

Linguistic scale for importance	Fuzzy numbers	Triangular fuzzy scale
Equally important	$\tilde{1}$	(1,1,3)
Weakly important	$\tilde{3}$	(1,3,5)
Strongly important	$\tilde{5}$	(3,5,7)
Very strongly important	$\tilde{7}$	(5,7,9)
Extremely important	$\tilde{9}$	(7,9,9)

proposed by Buckley (1985) is adapted [15]. Fuzzy-AHP is an extension of AHP which uses triangular fuzzy membership functions to deal with uncertainty rather than Saaty's crisp 1-9 scales [100]. Table 6.1 shows the linguistic terminations and their corresponding fuzzy numbers.

The evaluation matrix A is constructed through the pairwise comparisons of criteria based on table6.1. Matrix A is an $n \times n$ matrix where n is number of criteria.

$$\tilde{A} = \begin{bmatrix} \tilde{a_{11}} & \dots & \tilde{a_{1j}} & \dots & \tilde{a_{1n}} \\ \vdots & & & & \vdots \\ \tilde{a_{i1}} & \dots & \tilde{a_{ij}} & \dots & \tilde{a_{in}} \\ \vdots & & & & \vdots \\ \tilde{a_{n1}} & \dots & \tilde{a_{nj}} & \dots & \tilde{a_{nn}} \end{bmatrix} \tag{6.5}$$

In 6.5, $\tilde{a_{ij}} \bigodot \tilde{a_{ji}} = 1$. $\bigodot$ is matrix multiplication symbol and $\bigoplus$ is symbolizing matrix summation operation. Next, by applying geometric mean method fuzzy weights of each criterion is obtained as follows:

$$\tilde{w_i} = \tilde{u_i} \bigodot \left(\tilde{u_1} \bigoplus \tilde{u_2} \bigoplus \cdots \bigoplus \tilde{u_n}\right)^{-} 1 \tag{6.6}$$

where;

$$\tilde{u_i} = \left(\tilde{a_{i1}} \bigodot \tilde{a_{i2}} \bigodot \cdots \bigodot \tilde{a_{in}}\right)^{\frac{1}{n}} \tag{6.7}$$

Fuzzy weights are defuzzified by Center of Area (COA) defuzzification method so that the crisp weight of each criterion is concluded. These weights indicate the customer preferences and will be used in later in Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) method. The concept of TOPSIS is that the chosen alternative should be closest to the positive ideal solution and the farthest from negative ideal solution [99].

The performances of alternatives with respect to criteria are expressed in a matrix known as decision matrix *'X'*. Performance matrix is a $m \times n$ matrix where m is the number of enterprises associated with n number of criteria. The performance score of each criteria has its own unit, to eliminate the effects of unit normalization is essential. Normalized performance matrix is constructed as:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \tag{6.8}$$

As a result of multiplying normalized performance matrix by criteria weights (obtained from fuzzy-AHP), the weighted normalized performance matrix is obtained:

$$\nu_{ij} = w_j r_{ij} \tag{6.9}$$

The next step is to determine the Positive ideal solution (PIS) and negative ideal solution (NIS) taking into consideration Equations 6.10 and 6.11.

$$PIS = A^{+} = (max._i \nu_{ij} \mid j \in J), \left(max._i v_{ij} \mid j \in J'\right) \mid i = 1, 2, 3, \cdots, m \tag{6.10}$$

$$NIS = A^{-} = (max._i \nu_{ij} \mid j \in J), \left(max._i v_{ij} \mid j \in J'\right) \mid i = 1, 2, 3, \cdots, m \tag{6.11}$$

where; $J = \{j = 1, 2, \cdots, n \mid j$ associated with benefit criteria $\}$ and $J' = \{j = 1, 2, \cdots, n \mid j$ associated with cost criteria $\}$. The separation value of each alternative from PIS and NIS is measured by Euclidean distance as follows:

$$s_{i^+} = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_i{}^{+})^2} \tag{6.12}$$

$$C_i{}^{+} = \frac{S_i{}^{-}}{\left(S_i{}^{+} + S_i{}^{-}\right)} \tag{6.13}$$

where $C_i{}^{+} \in [0, 1]\ \forall i = 1, \cdots, n$. Preference order of alternatives is ranked according to the deceasing order of $C_i{}^{+}$. To sum up, Fuzzy-AHP-TOPSIS model which provides the ranked list of enterprises is based on implementation of customer preferences obtained from fuzzy-AHP method in TOPSIS's conventional model.

6.5 Management Board Decision

As a result of Fuzzy-AHP TOPSIS model the ranked list of enterprises are obtained and the winner is identified. The list is presented to the management board in order to get the final confirmation. Lastly the winner of each task is announced. Partner selection algorithm follows all the five stages for each task of the main project in order to select winner of each task. When all the partners responsible for fulfilling the tasks are determined, VE forms up and operation phase starts.

6.6 Summary

As a summary, proposed partner selection method (Fuzzy-AHP-TOPSIS hybrid method) did not applied in VE system any before. This is a new approach for VE partner selection which aims to select the most suitable partners for VE consortium. In this approach first, based on customer preferences using Fuzzy-AHP method weights for criteria are calculated. According to the obtained weights and applying TOPSIS method enterprises points are calculated. An enterprise with the highest point is selected as the winner enterprise, and other enterprises are ranked according to their gained points.

CHAPTER 7

INTEGRATION OF OMAVE SYSTEM COMPONENTS

7.1 Jena Platform

Jena is a java-based open source framework to develop semantic web applications. Jena provides an environment for coding and programming applications for RDF, RDFS, OWL, SPARQL, Gleaning Resource Descriptions from Dialects of Languages (GRDDL) and reason inference models by jena reasoning engines. Jena framework supports various internal reasoners as well as Pellet reasoner. Jena serializes graphs in RDF/XML, Turtle[1], N3[2] and relational database formats [39]. RDF statements or triples are containing three parts;

- subject which is the starting point of relation arc
- predicate which is the property that labels the arc
- and object which is the end point of arc. It could be resource or literal.

Arc could be assumed as a curved line which connects start point (Subject)to the end point (object) through a relation called predicate. RDF models are data structures containing RDF triples or model statements. These statements relate RDF Nodes with assigned relations in the model. RDF Graphs illustrate these nodes and their relations. RDF Nodes are in fact ontology model resources and entities and the relations between them are ontology model properties.

Jena platform provides several tools and Application Programming Interface (APIs) to handle RDF and OWL files and models like, create, write, read, edit RDF and OWL models, control prefixes, query a model and operate on the models. The interaction between Jena APIs is shown in Figure 7.1 [39].

For example, here there is a model creation code in Jena;

```
    OntModel m = ModelFactory.createOntologyModel(OntModelSpec.OWL_MEM);
```

[1] Turtle is a compact text form of RDF graphs [7]
[2] non-XML more compact and human readable format of RDF models [10]

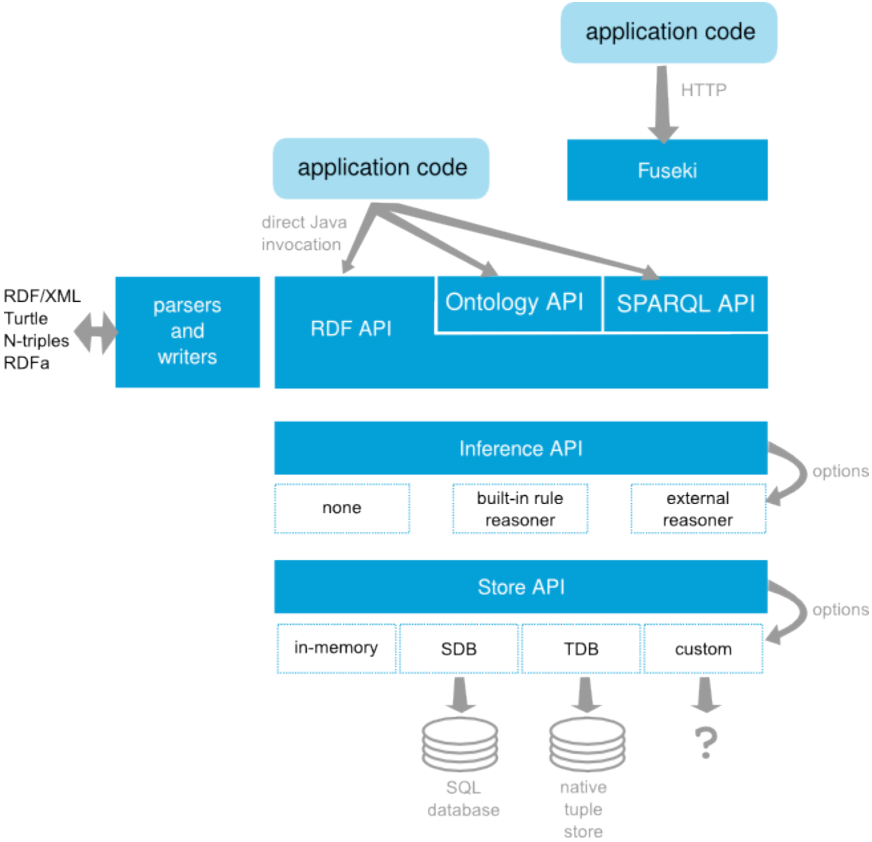


Figure 7.1: Apache Jena APIs Interaction [39]

Part of code that used to create and load the VE ontology model to TDB triple data store in Jena is shown in A. This part of code shows that the created model is going to be loaded to TDB triple store. In the following section(7.1.1) more detailed information about TDB data stores are given [39].

7.1.1 TDB Triple Data Store

TDB is a Jena platform component, which deals with store and query over RDF data structures. Using Jena API, TDB store is accessible and manageable. Benefiting from write-ahead-logging transaction, TDB data set is protected against corruption, system crashes and unexpected process terminations. Regarding TDB data sets there is a restriction. Access to these data sets is possible only from one Java Virtual Machine (JVM) at the same time Otherwise corruption may occur. Early versions of TDB is automatically protected from multi JCM connections. If multiple applications have to use TDB Jena fuskei component usage is recommended by apache-Jena [39].

TDB datasets do consist of the following conditions;

Node Table

Node table keeps RDF model nodes information with two mapping. One mapping is from Node to NodeId which is used in loading Jena RDF data to TDB store and the second one is NodeId to Node, is employed during a query execution on TDB stores to reach to Jena RDF data.

Triple and Quad Indexes
Default graphs are kept in triple indexes which includes 3 NodeIds for a Jena statement. in named graphs quads are used.

Prefix Table
For serializing and presenting Jena statements as triples in RDF/XML or turtle format this prefixes table is used.

As TDB data sets are stored as triples (or statements, in order to query over these type of data sets) a semantic query language called Protocol and RDF Query Language (SPARQL) is used. Users may run queries over nonSQL RDF databases in the shape of triples (subject, predicate, object). It is possible to represent RDF data in normal SQL relational databases. In that case, it is a table with three columns; the subject column, the predicate column and the object column [39].

7.1.2 Comparing VE system with TDB triple data store and SQL relational database

In order to compare the system performance of relational Structured Query Language (SQL) database base and SPARQL based systems, VE ontology model mapped to a SQL relational database. SQL database has a single table with three columns (OMAVE-SQL) and it was compared to the normal SPARQL based VE system (OMAVE-SPARQL) performance. To compare two systems' performances different criteria; Read, write and query of data were considered. In OMAVE-SPARQL reading, writing and query operations are realized by SPARQL language. In OMAVE-SQL, query language is Structured Query Language (SQL). Reading and data search performance of OMAVE-SQL is relatively better than OMAVE-SPARQL especially when data volume is large. This is not unexpected issue. Triple store is a young and somehow immature technology therefore in order to support systems with larger database different types of data stores like Fuseki and etc are used.

But in writing operation there is a problem in OMAVE-SQL. When it comes to write information, by writing new inputs to table, the backing model remains outdated and new data do not transfer to model. If data are written directly to the model, SQL table remains outdated. Third way is to write data to model and then transfer new added statements to data table. However in this method too much time is needed to check the model and fin new added statements. This method is not reasonable to be applied for larger models. By considering these conditions continuing to use OMAVE-SPARQL seems reasonable until the end of pilot test bed. Later, in the next phase, a new algorithm to write data directly to SQL tables in the form of triple statements will be created and the model will be updated periodically. This procedure enhances system performance and keeps system model updated.

7.2 Virtual Enterprise System and Applications

As described in the last section (7.1) Jena was the main platform to develop, OMAVE system tools, and applications. For this purpose different type of tools, dashboards and applications for monitoring, add, remove or edit system information, Reasoning, managing system components were developed. Here in this section each system component and their functionality are outlined.

Information Monitoring
: This part of system is designed to illustrate different entities conditions and monitor updated information. Retrieved data are directly coming from TDB data store which is updating continuously according to the system model. All predicted OMAVE categories and their members (model individuals) are available here, and system administrator may have access to all model information without engaging complicated ontology and Jena platform characteristics.

Model Modification
: System user may add,remove or edit all information on the model bu using simple and user friendly graphical user interfaces. It is available also for system administrator to reconfigure even model architecture, add, remove classes or properties based on new requirements.

Search Engines
: After system test and verification step, a big data volume is going to be entered in this system. This may cause lots of problems for system administrator or even users to find their required data and information by only looking up to the information monitoring tables. Therefore in order to solve this problem, two types of search engines were developed for OMAVE system.

System Management
: This system component is developed to manage projects, parts, tasks, and multi agent based partner selection processes.

Additional Components
: There are available infrastructure to add new features to the system and control panels for new added components in this system. One of the these components is eco-friendly manufacturing and ecologic performance analyzer of enterprises. Energy, waste water, hazardous materials, waste treatment infrastructures and Supervisory Control and Data Acquisition (SCADA) system inputs are already designed and established in the model. Consequently they are available in Jena to be integrated with required dashboards.

The other component is systems integration platform for Manufacturing Execution Systems (MES), Enterprise Resource Planning (ERP) and Product Life cycle Management (PLM). These features also could be integrated to the main model through modeled class axioms. Manufacturing processes and machine tool abilities and capacity information nodes and properties are already set on the system but for now these information are entered to system manually. But if there is a suitable automated ERP or MES system in companies it could be easily integrated to this platform through VE ontology model, any tabular or XML based data stores or even SQL relational databases.

As mentioned before this system's main characteristic is its high flexibility and platform independent architecture, which offer services to the users based on software as a service (SaaS) concept, through developed web services and agents.

7.2.1 OMAVE Information Monitoring

Information monitoring was designed to display updated data and knowledge about system entities to the system administrator and users. Depicted information here are directly fetched from the TDB data store. The important issue here is that the retrieved data from the TDB data store is not based on any known indexes like SQL relational data bases. Based on user selection, system instantly refers to TDB store and depending on the selected object, scans data store for any object or relation related to the selection. Then it classifies and display final results through user friendly Graphical User Interfaces (GUIs) to the user.

As search and scanning the model is based on uncertainties, OMAVE ontology model must be developed in a very generic and flexible way to respond all possible types of searches and data. This is the key point in designing OMAVE ontology model. For instance, number and type of properties and relations for a SME is completely different from a machine tool, infrastructure or etc. Even in a same category, number of different individuals assigned properties also differ from one another.

As depicted in Figure 7.2 from left menu bar, the type of category is selected. If there are any sub categories a query for user selection subcategories started and at the same time system is searching for the selected categories individuals. The SPARQL query for finding any available subcategories is illustrated in Appendix B. Referring the code, it can be seen that the population of left side bar is completely depend on the model data and knowledge at the time the list is going to be populated, type and name of classes and their belonging individuals could be changed at any time. Code sample in Appendix C is part of the query to find out the selected classes assigned properties. These properties in Figure 7.2 are the columns in front of enterprise names such as; Acceptance Rate, Address, After Sale Service and etc. Appendix D is a sample code to query for list of individuals in a class in owl model.

The same queries go for a new node selection Figure 7.3. This time, the selected node is a Multi tasking machine tool class. There are two individuals in this class, *Mazak Integrex J-200* and *Mazak Integrex i-200 ST* with different list of properties. For manufacturing processes also different characteristics and properties is assigned as it is shown in Figure 7.4.

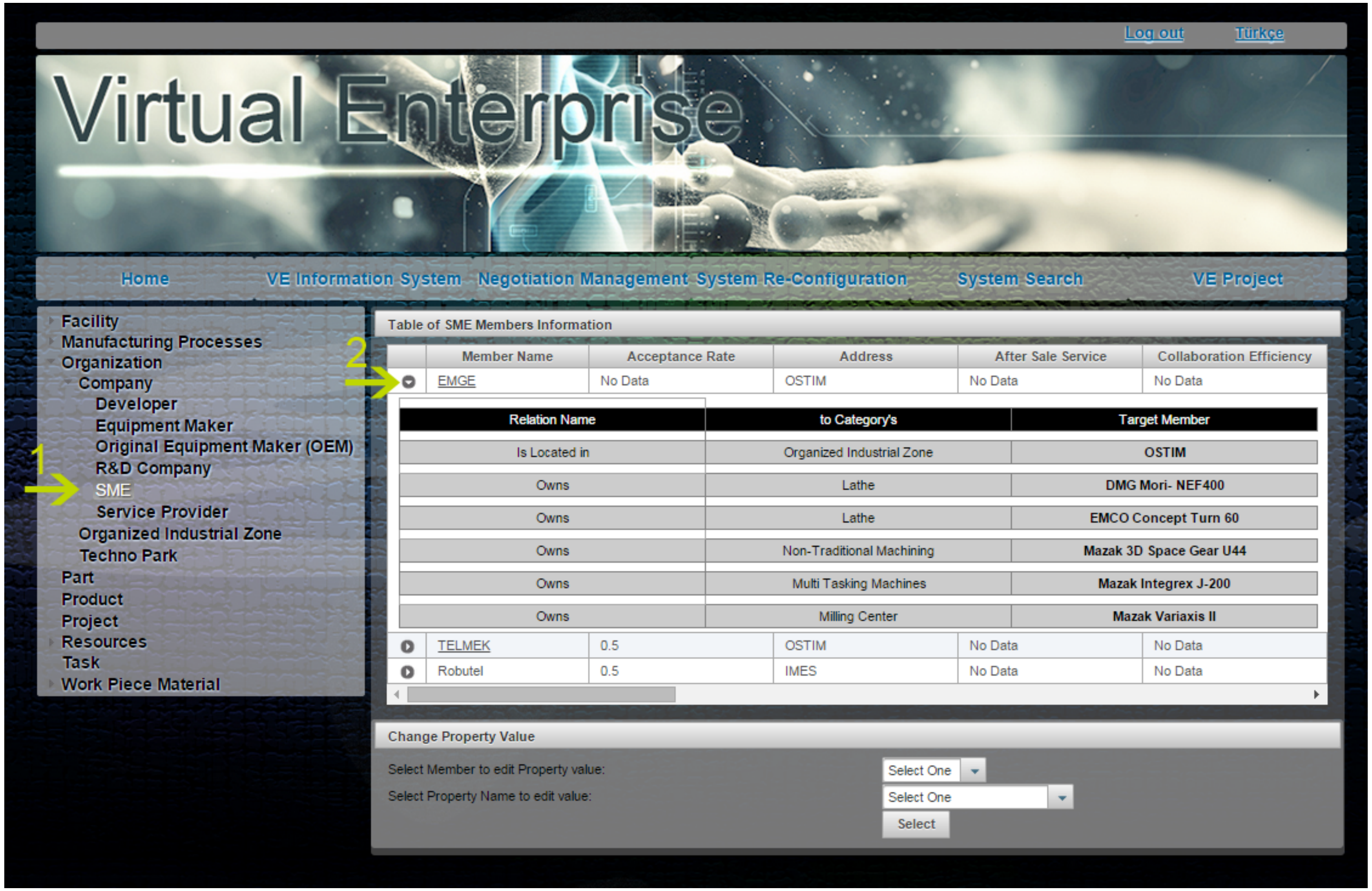


Figure 7.2: SME information screen

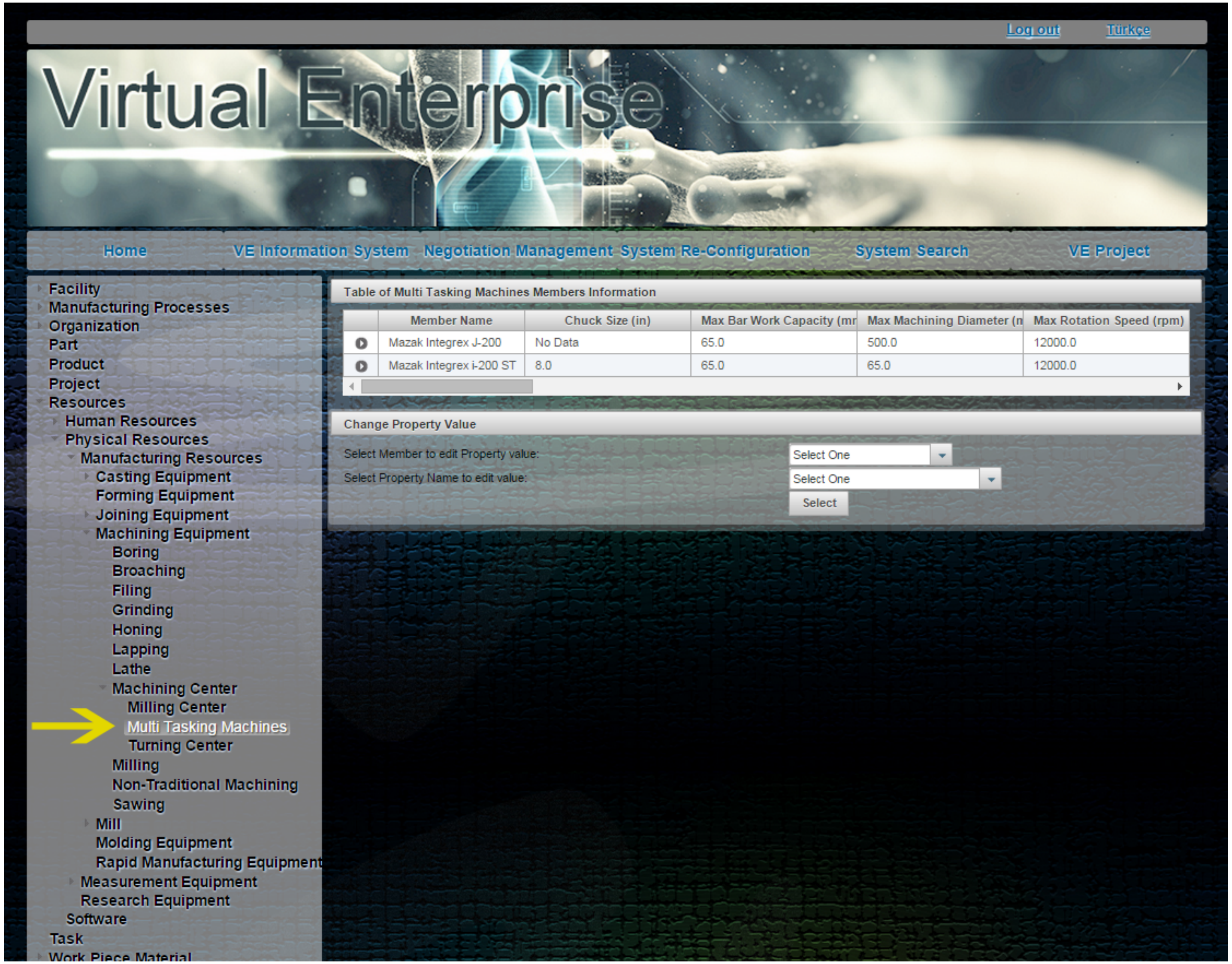


Figure 7.3: Multi tasking machine tool information screen

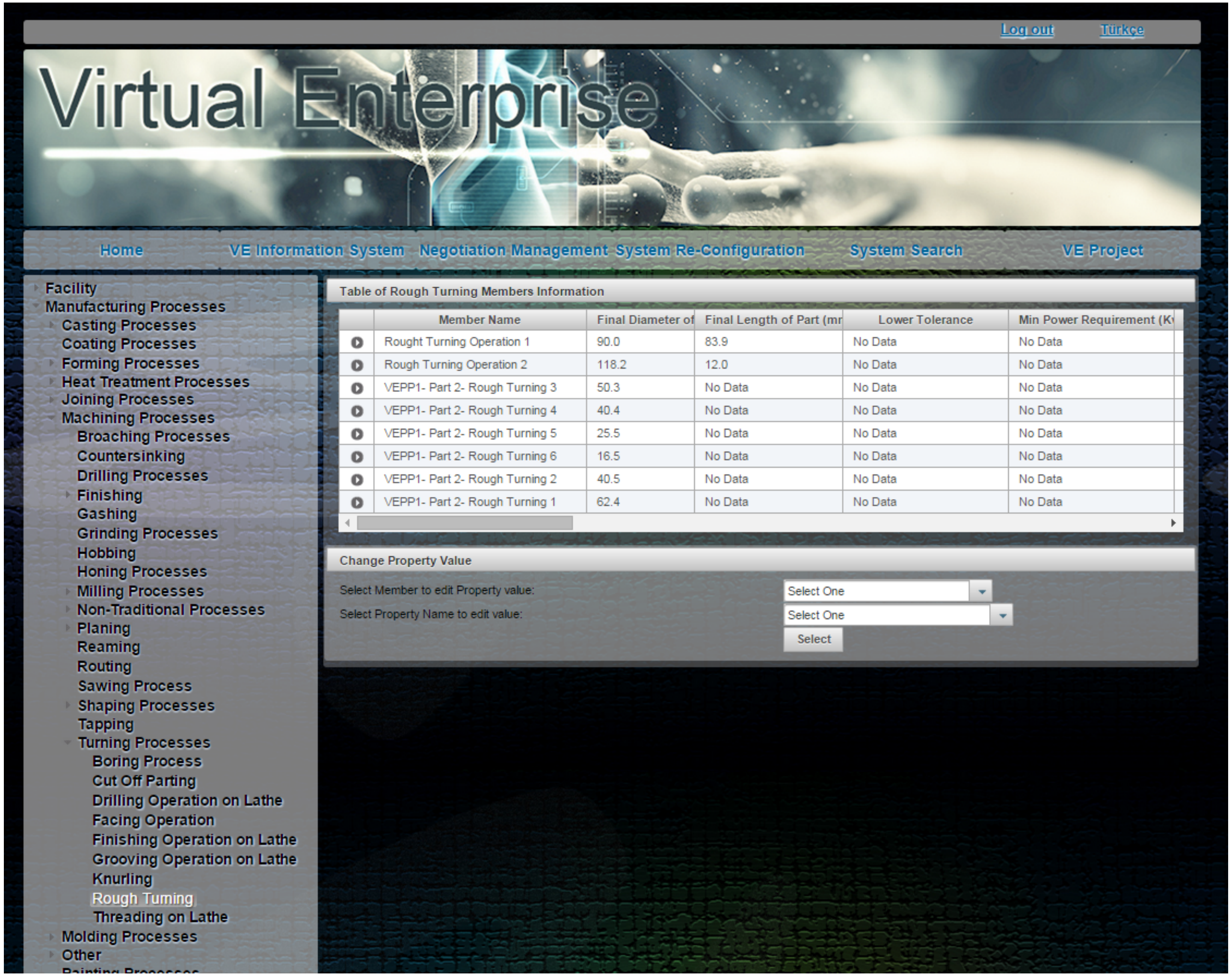


Figure 7.4: Rough turning processes information screen

Figure 7.5: Class reconfiguration screen

7.2.2 OMAVE Model Modification

To make required system re-configurations, special tools for model editing were developed. These tools enable system administrator to change, edit, update all the OMAVE backing model structure, entities and literal properties using simple, user friendly GUIs. This part is divided into four separate section;

Class Reconfiguration Section

From this panel administrator is able to add new classes or remove existing classes from the ontology model simply by selecting or entering class name.

