MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

COOPERATION & COORDINATION OF DISTRIBUTED INTELLIGENT AGENTS FOR MANUFACTURING SYSTEMS

BANU ÇALIŞ

Ph.D. THESIS

Department of Industrial Engineering

Thesis Supervisor

Prof. Dr. Ercan ÖZTEMEL

ISTANBUL, 2015

MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

COOPERATION & COORDINATION OF DISTRIBUTED INTELLIGENT AGENTS FOR MANUFACTURING SYSTEMS

BANU ÇALIŞ

(724408001)

Ph.D. THESIS

Department of Industrial Engineering

Thesis Supervisor

Prof. Dr. Ercan ÖZTEMEL

ISTANBUL, 2015

MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES IN PURE AND APPLIED SCIENCES

Banu Çalış, a Doctor of Philosophy student of Marmara University Institute for Graduate Studies in Pure and Applied Sciences, defended her thesis entitled "**COOPERATION & COORDINATION OF DISTRIBUTED INTELLIGENT AGENTS FOR MANUFACTURING SYSTEMS**", on **12.10.2015** and has been found to be satisfactory by the jury members.

JuryMembers

Prof. Dr. Ercan ÖZTEMEL (Advisor)
Marmara University ....

Prof. Dr. Cemalettin KUBAT (Jury Member)
Sakarya University ....

Asst. Prof. Dr. S.Serdar YÖRÜK (Jury Member)
Marmara University..

Prof. Dr. M. Akif EYLER (Jury Member)
Marmara University..

Prof. Dr. Selim ZAİM (Jury Member)
İstanbul Technical University..

APPROVAL

Marmara University Institute for Graduate Studies in Pure and Applied Sciences Executive Committee approves that Banu Çalış be granted the degree of Doctor of Philosophy in Department of Industrial Enginnering, Industrial Engineering Program on 19.10.2015..... (Resolution no: 2015/24-02...).

Director of the Institute
Prof. Dr. Uğur YAHŞİ

TABLE OF CONTENT

ÖZET

Zeki etmenlerin otomasyona bağlı karar verme sistemlerinin tasarımında ve operasyonel uygulama alanlarındaki kullanımı, günden güne daha popülerlik kazanmaktadır. Tanım olarak etmenler, bağımsız olarak çalışabilen, ilgili olayları yönetebilen ve bu olaylara bağlı sorunlara karşılık gelen problemlere mümkün olan en iyi çözümü üretme yeteneğine sahip bilgisayar tabanlı sistemlerdir.

Günümüz gelişmeleri, imalat sektöründe tamamiyle otomasyona dayalı makine tasarımlarına da ihtiyaç olduğunu göstermektedir ancak sadece otomasyon sistemleri etken bir imalat sistemi için yetersizdir. Zeki yeteneklere sahip üretim sistemlerinin artması, özellikle işçi verimliliğinin arttırılmasında, verimliliğin güvence altına alınmasında ve ürün/süreç kalitesinde ciddi avantajlar sağlamaktadır.

Üretimin tasarım, planlama, izleme, test etme, depolama gibi çeşitli fonksiyonların bir araya gelmesi ile oluşması ve bu fonksiyonların otonom sistemlere dönüştürülmesi, insansız fabrikalar için önemli bir katkı oluşturmaktadır. Bu ise ancak sağlıklı ve bütünleşik bir yapı ile olabilir.

Fonksiyonlar arası entegrasyonun sağlanabildiği bir üretim sistemi elde edebilmek için, bütünleşik zeki etmen tabanlı sistemlere ihtiyaç vardır. Etmenler için bazı bilgi paylaşım çevreleri yaratma çalışmaları olmasına rağmen, iş birlikteliği ve koordinasyon stratejilerine bağlı olarak kolaylıkla uygulanabilecek entegre sistemlere olan ihtiyaç hala mevcuttur. Burada temel odak noktası, amaç veya sonuç odaklı iş birlikteliğinin sağlanmasıdır, başka bir deyişle etmenlerden artık sadece bilgi sağlama va paylaşımında bulunma değil, ayrıca sonuçları takip ederek kendi bilgi seviyelerini güncellemeleri beklenir.