Figure 7.5 is illustrating the class add/removal panel. Entered new class will be added directly to VE ontology model and the result will be populated on information monitoring screen.

Individuals Reconfiguration Section

Individual reconfiguration panel enables, add or remove members to or from classes of the developed OMAVE model.

As it is clear from Figure 7.6 administrator only picks the class that the individual is going to be added and then adds the individual by entering the name and language labels. The important issue in adding new class or individual to the model is that, in both cases model automatically assigns super classes' all data properties and relations.

For example if a new Turning center is going to be added to the system (Under the class of Resources/Manufacturing Resources/Machining Tools/Machining Centers/Turning Centers) all super classes' data properties and relations will be assigned to the new added entity automatically.

Figure 7.6: Individual reconfiguration screen

Reasoning procedure runs immediately after adding new individual or changing any property of the individuals automatically. If assume again the turning center example, immediately after adding this machine tool, system tries to find which manufacturing processes are capable of being related to this new added machine tool. Therefore after selecting machine tool owner, selected enterprise will be connected to those manufacturing processes and will be nominated as one of the capable companies to execute those related manufacturing processes.

Data Type Properties Re Configuration Section

Data properties as described before in sections 4.3.3 and 4.3.3.1 are properties which relate data values to individuals or classes.

In this case additional to add/remove panels, property assignment panels are also considered. After adding a new property to OMAVE model, it should be related to specific desired classes. By assigning this property to target class all sub class and member individuals may inherit that property accordingly. Thus a new panel was added to this section to empower administrator to edit properties assignment.

The other important issue in the case of data properties is the value of property for a special individual. Administrator may edit the property value of individuals directly from information monitoring panel (Figure 7.8).

Predicate or Object Properties Reconfiguration Section

Another reconfiguration panel designed for OMAVE system is object property reconfiguration. Object properties declare relations (predicates) in ontology model. These

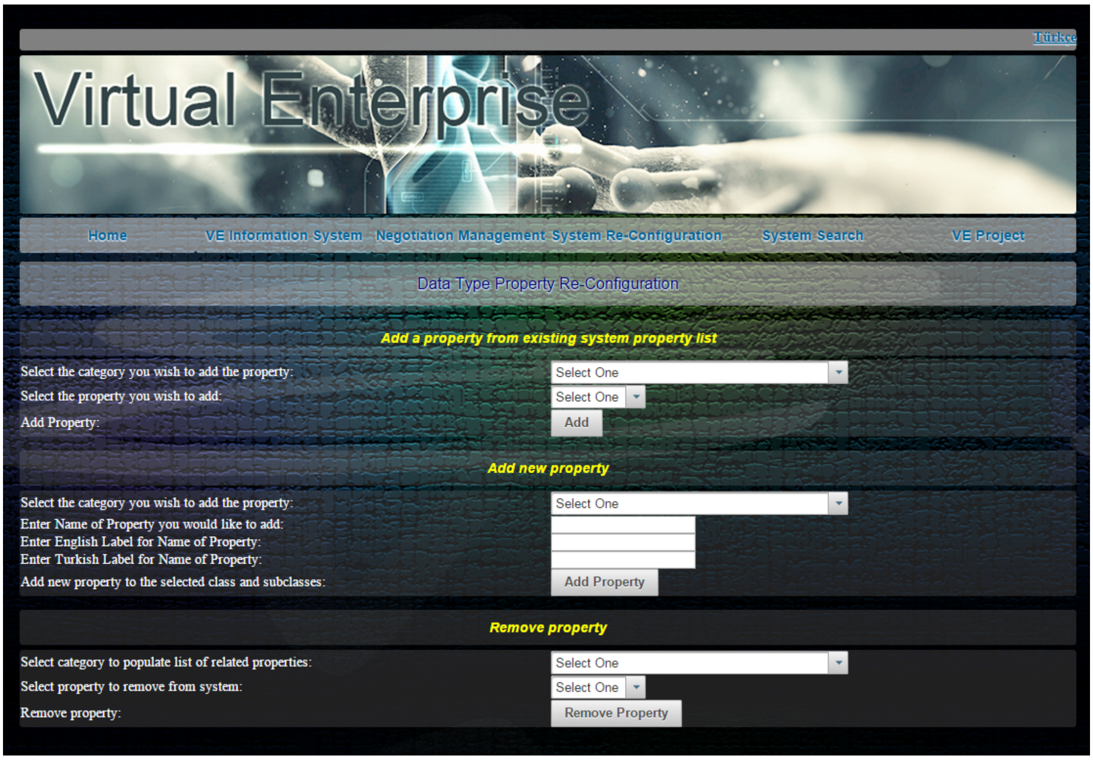


Figure 7.7: Data type property reconfiguration screen

Figure 7.8: Edit data type property value screen

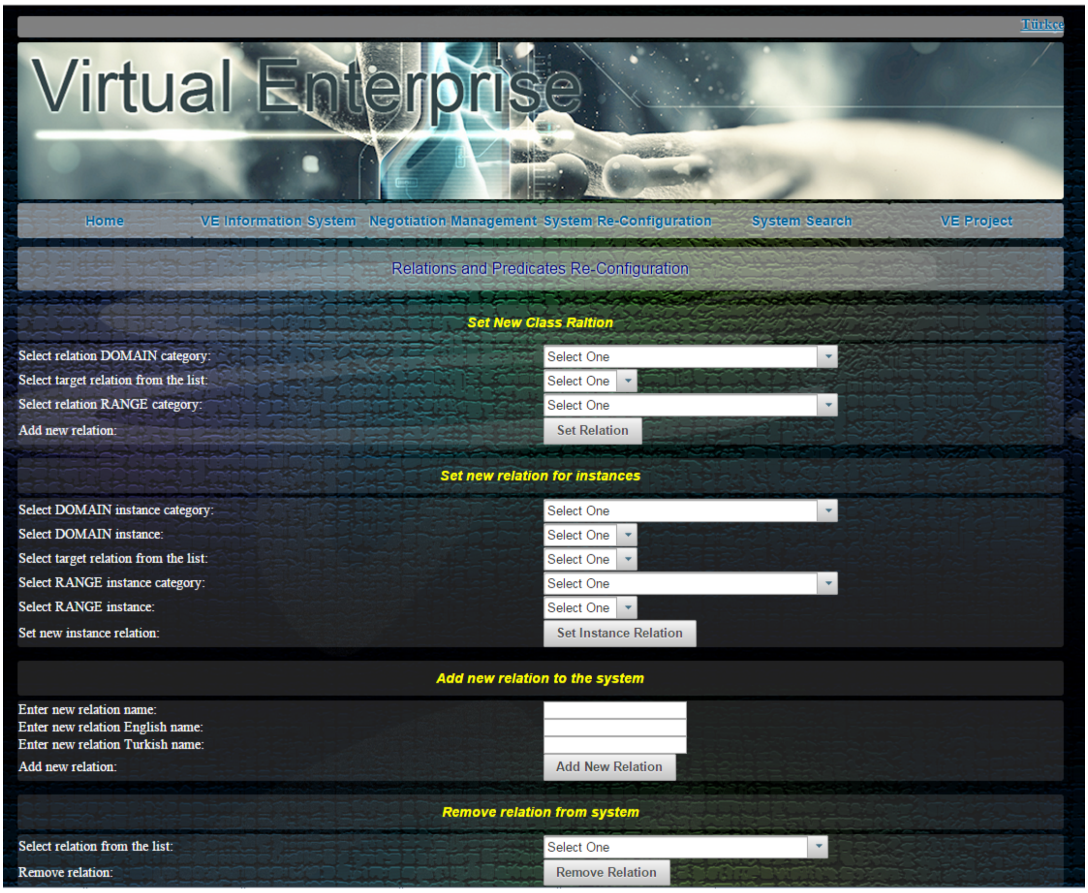


Figure 7.9: Object property reconfiguration screen

relations are highly dynamic in virtual enterprise systems and they are are changing all the time. One of advantages of ontology based VE model over other models is its high flexibility in editing, changing and creating new relations without any databases or information model reconstruction.

In this panel, as shown in Figure 7.9, 4 separate sections are available; connect two or more classes using available relations from the system; connecting two or more individuals by available relations in the system; Defining completely new relation to the system, then using first or second sections connect two entities to each other; Last one is for removing any relation from the system. In the case of removing a relation which already connects classes or entities to each other, those entities will be disconnected.

7.2.3 OMAVE Search Engines

In the next phase if Virtual Enterprise project in OSTIM organized industrial zone, 25-30 enterprises' information then, OSTIM Defense Industry Cluster or in Turkish (OSTIM Savunma Sanayi Kumelenmesi (OSSA)), and finally all OSTIM Clusters (OSTIM Renewable Energy Cluster, OSTIM Bio Medical Cluster, Constrauction Machines Cluster, Rail Industry Cluster)

Figure 7.10: Simple VESSE Configuration and View

will be added to this system. This means more than 500 active enterprises with huge amount of data and dynamic information. Searching for a specific information or data from this massive data store is truly difficult. In order to overcome this problem, two types of search engines special for OMAVE system were developed.

Two types of search engines are provided in OMAVE system. Fist type is general system search engine; and second type is specific for manufacturing processes search engine. Virtual Enterprise System general Search Engine (VESSE) was designed in two modes (simple and advanced) to search for any type of data and information in OMAVE TDB data store. For example, in Figure 7.11 is showing out coming results for key word *'Mazak'*.

VESSE advanced mode is designed for only organizations and resources search till now. It will be developed more in the next phases, but as the pilot test bed is concentrating on Machining and Forming processes in OSTIM and Case Study is focusing on these parameters, advanced search section is primarily developed to cover these processes.

In Figure 7.12, from subject type selection drop down menu bar, manufacturing resources or organizations could be selected, then if the manufacturing equipment choice is selected, two types of machining or forming machine tools will be available for search. If organization is chosen, machining tool owners selection will be shown in second drop down menu bar. Here user may select types of machine tool, or enter minimum and maximum values for specific criteria given in the lower part of the page and filter out the search results.

For instance, machining center type of machine tools are filtered in Figure 7.13. List of results and their properties is populated in the Results section. Second search type in OMAVE system is manufacturing processes specific search engine.

VESSE advanced search dashboard illustrated in Figure 7.14 was designed to follow the situation of project tasks and to discover what type of machine tools are available to realize

Figure 7.11: Simple VESSE Results for Mazak word

Figure 7.12: VESSE advanced search mode for a machining equipment

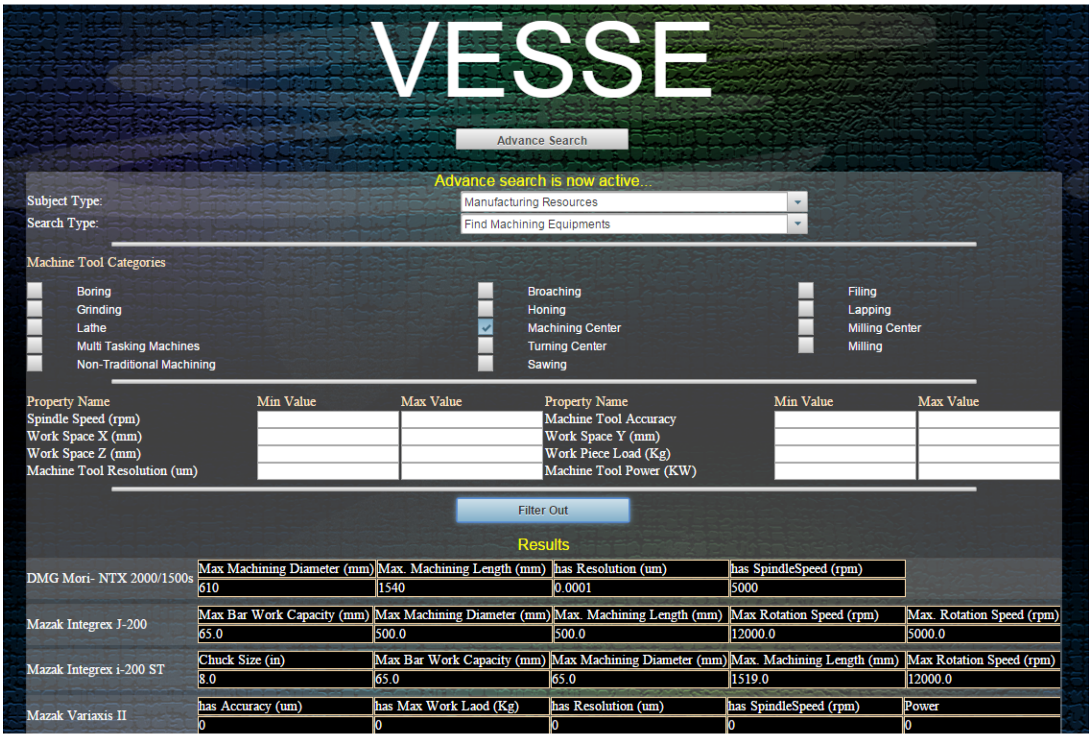


Figure 7.13: VESSE advanced search mode for a machining centers

these tasks, and which enterprise is the owner of these machine tools. This is the main filtering section of enterprises to enter negotiation process of a specific task. The procedure for finding task enablers and potential partners of a task, starts with selecting the project. Each project may include several assemblies. Based on selected Project, assembly products will be listed, and administrator may select one of these assembly products. Obviously each assembly product could be separated to several parts and consequently each part should follow special manufacturing procedures to be produced. Based on these stages, step based selection of drop down menu bars were designed. After selecting a special manufacturing task, it's processes, processes' properties and available machine tools for this task and finally list of machine tools' owner companies are populated.

7.2.4 OMAVE System Management

Auction management panel is here. Administrator is able to list all tasks, find potential partners, give order to start the negotiation procedure and finally find the auction winners. In Negotiation Management Screen (Figure 7.15) like Manufacturing process search section, first the part which is going to be auctioned must be selected. Then all the tasks in the selected part are listed. By clicking on open for bidding, an invitation letter, including following documents;

1. Included manufacturing processes information of the task

2. Part design document
3. Part's related assembly design document
4. Agreement and conditions regarding project and task negotiation
5. All additional documents regarding task

are sent to all potential partners (named enterprises in the last column). The letter and attached documents are sent to the named enterprises and task customer which is the project customer. If enterprises reply to email, an enterprise type agent for that enterprise will be deployed. This agent will manage the auction process in the name of assigned enterprise. Enterprises will be guided to their agent's GUI, and they have to give the agent's required information. Definitely customer agents GUI and it's required information are completely different.

Detailed information about the process of assigning agents, agents action and auction process is given in Section 8. If enterprises are eager to enter negotiation procedure, they have to respond email before announced deadline. After deadline if number of volunteer enterprises to enter bargaining is above the threshold (minimum number of enterprises to enter negotiation), administrator starts auctioning process by clicking on deploy agents button on management screen. Multiple agents start to negotiate and winner enterprise for the chosen task will be announced by task manager agent. Ranked list of bidding enterprises and their gained points will be tabulated on system administrator screen. Final decision about job allocation for tasks will be taken by management board of OSTIM technology according to the announced ranked list of enterprises.

7.3 Summary

In Chapter 7 developed various system components and their integration to each other were discussed. These system components are all developed based on system ontology model. According to illustrated OMAVE system hierarchy in Figure 2.4, to satisfy discussed operations and processes in Chapter 3.4 and meanwhile fulfill agent based system requirements, the integration process accomplished. This integration procedure realized using different software platforms like, Jena and JADE. In order to interact with system users, different user friendly and simple GUIs were developed. These GUIs were also discussed in details in this chapter.

Figure 7.14: Manufacturing process search screen

Figure 7.15: Negotiation management screen

CHAPTER 8

CASE STUDY

In order to test and validate OMAVE system's operation, a consortium with the aim of producing a sample assembly product as a final product is considered. The main target in this case study is to verify the performance of system components during proposed OMAVE life cycle. In this chapter, OMAVE platform components' performance is presented, evaluated and the final results are discussed. System performance evaluation process is as follows;

VE Formation Phase

- Entering project data into OMAVE system
- Creating Task groups based on Manufacturing Process Planning
- OWL DL reasoning for created task groups
- Invite candidate enterprises for auction process
- Start agent based partner selection process
- Present ranked list of enterprises to management board
- Select winner
- Form up Consortium with tasks' winners

VE Design

- Design ordered product in details concurrently using CATIA V6 web based design tool
- Project Process Planning

VE Operation

- Start OMAVE operation
- Follow OMAVE operation using ENOVIA PLM Tool
- Deliver produced parts of all tasks
- Assemble final product
- Test and quality control

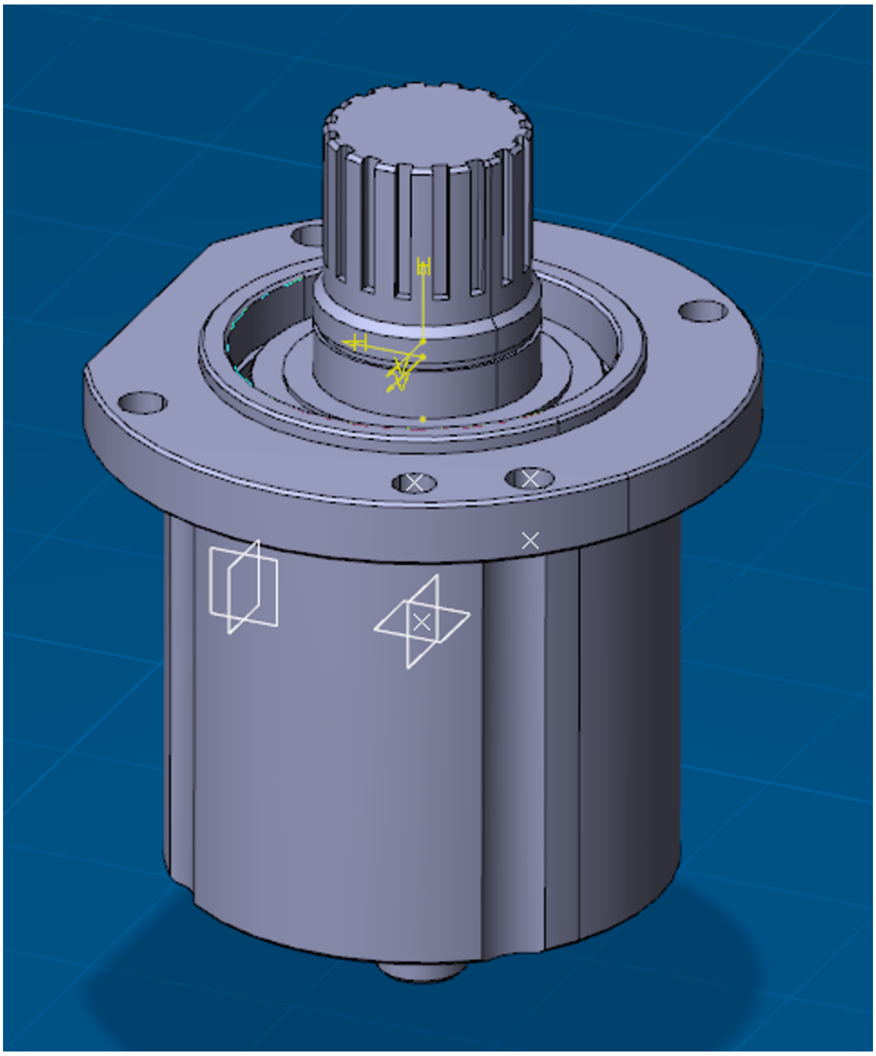


Figure 8.1: Case Study Product Assembly on CATIA V6

- Deliver tested and verified final product to the customer

VE Dissolution

- Performance evaluation of project partners

8.1 Ordered Product

Case study assembly product shown in Figure 8.1. This assembly consists of 12 separate parts where the exploded view is illustrated in Figure 8.2.

The target is to design product collaboratively on CATIA V6 platform with contributions of Experts from Enterprises and researchers from Universities. According to the product design, for each part of product, process plans will be developed.

Process planning information for each part, all design and engineering documents will be up-

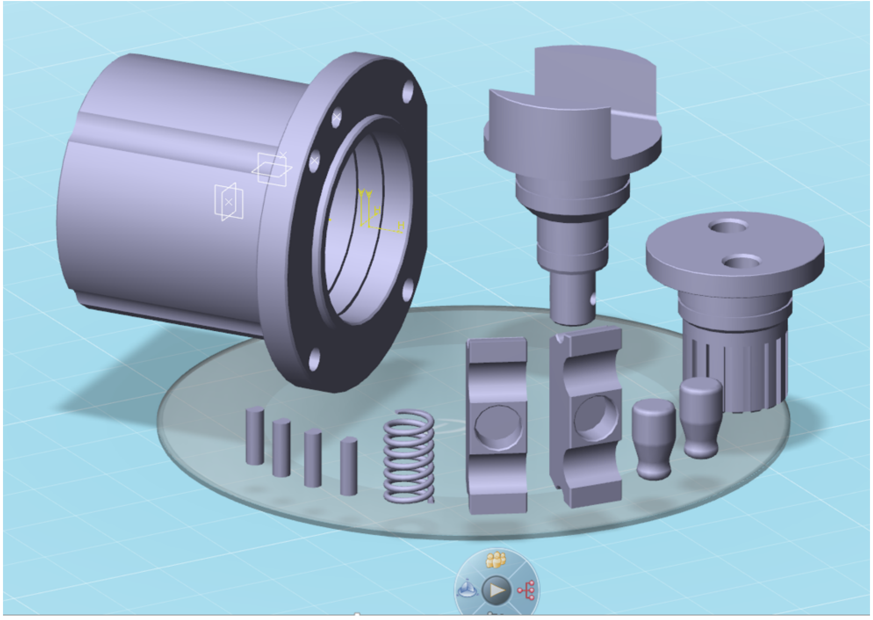


Figure 8.2: Case Study Product Parts Exploded View on CATIA V6

loaded to OMAVE system manually. Later on respectively, automatically resource allocation for tasks, inefficient enterprise elimination, multi agent based partner selection, VE form up, VE operation and dissolution phases is expected to be accomplished.

8.2 Product Design

Product order by customer could be submitted with(or without) product design. If order is without design, a consortium for designing new product should be formed. For this reason, customer order is divided into two separate order, one for product design and second for manufacturing processes. Enterprises with design capability or research and development start ups with reacquired capabilities may enter into design consortium. In this approach design partners and customer develop new product design through web based CATIA V6 design tool. After getting final design approval from customer, verified design will be the input for the next manufacturing consortium. Based on this design and related process planning, task groups are formed up and OMAVE administrator informs potential partners to put their bids and starts OMAVE formation phase. In second scenario, which is the case mostly in organizations like OSTIM, customer provides design and process planning for the ordered product. In this case according to the provided design and process plans task groups are shaped and potential partners are informed to submit their bids.