Çalışmanın amacı, özellikle üretimde, imalat sistemlerinde zeki etmenlerin işbirliği ve koordinasyonu üzerine bir yöntem geliştirilmesidir.

Çalışma kapsamında, zeki imalat sistemlerinde planlama, üretim, stok yönetimi ve satın alma fonksiyonlarını entegre eden, amaç odaklı ve hiyerarşik bir model önerilmiştir. Çalışmanın literatüre sağladığı en belirgin katkı, amaçları süreçlerdeki operasyonlara, operasyonları makinalara, makinaları üretilen ürün tiplerine, ürünleri ham maddelerine ve hammaddeleri tedarikçilere ilişkilendirerek ölçen entegre bir yapı oluşturmasıdır.

Önerilen model, kabin üretimi yapan gerçek bir üretim sisteminde uygulanmış ve uygulama sonuçları modelin, sistemin etkinliğini ve verimliliğini açık bir şekilde iyileştirdiğini göstermiştir.

Anahtar Kelimeler: Etmen, Etmen Tabanlı Sistemler, Üretim

Eylül, 2015 **Banu ÇALIŞ**

ABSTRACT

Utilization of intelligent agents in the design of automation-based decision making systems and operational application field has been gaining popularity day by day. The agents, by definition, are computer-based systems, which could operate independently, manage the relevant events and have the ability to generate the best possible solution to troubles that correspond to the problems depending on these events.

Today's developments illustrate that machinery designs that depend on automation completely in the manufacturing sector are needed however automation systems only is not sufficient for an effective manufacturing system. Increase in manufacturing systems that have intelligent abilities enables especially improvement of worker efficiency, assuring efficiency and providing critical advantages in product/process quality.

Generation of production with the gathering of various functions such as design, planning, monitoring, testing and storage and turning these functions to autonomous systems makes crucial contribution to unmanned factories. This in turn can happen only by a healthy and integrated structure.

In order to obtain a manufacturing system where inter-functional integration is managed, integrated intelligent agent-based systems are needed. Although there are studies to create some information sharing settings for the agents, there is still need for integrated systems which could be applied conveniently depending on work synergy and coordination strategies. The fundamental focus point here is ensuring work synergy focusing on purpose or corollary, in other words, it is expected from the agents not only supplying information or sharing but also monitoring the results and updating their own knowledge levels.

The purpose of the study is to develop a method on cooperation and coordination of intelligent agents in manufacturing systems, specifically in manufacturing.

In the scope of this thesis, a hierarchical intelligent based model that entegrates planning, manufacturing, stock management and procurement functions of manufacturing systems.

The proposed model is different from other studies in the literature in terms of applying an integrated and interoperable structure including an opportunity for measurability of suppliers; operations included in the processes, machinery for operation execution, products manufactured using available machinery and supplies and raw materials used for the manufacturing of the products.

A real case study is implemented in a cabinet manufacturing company. The results showed that proposed model prominently increased the efficiency and effectiveness of the system.

Key Words: Agent, Agent Based Systems, Manufacturing

September, 2015 **Banu ÇALIŞ**

CLAIM OF ORIGINALITY

New improvement directions in technology increased the competition between firms. To respond the diversified needs of customers the different manufacturing functions such as planning, manufacturing and quality have to be integrated in an efficient manner. The underlying idea of this integration is possible with the introduction of intelligent manufacturing systerms. As a part of intelligent manufacturing system, Agent technology is well equipped to perceive environment, reason about it and act accordingly. Agents are now proven to be capable of having been able to make better decisions under very complex situations, behave as much flexible as possible, and prevent misuse of knowledge with excellent communication capabilities.

As well known, the multi agent systems in manufacturing process provide well-defined integration and automation and can ensure several advantages of distributed manufacturing.

In this study, it is shown that a well-defined set-up of integrated manufacturing environment with self-decision making capability and modelling can be established. This certainly increase the efficiency of manufacturing systems through employing self-organization, pro-activeness and reactiveness properties of agents with related experience and knowledge as well as inter-agent interactions.

The prototype multi agent based manufacturing systems with real time information flow is introduced in this thesis. The proposed model is called as IMPACT (Intelligent Manufacturing Planning with Agent-based Control Technology).

The utilization, coordination and communication among the manufacturing agents are demonstrated in such a way that; manufacturing span is decreased, manufacturing cost caused by purchasing of raw materials, operation scheduling, WIP (Work-in-Process), overstock etc. is reduced. The quality level of both purchased and manufactured products is increased. The control on manufacturing, scheduling, supplier selection, machine usage, operations and process management etc. is assured.

A sustainable ability of assessment of performance of manufacturing goals, processes, operations, machines, products and suppliers is ensured. The study also highlighted a clear road map for intelligent agent-based manufacturing planning.

September, 2015 **Prof. Dr. Ercan ÖZTEMEL** **Banu ÇALIŞ**

ABBREVIATIONS

ABM : Agent-Based Modeling

ABMS : Agent Based Manufacturing System

ACL : Agent Communication Language

AGV : Automated Guided Vehicle

AIP : Agent Interaction Protocols

AMIS : Agile Manufacturing Information System

AMS : Agent Management System

AP : Agent Protocols

BDI : Belief-Desire-Intention

BOID : Belief-Obligation-Intention-Desire

CORBA : Common Object Request Broker Architecture

CSAL : Cloud Storage Abstraction Layer

DAI : Distributed Artificial Intelligence

FIPA : the Foundation for Physical Agents

FTTH : Fiber To The Home

ECNpro : the Extended Contract-Net- like multilateral Protocol

GA : Genetic Algorithm

IMPACT : Intelligent Manufacturing Planning using Agent-based Control Technology

IMAV : Intelligent multi-agent model based on virtualization rules for resource virtualization

ITS : IMPACT Task Set

KAoS : the Knowledge-able Agent-oriented System

KPI : Key Performance Indicator

KQML : Knowledge Query and Manipulation Language

MAS : Multi Agent Systems

MS : Manufacturing Systems

MTS : Message Transport Service

NA : Node Agents

NP	**:** Non-Deterministic Polynomial
N. R. M.T	**:** Needed Raw Material Type
OMC	**:** Online Manufacturing Control
OMG	**:** Object Manager Group
ORB	**:** Object Request Brokers
PM	**:** Physical Machine
REMIMS	**:** Reference Model for Intelligent Integrated Manufacturing Systems
RFQ	**:** Requests for Quote
RPC	**:** Remote Procedure Call
TAC	**:** Trading Agent Competition
TCP	**:** Transport Control Protocol
VM	**:** Virtual Machine

TABLE OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Today; business is characterized by an ever increasing pace in production, decreasing product cycle times, and an increasing shift from mass production to mass customization. Companies must be more responsive to market changes and more sensitive to customer needs in order to survive and continue to operate and sustain their business goals. This obviously necessitates systematic monitoring and control over manufacturing systems.

Traditional monitoring and control of manufacturing systems are usually based on deterministic algorithms. However, since customers' expectations are dynamic in their nature and, manufacturing systems also need to be more responsive to the changing conditions through concurrently balancing and optimizing multiple manufacturing constraints. In order to achieve this, companies must have flexible manufacturing system to be able to respond technological and service expectations.

Having being able to control manufacturing environment strictly requires the related departments operate in optimal condition as a whole and cannot work independently. This makes it difficult to control all stages of flexible manufacturing systems. Manufacturers therefore need to provide coordination and cooperation between all units after all stages of manufacturing. At this point, integration between information and technology becomes the main concern. There is a strong need for software programs to collect data and perform data analysis in order to get the best information flow with balanced overall performance and to optimize respective manufacturing processes.

There are three main paradigms of manufacturing. First one is about the flexibility of the system. For example, a mobile phone is basically mass-produced (masscustomization). In a regular shop, a particular type of phone is ordered from the catalog. When cell phone technology is considered in order to show the role of the flexible manufacturing for answering rapidly to customer requests, it is observed that a flexible manufacturing is not possible yet at the level of personalization of the products. For example, a sound camera is to be replaced with memory card on a standard phone, it is impossible by design. If a phone manufacturer that has a large list of side-items ("modules") that can be added in any combination to the main entrée and the combination of “modules” can be served personally, this is the core content of personalization. Personalization is only possible the flexibility of the system and the ability to response to customer demands in a flexible manner.