These two scenarios are depicted in Figure 8.3 below;

In this case study, a combination of two scenarios are considered. In this case study collaborative product design developed on CATIA V6 collaborative design and ENOVIA PLM platform with the participation of researchers from universities and experts from OSTIM com-

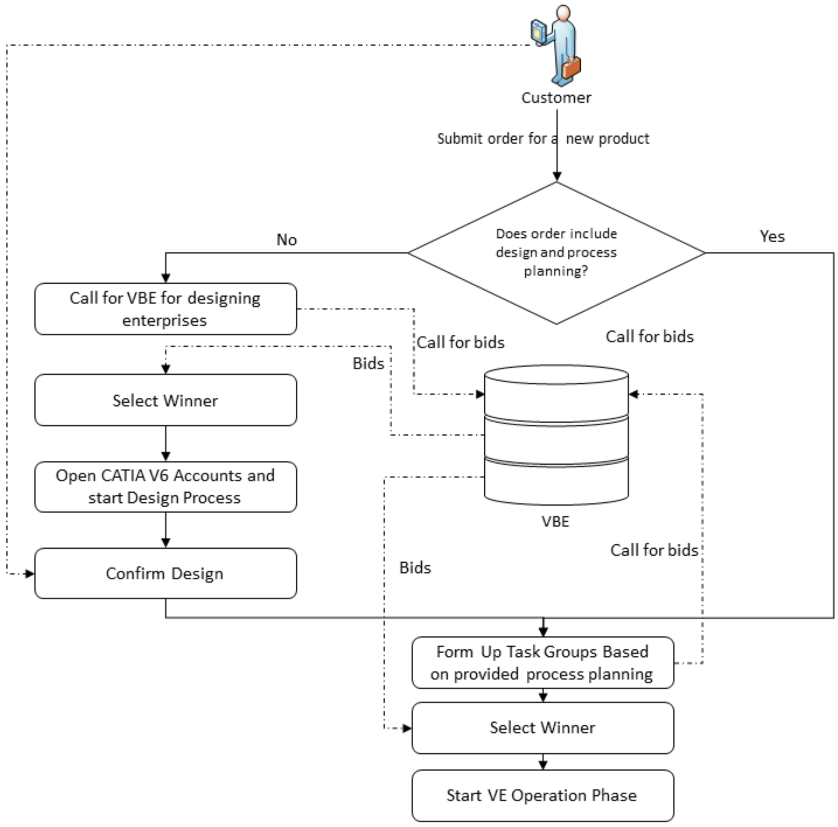


Figure 8.3: Order Design Scenarios

panies with designing capabilities.

8.2.1 ENOVIA V6 PLM

Dassault CATIA V6 is digital 3D design platform which provides special features for different industrial applications and sectors. There are special professional features for transportation and mobility, aerospace and defense, marine and offshore, industrial equipment, high tech, consumer goods-Retail, consumer packaged goods, life sciences and finally Energy, process and utilities.

The new version of Dassault CATIA V6 design tool is backed by ENOVIA V6 PLM tool. ENOVIA is online collaborative environment that enable users to contribute in design from anywhere at anytime. By using ENOVIA V6 platform, secure and traceable data exchange could be realized between stakeholders. If an authorized user connects to CATIA V6 the 3DXML[1] file directly is sent by V6 authoring application to the user client and the required information (based on user roles) will be available. There are different roles in ENOVIA V6 platform which are assigned by administrator to users. Some of these roles are predefined

[1] The 3DXML file type is primarily associated with '3D XML' by Dassault Systemes. 3D XML is a universal lightweight XML-based format for sharing of 3D data. With 3D XML, PLM information can be incorporated into technical documentation, maintenance manuals, marketing brochures, websites, email communications and other everyday uses. Note: Files created with 3dprintscreen.exe may differ from those created with other software.

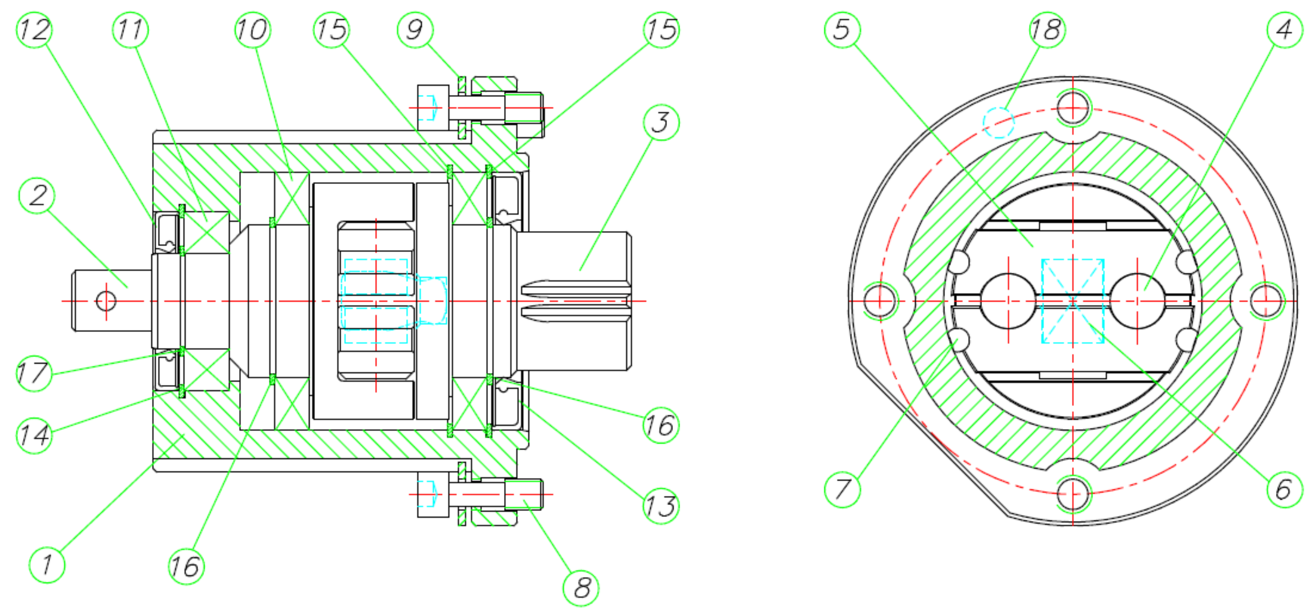
by ENOVIA and there is possibility to define more special roles by system administrator according to the project requirements [116].

ENVOIA V6 simplifies the view and management of project team members' task assignments through **"My Calendar"**. **"My Calendar"** provides an aggregated view of project tasks, risks, issues, meetings and route tasks for quick global assessment. Users' assignments can be visualized in daily, weekly and monthly views based on task due date and start date helping to better manage priorities. From these views users can also directly access assignment properties and work on their tasks improving overall usability. It also introduces the possibility to configure requirement definitions using product features options in addition to the already available date and product revision based effectiveness. These powerful requirement configuration capabilities enable the creation of a highly coherent design through the use of a fully configured Requirement, Functional, Logical and Physical (RFLP) product definition model. This rich RFLP model leverages a single requirements structure and variants dictionary to manage all product or system variants [116].

8.2.2 Design and Process Planning of Ordered Product on CATIA V6

As mentioned before, ordered product is containing 12 parts (Figure 8.2). This Figure is captured from CATIA V6 design screen. All these parts were designed on collaborative design platform then each part's process planning was developed. Designed parts and their process planning is illustrated in table 8.1.

These process plans and part design are added to the system using provided, system modification interfaces. These interfaces described in section 7.2.2. In order to check and verify system performance KNM1 and KNM2 parts information and their process plans were added to the system.



P. NO	ADET	TANIM	MALZEME	STOK NO
18	1	DÜZ TESPİT PİMİ ø8m6x16	HAZIR	DIN 6325
17	1	ÖZEL SPİRAL YAYLI SEGMAN ø29xø23.4x0.5	YAY ÇELİĞİ	ÖZEL İMALAT
16	2	ÖZEL SPİRAL YAYLI SEGMAN ø43.8xø38x0.7	YAY ÇELİĞİ	ÖZEL İMALAT
15	2	ÖZEL SPİRAL YAYLI SEGMAN ø73.2xø66x0.6	YAY ÇELİĞİ	ÖZEL İMALAT
14	1	ÖZEL SPİRAL YAYLI SEGMAN ø51.2xø44x0.7	YAY ÇELİĞİ	ÖZEL İMALAT
13	1	YAĞ KEÇESİ 40x68x8	HAZIR	CR–15854
12	1	YAĞ KEÇESİ 25.4x47.02x6.35	HAZIR	CR–10071
11	1	SABİT BİLYALI YATAK	25x47x12	ORS 6005
10	2	SABİT BİLYALI YATAK	40x68x9	ORS 16008
9	4	RONDELA A 8.4 St (ø8.4xø16x1.6)	BEYAZ ÇİNKO KAPLI ÇELİK	DIN 125

P. NO	ADET	TANIM	MALZEME	STOK NO
8	4	HAPİS VİDA	HAZIR İMBUS CİVATA	..2734/KNM8
7	4	ÖZEL PİM	CIVA ÇELİĞİ	..2734/KNM7
6	1	BASKI YAYI	PASLANMAZ YAY ÇELİĞİ	..2734/KNM6
5	2	ÇENE	4140 ÇELİK	..2734/KNM5
4	2	DÖNDÜRME PİMİ	4140 ÇELİK	..2734/KNM4
3	1	KANALLI MİL	4140 ÇELİK	..2734/KNM3
2	1	KADEMELİ MİL	4140 ÇELİK	..2734/KNM2
1	1	GÖVDE YATAK	4140 ÇELİK	..2734/KNM1

Figure 8.4: Assembled Design of ordered Product

Table 8.1: Designed product parts and their process planning

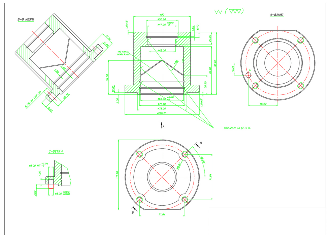
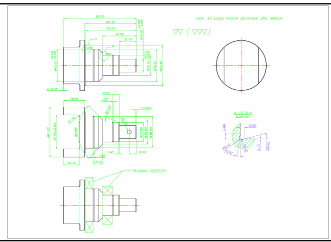
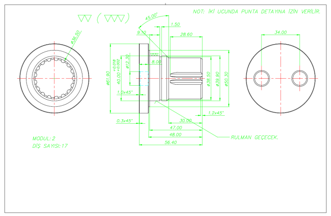
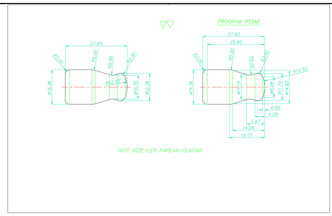
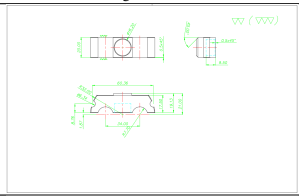
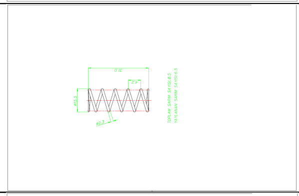
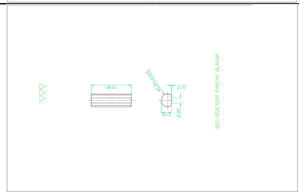
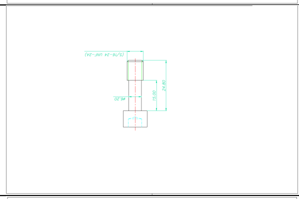
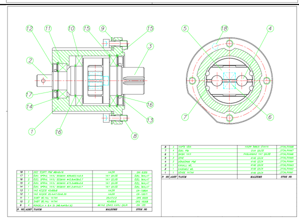
Part Name	Part Detailed Design	Part Process Plan	
		Process Name	KNM1 Process Plan
Part KNM1		Name	KNM1 Shaft Bearing
		Raw Material	4140 Quenched and Tempered Steel 42 CRMO 4 bar
		Raw Material Size	$\phi 120 \times 105mm$
		Operation 1	Turning part length $\phi 90$ and $\phi 118.2$ features
		Operation 2	Milling Operation of $\phi 111.8$ chord feature
		Operation 3	Drilling $4 \times 5/16 - 24$ UNF-28 holes
		Operation 4	Turning $\phi 78$ feature
		Operation 5	Channel $\phi 71.6$, $\phi 68$, $\phi 42$, $\phi 47$, and $\phi 50.6$
		Operation 6	Milling 1 mm width lubrication channels on 5 axis milling machine
		Operation 7	Heat Treatment, hardening operation $HRC50 \pm 2$
		Operation 8	Grinding Operation of inner surfaces
		Operation 9	Coating: Epoxy body coat, MILP 53022 TYPE 2, 1 Layer, 0.025 mm thickness. EPOXY MILC 22750 compatible white color 17925 FET-STD-595 compatible 2 layers 0.04 mm -0.05 mm thickness.
		Process Name	KNM2 Process Plan
Part KNM2		Name	KNM2 Stepping Shaft
		Raw Material	4140 Quenched and Tempered Steel 42 CRMO 4 bar
		Raw Material Size	$\phi 65 \times 95mm$
		Operation 1	Turning $\phi 61.9$, $\phi 40$, $\phi 49.8$, $\phi 39.9$, $\phi 25$, $\phi 16$, features with remaining 0.5 mm for finishing after heat treatment operation
		Operation 2	Milling Operation of 42 mm feature
		Operation 3	Heat Treatment, hardening operation $HRC50 \pm 2$
		Operation 4	Grinding specified features
		Operation 5	Drilling Operation of $\phi 5.0mm$ hole
		Operation 6	Blackening operation
		Process Name	KNM3 Process Plan
Part KNM3		Name	KNM3 Timing Shaft
		Raw Material	4140 Quenched and Tempered Steel 42 CRMO 4 bar
		Raw Material Size	$\phi 65 \times 60mm$
		Operation 1	Turning $\phi 61.9$, $\phi 50.30$,$\phi 40$, $\phi 39.9$, features with remaining 0.5 mm for finishing after heat treatment operation
		Operation 2	Turning 1.5 mm $\phi 38.5$ Groove
		Operation 3	Drilling and boring $\phi 30mm$
		Operation 4	Using CNC Lathe and NC programming machine inside and outside gears
		Operation 5	Heat Treatment, hardening operation $HRC50 \pm 2$
		Operation 6	Finishing and drilling $2 \times \phi 12.30mmholes$
		Process Name	KNM4 Process Plan
Part KNM4		Name	KNM4 turning pin
		Raw Material	4140 Quenched and Tempered Steel 42 CRMO 4 bar
		Raw Material Size	$\phi 65 \times 60mm$
		Operation 1	Using CNC Lathe and NC programming whole part surface is being machined

Table 8.1: Designed product parts and their process planning (cont'd)

Part Name	Part Detailed Design	Part Process Plan
Part KNM5		**Process Name**: KNM5 Process Plan Name: KNM5 Raw Material: 4140 Quenched and Tempered Steel 42 CRMO Raw Material Size: $66mm \times 90mm \times 21mm$ (For Two Jaw and Two Pins) Operation 1: Milling Operation for preparing cuboid materials Operation 2: Milling $\phi16.2mm$ holes Operation 3: Two Jaws and two pins are cut out by wire EDM Machine
Part KNM6		**Process Name**: KNM6 Process Plan Name: KNM6 Spring Raw Material: Stainless Spring Steel Raw Material Size: $\phi15 \times 30mm$ Spring steel thickness $\phi2.3$ Operation 1: Outsourced
Part KNM7		**Process Name**: KNM7 Process Plan Name: KNM7 Raw Material: 4140 Quenched and Tempered Steel 42 CRMO Raw Material Size: $66mm \times 90mm \times 21mm$ (For Two Jaw and Two Pins) Operation 1: Milling Operation for preparing cuboid materials Operation 2: Milling $\phi16.2mm$ holes Operation 3: Two Jaws and two pins are cut out by wire EDM Machine
Part KNM8		**Process Name**: KNM8 Process Plan Name: KNM8 Raw Material: Allen Bolt Raw Material Size: $5/16 - 24$ UNF-2A Operation 1: Turning specified features on Allen bolt Operation 2: Blackening Operation
Part Assembly		

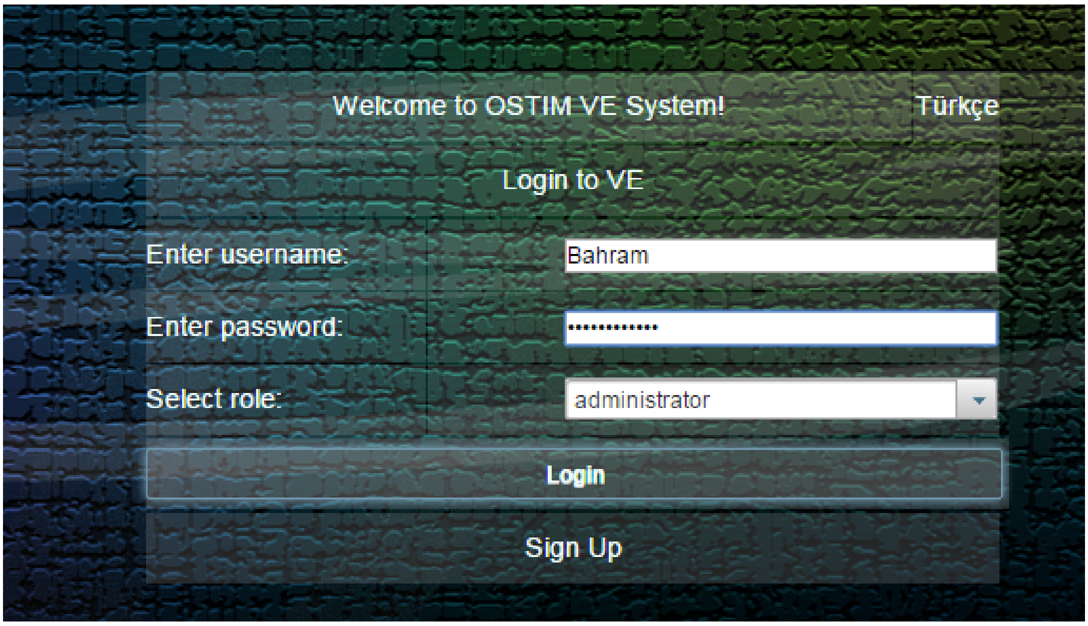


Figure 8.5: Administrator Login Panel

8.3 OMAVE Formation

Only system administrator is able to modify, add or remove data from OMAVE system. In order to enter VE system three user types were created;

- Administrator
- Enterprise
- Customer

After login to OMAVE system, each of these group will face a different GUI. To put data in system from designing and process planning phase, an administrator login is required (8.5). In order to add new information on the data store system administrator needs to refer to section***"System Re-Configuration"*** .

These parts are containing different manufacturing processes. These Manufacturing processes are mostly machining processes, and there are also some heat treatment and blackening processes. In pilot test bed there are not any available enterprise for realizing heat treatment or blackening processes. Therefore only machining processes are considered here.

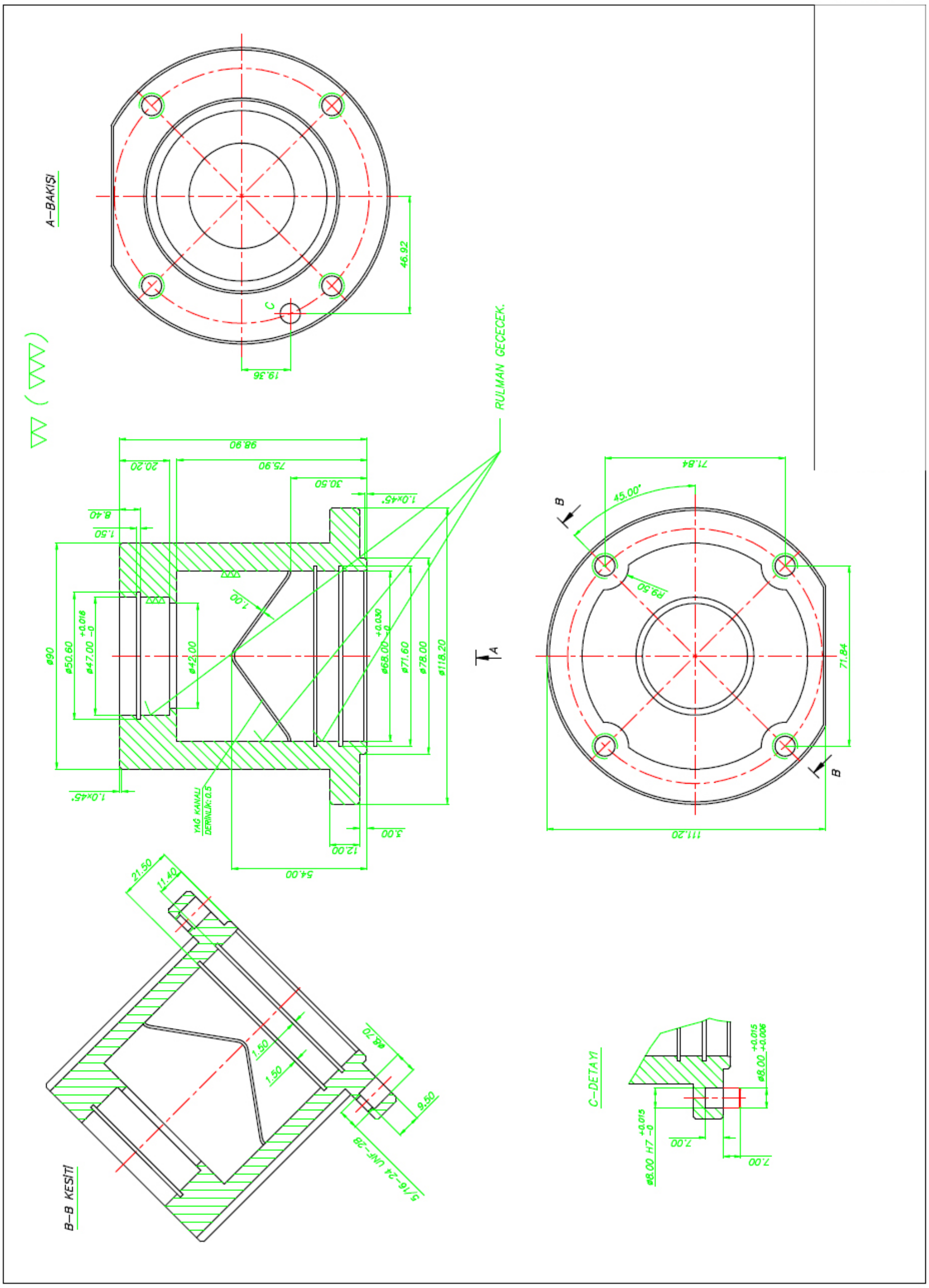


Figure 8.6: Part KNM1 detailed design from CATIA V6

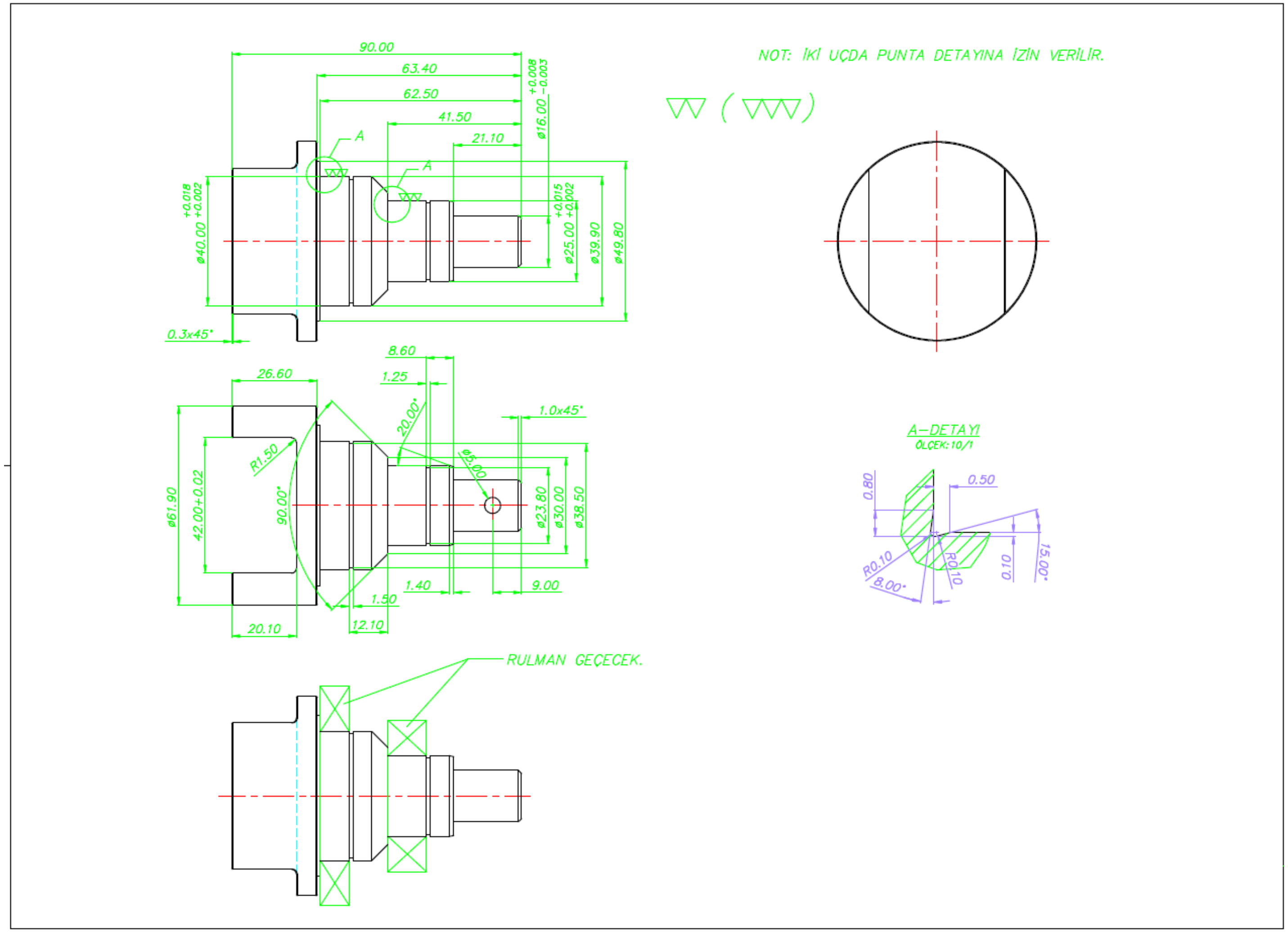


Figure 8.7: Part KNM2 detailed design from CATIA V6

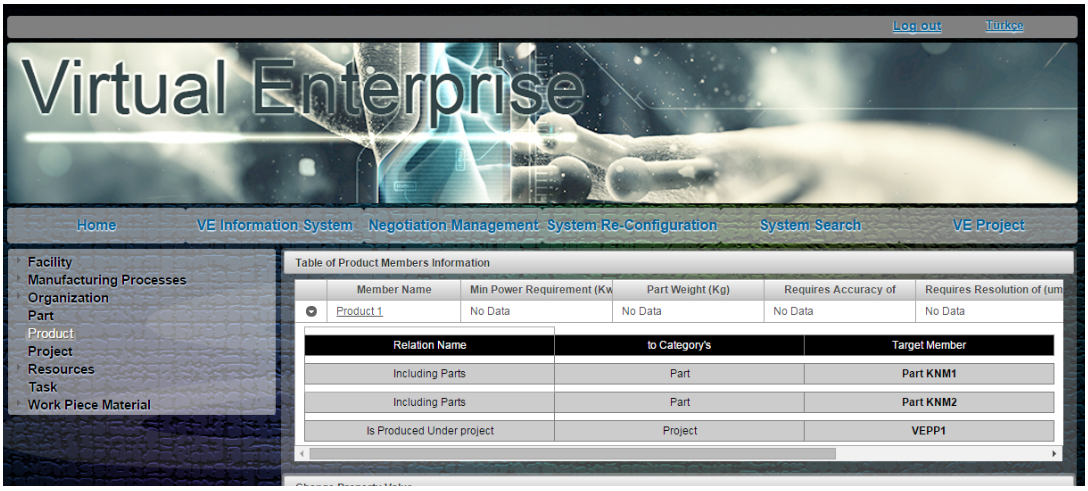


Figure 8.8: Added Product1 to the system

8.3.1 Sample CASE STUDY Parts KNM1 and KNM2

Part KNM1 information entered to OMAVE system using OMAVE modification interfaces. In order to enter all information to the system, under project category, an individual in the name of ***'VEEPP1'*** (which is an abbreviation of Virtual Enterprise Pilot Project) was created. This project has one assembly product. Therefore, under product class, a new individual called ***'Product 1'*** was defined. These two entities should be related to each other by two reverse relations ***'has a product'*** and ***'is in project'***. These relations relates VEPP1 and Product 1 in this way;

$$VEPP1 \xrightarrow{hasProduct} Product1$$

and reverse relation is:

$$Product1 \xrightarrow{isinProject} VEPP1$$

These two relations also were added through predicate configuration interface. Added individuals and their relations are shown in Figures 8.8 and 8.9;

The same procedure was followed to add ***'Part KNM1'*** and ***'Part KNM2'*** and their tasks, according to the developed product process planning. After adding parts and tasks, their relations, data properties also must be entered to the system.