The second is about increasing the frequency of new product introductions. Incereasing this frequency is only possible with rapid reconfiguration that shortens the manufacturing system lead-time.

And the last one is about changes in manufacturing technologies, methods and methodologies to increase quality level of the products. There is a need for how they can achieve manufacturing goals by taking the initiative and making use of pre-compiled plans. Also they need a capability of perceiving their environment in order to be able to respond changes. These are necessary in order to keep the required quality level and maintain or even reduce the costs as well as keeping the market share if not possible to increase autonomous systems such as intelligent systems the problems of control and management requirements is not also possible.

Also as a new topic unmanned manufacturing environment is becoming more and more popular. There are several approaches in the literature in order to realize. First approach is a real-time process control approach which means that the modifications to the process can occur only after the processing the previous part. The second one is a real-time process control approach means that adjust or modify the process to manage unmanned manufacturing while the process is taking place.

Considering the above descriptions, clearly seen that there is an overall tendency of technology in the development of industrial robots by a reason of new production concepts brought forward in manufacturing industry. The main motivation for using robot control in industry is to reduce the production cost and satisfy easy controlled manufacturing operations.

Similarly, there is a need for machine-to-machine communication systems such as intelligent and robotic environment to satisfy overall control of the system for distributed manufacturing systems. Automated decision making systems are unavoidable by a reason of technological tendency of the manufacturing environments.

As a result of the given explanation above, Agent technology is well equipped to perceive environment, reason about it and act accordingly. Agents are now proven to be capable of having been able to make better decisions under very complex situations, behave as much flexible as possible, and prevent misuse of knowledge with excellent communication capabilities.

Based on the explanations above, the main focus of this thesis is to provide information and a prototype of dynamic, flexible and intelligent model.

The prototype model **proposed is so called as IMPACT (I**ntelligent **M**anufacturing **P**lanning using **A**gent-based **C**ontrol **T**echnology**) and** has a real-time information flow (available at any time to many people). In generating the model, the intelligent agent technology is employed. As well known, the agents may work together and share their knowledge related to manufacturing activities. Multi agent systems in manufacturing process provide well defined integration and automation and can ensure several advantages of distributed manufacturing.

1.1. Main Motivation of Thesis

Planning and controlling of manufacturing systems needs to numerous specialized experience such as manufacturing planning, procurement, profit-analysis, scheduling, material management, appropriate supplier selection etc. Each area of expertise necessitates the following capabilities.

- determining related data,
- data collection, and
- tactful handling and interpretation of respective data
- an ability to take an "integrated view"
- places tasks in a logical sequence
- having a pro-active nature
- preparation of technical reports
- quantifying and tracking all scope of work

There is a need for a cooperated and coordinated system to handle and process of the tactful handling and interpretation of knowledge of the domain expertise.

In distributed manufacturing systems, decision maker of each department uses his/her own experience for the attainment of company objectives keeping cooperation and communication channels between different departments. This requires well-established co-operation procedures especially when the products and processes are very complex and with countable varieties.

Agent-based technology on the other hand, enables or supports the construction of this type of co-operation channels by their self-organized properties and flexible attributes as well as knowledge processing capabilities.

Some researcher investigated the advantages of agent based technology in manufacturing systems. A few of them focused their attention to integrated structures taking supplier's model, operations model, machine model, product model, process model and goal model as outlined in this thesis into account. Related literature review is discussed in detail in Chapter Two. However, a brief summary is provided in Table 1.1 for clarification purpose.