Entered data properties are really important in the automatic reasoning process. OWL-DL reasoning goes through entered data properties and relations to find out which machine tool is capable of operating that specific task and based on this reasoning task groups were created.

In Figure 8.10 entered design information of three parts (KNM1, KNM2, and KNM3) is represented. As it can be seen from this Figure, ***'Part has task'***, ***'Is Used in Product'***, relations and their reverse relations (***'Task is belonging to Part'***, ***'Including Parts'***) relations also were

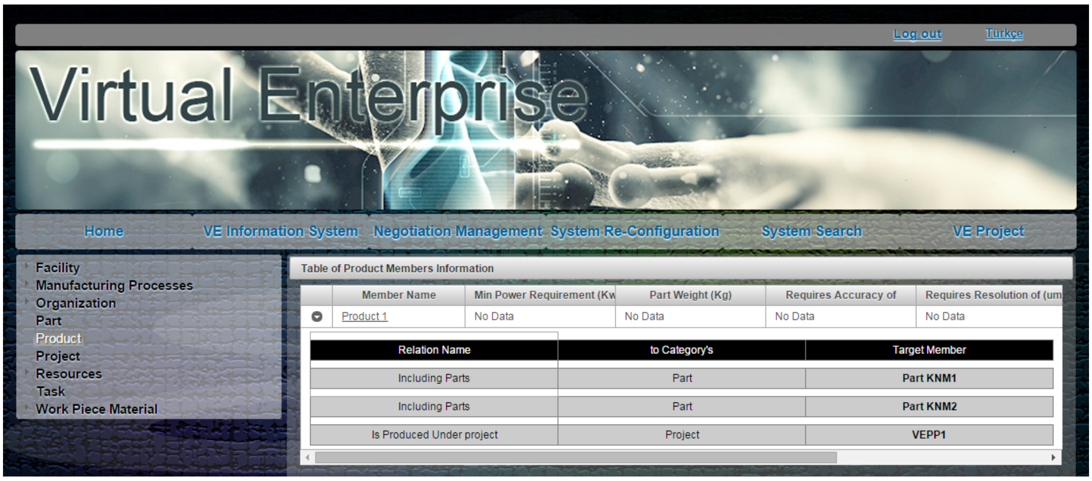


Figure 8.9: Added Project1 to the system

added to the system. One of the sample relations between part and task is shown below;

$Part\ KNM1 \xrightarrow{parthastask} VEPP1 - Part\ KNM1 - Milling\ Task\ 1$
reverse relation:
$VEPP1 - Part\ KNM1 - Milling\ Task\ 1 \xrightarrow{TaskisbelongingtoPart} Part\ KNM1$

Next step is to add one more deeper layer which is processes layer. Each task has one or more manufacturing processes included. All the manufacturing processes, their relations, and data properties, were entered to the system. The results are illustrated in Figure 8.11.

and there is a sample entered relation for tasks and manufacturing processes;

$VEPP1\ Part\ KNM1\ Task1 \xrightarrow{Task\ has\ Process} Rough\ Turning\ Operation\ 1$
reverse relation:
$Rough\ Turning\ Operation\ 1 \xrightarrow{Porcess\ is\ in\ Task} VEPP1\ Part\ KNM1\ Task1$

8.3.1.1 Automated Enabler Assignment

After adding each entity to the system, OWL DL reasoning run once and checks all entered properties, relations. Based on these information and given OWL DL rules which discussed in Section 4.3.4 and listed in Table 4.2 the appropriate machine tools for tasks are selected. For instance as it is illustrated in Figure 8.12 *Turning Operation 3* as an instance for a manufacturing process is related to 3 lathes and 3 Multi Tasking Machines, through ***Is Enabled By*** relation. This process is in task *'VEPP1- Part KNM1- Turning Task 2'*. This means that each of these machines are able to process this manufacturing operation and their characteristics may satisfy these process requirements.

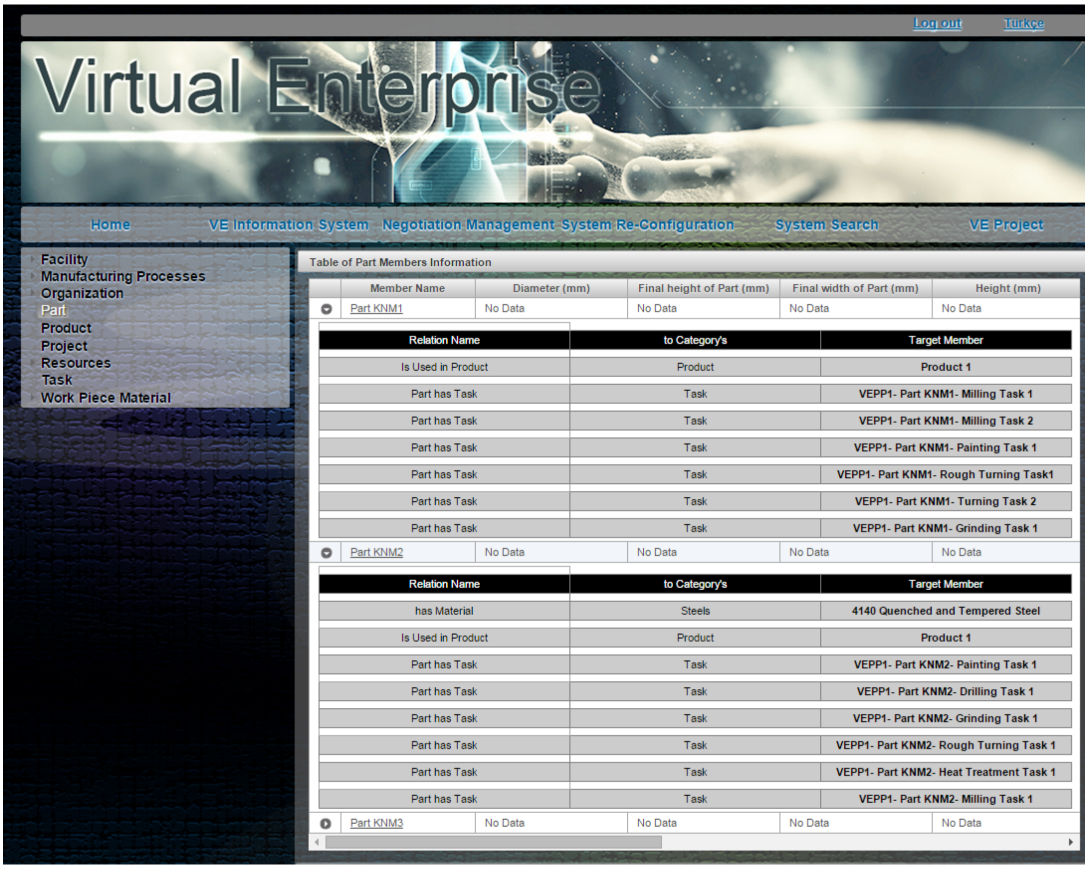


Figure 8.10: Added parts to the system

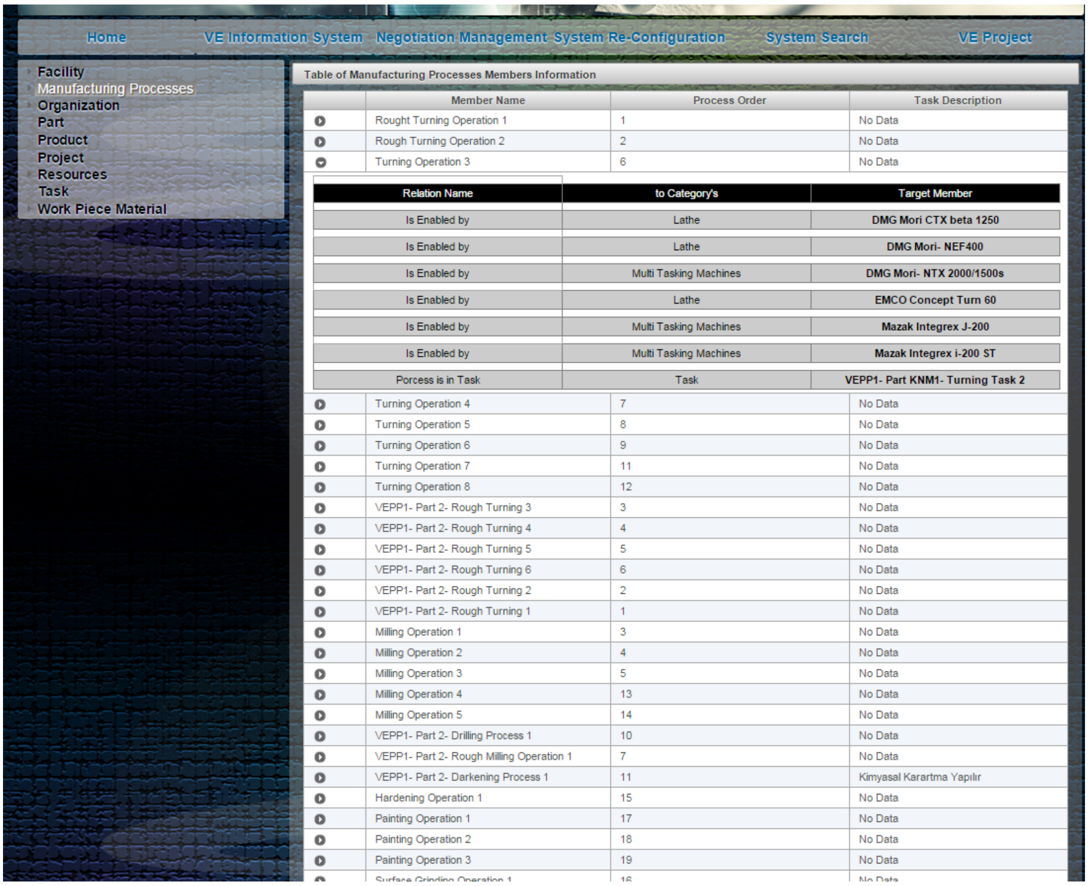


Figure 8.11: Added manufacturing processes to the system

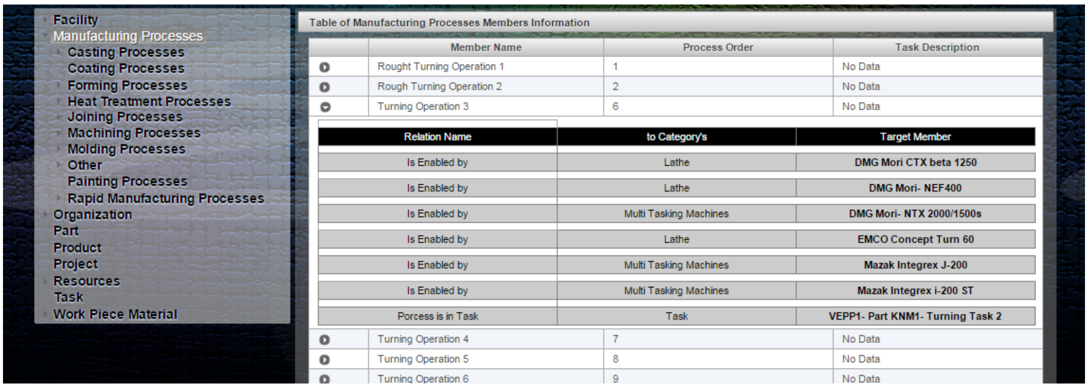


Figure 8.12: List of available machine tools for Turning Operation 3

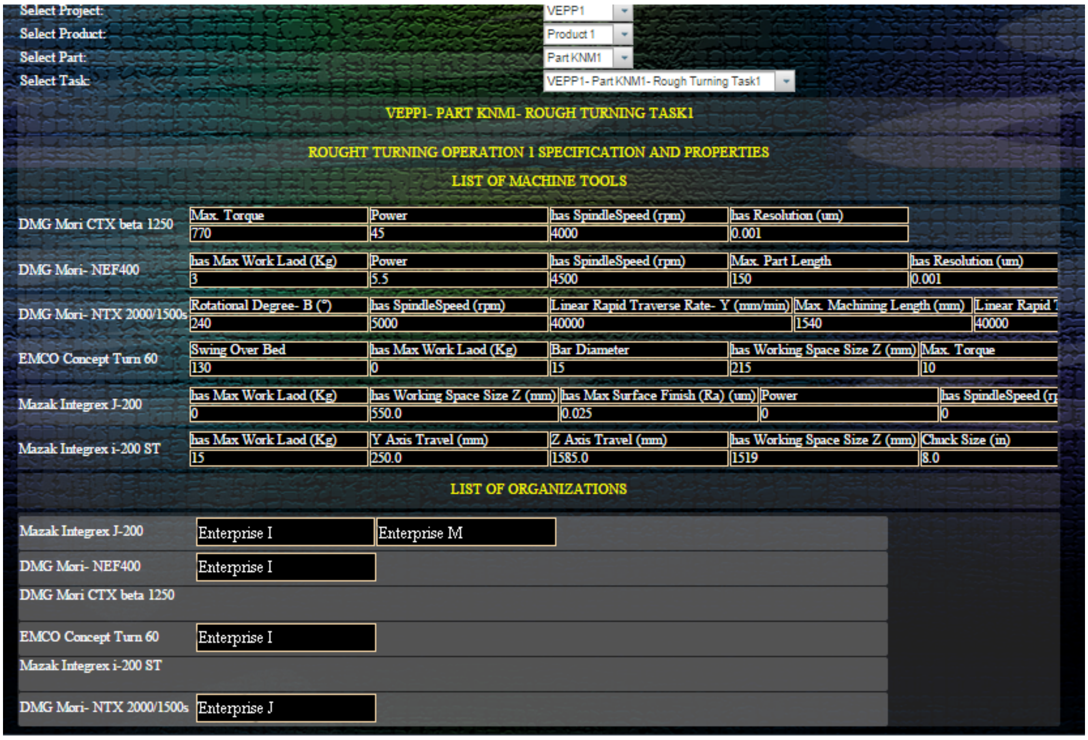


Figure 8.13: List of assigned machine tools for tasks

A Task is containing multiple manufacturing operations. Each manufacturing operation needs its own properties and according to these properties, special suitable machine tools could be assigned. If multiple successive operations from same category (for instance multiple successive turning operations) appear in a part manufacturing process planning, these operation will be packed into one task. To assign machine tools to a combined task, a unified manufacturing operation, (in this sample one turning operation, with most strict requirements) is considered. In other words, the union of selected machine tools for all operations is selected. Task enablers list is in fact the common machine tools between all operations in the task (which is combination of multiple similar operations with different requirements). In Figure 8.13 list of assigned machine tools for task ***VEPP1- PART KNM1- ROUGH TURNING TASK1*** is depicted. All selected machine tools specifications is listed as well.

The most important goal to relate these processes to machine tools is, to find enterprises which are capable of realizing these manufacturing processes. Because all these are done just to see who is capable of doing this specific manufacturing process and invite them to join auctioning procedure and be a partner of forthcoming OMAVE consortium. Again from Figure 8.13, it can be seen that, below the list of enablers (list of machine tools) there is a list of machine tools' owner enterprises. Separately, each of these machine tools and their owners are listed.

All this task- machine tool assignment process was done by semantic rules of OWL-DL reasoning engines. Rules were developed based SWRL format and in order to be compiled and parsed in Jena platform they all were written in Jena format. In Appendix E a sample Jena-formatted code of process- machine tool assignment rule is shown. There are 7 rules for this purpose which are listed in 4.2.

8.3.1.2 Partner Selection

For ordered product like *Product1*, there is a list of Parts and their associated tasks and lists of potential partners for each task of each part. For each part a VE consortium will be formed up and the partners of consortium are auction winners.

In this case, study for producing *Part KNM1* and *Part KNM2*, two VE consortium were developed. Maximum number of OMAVE consortium partners could be the number of tasks of that part. This means that for *Part KNM1*, maximum number of partner enterprises could be 6 enterprises.

To start the partner selection procedure, system administrator goes to Negotiation Management section of OMAVE system. System administrator, selects the target task to open for auctioning process. As it illustrated in Figure 8.14 by selecting a part (here Part KNM1 is selected) list of tasks are populated. Just by selecting target task and clicking on ***'Open for Bidding'*** button in the bottom of the page the negotiation procedure starts.

By opening the task for bidding following steps run one after each other;

1 Contacting potential partners listed as organizations in the the tasks table

2 Send enterprises documents, including negotiation conditions, prepared auctioning agreements, technical drawings, and any other complementary documents regarding task.

3 Send project customer task opening warning and collect required information from customer about task auctioning.

4 Get responses from enterprises

5 Get customer respond

6 If number of responding and volunteer enterprises reaches minimum number for bargaining, auctioning commencement order is declared by administrator.

7 Multi agent based partner selection procedure starts

8 In the case of finalizing auctioning fruitfully, ranked list of bargaining enterprises and winner enterprise are announced.

9 Winner enterprise as one of the part VE partners is disclosed.

10 Agreements are signed

In order to invite and inform enterprises about negotiation operation, a formatted email containing task information is sent to all the listed enterprises. Besides, an email is sent to the

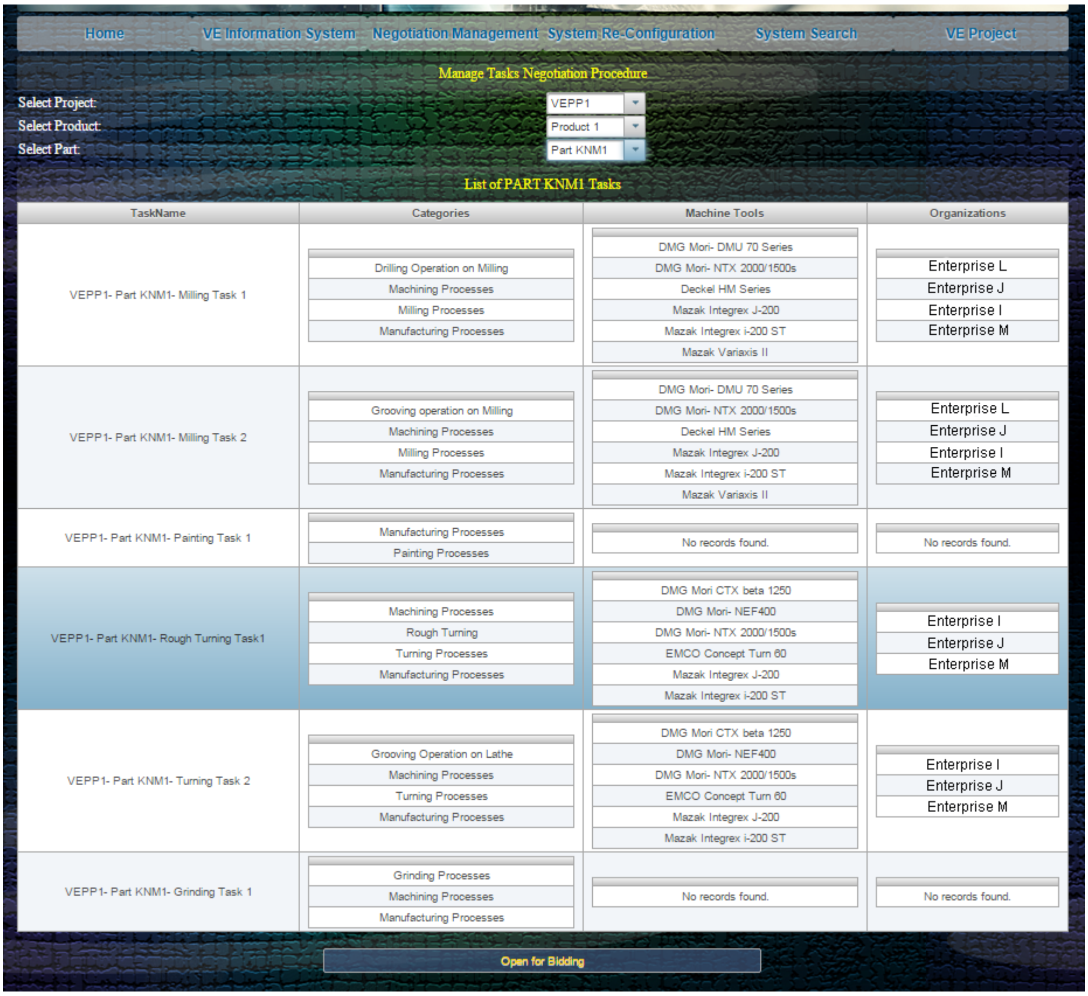
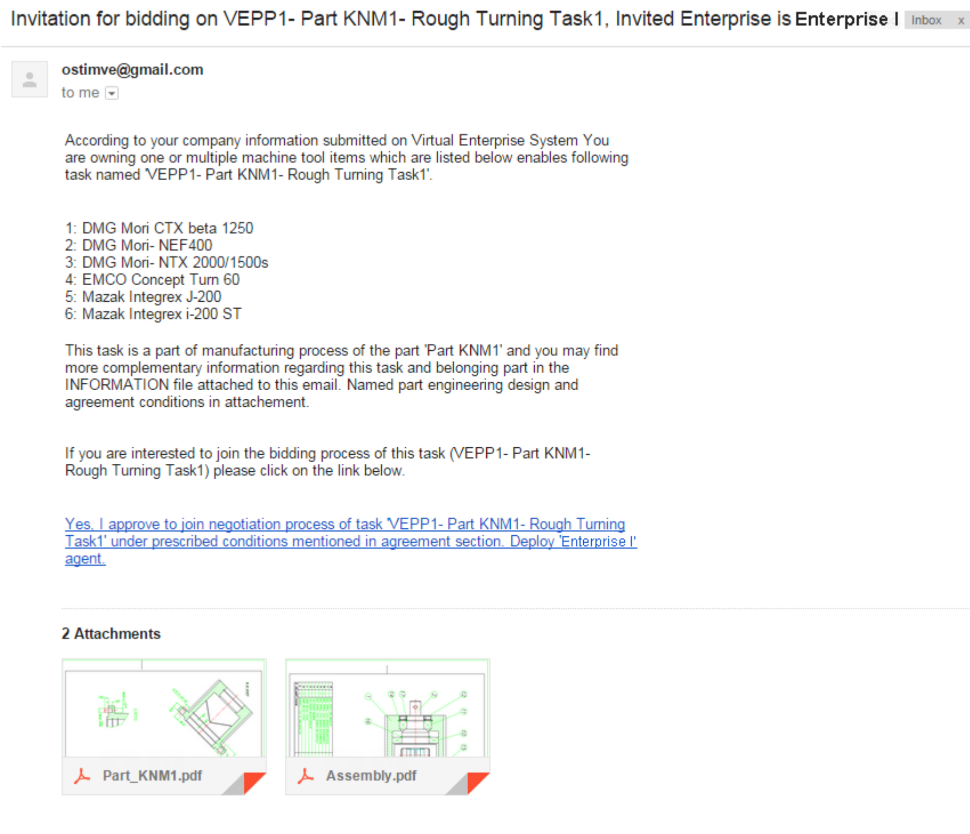


Figure 8.14: Starting Negotiation in VE system



Invitation for bidding on VEPP1- Part KNM1- Rough Turning Task1, Invited Enterprise is Enterprise I Inbox x

ostimve@gmail.com
to me

According to your company information submitted on Virtual Enterprise System You are owning one or multiple machine tool items which are listed below enables following task named 'VEPP1- Part KNM1- Rough Turning Task1'.

1: DMG Mori CTX beta 1250
2: DMG Mori- NEF400
3: DMG Mori- NTX 2000/1500s
4: EMCO Concept Turn 60
5: Mazak Integrex J-200
6: Mazak Integrex i-200 ST

This task is a part of manufacturing process of the part 'Part KNM1' and you may find more complementary information regarding this task and belonging part in the INFORMATION file attached to this email. Named part engineering design and agreement conditions in attachement.

If you are interested to join the bidding process of this task (VEPP1- Part KNM1- Rough Turning Task1) please click on the link below.

Yes, I approve to join negotiation process of task 'VEPP1- Part KNM1- Rough Turning Task1' under prescribed conditions mentioned in agreement section. Deploy 'Enterprise I' agent.

2 Attachments

Part_KNM1.pdf
Assembly.pdf

Figure 8.15: Enterprise invitation letter sample

project customer as well. These two types of emails have different formats and different types of documents are attached. An enterprise invitation letter sample is illustrated in Figure 8.15.

If invited enterprise is eager to join auctioning procedure, just by clicking over provided link, enterprise agent will be activate. An authorized person of invited enterprise will encounter a login page like as shown in Figure 8.16. If login process was successful enterprise agent GUI will burst into sight. Enterprise agent asks some critical information and data through its GUI from enterprise authorities(Figure 8.17). These information shape enterprise auction policy, and form enterprise agent's character. An enterprise agent uses special algorithm (as described in Section 5.3) that is highly affected by enterprise parameters. There are two types of enterprise parameters. One is static parameters such as, enterprise past performance, quality, service and after sale service performances which are obtained from system data store and the second type is dynamic parameters which are provided by enterprise authorities, using this agent interface. Dynamic parameters are like, min price, starting offer, delivery time, and strategy.

An agent with better static parameters, definitely will be more competitive and it may handle negotiation procedure much better. On the other hand an agent matched with an enterprise with low static parameters and noncompetitive offers would behave poorly and will have a week characteristic against rivals in auctioning procedure.

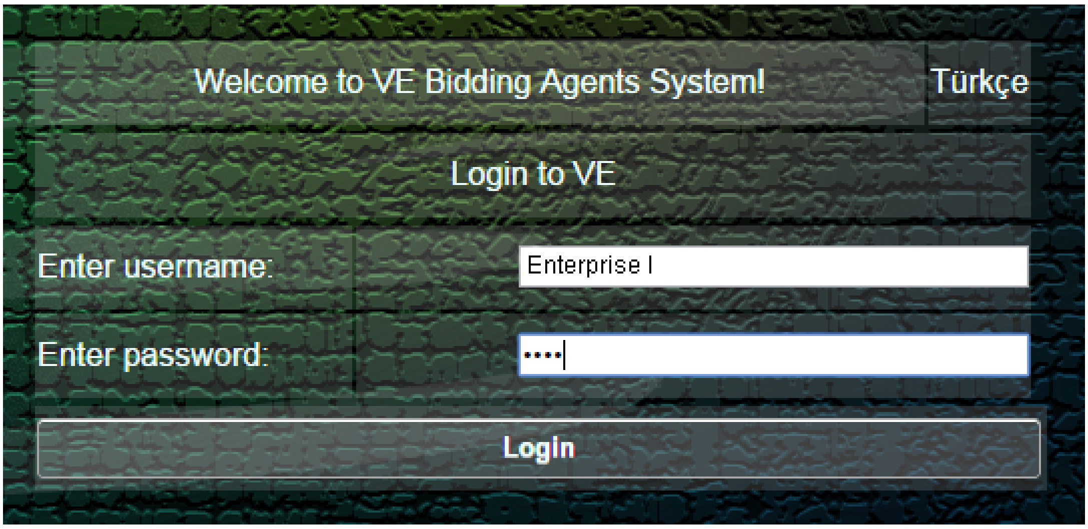


Figure 8.16: Enterprise login page

During each iteration, task manager agent announces to all competing enterprise agents, the last iteration's best price. Agents try to bid lower price to catch the next best bid which may be the last bid, because suddenly their bid may go below customer offered priceand auction process ends up. There are some restrictions for agents to put their next bid. As described in Section 5.3, agents are only allowed to bid in their own defined secure bidding gap which is shaped based on static and dynamic parameters of enterprise. Stronger enterprises have wider range for bidding however weaker enterprises have narrower gap [2].

Wider secure gap for bidding increases the chance of enterprise agent to win the auction and vice versa. If weaker enterprises are intending to win the negotiation any way, and challenge stronger companies they need to offer much competitive offers and follow more aggressive strategies. Absolutely it is not the case when the stronger enterprise bid lower prices as well. Because in this situation weaker enterprises will loose their only chance to win the negotiation. Otherwise they still have hope to win the bargaining procedure. There are lots of possibilities in this type of auctioning and it is really hard to predicate the winner enterprise and this unpredictability is an advantage of this auctioning system.

After getting requested parameters from belonging enterprise authorities, enterprise agent waits for the start signal from task manager agent. System administrator sets up a deadline, for incoming bidding approvals from enterprises and customer when opens tasks for bidding. On the pointed deadline task manager agent checks customer and enterprises' agents. If customer and acceptable number of enterprises were eager to join negotiation procedure, task manager agent orders to start the auction. Figure 8.18 shows a sample customer invitation letter. Customer enterprise is one side of negotiation in all tasks' auctioning procedure. Therefore the information regarding all tasks is sent to customer.

[2] Strong enterprise does not means bigger or richer enterprise. Likewise weak enterprise does not mean a poor and small enterprise. Strong and weak terms here mean, enterprise with better performance, efficiency, and more discipline versus enterprises those have poor performances, undisciplined and inefficient working policy

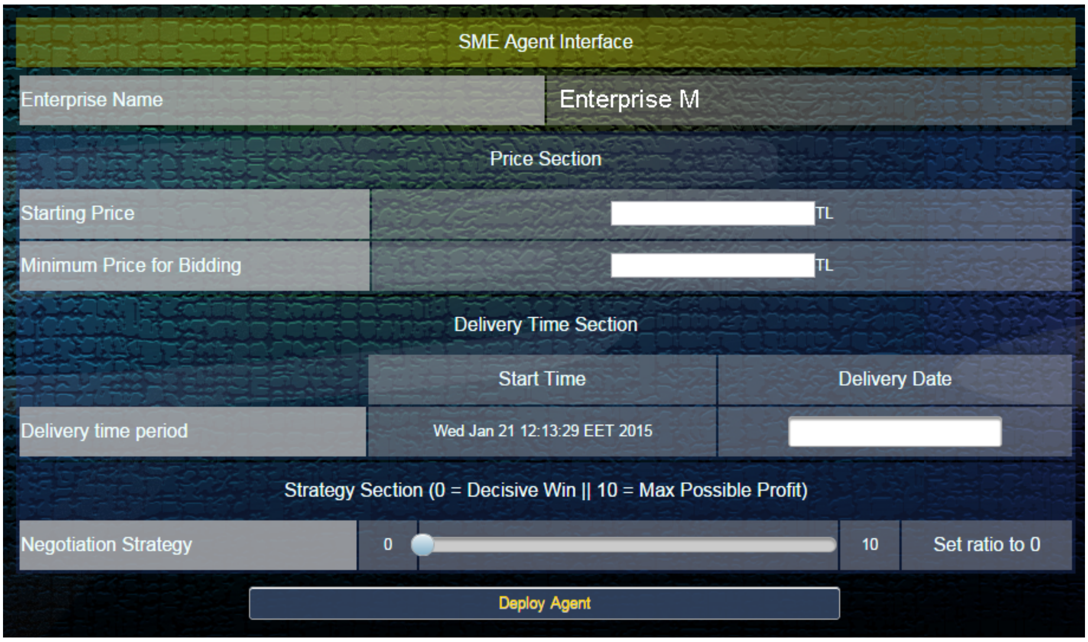


Figure 8.17: Enterprise agent interface

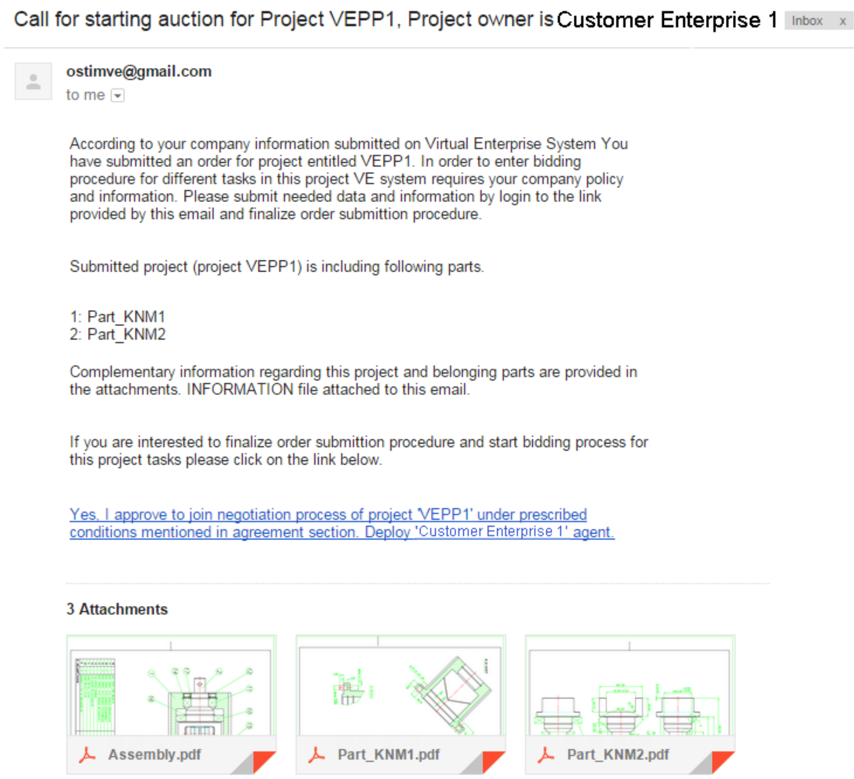
Customer login page is similar to enterprises login page. However, customer agent interface and questions type are quite different than enterprise agent's interface and questions. Customer agent questions are divided into two main sections (Figure 8.19). Upper part is criteria weight evaluation section. In this section customer is asked to (pairwise) compare 4 main criteria (as described in details in Section 6). After setting all pairwise comparisons, system check for consistency of selections, if it is consistent, each criteria weights will be calculated and displayed. Customer dynamic parameters are asked in the second part of the form. After approving all inputs by customer, and clicking over deploy agent, customer agent will be deployed. The pilot test is focusing on machining operations, specially on turning, milling and drilling operations.

8.3.1.3 Established VE Consortium for Test Parts KNM1 and KNM2

Test runs for establishing virtual enterprise consortium were accomplished for parts KNM1 and KNM2 of pilot product. In order to produce each part of product, it is compulsory to create an OMAVE consortium. Consortium members are winners of auctions of each task in a part. Parts KNM1 and KNM2 are containing following tasks;

1) Part KNM1

1-1) Milling Task 1
1-2) Milling Task 2
1-3) Painting Task 1
1-4) Rough Turning Task 1



Call for starting auction for Project VEPP1, Project owner is Customer Enterprise 1 Inbox x

ostimve@gmail.com
to me

According to your company information submitted on Virtual Enterprise System You have submitted an order for project entitled VEPP1. In order to enter bidding procedure for different tasks in this project VE system requires your company policy and information. Please submit needed data and information by login to the link provided by this email and finalize order submittion procedure.

Submitted project (project VEPP1) is including following parts.

1: Part_KNM1
2: Part_KNM2

Complementary information regarding this project and belonging parts are provided in the attachments. INFORMATION file attached to this email.

If you are interested to finalize order submittion procedure and start bidding process for this project tasks please click on the link below.

Yes, I approve to join negotiation process of project 'VEPP1' under prescribed conditions mentioned in agreement section. Deploy 'Customer Enterprise 1' agent.

3 Attachments

Assembly.pdf Part_KNM1.pdf Part_KNM2.pdf

Figure 8.18: Customer invitation letter sample

1-5) Turning Task 2

1-6) Grinding Task 2

2) Part KNM2

1-1) Rough Turning Task 1

1-2) Milling Task 1

1-3) Heat Treatment Task 1

1-4) Drilling Task 1

1-5) Grinding Task 1

1-6) Painting Task 1

As there were not sufficient information regarding blackening, heat treatment and grinding equipment from pilot platform members, the focus was on turning, milling and drilling tasks. In fact, there are not any differences in partner selection process characteristics and procedures in various manufacturing processes. There were 12 enterprises; Enterprise A, B, C, ..., and L. There were also three different customers with different expectations from enterprises. Customer Enterprise 1, Customer Enterprise 2 and Customer Enterprise 3 were considered to be three main customers for this system and it was assumed that each of these customers ordered the same product but with different partner selection attitudes. Therefore, according to customer requirement different final enterprise ranks will be populated. System was checked for

Figure 8.19: Customer agent interface

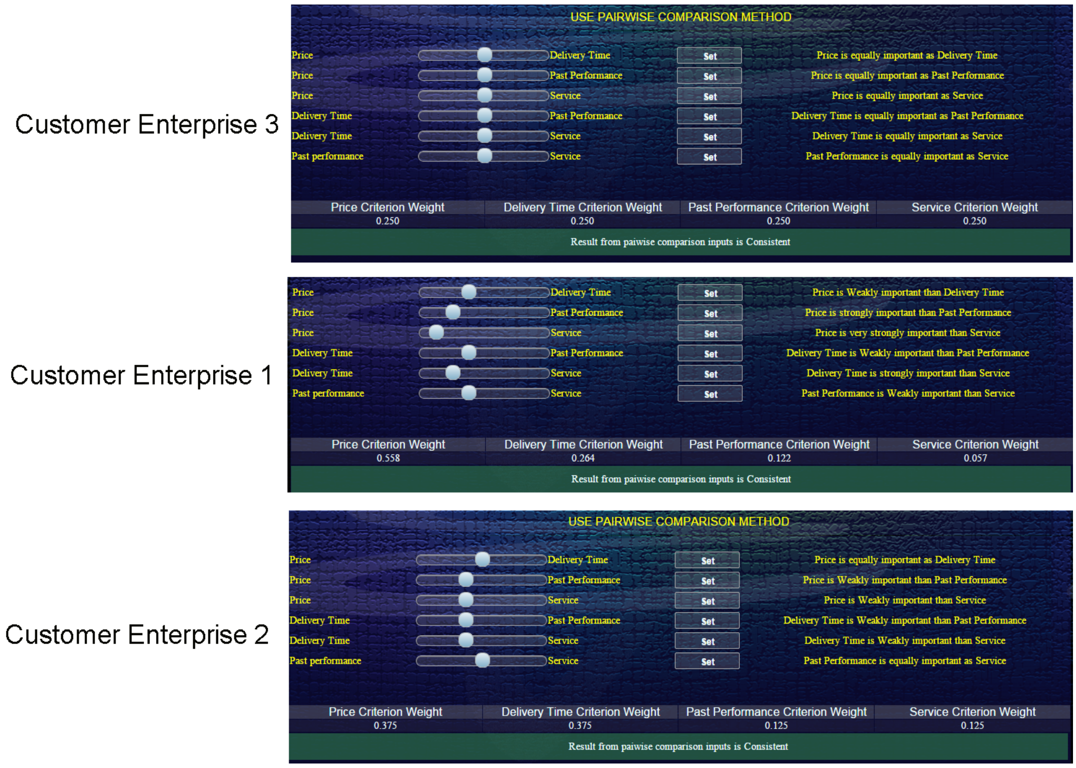


Figure 8.20: Three different customer configurations

different customer requirements with the same inputs from enterprises in each task auction. Figure 8.20 shows different customer policies of customers. These configurations resulted weights also is illustrated as follows;

8.4 Part KNM1

For part KNM1, based on these customer preferences 12 auctions were held. 8 enterprises joined to bid for Turning Task 1. These 8 companies proposed price, their delivery time, past performance and service points also are given in Table 8.2.

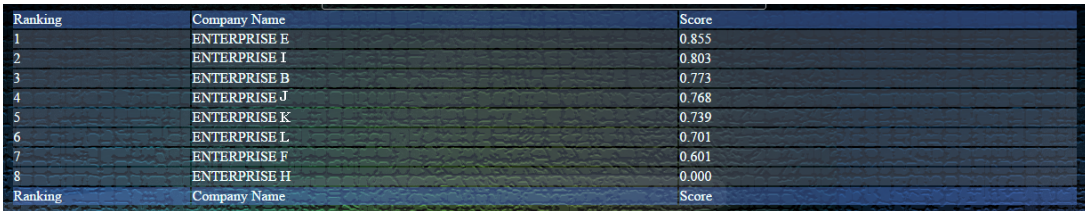
8.4.1 Part KNM1- Turning Task 1 Auction

According to these inputs ranked list of enterprises for KNM1-Turning Task 1 is given in Table 8.3. As these 8 enterprises satisfy the requirements of Turning Task 1, they are allowed by system automatically to enter the bidding procedure of this task.

Ranked enterprises in the system is shown in Figure 8.21. Results of auctioning for ***Part KNM1's Turning Task 1*** shows that according to the customer preferences the winner enterprise changes. If customer preferences is like *CUSTOMER ENTERPRISE 1*, the winner enterprise will be ***Enterprise E***. If customer preferences is like*CUSTOMER ENTERPRISE*

Table 8.2: Enterprise bids for KNM1- Turning Task 1

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE B	9,000 ₺	4	0.6	0.7
ENTERPRISE E	15,000 ₺	6	0.5	0.5
ENTERPRISE F	8,000 ₺	3	0.5	0.6
ENTERPRISE H	12,000 ₺	4	0.5	0.5
ENTERPRISE I	10,000 ₺	3	0.9	0.9
ENTERPRISE J	10,500 ₺	3	0.9	0.9
ENTERPRISE K	10,000 ₺	4	0.9	0.9
ENTERPRISE L	12,000 ₺	2	0.9	0.9



Ranking	Company Name	Score
1	ENTERPRISE E	0.855
2	ENTERPRISE I	0.803
3	ENTERPRISE B	0.773
4	ENTERPRISE J	0.768
5	ENTERPRISE K	0.739
6	ENTERPRISE L	0.701
7	ENTERPRISE F	0.601
8	ENTERPRISE H	0.000
Ranking	Company Name	Score

Figure 8.21: Ranked list of companies after auctioning- For ASELSAN customer configuration

Table 8.3: Results of KNM1- Turning Task 1 auctioning

Customer Enterprise 1			Customer Enterprise 2			Customer Enterprise 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE F	0.486	1	ENTERPRISE L	0.552	1	ENTERPRISE I	0.788
2	ENTERPRISE I	0.483	2	ENTERPRISE I	0.541	2	ENTERPRISE L	0.784
3	ENTERPRISE L	0.482	2	ENTERPRISE J	0.536	3	ENTERPRISE J	0.771
4	ENTERPRISE J	0.475	4	ENTERPRISE F	0.522	4	ENTERPRISE K	0.649
5	ENTERPRISE K	0.451	5	ENTERPRISE K	0.508	5	ENTERPRISE F	0.559
6	ENTERPRISE B	0.450	6	ENTERPRISE B	0.496	6	ENTERPRISE B	0.531
7	ENTERPRISE H	0.394	7	ENTERPRISE H	0.452	7	ENTERPRISE H	0.375
8	ENTERPRISE E	0.271	8	ENTERPRISE E	0.337	8	ENTERPRISE E	0.087

Table 8.4: Enterprise bids for KNM1- Turning Task 2

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE B	10,700 ₺	5	0.6	0.7
ENTERPRISE E	12,500 ₺	5	0.5	0.5
ENTERPRISE F	10,200 ₺	4	0.5	0.6
ENTERPRISE H	11,500 ₺	5	0.5	0.5
ENTERPRISE I	11,000 ₺	4	0.9	0.9
ENTERPRISE J	11,200 ₺	4	0.9	0.9
ENTERPRISE K	10,700 ₺	5	0.9	0.9
ENTERPRISE L	14,400 ₺	3	0.9	0.9

Table 8.5: Results of KNM1- Turning Task 2 auctioning

CUSTOMER ENTERPRISE 1			CUSTOMER ENTERPRISE 2			CUSTOMER ENTERPRISE 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE I	0.354	1	ENTERPRISE L	0.316	1	ENTERPRISE I	0.781
2	ENTERPRISE J	0.350	2	ENTERPRISE I	0.294	2	ENTERPRISE J	0.775
3	ENTERPRISE L	0.343	3	ENTERPRISE J	0.287	3	ENTERPRISE L	0.709
4	ENTERPRISE F	0.342	4	ENTERPRISE K	0.261	4	ENTERPRISE K	0.644
5	ENTERPRISE K	0.324	5	ENTERPRISE F	0.245	5	ENTERPRISE B	0.397
6	ENTERPRISE B	0.304	6	ENTERPRISE B	0.205	6	ENTERPRISE F	0.351
7	ENTERPRISE H	0.277	7	ENTERPRISE H	0.153	7	ENTERPRISE H	0.219
8	ENTERPRISE E	0.256	8	ENTERPRISE E	0.110	8	ENTERPRISE E	0.202

2, the winner will be ***Enterprise I*** and for preferences like *CUSTOMER ENTERPRISE 3*, the winner will be ***Enterprise I***. As it can be seen from Table 8.3 the whole ranking list is changed for different customers.

8.4.2 Part KNM1- Turning Task 2 Auction

Incoming bids for *KNM1- Turning Task 2* is given in Table 8.4. Based on these bids enterprise entered the auctioning procedure. As both tasks are similar and have the same manufacturing requirements nominated enterprises for auctioning process are the same.

Results of the negotiation and ranked list of enterprises for Turning Task 2 is shown in Table 8.5. In OMAVE system the ranked list of enterprises is shown like Figure 8.5.

For different customer configurations, the ranked list of enterprises were different. But the auctioning winner for two of cases were the same and ***ENTERPRISE I*** won first and third negotiations. In second negotiation winner was ***ENTERPRISE L***.

Table 8.6: Enterprise bids for KNM1- Milling Task 1

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE A	14,900 ₺	5	0.7	0.7
ENTERPRISE C	13,800 ₺	7	0.7	0.6
ENTERPRISE H	14,400 ₺	5	0.5	0.5
ENTERPRISE I	15,000 ₺	4	0.9	0.9
ENTERPRISE J	14,800 ₺	4	0.9	0.9
ENTERPRISE K	14,500 ₺	5	0.9	0.9
ENTERPRISE L	15,800 ₺	3	0.9	0.9

Table 8.7: Results of KNM1- Milling Task 1 auctioning

CUSTOMER ENTERPRISE 1			CUSTOMER ENTERPRISE 2			CUSTOMER ENTERPRISE 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE L	0.461	1	ENTERPRISE L	0.462	1	ENTERPRISE L	0.890
2	ENTERPRISE J	0.438	2	ENTERPRISE J	0.396	2	ENTERPRISE J	0.816
3	ENTERPRISE I	0.437	3	ENTERPRISE I	0.395	3	ENTERPRISE I	0.812
4	ENTERPRISE K	0.406	4	ENTERPRISE K	0.309	4	ENTERPRISE K	0.668
5	ENTERPRISE A	0.389	5	ENTERPRISE A	0.266	5	ENTERPRISE A	0.500
6	ENTERPRISE H	0.380	6	ENTERPRISE H	0.245	6	ENTERPRISE H	0.339
7	ENTERPRISE C	0.314	7	ENTERPRISE C	0.104	7	ENTERPRISE C	0.249

8.4.3 Part KNM1- Milling Task 1 Auction

Due to change in job properties, the companies having turning centers or lathes may not join this auction procedure and were automatically eliminated. Only companies with machine tools capable of doing milling operations were invited to put their bids for this task negotiation. In this auctioning 7 companies were eligible to enter the bidding procedure and incoming bids from these enterprises is shown in Table 8.6.

For KNM1-Milling Task 1 enterprise bids are given in table8.7.

The auctioning results for ***Part KNM1's Milling Task 1*** is illustrated in Table 8.7. The most important part of these results is the winner of auctions. In all the negotiations winner enterprise is the same. This means that ***ENTERPRISE L*** as the winner of all tasks had very competitive parameters.In other words, Enterprise L is a good company and entered this negotiation procedure with very competitive dynamic parameters like minimum price, delivery time.

8.4.4 Part KNM1- Milling Task 2 Auction

Here, almost everything is like Milling Task 1 auction, except enterprises bids are different. These two tasks are both milling operations with similar conditions. Therefore, it is quite

Table 8.8: Enterprise bids for KNM1- Milling Task 2

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE A	18,700 ₺	8	0.7	0.7
ENTERPRISE C	18,600 ₺	9	0.7	0.6
ENTERPRISE H	19,900 ₺	6	0.5	0.5
ENTERPRISE I	20,000 ₺	7	0.9	0.9
ENTERPRISE J	20,500 ₺	6	0.9	0.9
ENTERPRISE K	20,880 ₺	5	0.9	0.9
ENTERPRISE L	19,100 ₺	6	0.9	0.9

Table 8.9: Results of KNM1- Milling Task 2 auctioning

CUSTOMER ENTERPRISE 1			CUSTOMER ENTERPRISE 2			CUSTOMER ENTERPRISE 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE K	0.409	1	ENTERPRISE L	0.470	1	ENTERPRISE L	0.635
2	ENTERPRISE L	0.408	2	ENTERPRISE K	0.468	2	ENTERPRISE K	0.632
3	ENTERPRISE J	0.390	3	ENTERPRISE J	0.452	3	ENTERPRISE J	0.627
4	ENTERPRISE I	0.377	4	ENTERPRISE I	0.441	4	ENTERPRISE I	0.624
5	ENTERPRISE H	0.376	5	ENTERPRISE H	0.426	5	ENTERPRISE A	0.572
6	ENTERPRISE A	0.345	6	ENTERPRISE A	0.409	6	ENTERPRISE C	0.551
7	ENTERPRISE C	0.316	7	ENTERPRISE C	0.383	7	ENTERPRISE H	0.546

normal that the same list of eligible companies put their bids for entering the auctioning procedure. Table 8.8 shows the incoming bids from nominated enterprises for ***Part KNM1's Milling Task 2***.