Table 1.1. Recent studies of Multi Agent Based Manufacturing solutions (Part 1)

Area	**Explanation of Study**	**Algorithm / Industry**	**Article**
Scheduling	Focused on developing an abstract optimized scheduling model, seeking to minimize total production cost where plans and	Algorithm	Burke & Prosser (1991)
Scheduling	Appropriate coordination technique in responding in real time to problems in job shop scheduling is emphasized as a need for distributed systems.	Algorithm	Liu & Sycara (1997)
Scheduling	The higher frequency of learning may help an agent to quickly adapt to variations on the shop floor.	Algorithm	Pendharkar. (1999).
Production Control	Issues identified in production control including; overcoming constraint in the architecture of non-adoptive control systems through modifying hierarchical and non-hierarchical relationships between agents	Algorithm	Ottoway & Burns (2000)
Enterprise integration	This research constructed an agent-based control framework, which can be used in the collaborative manufacturing environment that has already been recognized and is practiced in many sectors of manufacturing industry. This control framework is able to satisfy multiple goals, which are to deliver lower penalties for the whole system and the two different order types simultaneously.	Industry	Lu & Yih (2001)
Scheduling	The system is composed of the agent and the simulated environment (SE). The agent is able to perform dynamic scheduling based on the available information provided by the SE. It makes decision for selection of the most appropriate dispatching rule in real time. At the end of training, the agent gives better results than the traditional alternatives (SPT, COVERT, and CR rules).	Algorithm	Aydin & Oztemel (2000)
Scheduling	An experimental approach for performance analysis of a multi-agent system for job routing in job-shop settings. Some simple but practical mechanisms are proposed and implemented	Algorithm	Usher (2003)
CRM and SCM	The study is considering intra- and cross enterprise integrating, examined the possible correlation of companies' production and outsourcing plans. The computerization of the customer–supplier management process helps the enterprise to implement a CRM strategy fully in each department, to achieve a close relationship with suppliers/partners through integration with the SRM strategy, and consequently increase the manufacturers' own competitiveness, reputation and revenue in the market.	Industry	Choy et al. (2004)
Scheduling	A parallel implementation of modular simulated annealing algorithm (MSA) is presented. In order to run the parallel MSA , multi agent systems is used. The empirical results show that the method is quite successful comparing to the sequential version of MSA.	Algorithm	Aydın & Fogarty (2004)
Supply Chain	A multi agent based distributed coordination of a supply chain model is presented based on 11 agent per node of the supply chain that could act as a company representative.	Algorithm	Chan & Chan (2004)
Process Planning and Scheduling	Specified a number of multi objective decision making problems considering optimal tasks assignment and system architecture	Algorithm	Deshpande & Cagan (2004)

Table 1.1. Recent studies of Multi Agent Based Manufacturing solutions (Part 2)

Area	**Explanation of Study**	**Algorithm / Industry**	**Article**
Supply Chain	A multi agent system approach is proposed for solving problems in a supply chain network. The major benefits of this modelling technique can be easily formulated. First of all, computation effort can be reduced dramatically. Authors emphasized that if the multi-agent paradigm is adopted, the decision-making is only responsible to evaluate certain alternatives (bids) against its own constraint	Algorithm	Chan & Chan (2005)
Supply Chain	A Holonic supply chain management system for Telephone Manufacturing Industry is proposed, the multi agent system is generated based on 7 types different agent.	Algorithm	Ulieru and Cobzaru (2005)
Production Control	This paper reports on the design and realization of an agent-based intelligent control system for industrial manufacturing systems, which covers both conventional mechatronics equipment and embedded control agents. The agent-based control software components are implemented as Windows NT services and logic control programs and interfaced via distributed component object model (DCOM) and Ethernet. Results of the application in an industrial manufacturing environment are used to show the effectiveness of the proposed "Holonic" approach.	Industry	Colombo et al. (2006)
Process Planning and Scheduling	A hybrid -based multi agent system for integrated process planning with scheduling and rescheduling is presented in job shops or similar kinds of flexible manufacturing environments. A negotiation protocol is generated to control interaction and communication among agents.	Algorithm	Wong et al. (2006)
Scheduling	Neural reinforcement learning method is used and the empirical evaluation on large-scale benchmark problems leads to the conclusion that problems of current standards of difficulty can very well be effectively solved by the learning method. A disadvantage of this reactive scheduling approach is the fact that only non-delay schedules can be produced, which in many cases prohibits finding optimal schedules.	Algorithm	Gabel & Riedmiller (2007)
Process Planning and Scheduling	The study shows a bidding-based multi-agent system for solving integrated process planning and scheduling. The proposed architecture consists of various autonomous agents capable of communicating (bidding) with each other and making decisions based on their knowledge	Algorithm	Shukla et al. (2008)
Planning and Control	A Multi agent based agile manufacturing planning and control system ans exceptions are presented and RFID- based control module is used for manufacturing system The development of an AMPCS for an automated manufacturing cell demonstrates that the integration of RFID technique and MAS in developing an agile manufacturing planning and control system can really possess the characteristics of visibility, accountability, track ability, responsiveness, and flexibility in a distributed and dynamic manufacturing system	Algorithm	Wang & Lin (2009)
Scheduling	The simulation results show that the principle of the algorithms is simple, their computational quantity is small, and the algorithms can be applied to multi-batch dynamic scheduling with unpredictable entry time due to their favorable potential. It is suitable for multi-objective scheduling problems that need to consider average delay time and delivery cut-off time. The scheduling results of wasp colony algorithms are better than those of static scheduling algorithms for divided scheduling batch by batch.	Algorithm	Cao et al. (2009)