As it is shown in table 8.8 KNM1-Milling Task 2 input information are different than KNM1-Milling Task 1 and here is the results (8.9).

Again here like ***Part KNM1's Milling Task 1***, Enterprise L's performance was very good. In this task (***Part KNM1's Milling Task 2***) in the first customer configuration the performance of ***Enterprise K*** was much better and won the negotiation but it seems that it was not enough competitive to bit enterprise L in the other negotiations and ***Enterprise L*** won those auctions.

8.4.5 Part KNM1- Consortium

Remaining tasks of KNM1 (Painting and Grinding tasks)due to lack of information could not be auctioned. However, technically they are not different than auctioned tasks. In those tasks also the same parameters, and the same auctioning procedures would be applied. In this case, it is considered that, Enterprise X is the winner of Painting task and Enterprise Y is the winner for Grinding task. Therfore, the consortium needed to produce part ***KNM1*** for different customer configurations is shown in Table 8.10.

Table 8.10: Customer configuration based KNM1 consortium results

Consortium Members for Different Customer Configurations for Part KNM1		
Customer Enterprise Name	Task Name	Winner Enterprise Name
Customer Enterprise 1	Turning Task 1	Enterprise F
	Turning Task 2	Enterprise I
	Milling Task 1	Enterprise L
	Milling Task 2	Enterprise K
	Painting Task 1	Enterprise X
	Grinding Task 1	Enterprise Y
Customer Enterprise 2	Turning Task 1	Enterprise L
	Turning Task 2	Enterprise L
	Milling Task 1	Enterprise L
	Milling Task 2	Enterprise L
	Painting Task 1	Enterprise X
	Grinding Task 1	Enterprise Y
Customer Enterprise 3	Turning Task 1	Enterprise I
	Turning Task 2	Enterprise I
	Milling Task 1	Enterprise L
	Milling Task 2	Enterprise L
	Painting Task 1	Enterprise X
	Grinding Task 1	Enterprise Y

List of PART KNM2 Tasks

TaskName	Categories	Machine Tools	Organizations
VEPP1- Part KNM2- Painting Task 1	Manufacturing Processes Painting Processes	No records found.	No records found.
VEPP1- Part KNM2- Drilling Task 1	Drilling Processes Machining Processes Manufacturing Processes	DMG Mori- DMU 70 Series DMG Mori- NTX 2000/1500s Deckel HM Series Mazak Integrex J-200 Mazak Variaxis II Mazak Integrex i-200 ST	Enterprise L Enterprise A Enterprise J Enterprise H Enterprise K Enterprise C Enterprise I
VEPP1- Part KNM2- Grinding Task 1	Grinding Processes Machining Processes Manufacturing Processes	No records found.	No records found.
VEPP1- Part KNM2- Rough Turning Task 1	Machining Processes Rough Turning Turning Processes Manufacturing Processes	DMG Mori CTX beta 1250 DMG Mori- NEF400 DMG Mori- NTX 2000/1500s Mazak Integrex J-200 EMCO Concept Turn 60 Mazak Integrex i-200 ST	Enterprise F Enterprise I Enterprise E Enterprise J Enterprise H Enterprise K Enterprise L Enterprise B
VEPP1- Part KNM2- Heat Treatment Task 1	Heat Treatment Processes Manufacturing Processes Hardening Processes	No records found.	No records found.
VEPP1- Part KNM2- Milling Task 1	Machining Processes Milling Processes Rough Milling Process Manufacturing Processes	DMG Mori- DMU 70 Series DMG Mori- NTX 2000/1500s Deckel HM Series Mazak Integrex J-200 Mazak Variaxis II Mazak Integrex i-200 ST	Enterprise L Enterprise A Enterprise J Enterprise H Enterprise K Enterprise C Enterprise I

Figure 8.22: List of tasks in part KNM2

8.5 Part KNM2

The conditions and customers for KNM2 auctions is also exactly like KNM1 but bidding companies and bids are different. In Figure 8.22 list of KNM2 tasks are illustrated. For part KNM2, Turning Task 1, Milling Task 1, and Drilling Task 1 are auctioned and the results are given in the following section.

8.5.1 KNM2- Turning Task 1

The contributing enterprises for turning task in KNM2 are exactly the same companies in KNM1 turning tasks. Because the part specifications are very close to each other, machine tools elimination due to part tolerances, size, power requirement and etc. lead to the exactly the same machine tools list. Therefore, the same machine tool owners are going to join the negotiation procedure. In Table 8.11 enterprise bids and their conditions are depicted;

And according to these bids the results are shown in Table 8.12;

Actually, by skimming through the incoming bids from volunteer enterprises, it could be prognosticated that Enterprise J's offer is much competitive than others. There were not any surprises in auctioning results and ***ENTERPRISE J*** was the winner for all customer configurations. But the ranking list of enterprises changed radically by customer configurations

Table 8.11: Enterprise bids for KNM2- Turning Task 1

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE B	11,400 ₺	4	0.6	0.7
ENTERPRISE E	11,000 ₺	5	0.5	0.5
ENTERPRISE F	11,750 ₺	3	0.5	0.6
ENTERPRISE H	11,250 ₺	4	0.5	0.5
ENTERPRISE I	11,500 ₺	4	0.9	0.9
ENTERPRISE J	11,800 ₺	3	0.9	0.9
ENTERPRISE K	11,200 ₺	5	0.9	0.9
ENTERPRISE L	11,000 ₺	6	0.9	0.9

Table 8.12: Results of KNM2- Turning Task 1 auctioning

CUSTOMER ENTERPRISE 1			CUSTOMER ENTERPRISE 2			CUSTOMER ENTERPRISE 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE J	0.429	1	ENTERPRISE J	0.412	1	ENTERPRISE J	0.935
2	ENTERPRISE F	0.402	2	ENTERPRISE F	0.347	2	ENTERPRISE I	0.792
3	ENTERPRISE I	0.395	3	ENTERPRISE I	0.328	3	ENTERPRISE K	0.632
4	ENTERPRISE B	0.376	4	ENTERPRISE B	0.277	4	ENTERPRISE L	0.525
5	ENTERPRISE H	0.369	5	ENTERPRISE H	0.260	5	ENTERPRISE B	0.506
6	ENTERPRISE K	0.356	6	ENTERPRISE K	0.230	6	ENTERPRISE F	0.476
7	ENTERPRISE E	0.331	7	ENTERPRISE L	0.171	7	ENTERPRISE H	0.369
8	ENTERPRISE L	0.310	8	ENTERPRISE E	0.152	8	ENTERPRISE E	0.252

specifically for third customer configurations.

8.5.2 KNM2- Milling Task 1

Enterprises with the capability of milling operation volunteered to bid for this task. As the milling operation specifications were very close to KNM1-Milling tasks specifications, the same nominated list of enterprises were populated by the system. As the milling process and shape of part were different, obviously, it affected incoming price and delivery time bids. Enterprises offered bids are shown in Table 8.13.

KNM2 Milling task 1 auction results are shown in table 8.14;

ENTERPRISE I won all the negotiations easily. However, since enterprises bids were very close to each other any changes in customer configurations, affected the ranked list of enterprises severely. For example, Enterprise H in first customer configuration was the second best enterprise but, in third customer configuration it became 6th enterprise. This is a very good example to show the effect customer preferences on negotiation procedure.

Table 8.13: Enterprise bids for KNM2- Drilling Task 1

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE A	12,900 ₺	5	0.7	0.7
ENTERPRISE C	12,400 ₺	7	0.7	0.6
ENTERPRISE H	13,500 ₺	4	0.5	0.5
ENTERPRISE I	13,000 ₺	5	0.9	0.9
ENTERPRISE J	15,000 ₺	5	0.9	0.9
ENTERPRISE K	12,800 ₺	6	0.9	0.9
ENTERPRISE L	12,500 ₺	7	0.9	0.9

Table 8.14: Results of KNM2- Milling Task 2 auctioning

CUSTOMER ENTERPRISE 1			CUSTOMER ENTERPRISE 2			CUSTOMER ENTERPRISE 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE I	0.380	1	ENTERPRISE I	0.309	1	ENTERPRISE I	0.818
2	ENTERPRISE H	0.377	2	ENTERPRISE H	0.306	2	ENTERPRISE J	0.753
3	ENTERPRISE A	0.365	3	ENTERPRISE J	0.291	3	ENTERPRISE K	0.681
4	ENTERPRISE J	0.362	4	ENTERPRISE A	0.264	4	ENTERPRISE L	0.583
5	ENTERPRISE K	0.349	5	ENTERPRISE K	0.243	5	ENTERPRISE A	0.569
6	ENTERPRISE L	0.317	6	ENTERPRISE L	0.212	6	ENTERPRISE H	0.429
7	ENTERPRISE C	0.302	7	ENTERPRISE C	0.148	7	ENTERPRISE C	0.329

8.5.3 KNM2- Drilling Task 1

Seven enterprises were nominated by system based on their resources to enter the bidding process for KNM2- Drilling task. In Table 8.15 volunteer enterprises' bids for this task is shown. The same scenario is also repeated here. Generally bids are close to each other therefore, enterprises past performance and service parameters become much more important for making decisions.

The results of KNM2- Drilling task negotiation are tabulated in Table 8.16. ***ENTERPRISE J*** once again won the negotiations. But note the ranking list. Like other negotiation processes

Table 8.15: Enterprise bids for KNM2- Drilling Task 1

Enterprise Name	Proposed Price	Delivery Time(days)	Past Performance	Service
ENTERPRISE A	12,900 ₺	5	0.7	0.7
ENTERPRISE C	12,400 ₺	7	0.7	0.6
ENTERPRISE H	13,500 ₺	4	0.5	0.5
ENTERPRISE I	13,000 ₺	5	0.9	0.9
ENTERPRISE J	13,200 ₺	4	0.9	0.9
ENTERPRISE K	12,800 ₺	6	0.9	0.9
ENTERPRISE L	12,500 ₺	7	0.9	0.9

Table 8.16: Results of KNM2- Drilling Task 1 auctioning

CUSTOMER ENTERPRISE 1			CUSTOMER ENTERPRISE 2			CUSTOMER ENTERPRISE 3		
Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points	Rank	Enterprise Name	Gained Points
1	ENTERPRISE J	0.481	1	ENTERPRISE J	0.517	1	ENTERPRISE J	0.795
2	ENTERPRISE I	0.439	2	ENTERPRISE I	0.412	2	ENTERPRISE I	0.691
3	ENTERPRISE H	0.424	3	ENTERPRISE H	0.365	3	ENTERPRISE L	0.526
4	ENTERPRISE C	0.400	4	ENTERPRISE L	0.277	4	ENTERPRISE H	0.476
5	ENTERPRISE A	0.397	5	ENTERPRISE C	0.265	5	ENTERPRISE K	0.418
6	ENTERPRISE L	0.396	6	ENTERPRISE A	0.263	6	ENTERPRISE A	0.417
7	ENTERPRISE K	0.355	7	ENTERPRISE K	0.217	7	ENTERPRISE C	0.392

described before, customer preferences and configurations become more and more important when incoming bids from enterprises are more similar to each other. Therefore, ranked list of enterprises get affected by any changes in customer parameters. But this changes does not have too much influences on enterprises with more competitive bids. This is also observable from Table 8.16. First two enterprises (Enterprise I and Enterprise J) kept their places in all ranking lists.

8.5.4 Part KNM2- Consortium

Like KNM1 consortium because of lack of information and absence of companies with the capabilities of Blackening , heat treatment and grinding related tasks negotiation procedure did,t accomplished. However, exactly the same procedure is valid for these tasks as well. Here, it is assumed that Enterprise Z is the winner enterprise for blackening task, Enterprise W is winner of heat treatment task and Enterprise V is the winner enterprise in grinding negotiation. By these assumptions and getting results from accomplished auctions it is possible to form the consortium for manufacturing KNM2. Looking at results Table 8.17, for different customer preferences, same consortium was formed up. A consortium containing (Enterprises; I,J, Z, V and W) is the solution for these customer preferences for part KNM2.

8.6 OMAVE operation

After signing agreements by stakeholders of formed OMAVE partners and customer, Dassault system accounts are assigned to all partners. According to partners' duties, they are authorized to view, edit, manage or supervise a particular job associated with product. All design and analyze information and data from CATIA design tool are supported in ENOVIA PLM and project administrator according to the agreed VE scheduling, establish project Gantt charts and designate project plan. Figure 8.23 shows a roles control panel of ENOVIA PLM tool.

All partners, system administrator and customer follow project plan and scheduling using their appointed Dassault accounts. Figure 8.24 displays a sample ENOVIA project management screen created for pilot VE project.

Table 8.17: Customer configuration based KNM2 consortium results

Consortium Members for Different Customer Configurations for Part KNM2		
Customer Enterprise Name	Task Name	Winner Enterprise Name
Customer Enterprise 1	Turning Task 1	Enterprise J
	Milling Task 1	Enterprise I
	Drilling Task 1	Enterprise J
	Grinding Task 1	Enterprise W
	Blackening Task 1	Enterprise Z
	Heat Treatment Task 1	Enterprise V
Customer Enterprise 2	Turning Task 1	Enterprise J
	Milling Task 1	Enterprise I
	Drilling Task 1	Enterprise J
	Grinding Task 1	Enterprise W
	Blackening Task 1	Enterprise Z
	Heat Treatment Task 1	Enterprise V
Customer Enterprise 3	Turning Task 1	Enterprise J
	Milling Task 1	Enterprise I
	Drilling Task 1	Enterprise J
	Grinding Task 1	Enterprise W
	Blackening Task 1	Enterprise Z
	Heat Treatment Task 1	Enterprise V

In order to get the updated data and information from enterprises activities, project progress reports and real time machinery capacity, in the later phases of OMAVE project agents interaction between ENOVIA database, OMAVE data store and enterprises established ERP, MRP or MES systems will be developed. To manage VE life cycle operation phase effectively these tool are missing. It also is considered to add a new web based light ERP, MRP systems beside OMAVE system to feed online, up to dated data and information from member enterprises to OMAVE data store. The system, which is going to be developed has special characteristics somehow different than available ERP systems. Considering the type of enterprises, the ERP system should require minimum maintenance costs in client side (enterprise). It should also have very simple and straightforward interfaces. In developed OMAVE system all data and information from enterprises are entering manually to ENOVIA system. But in the future, in order to guarantee the security and validity of data) absolutely it is necessary to automatize this information transform between enterprises and OMAVE system components.

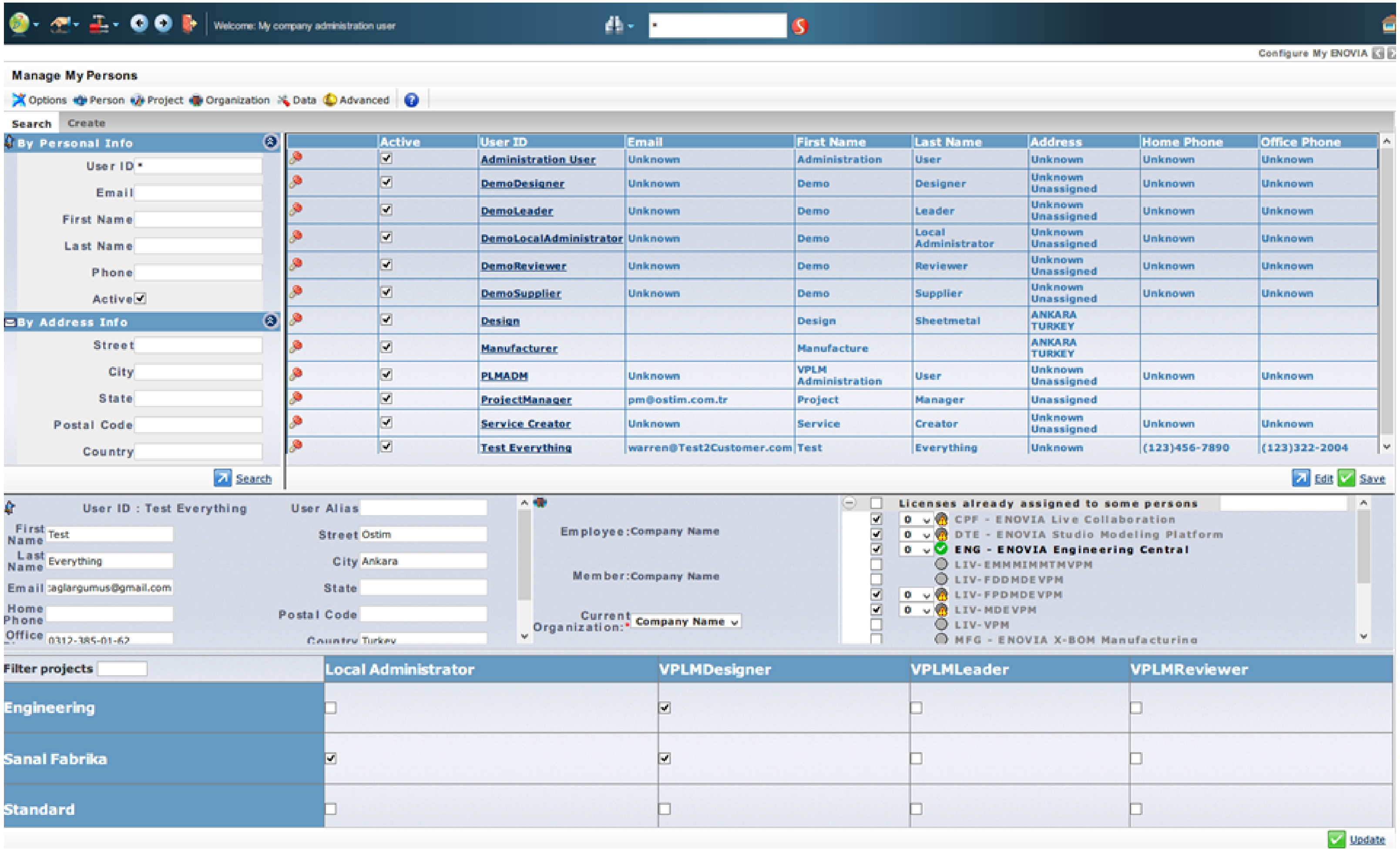


Figure 8.23: Roles control panel in ENOVIA PLM

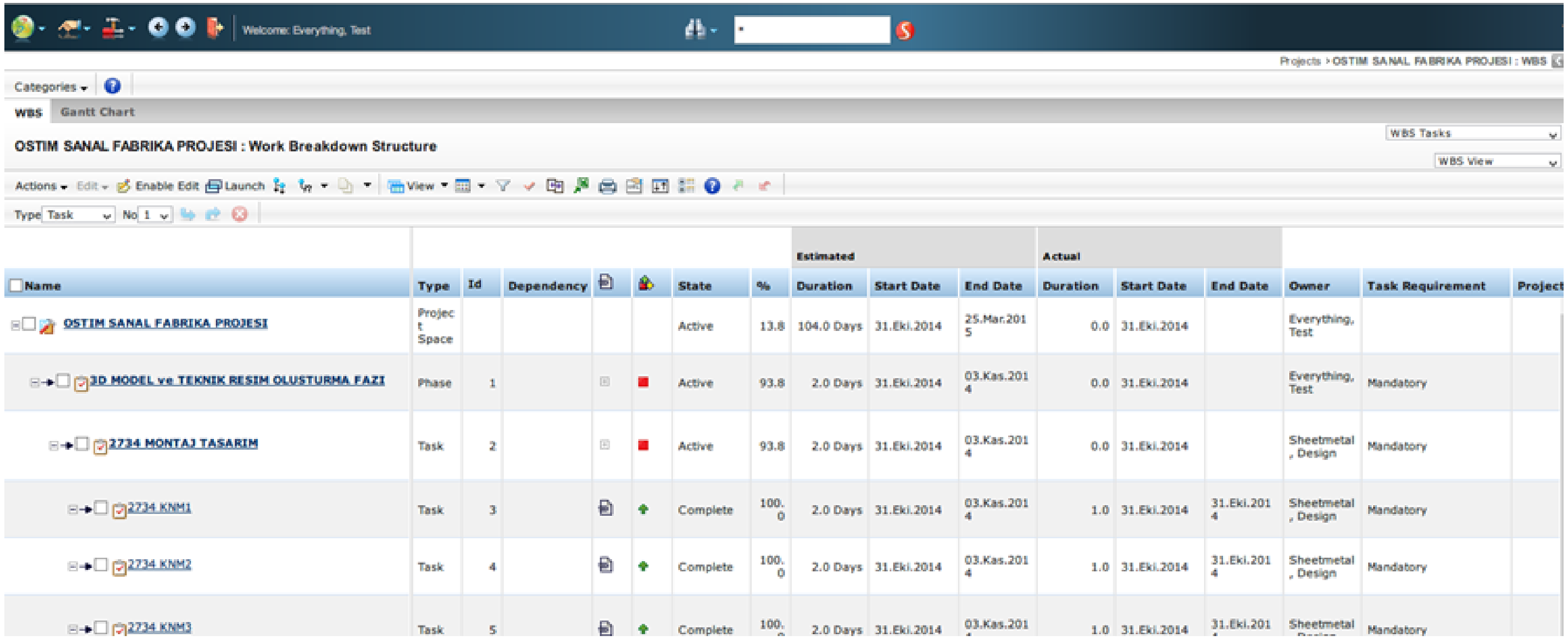


Figure 8.24: ENOVIA project manager panel

8.7 OMAVE Dissolution

Regarding the definition of VE, as a temporary consortium, once the project's goal is accomplished VE system will be dissolved. But the performance records of the completed tasks' partners should be evaluated and stored in system data-base to update and enrich the enterprises information.

When the product is handed over to the customer, a feedback oriented system gathers customers' satisfaction level based on partner enterprises communication skills, responsiveness and after-sale service. Customers give their ideas regarding these abstract factors via answering a questionnaire. Apart from the subjective factors, the product quality and on time delivery of product also should be reported. These data reflect the enterprises' commitment level. Responsible enterprises which could not deliver the qualified products on time will be penalized by decreasing their past performance score. Past performance of an enterprise is affected by two parameters; quality and on time delivery ratio, which are formulated as follows:

The quality and on time delivery scores of an enterprise could be calculated from equations 8.1 and 8.2.

$$QS = \frac{P_{acc}}{P_{TO}} \tag{8.1}$$

$$DS = 1 - r_i e^{l_i} \tag{8.2}$$

where;

$$r_i = \frac{D_{TP}}{P_{TO}} \tag{8.3}$$

$$l_i = \frac{d}{D} \tag{8.4}$$

QS : Enterprise quality score
DS : Enterprise on time delivery score
P_{acc} : Number of accepted parts
P_{TO} : Total number of ordered parts
D_{TP} : Number of late delivered parts
d : Delay duration
D : Total delivery time

In equation 8.2 term $r_i e^{l_i}$ formulates the penalty function of late delivery with an exponential function. As shown in Figure 8.25. The penalty increases exponentially by increasing the delay duration.

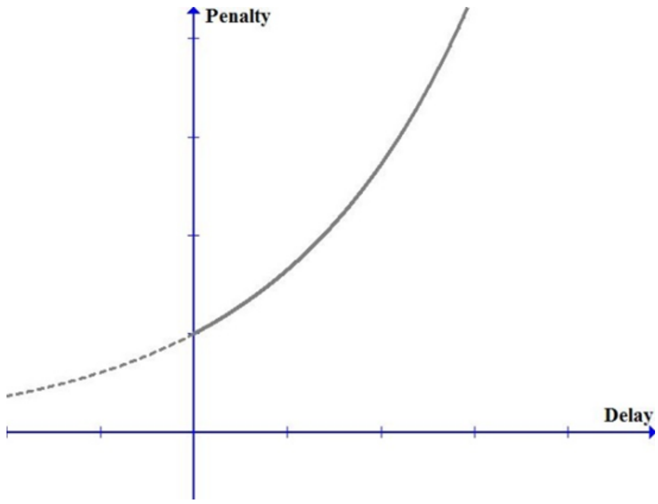


Figure 8.25: Penalty diagram

These data would return the past performance score of each enterprise. When a new VE consortium is going to be formed up in the future, these updated data will be called from system data base. Mentioned partner selection examples, were a proof of concept for proposed VE partner selection approach. Obviously, instead of manufacturing processes, any type of tasks or concepts could be replaced. This task formation is depend on the rules that system administrator defined to the system model. Here, in this example it is defined to form tasks based on a group of similar manufacturing processes. In other words, a task is a combination of similar manufacturing operations (Turning or milling or etc.). This concept could be replaced. For instance, instead of Turning task it is possible to put a part and a whole part could be opened for auctioning.

8.8 Summary

To verify system tools performance a sample use case product by taking advantage of developed OMAVE were manufactured. System performed as predicted and the results were acceptable. Sample product was containing 8 parts and for each part a VE consortium was created. Consortium members were the winners of tasks negotiations. Each task here is containing similar manufacturing processes. This sample use case is a proof of concept and does not necessarily means that each task should consists of manufacturing processes. Tasks may acquire all forms of components including manufacturing processes, parts, assembly products or whatever system administrator desires. Different steps of OMAVE system were tested and their performances were verified, in this use case study.

CHAPTER 9

CONCLUSIONS AND RECOMMENDATIONS

The idea of establishing a collaboration framework for small and medium sized enterprises to increase their competitiveness and the quality of products started from around three decades ago. Different attempts and researches were conducted in this era and some of them gained success in creating at least some tools to enhance VE creation and forming procedures.

The main problem regarding VE platforms is their highly dynamic nature and uncertainties in the system. It is some how easy to form up a VE system for a special order or project but established platform is not suitable enough to respond a new order comprehensively. Each product and each order has its own characteristics and specifications, even in some cases two consecutive orders has totally different or opposite specifications. In such cases developed structure fails and is unable to respond new requirements.

The other important shortcoming in VE systems is inventory management. Various types of enterprises, machine tools, manufacturing processes, products have to be handled. Setting a system for a special type of manufacturing procedure, is a big mistake and it will result in collapsing the system when it faces a new type of product with new type of requirements and completely different type of machine tools. This will force designers to change system hierarchy, structure and rebuild it from zero and this is a very expensive, time consuming.

Improvements in computer sciences, networking and ICT technologies bring new opportunities to overcome these obstacles by implementing different types of tools and developing a new form of collaboration platforms. One of these recently developed technologies is in the field of data bases and data management systems. In traditional relative data bases, any sudden radical changes in type and property of data will end in changing formed data base and developers were needed to restructure data base and schema. This means they are going to create a completely new system, so re usability of these systems specially in dynamic and changing environments like VE systems is very low. This was one of the limitations in enhancing VE systems flexibility.

Development of triple data stores is one of these advancements. Instead of keeping data in tables, relate them through different type of relations and create a very complicated data bases, the idea of keeping data and knowledge in form of triples or quads is very exciting.

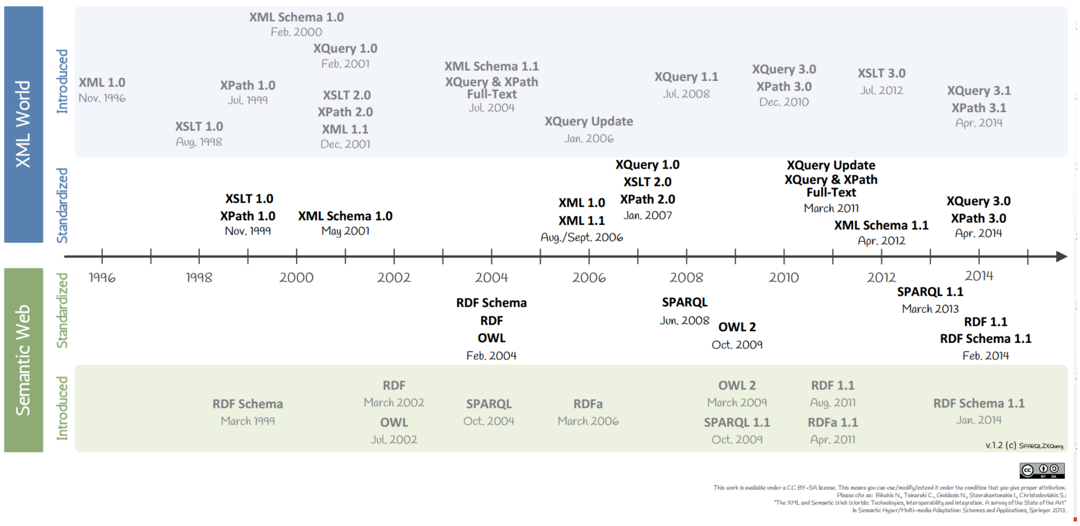


Figure 9.1: XML and Semantic Web World [82]

This means what everything could be defined and related using triples. Then it is possible to change and reconfigure system in any ways in any forms without loosing information and need for system reconstruction.

The other advancement is the progress in development of semantic infrastructures and web based tools to interact more efficiently and faster with clients and development of machine interpretable languages. The first attempts to enable machines to read and understand contents dates back to 1996 and started by introducing EXtended Markup Language (XML), Comma Separated Values (CSV) and Java Script Object Notation (JSON). But later on by increasing concentration on development of semantic web, W3C introduced RDF as metadata data models, which is a concept to keep data in triples format using XML serializations. This technology afterward extended under RDF Schema, RDFa and OWL languages. Figure 9.1 shows the time line of development and enhancement of machine readable formats and semantic web rapid formation and completion [82].

In this research, a new approach to VE systems is proposed by using the forementioned technologies above to face the flexibility problem of VE systems. Based on this proposal a new VE system developed and implemented in OSTIM Technology. OSTIM organized industrial zone is a vast industrial area in Ankara- Turkey that includes more than 5000 active enterprises in different industrial sectors. The manufacturing enterprises in OSTIM are divided into five main clusters by management, (namely aviation and defense cluster, bio Medicine cluster, renewable energy cluster, rail industry cluster, construction machines cluster). Enterprise with different automation levels, produce different types of products. This would be a very massive project to cover all enterprise here. This project is separated into 5 main phases. In each step has its own targets:

First Phase Develop VE system and implement in three pilot enterprises (This research)

Second Phase Develop system to cover 25-35 enterprises from Aviation and defense cluster (OSSA)

Third Phase Develop to cover all enterprises from OSSA

Forth Phase Cover all enterprises in OSTIM

Fifth Phase Serve to enterprises out of OSTIM flexible

This research covers up the first phase of this five stepped project. In this research, to achieve the aimed flexibility a machine readable and understandable infrastructure for VE was established. First step to establish such an infrastructure was the development of an ontology domain model of VE framework in OWL-DL format using protege ontology editor. All structural concepts, entities, relations, properties, rules and functions defined in this domain model.

A Triple Data Base (TDB) data store was created and benefiting from Apache Jena platform, VE system required management, administrative tools and GUIs were developed. User interfaces for developed web application formed on Java Server Faces (JSF) technologies. RDF query language SPARQL which is a semantic query language also was implemented in this framework to retrieve, update and manipulate data stored in TDB data stores of VE.

In order to make a flexible, neutral, objective and unpredictable negotiation process for OMAVE partner selection process, a multi agent based platform was created. This platform was developed on Java Agent Development Environment (JADE). Agents behavior and their decision making algorithms are different based on their associated enterprises. According to each enterprise conditions and parameters like (their past performance, quality performance, service, proposed min and starting prices, delivery times and the enterprise negotiation policy) enterprise's representing agent behavior and its functionality differ.

Agents' interactions and their decision mechanism is supported by agent ontology. Agent ontology is different than OMAVE domain ontology. Developed agent ontology defines, concepts, rules and transmitting message formats for agents. Agents, at the same time gain enterprises static information like(past performance, service and quality performance values) from developed VE domain ontology model and dynamic parameters like(prices, delivery times and negotiation policy) using provided special agent GUIs from authorities in enterprises.

A multi criteria decision making algorithm supports multi agent negotiation process to select the most appropriate partners. In This decision making procedure Fuzzy-AHP-TOPSIS approach is used. These algorithms are coupled to agents platform and working together. In each bidding iteration, these algorithms obtain negotiation results from agents, analyze and rank agents and wait for a new bidding round. All developed web applications (OMAVE tools), GUIs and agents are developed using Java Enterprise Edition programming language. To increase system efficiency, new version of Dassault Systems CAD/CAM/CAE and PLM tool is implemented in VE system. The new software specification, which makes it suitable for VE infrastructure, is it's working principle. This tool has a web based infrastructure and is installed on OSTIM technology central server and all clients are connecting to this server and according to the assigned licenses users may join consortium and work on the same project, product or part simultaneously and remotely. All project data are stored on server. The other important feature regarding this tool is ENOVIA which acts as a system information backbone and PLM tool. All design, engineering product life cycle data, and even project management data are kept in ENOVIA. In order to boost up VE efficiency and performance, and enhance VE operation phase management capabilities, it is required to establish the interaction between ENOVIA data base and VE data store.

9.0.1 Main Contributions of This Dissertation

The main contribution of this research in VE literature are as follows:

- Introducing a completely new approach for design and development of a flexible VE data structure, by benefiting from triple data stores advantages and OWL DL machine readable semantic rules.
- Ontology based modeling of VE systems. This modeling approach provided base requirements for developing semantic infrastructure of VE systems.
- Development of a multi agent based auctioning platform and a negotiation ontology for agents to be used in VE formation phase.
- Using a completely new multi criteria decision making approach for partner selection phase of VE. In this approach a Fuzzy-AHP-TOPSIS algorithm was developed by contribution of researchers from METU and TOBB ETU.

9.0.2 Future Works and Recommendations

Developed system meets the basic VE requirements and pilot test outcomes shows that this system is operational and is capable to respond VE system necessities. But in order to enhance system efficiency and performance it is needed to integrate some additional tools and applications like; enterprise ERP, MES systems to OMAVE system in the future.

1. One of the main concerns about this system is about collecting needed information and data from enterprises with lower levels of automation in a secure and continuous way automatically. One of the solutions for this trouble is to create auxiliary modules for OMAVE system like; web based, light and easy to use ERP, MRP, MES systems systems integrated to OMAVE main platform and encourage small and micro enterprises to use these systems. The integrated platform may establish a smooth flow of data in OMAVE system and helps to get real time data from partners and keep system data store updated.

2. The other issue is to integrate design tools like; CATIA V6 with process planning and scheduling tools of VE. In this way development of a design ontology by implementing STEP standards could be tested for simple round parts and later on more complicated prismatic parts.

3. Integrating information management tools is the another proposition of author to be implemented in OMAVE system. To manage OMAVE projects specially during the operation phase detailed online information flow between system components is highly required. To monitor enterprise's conditions and capacity integrated ERP or MRP systems may help VE operation manager, and to follow product data, PLM devices like ENOVIA could be integrated to OMAVE operation management system and act synchronized with OMAVE project manager smoothly.

REFERENCES

[1] N. Aeronautics and S. Administration. Nasa systems engineering handbook. Technical Report NASA/SP-2007-6105- Rev1, NASA Headquarters, Paris, December 2007.

[2] A. Aerts, N. Szirbik, and J. Goossenaerts. A flexible, agent-based {ICT} architecture for virtual enterprises. *Computers in Industry*, 49(3):311 – 327, 2002.

[3] Y. Al-Safi and V. Vyatkin. An ontology-based reconfiguration agent for intelligent mechatronic systems. In V. Mařík, V. Vyatkin, and A. Colombo, editors, *Holonic and Multi-Agent Systems for Manufacturing*, volume 4659 of *Lecture Notes in Computer Science*, pages 114–126. Springer Berlin Heidelberg, 2007.

[4] A. Alzaga and J. Martin. A design process model to support concurrent project development in networks of smes. In L. Camarinha-Matos and H. Afsarmanesh, editors, *Infrastructures for Virtual Enterprises*, volume 27 of *IFIP — The International Federation for Information Processing*, pages 307–318. Springer US, 1999.

[5] S. Ambler. Uml 2 sequence diagrams," agile modeling, 2003. Available from: http://www.agilemodeling.com/artifacts/sequenceDiagram.htm/.

[6] S. as-a Service Executive Council. Software-as-a-service; a comprehensive look at the total cost of ownership of software applications, 2006.

[7] D. Beckett and T. Berners-Lee. Turtle - terse rdf triple language, March 2011. http://www.w3.org/TeamSubmission/turtle/.

[8] D. Bell. Uml basics: An introduction to the unified modeling language, 2003. Available from:http://www.ibm.com/developerworks/rational/library/769.html/.

[9] F. L. Bellifemine, G. Caire, and D. Greenwood. *Developing multi-agent systems with JADE*, volume 7. John Wiley & Sons, 2007.

[10] T. Berners-Lee and D. Connolly. Notation3 (n3): A readable rdf syntax, March 2011. http://www.w3.org/TeamSubmission/n3/.

[11] P. Bernus and L. Nemes. A framework to define a generic enterprise reference architecture and methodology. *Computer Integrated Manufacturing Systems*, 9(3):179 – 191, 1996.

[12] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau. Extensible markup language (xml) 1.0 (fifth edition), November 2008. http://www.w3.org/TR/REC-xml/sec-notation.

[13] C. Bremer, F. Michilini, J. Siqueira, and L. Ortega. Virtec: An example of a brazilian virtual organization. *Journal of Intelligent Manufacturing*, 12(2):213–221, 2001.

[14] J. Browne and J. Zhang. Extended and virtual enterprises – similarities and differences. *International Journal of Agile Management Systems*, 1(1):30–36, 1999.

[15] J. Buckley. Fuzzy hierarchical analysis. *Fuzzy Sets and Systems*, 17(3):233 – 247, 1985.

[16] J. A. Byrne, R. Brandt, and O. Port. The virtual corporation, February 1993. Available from: http://www.businessweek.com/stories/1993-02-07/the-virtual-corporation/.

[17] L. Camarinha-Matos and H. Afsarmanesh. Virtual enterprise modeling and support infrastructures: Applying multi-agent system approaches. In M. Luck, V. Mařík, O. Štěpánková, and R. Trappl, editors, *Multi-Agent Systems and Applications*, volume 2086 of *Lecture Notes in Computer Science*, pages 335–364. Springer Berlin Heidelberg, 2001.

[18] L. Camarinha-Matos, H. Afsarmanesh, C. Garita, and C. Lima. Towards an architecture for virtual enterprises. *Journal of Intelligent Manufacturing*, 9(2):189–199, 1998.

[19] L. M. Camarinha-Matos and H. Afsarmanesh. Elements of a base {VE} infrastructure. *Computers in Industry*, 51(2):139 – 163, 2003. Virtual Enterprise Management.

[20] L. M. Camarinha-Matos and H. Afsarmanesh. A framework for virtual organization creation in a breeding environment. *Annual Reviews in Control*, 31(1):119 – 135, 2007.

[21] R. Chalmeta and R. Grangel. Performance measurement systems for virtual enterprise integration. *International Journal of Computer Integrated Manufacturing*, 18(1):73–84, 2005.

[22] B. Chandrasekaran, J. R. Josephson, and V. R. Benjamins. The ontology of tasks and methods, 1998.

[23] A. Charnes, W. Cooper, and E. Rhodes. Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2(6):429 – 444, 1978.

[24] R.-S. Chen and M. A. Tu. Development of an agent-based system for manufacturing control and coordination with ontology and {RFID} technology. *Expert Systems with Applications*, 36(4):7581 – 7593, 2009.

[25] Y. Cho, C. Leem, and K. Shin. An assessment of the level of informatization in the korea mold industry as a prerequisite for e-collaboration: an exploratory empirical investigation. *The International Journal of Advanced Manufacturing Technology*, 29(9-10):897–911, 2006.

[26] K. H. Choi, D. S. Kim, and Y. H. Doh. Multi-agent-based task assignment system for virtual enterprises. *Robotics and Computer-Integrated Manufacturing*, 23(6):624

– 629, 2007. 16th International Conference on Flexible Automation and Intelligent Manufacturing.

[27] G. Cândido and J. Barata. A multiagent control system for shop floor assembly. In V. Mařík, V. Vyatkin, and A. W. Colombo, editors, *Holonic and Multi-Agent Systems for Manufacturing*, volume 4659 of *Lecture Notes in Computer Science*, pages 293–302. Springer Berlin Heidelberg, 2007.

[28] s. COMMISSION RECOMMENDATION of 6 May 2003 concerning the definition of micro and medium-sized enterprises. Commission. *Official Journal of the European Union*, L 124(C(2003) 1422):36–41, 2003.

[29] S. Cranefield, S. Haustein, and M. Purvis. Uml-based ontology modelling for software agents. *The Information Science Discussion Paper Series*, 2001(07), May 2001.

[30] R. Daviddrajuh and Z. Deng. Identifying potential suppliers for formation of virtual manufacturing systems. In *Proceedings of the World Computer Congress, Track on Information Technology for Business Management*, 2000.

[31] D. de Chambers, F. Báguena, and M. G. Fernández. *Production Planning and Control Information System for the Engineering and Make to Order Environment. A Virtual Enterprise Approach*, volume 56 of *IFIP — The International Federation for Information Processing*, pages 333–340. Springer US, 2001.

[32] V. Dominik and L. Peter. Distributed model management platform for virtual enterprise networks. In *Network-Centric Collaboration and Supporting Frameworks*, volume 224 of *IFIP International Federation for Information Processing*, pages 507–514. Springer US, 2006.

[33] Eurostat. Europe in figures, eurostat yearbook 2012. Technical report, Eurostat European Commision, Luxembourg, 2012.

[34] F. Fadel, M. S. Fox, and M. Gruninger. A resource ontology for enterprise modelling. In *Third Workshop on Enabling Technologies-Infrastructures for Collaborative Enterprises*, pages 117–128, 1994.

[35] W. Feng and R. Wang. Multi-agent based virtual enterprise. In *International Conference on Computer Technology and Development. (ICCTD '09)*, 2012.

[36] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Iiop: Omg's internet inter-orb protocol- a brief description, June 1999. https://tools.ietf.org/html/rfc2616.

[37] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Iiop: Omg's internet inter-orb protocol- a brief description, June 1999. https://tools.ietf.org/html/rfc2616.

[38] S. C. for Biomedical Informatics Research. Protege. Available from: http://protege.stanford.edu/overview/.

[39] T. A. S. Foundation. Apache jena a free and open source java framework for building semantic web and linked data applications. https://jena.apache.org/index.html.

[40] M. Fowler. *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. The Addison-Wesley object technology series. Addison-Wesley, 2004.

[41] U. Franke. *Managing Virtual Web Organizations in the 21st Century: Issues and Challenges: Issues and Challenges*. Idea Group Pub., 2001.

[42] A. Frenkel, H. Afsarmanesh, C. Garita, and L. Hertzberger. Supporting information access rights and visibility levels in virtual enterprises. In L. Camarinha-Matos, H. Afsarmanesh, and R. Rabelo, editors, *E-Business and Virtual Enterprises*, volume 56 of *IFIP — The International Federation for Information Processing*, pages 177–192. Springer US, 2001.

[43] S. Friedenthal, A. Moore, and R. Steiner. *A Practical Guide to SysML: The Systems Modeling Language*. The MK/OMG Press. Elsevier Science, 2011.

[44] B. Glimm, I. Horrocks, B. Motik, G. Stoilos, and Z. Wang. Hermit: an owl 2 reasoner. *Journal of Automated Reasoning*, pages 1–25, 2013.

[45] GLOBEMEN. Global engineering and manufacturing in enterprise networks- globemen. Announcing the standard for, VTT Technical Research Centre of Finland,, Helsinky, December 2002.

[46] S. Goldman, R. Nagel, and K. Preiss. *Agile Competitors and Virtual Organizations: Strategies for Enriching the Customer*. Industrial Engineering Series. Van Nostrand Reinhold, 1995.

[47] H. Goranson. *The Agile Virtual Enterprise: Cases, Metrics, Tools*. Quorum Books, 1999.

[48] H. Gou, B. Huang, W. Liu, Y. Li, and S. Ren. Agent-based virtual enterprise modeling and operation control. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, volume 3, pages 2058–2063 vol.3, 2001.

[49] C. F. Gray and E. W. Larson. *Project Management- The Managerial Process*. McGraw- Hill, Irwin, 5th ed. edition, 2000.

[50] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199 – 220, 1993.

[51] T. R. Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International Journal of Human-Computer Studies*, 43(5–6):907 – 928, 1995.

[52] A. Gunasekaran. *Agile Manufacturing: The 21st Century Competitive Strategy: The 21st Century Competitive Strategy*. Elsevier Science, 2001.

[53] J. Heflin. Owl web ontology language, February 2004. Available from: http://www.w3.org/TR/owl-features/.

[54] B. E. Hirsch, T. Kuhlmann, Z. K. Marciniak, and C. Masow. Information system concept for the management of distributed production. *Computers in Industry*, 26(3):229 – 241, 1995. Computer Integrated Manufacturing and Industrial Automation.

[55] D. Hollingsworth. Workflow management coalition the workflow reference model, 1994. Available from: http://www.aiai.ed.ac.uk/project/wfmc/ARCHIVE/DOCS/refmodel/rmv1-16.html/.

[56] I. Horrocks. The fact system. In *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 307–312. Springer, 1998.

[57] I. Horrocks, P. F. P. Schneider, H. Boley, S. Tabet, B. Gorosof, and M. Dean. Swrl: A semantic web rule language combining owl and ruleml, May 2004. Available from: http://www.w3.org/Submission/SWRL/.

[58] F.-S. Hsieh and J.-B. Lin. Context-aware workflow management for virtual enterprises based on coordination of agents. *Journal of Intelligent Manufacturing*, 25(3):393–412, 2014.

[59] X. Huang, Y. Wong, and J. Wang. A two-stage manufacturing partner selection framework for virtual enterprises. *International Journal of Computer Integrated Manufacturing*, 17(4):294–304, 2004.

[60] IDEF. Idef integrated definition methods, 2010. Available from: http://www.idef.com.//.

[61] IDEF0. Integration definition for function modeling (idef0). Announcing the Standard for 183, Draft Federal Information Processing Standards Publication, December 1993.

[62] H. S. Jagdev and K.-D. Thoben. Anatomy of enterprise collaborations. *Production Planning and Control*, 12(5):437–451, 2001.

[63] P. V. Jan de Kok, W. Verhoeven, N. Timmermans, T. Kwaak, J. Snijders, and F. Westhof. Do smes create more and better jobs? Technical report, EIM Business & Policy Research, Zoetermeer, November 2011.

[64] T. Kaihara, S. Fujii, and K. Iwata. Virtual enterprise coalition strategy with game theoretic multi-agent paradigm. *{CIRP} Annals - Manufacturing Technology*, 55(1):513 – 516, 2006.

[65] A. P. Klen, R. J. Rabelo, L. M. Spinosa, and A. C. Ferreira. Integrated logistics in the virtual enterprise: The prodnet-ii approach. In *Proceedings of 5th IFAC Workshop on Intelligent Manufacturing Systems (IMS'98)*, 1998.

[66] H. Knublauch, D. Oberle, P. Tetlow, and E. Wallace. A semantic web primer for object-oriented software developers, 2006.

[67] G. Koppensteiner, M. Merdan, A. Zoitl, and B. Favre-Bulle. Ontology-based resource allocation in distributed systems using director facilitator agents. In *Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on*, pages 1721–1726, June 2008.

[68] G. H. Kruger, A. J. Shih, D. G. Hattingh, and T. I. van Niekerk. Intelligent machine agent architecture for adaptive control optimization of manufacturing processes. *Advanced Engineering Informatics*, 25(4):783 – 796, 2011. Special Section: Advances and Challenges in Computing in Civil and Building Engineering.

[69] T. Kunimoto. Lecture note on auctions. *Department of Economics, McGill University*, 2008.

[70] A. Lazcano, G. Alonso, C. Schuler, and C. Schuler. The wise approach to electronic commerce. *International Journal of Computer Systems Science & Engineering, special issue on Flexible Workflow Technology Driving the Networked Economy*, 15:2000, 2000.

[71] M. Lim and Z. Zhang. A multi-agent system using iterative bidding mechanism to enhance manufacturing agility. *Expert Systems with Applications*, 39(9):8259 – 8273, 2012.

[72] H. Lin and J. Harding. A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration. *Computers in Industry*, 58(5):428 – 437, 2007.

[73] E. Lukacs. The economic role of smes in world economy, especially in europe. Technical report, European Integration Studies, Miskolc, Hungary, 2005.

[74] B. M., K. S., R. Yacine, and V. M. An open system for inter-enterprise information management in dynamic virtual enterprises. In *Proceedings of the 2nd World Wide ECCE Symposium Information and Communication Technology in the Practice of Building and Civil Engineering*, pages 1001–1006, 2001.

[75] W. Mangra. *Ontology based methodology for domain modelling*. Master (master of science in information systems), Athabasca, Alberta, Athabasca, Alberta, April 2006.

[76] R. J. Mayer, J. W. Crump, R. Fernandes, A. Keen, and M. K. Painter. Information integration for concurrent engineering (iice) compendium of methods report. Interim technical rept. Feb 1991-Mar 1995 ADA531128, KNOWLEDGE BASED SYSTEMS INC COLLEGE STATION TX, June 1995.

[77] I. McCarthy and M. Menicou. A classification schema of manufacturing decisions for the grai enterprise modelling technique. *Computers in Industry*, 47(3):339–355, 2002.

[78] G. Meditskos and N. Bassiliades. Dlejena: A practical forward-chaining owl 2 rl reasoner combining jena and pellet. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(1), 2010.

[79] M. Merdan, G. Koppensteiner, I. Hegny, and B. Favre-Bulle. Application of an ontology in a transport domain. In *Industrial Technology, 2008. ICIT 2008. IEEE International Conference on*, pages 1–6, April 2008.

[80] A. Molina and M. Flores. Exploitation of business opportunites: The role of the virtual enterprise broker. In L. Camarinha-Matos, H. Afsarmanesh, and R. Rabelo, editors, *E-Business and Virtual Enterprises*, volume 56 of *IFIP — The International Federation for Information Processing*, pages 269–280. Springer US, 2001.

[81] J. Mun, M. Shin, K. Lee, and M. Jung. Manufacturing enterprise collaboration based on a goal-oriented fuzzy trust evaluation model in a virtual enterprise. *Computers & Industrial Engineering*, 56(3):888 – 901, 2009. Intelligent Manufacturing and Logistics.

[82] B. N., T. C., G. N., S. I., and C. S. The xml and semantic web worlds: Technologies, interoperability and integration. a survey of the state of the art. In *Semantic Hyper/Multi-media Adaptation: Schemes and Applications*, pages 319–360. Springer, 2013.

[83] Y. E. Nahm and H. Ishikawa. A hybrid multi-agent system architecture for enterprise integration using computer networks. *Robotics and Computer-Integrated Manufacturing*, 21(3):217 – 234, 2005.

[84] O. Noran. Towards a meta-methodology for collaborative networked organisations. In L. Camarinha-Matos, editor, *Virtual Enterprises and Collaborative Networks*, volume 149 of *IFIP International Federation for Information Processing*, pages 71–78. Springer US, 2004.

[85] T. J. Norman, A. Preece, S. Chalmers, N. R. Jennings, M. Luck, V. D. Dang, T. D. Nguyen, V. Deora, J. Shao, W. Gray, and N. J. Fiddian. Agent-based formation of virtual organisations. *Knowledge-Based Systems*, 17(2–4):103 – 111, 2004. {AI} 2003, the Twenty-third {SGAI} International Conference on Innovative Techniques and Applications of Artificial Intelligence.

[86] N. F. Noy, D. L. McGuinness, et al. Ontology development 101: A guide to creating your first ontology, 2001.

[87] H. . Ünver and B. L. Sadigh. *Small and Medium Enterprises: Concepts, Methodologies, Tools, and Applications*, chapter An agent-based operational virtual enterprise framework enabled by RFID, pages 198–215. IGI Global, Web., 2013.

[88] K. H. Park and J. Favrel. Virtual enterprise-information system and networking solution. *Computers & Industrial Engineering*, 37(12):441–444, 1999. Proceedings of the 24th international conference on computers and industrial engineering.

[89] D. D. Paro. Wireless application protocol (wap): "what is it all about....how does it work", 2001.

[90] B. Parsia and E. Sirin. Pellet: An owl dl reasoner. In *In Proceedings of the International Workshop on Description Logics*, page 2003, 2004.

[91] S. A. Petersen, M. Divitini, and M. Matskin. An agent-based approach to modelling virtual enterprises. *Production Planning and Control*, 12(3):224–233, 2001.

[92] R. Rabelo, H. Afsarmanesh, and L. Camarinha-Matos. Federated multi-agent scheduling in virtual enterprises. In L. Camarinha-Matos, H. Afsarmanesh, and R. Rabelo, editors, *E-Business and Virtual Enterprises*, volume 56 of *IFIP — The International Federation for Information Processing*, pages 145–156. Springer US, 2001.

[93] R. Rabelo, L. Camarinha-Matos, and H. Afsarmanesh. Multiagent perspectives to agile scheduling. In L. Camarinha-Matos, H. Afsarmanesh, and V. Marik, editors, *Intelligent Systems for Manufacturing*, volume 1 of *IFIP — The International Federation for Information Processing*, pages 51–66. Springer US, 1998.

[94] R. Rabelo, L. Camarinha-Matos, and H. Afsarmanesh. Multi-agent-based agile scheduling. *Robotics and Autonomous Systems*, 27(1–2):15 – 28, 1999. Multi-Agent Systems Applications.

[95] R. J. Rabelo. Interoperating standards in multiagent agile manufacturing scheduling systems. *Int. J. Comput. Appl. Technol.*, 18(1-4):146–159, July 2003.

[96] R. J. Rabelo, L. M. Camarinha-Matos, and R. V. Vallejos. Agent-based brokerage for virtual enterprise creation in the moulds industry. In *E-business and Virtual Enterprises*, pages 281–290. Springer, 2001.

[97] A. Rocha and E. Oliveira. An electronic market architecture for the formation of virtual enterprises. In L. Camarinha-Matos and H. Afsarmanesh, editors, *Infrastructures for Virtual Enterprises*, volume 27 of *IFIP — The International Federation for Information Processing*, pages 421–432. Springer US, 1999.

[98] J. Rush. Ibm i and saas positioning ibm i as a software as a service platform, 2010. Available from: https://www-304.ibm.com/partnerworld/wps/servlet/ContentHandler?contentId=XhrjdgScEXMiPCA$cnt&roadMapI.

[99] T. L. Saaty. *What is the analytic hierarchy process?* Springer, 1988.

[100] T. L. Saaty. Decision making with dependence and feedback: The analytic network process, 1996.

[101] B. Sadigh, H. Ünver, and S. Kılıç. Design of a multi agent based virtual enterprise framework for sustainable production. In G. Putnik and M. Cruz-Cunha, editors, *Virtual and Networked Organizations, Emergent Technologies and Tools*, volume 248 of *Communications in Computer and Information Science*, pages 186–195. Springer Berlin Heidelberg, 2012.

[102] B. L. Sadigh, F. Arikan, A. Ozbayoglu, H. Unver, and S. Kilic. A multi-agent system model for partner selection process in virtual enterprise. *Procedia Computer Science*, 36(0):367 – 372, 2014. Complex Adaptive Systems Philadelphia, {PA} November 3-5, 2014.

[103] B. L. Sadigh, H. O. Unver, E. Dogdu, and S. E. Kilic. An ontology based model for virtual enterprise. In *Proceedings of 15th International Conference on Machine Design and Production, UMTIK 2012*, pages 885–898, Ankara, 19–22 June 2012 2012.

[104] B. L. Sadigh, H. Ö. Ünver, and S. E. Kılıç. Design of a multi agent based virtual enterprise framework for sustainable production. In *Virtual and Networked Organizations, Emergent Technologies and Tools*, pages 186–195. Springer, 2012.

[105] F. Sandakly, J. Garcia, P. Ferreira, and P. Poyet. Perdis: An infrastructure for cooperative engineering in virtual enterprise. In L. Camarinha-Matos and H. Afsarmanesh, editors, *Infrastructures for Virtual Enterprises*, volume 27 of *IFIP — The International Federation for Information Processing*, pages 319–332. Springer US, 1999.

[106] B. Sari. *Methodology Development for Small and Medium Sized Enterprise (SME) Based Virtual Enterprises*. Thesis (phd), Middle East Technical University, METU, Mech. Eng. Dept., 2006.

[107] B. Sari, T. Sen, and S. Kilic. Formation of dynamic virtual enterprises and enterprise networks. *The International Journal of Advanced Manufacturing Technology*, 34(11-12):1246–1262, 2007.

[108] B. Sari, T. Sen, and S. Kilic. Ahp model for the selection of partner companies in virtual enterprises. *The International Journal of Advanced Manufacturing Technology*, 38(3-4):367–376, 2008.

[109] C. Savage. *Fifth Generation Management: Co-creating Through Virtual Enterprising, Dynamic Teaming, and Knowledge Networking*. Business/Management / Butterworth-Heinemann. Butterworth-Heinemann, 1996.

[110] T. Schoenherr and V. A. Mabert. Online reverse auctions: Common myths versus evolving reality. *Business Horizons*, 50(5):373 – 384, 2007.

[111] M. E. Shalev and S. Asbjornsen. Electronic reverse auctions and the public sector – factors of success. *Journal of Public Procurement*, 10(3):428–452, March 2010.

[112] W. Shen, Q. Hao, S. Wang, Y. Li, and H. Ghenniwa. An agent-based service-oriented integration architecture for collaborative intelligent manufacturing. *Robotics and Computer-Integrated Manufacturing*, 23(3):315 – 325, 2007. International Manufacturing Leaders Forum 2005: Global Competitive Manufacturing.

[113] J. Siqueira and C. Bremer. Action research: The formation of a manufacturing virtual industry cluster. In L. Camarinha-Matos, H. Afsarmanesh, and R. Rabelo, editors, *E-Business and Virtual Enterprises*, volume 56 of *IFIP — The International Federation for Information Processing*, pages 261–268. Springer US, 2001.

[114] A. L. Soares, A. L. Azevedo, and J. P. de Sousa. Distributed planning and control systems for the virtual enterprise: organizational requirements and development life-cycle. *Journal of Intelligent Manufacturing*, 11(3):253–270, 2000.

[115] P. Spyns, R. Meersman, and M. Jarrar. Data modelling versus ontology engineering. *SIGMOD Rec.*, 31(4):12–17, Dec. 2002.

[116] D. Systems. Enovia, collaborative innovation. http://www.3ds.com/products-services/enovia/.

[117] TSE. Small and medium size enterprises statistics, 2013. Technical Report 15881, Turkish Statistical Institute, Turkey, November 2013.

[118] G. Tzeng and J. Huang. *Multiple Attribute Decision Making: Methods and Applications*. A Chapman & Hall book. Taylor & Francis, 2011.

[119] O. F. Valilai and M. Houshmand. A collaborative and integrated platform to support distributed manufacturing system using a service-oriented approach based on cloud computing paradigm. *Robotics and Computer-Integrated Manufacturing*, 29(1):110 – 127, 2013.

[120] A. Vargas, A. Boza, and L. Cuenca. Towards interoperability through inter-enterprise collaboration architectures. In *Proceedings of the 2011th Confederated International Conference on On the Move to Meaningful Internet Systems*, OTM'11, pages 102–111, Berlin, Heidelberg, 2011. Springer-Verlag.

[121] I. Villar-Medina, O. López-Ortega, and R. Hernández-Gómez. Implementation of a supervised learning technique in a multi-agent system for building production orders. *The International Journal of Advanced Manufacturing Technology*, 40(7-8):808–818, 2009.

[122] P. Vrba, M. Radakovič, M. Obitko, and V. Mařík. Semantic technologies: latest advances in agent-based manufacturing control systems. *International Journal of Production Research*, 49(5):1483–1496, 2011.

[123] W3C. The organization ontology, 2011. Available from: http://www.w3.org/TR/vocab-org//.

[124] D. Wang, S. V. Nagalingam, and G. C. Lin. Development of an agent-based virtual {CIM} architecture for small to medium manufacturers. *Robotics and Computer-Integrated Manufacturing*, 23(1):1 – 16, 2007.

[125] G. Wang, T. Wong, and X. Wang. A hybrid multi-agent negotiation protocol supporting agent mobility in virtual enterprises. *Information Sciences*, 282(0):1 – 14, 2014.

[126] S. Wang, W. Shen, and Q. Hao. An agent-based web service workflow model for inter-enterprise collaboration. *Expert Systems with Applications*, 31(4):787 – 799, 2006. Computer Supported Cooperative Work in Design and Manufacturing.

[127] X. Wang, T. Wong, and G. Wang. An ontological intelligent agent platform to establish an ecological virtual enterprise. *Expert Systems with Applications*, 39(8):7050 – 7061, 2012.

[128] X. Wang, T. Wong, and G. Wang. Service-oriented architecture for ontologies supporting multi-agent system negotiations in virtual enterprise. *Journal of Intelligent Manufacturing*, 23(4):1331–1349, 2012.

[129] A. Xiang, L. Liu, and Q. Luo. Vpeers: A peer-to-peer service discovery framework for virtual manufacturing organizations. *Computers in Industry*, 59(5):411 – 419, 2008.

[130] Z. Yang and H. Lin. Analysis of the virtual enterprise partner selection based on multi-agent system. In *Computer Science and Software Engineering, 2008 International Conference on*, volume 2, pages 516–519, Dec 2008.

[131] F. Ye and Y. Li. Multi-attribute decision-making model for virtual enterprise partners selection under uncertain information condition. In *Services Systems and Services Management, 2005. Proceedings of ICSSSM '05. 2005 International Conference on*, volume 2, pages 1162–1165 Vol. 2, June 2005.

[132] W. Zhang and J. Yin. Exploring semantic web technologies for ontology-based modeling in collaborative engineering design. *The International Journal of Advanced Manufacturing Technology*, 36(9-10):833–843, 2008.

[133] L. Zhou and R. Nagi. Design of distributed information systems for agile manufacturing virtual enterprises using {CORBA} and {STEP} standards. *Journal of Manufacturing Systems*, 21(1):14 – 31, 2002.

[134] A. Zwegers, M. Tolle, and J. Vesterager. Veram: virtual enterprise reference architecture and methodology. In *VTT SYMPOSIUM*, volume 224, pages 17–38. Citeseer, 2003.

APPENDIX A

LOADING OWL MODEL TO TDB DATA STORE

```
//Loading OWL Model to TDB-Jena
          Model model = dataset.getNamedModel(directory);
          OntModel ontmodel = ModelFactory.createOntologyModel(
              OntModelSpec.OWL_MEM, model);
          String source = "file:///users/bahram/tdb/myontology.owl";
          System.out.println("Load␣Model...");
          FileManager.get().loadModel(source);
          System.out.println("Loading␣completed.");
          System.out.println("Read␣Model...");
          FileManager.get().readModel(model, source);
          System.out.println("Reading␣Process␣Completed.");
```

APPENDIX B

SPARQL QUERY FOR CREATING CLASS HIERARCHY

Listing B.1: Query for sub classes

```
String queryString = "PREFIX␣rdf:␣<http://www.w3.org/1999/02/22-rdf-
    syntax-ns#>"
                   + "PREFIX␣owl:␣<http://www.w3.org/2002/07/owl#>"
                   + "PREFIX␣xsd:␣<http://www.w3.org/2001/XMLSchema#>"
                   + "PREFIX␣rdfs:␣<http://www.w3.org/2000/01/rdf-schema#>
                       "
                   + "PREFIX␣VEOnto:␣<http://www.semanticweb.org/
                       ontologies/2012/10/untitled-ontology-34#>"
                   + "SELECT␣?label␣?class␣"
                   + "␣WHERE␣{␣"
                   + "␣?class␣rdfs:subClassOf␣?<" + classURI + ">.␣"
                   + "␣FILTER␣(?class␣!=␣<" + classURI + ">)␣"
                   + "␣?class␣rdfs:label␣?label."
                   + "␣FILTER(␣langMatches(lang(?label),'en'))"
                   + "␣"
                   + "}"
                   + "␣ORDER␣BY␣?label";

          Query query = QueryFactory.create(queryString);
          QueryExecution qExec = QueryExecutionFactory.create(query,
              ontmodel);
          ResultSet results = qExec.execSelect();
          for (; results2.hasNext();) {
              QuerySolution soln = results.nextSolution();
              if (soln2.get("class") != null) {
                  String myLabel = soln.get("label").toString();
                  String classURI = soln.get("class").toString();
                  String labelName = myLabel.substring(0, myLabel.length
                      () - 3);
                  TreeNode secondLevelNode = new DefaultTreeNode(
                      labelName, firstLevelNode);
                  String queryString2 = "PREFIX␣rdf:␣<http://www.w3.org
                      /1999/02/22-rdf-syntax-ns#>"
                          + "PREFIX␣owl:␣<http://www.w3.org/2002/07/owl#>
                              "
                          + "PREFIX␣xsd:␣<http://www.w3.org/2001/
                              XMLSchema#>"
```

```
        + "PREFIX␣rdfs:␣<http://www.w3.org/2000/01/rdf-
            schema#>"
        + "PREFIX␣VEOnto:␣<http://www.semanticweb.org/
            ontologies/2012/10/untitled-ontology-34#>"
        + "SELECT␣?label␣?class2␣"
        + "␣WHERE␣{␣"
        + "␣?class2␣rdfs:subClassOf␣?<" + classURI + "
            >.␣"
        + "␣FILTER␣(?class2␣!=␣<" + classURI + ">)␣"
        + "␣?class2␣rdfs:label␣?label."
        + "␣FILTER(␣langMatches(lang(?label),'en'))"
        + "␣"
        + "}"
        + "␣ORDER␣BY␣?label";
Query query2 = QueryFactory.create(queryString2);
QueryExecution qExec2 = QueryExecutionFactory.create(
    query2, ontmodel);
ResultSet results2 = qExec2.execSelect();
....
```

APPENDIX C

SPARQL QUERY SAMPLE FOR CLASS PROPERTIES

Listing C.1: Query for the selected class properties

```
String queryString = "PREFIX␣rdf:␣<http://www.w3.org/1999/02/22-rdf-
    syntax-ns#>"
              + "PREFIX␣owl:␣<http://www.w3.org/2002/07/owl#>"
              + "PREFIX␣xsd:␣<http://www.w3.org/2001/XMLSchema#>"
              + "PREFIX␣rdfs:␣<http://www.w3.org/2000/01/rdf-schema#>"
              + "PREFIX␣VEOnto:␣<http://www.semanticweb.org/ontologies
                  /2012/10/untitled-ontology-34#>"
              + "SELECT␣?classURI␣?datatypeproperty␣?labelEN␣?labelTR␣"
              + "␣WHERE␣{␣"
              + "␣?classURI␣rdfs:label␣\"" + nodeString + "\"@en␣.␣"
              + "␣?datatypeproperty␣rdfs:domain␣?classURI␣.␣"
              + "␣?datatypeproperty␣rdf:type␣owl:DatatypeProperty␣.␣"
              + "␣?datatypeproperty␣rdfs:label␣?labelEN␣."
              + "␣FILTER(␣langMatches(lang(?labelEN),’en’))"
              + "␣?datatypeproperty␣rdfs:label␣?labelTR␣."
              + "␣FILTER(␣langMatches(lang(?labelTR),’tr’))"
              + "␣"
              + "}"
              + "␣ORDER␣BY␣?labelEN";

      Query query = QueryFactory.create(queryString);
      QueryExecution qExec = QueryExecutionFactory.create(query, ontmodel
          );
      ResultSet results = qExec.execSelect();
      for (; results.hasNext();) {
          QuerySolution soln = results.nextSolution();
          if (soln.get("datatypeproperty") != null) {
              String myLabelEN = soln.get("labelEN").toString();
              String myLabelTR = soln.get("labelTR").toString();
              String datatypepropertyURI = soln.get("datatypeproperty").
                  toString();
              selectedNodeDatatypePropertyENLabel = myLabelEN.substring
                  (0, myLabelEN.length() - 3);
              selectedNodeDatatypePropertyENLabels.add(
                  selectedNodeDatatypePropertyENLabel);
              selectedNodeDatatypePropertyTRLabel = myLabelTR.substring
                  (0, myLabelTR.length() - 3);
```

```
    selectedNodeDatatypePropertyTRLabels.add(
        selectedNodeDatatypePropertyTRLabel);
    selectedNodeDatatypePropertiesURI.add(datatypepropertyURI);
  }
}
```

APPENDIX D

QUERY FOR FINDING INDIVIDUALS IN A SELECTED CLASS

Listing D.1: Query for the selected class individuals

```
FileManager.get().loadModel(source);
    FileManager.get().readModel(model, source);

    String nodeString = TreeBean.nodeString;

    String queryString = "PREFIX␣rdf:␣<http://www.w3.org/1999/02/22-rdf
        -syntax-ns#>"
            + "PREFIX␣owl:␣<http://www.w3.org/2002/07/owl#>"
            + "PREFIX␣xsd:␣<http://www.w3.org/2001/XMLSchema#>"
            + "PREFIX␣rdfs:␣<http://www.w3.org/2000/01/rdf-schema#>"
            + "PREFIX␣VEOnto:␣<http://www.semanticweb.org/ontologies
                /2012/10/untitled-ontology-34#>"
            + "SELECT␣?instances␣?labelTR␣?labelEN␣?instanceComment"
            + "␣WHERE␣{␣"
            + "␣?classURI␣rdfs:label␣\"" + nodeString + "\"@en␣.␣"
            + "␣?instances␣rdf:type␣?classURI␣.␣"
            + "␣?instances␣rdfs:label␣?labelEN."
            + "␣?instances␣rdfs:label␣?labelTR."
            + "␣?instances␣rdfs:comment␣?instanceComment␣.␣"
            + "␣FILTER(␣langMatches(lang(?labelEN),’en’))"
            + "␣FILTER(␣langMatches(lang(?labelTR),’tr’))"
            + "␣"
            + "}";

    Query query = QueryFactory.create(queryString);
    QueryExecution qExec = QueryExecutionFactory.create(query, ontmodel
        );
    ResultSet results = qExec.execSelect();
    for (; results.hasNext();) {

        QuerySolution soln = results.nextSolution();

        String myLabelEN = soln.get("labelEN").toString();
        String myLabelTR = soln.get("labelTR").toString();
        String instanceURI = soln.get("instances").toString();
```

```
    String labelNameEN = myLabelEN.substring(0, myLabelEN.length()
        - 3);
    String labelNameTR = myLabelTR.substring(0, myLabelTR.length()
        - 3);
    String instanceComment = soln.get("instanceComment").toString()
        ;

    selectedNodeInstances.add(instanceURI);
    selectedNodeInstancesComments.add(instanceComment);
    selectedNodeIndividualENLabels.add(labelNameEN);
    selectedNodeIndividualTRLabels.add(labelNameTR);
}
```

APPENDIX E

TASK-MACHINE TOOL ASSIGNMENT RULES

Listing E.1: Task-MachineTool Assignment Rule Code Sample

```
processToMachineRules.append("␣"
                + "[multiTaskingMachineRules:␣(?mtm␣<http://www.w3.org
                  /1999/02/22-rdf-syntax-ns#type>␣<http://www.
                  semanticweb.org/ontologies/2012/10/untitled-
                  ontology-34#Multi_Tasking_Machines>),␣"
                + "(?y␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#type
                  >␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#Turning_Processes>),␣"
                + "(?z␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#type
                  >␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#Milling_Processes>),␣"
                + "(?w␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#type
                  >␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#Drilling_Processes>)␣"
                + "->␣"
                + "(?mtm␣<http://www.semanticweb.org/ontologies
                  /2012/10/untitled-ontology-34#enables>␣?y),␣"
                + "(?mtm␣<http://www.semanticweb.org/ontologies
                  /2012/10/untitled-ontology-34#enables>␣?z),␣"
                + "(?mtm␣<http://www.semanticweb.org/ontologies
                  /2012/10/untitled-ontology-34#enables>␣?w)␣]"
                + "␣[millingCentersRules:␣(?mc␣<http://www.w3.org
                  /1999/02/22-rdf-syntax-ns#type>␣<http://www.
                  semanticweb.org/ontologies/2012/10/untitled-
                  ontology-34#Milling_Center>),␣"
                + "(?a␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#type
                  >␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#Milling_Processes>),␣"
                + "(?b␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#type
                  >␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#Drilling_Processes>),␣"
                + "->␣"
                + "(?mc␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#enables>␣?a),␣"
                + "(?mc␣<http://www.semanticweb.org/ontologies/2012/10/
                  untitled-ontology-34#enables>␣?b)␣]"
```

```
            + "␣[turningCentersRules:␣(?tc␣<http://www.w3.org
                /1999/02/22-rdf-syntax-ns#type>␣<http://www.
                semanticweb.org/ontologies/2012/10/untitled-
                ontology-34#Turning_Center>),␣"
            + "(?a␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#type
                >␣<http://www.semanticweb.org/ontologies/2012/10/
                untitled-ontology-34#Turning_Processes>),␣"
            + "->␣"
            + "(?tc␣<http://www.semanticweb.org/ontologies/2012/10/
                untitled-ontology-34#enables>␣?a)␣]"
            + "␣[latheRules:␣(?l␣<http://www.w3.org/1999/02/22-rdf-
                syntax-ns#type>␣<http://www.semanticweb.org/
                ontologies/2012/10/untitled-ontology-34#Lathe>),␣"
            + "(?tp␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#
                type>␣<http://www.semanticweb.org/ontologies
                /2012/10/untitled-ontology-34#Turning_Processes>),␣
                "
            + "->"
            + "(?l␣<http://www.semanticweb.org/ontologies/2012/10/
                untitled-ontology-34#enables>␣?tp)␣]"
            + "␣[millingRules:␣(?m␣<http://www.w3.org/1999/02/22-
                rdf-syntax-ns#type>␣<http://www.semanticweb.org/
                ontologies/2012/10/untitled-ontology-34#Milling>),␣
                "
            + "(?mp␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#
                type>␣<http://www.semanticweb.org/ontologies
                /2012/10/untitled-ontology-34#Milling_Processes>),␣
                "
            + "(?dp␣<http://www.w3.org/1999/02/22-rdf-syntax-ns#
                type>␣<http://www.semanticweb.org/ontologies
                /2012/10/untitled-ontology-34#Drilling_Processes>),
                ␣"
            + "->"
            + "(?m␣<http://www.semanticweb.org/ontologies/2012/10/
                untitled-ontology-34#enables>␣?mp),␣"
            + "(?m␣<http://www.semanticweb.org/ontologies/2012/10/
                untitled-ontology-34#enables>␣?dp)␣]"
);
```

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Lotfi Sadigh, Bahram
Nationality: Iranian (IR)
Date and Place of Birth: 06.03.1981, Tabriz
Marital Status: Single
Phone: 0 553 012 0134
Fax:

EDUCATION

Degree	Institution	Year of Graduation
M.S.	University of Tabriz, Mechanical Engineering Department	2006
B.S.	University of Tabriz, Mechanical Engineering Department	2004
High School	Taleghani High School	2000

PROFESSIONAL EXPERIENCE

Year	Place	Enrollment
2007.03-2007.09	Iran Rail Industry Company	Production Line Supervisor

PUBLICATIONS

International Conference Publications

S. Nikghadam, B. L. Sadigh, A. M. Ozbayoglu, H. O. Unver, and S. E. Kilic. Partner selection in formation of virtual enterprises using fuzzy logic. In *In Proceedings of the International Conference on Operations Research and Enterprise Systems*, pages 82-88, Lisbon, 2015.

B. L. Sadigh, F. Arikan, A. Ozbayoglu, H. Unver, and S. Kilic. A multi-agent system model for partner selection process in virtual enterprise. *Procedia Computer Science*, 36(0):367-372, 2014. Complex Adaptive Systems Philadelphia, fPAg November 3-5, 2014.

Bahram Lotfi Sadigh, H. Ozgur Unver, Erdogan Dogdu and S. Engin Kilic, "Ontology based Virtual Enterprise System Domain Modeling", Proc. of the 24th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM), May 20 23, 2014, San Antonio, Texas.

H. O. Unver and B. L. Sadigh. *Small and Medium Enterprises: Concepts, Methodologies, Tools, and Applications*, chapter An agent-based operational virtual enterprise framework enabled by RFID, pages 198-215. IGI Global, Web., 2013.

B. L. Sadigh, H. Unver, and S. Kilic. Design of a multi agent based virtual enterprise framework for sustainable production. In G. Putnik and M. Cruz-Cunha, editors, *Virtual and Networked Organizations, Emergent Technologies and Tools*, volume 248 of Communications in Computer and Information Science, pages 186-195. Springer Berlin Heidelberg, 2012.

B. Lotfi Sadigh, H.Ö. Ünver, E. Doğdu, S.E. Kılıç (2012), An Ontology Based Model for Virtual Enterprise, *Proceedings of the 15th International Machine Design and Production Conference*, Pamukkale, Denizli, Turkey, pp. 885-891