Table 1.1. Recent studies of Multi Agent Based Manufacturing solutions (Part 3)

Area	Explanation of Study	Algorithm / Industry	Article
Process Planning and Scheduling	Ant Colony optimization algorithm is implemented on the established multi agent platform, artificial ants are implemented as software agents to solve integrated process planning and scheduling problem. Results show that purposed model approach is not competitive with negotiation based multi agent system approach	Algorithm	Leung et al. (2010)
Process Planning and Scheduling	The research has been conducted to develop an agent-based approach to facilitate the integration and optimization of these two systems. Process planning and scheduling functions are carried out simultaneously. An optimization agent based on an evolutionary algorithm has been developed to optimize and realize the proper decisions resulting from interactions between the agents. To verify the feasibility of the proposed approach, a number of experimental studies have been carried out to compare this approach with other previously developed approaches. The experimental results show that the proposed approach is very effective for the IPPS problem and achieves better overall optimization results.	Algorithm	Li et al. (2010)
E-Procurement	The study proposes an agent based approach, properly linked to production planning activities able to support negotiation im catalog based e market place. Author emphasized that Model effectiveness is high for high volume production environment.	Algorithm	Renna & Argoneto (2010)
Enterprise Management	In this study, a model demonstrated that hierarchical coordination approaches can be applied to the enterprise management domain. Based on this insight, we proposed and implemented a model-based approach that allows for the solution of distributed planning-related coordination problems in enterprise management. Software agents are used to implement the coordination algorithm.	Algorithm	Heyne & Mönch (2011)
Supply Chain	The advantages of a multi-agent SCRM model over conventional ICT tools for risk management are multitude. Thereal-time adaptability and the learning capability through algorithms that is embedded in the model can lead to a more efficient response to information asymmetries amongst supply chain partners	Algorithm	Giannakis&Louis (2011)

When the research some of which are provided in Table 1.1 is analyzed, it can be seen that the use of agent-based technology provides benefits over various traditional manufacturing functionalities. There is countless number of studies in the literature which use intelligent agent-based technology indifferent manufacturing functions. Although, the most of them are on scheduling, process planning and control of manufacturing systems, only few of them are reported to be implemented in real manufacturing environments. Similarly, there seems to be not much study in the literature which performs multi-agent manufacturing as modelled and implemented in the way as proposed model called IMPACT integrating six capabilities related to manufacturing mainly, Goals, Processes, Operations, Machines, Products and Suppliers.

1.2. Main Objectives of the thesis

The main objectives of this thesis are listed below:

- To set-up well defined integrated manufacturing environment with self-decision making capability and modelling agents Increasing efficiency of manufacturing systems using agents' self-organization, pro-activeness and reactiveness properties with related experience and knowledge as well as inter-agent interaction mechanisms
- To demonstrate the role of communication and coordination between manufacturing agents with decreasing communication problems between different units of manufacturing
- To decrease manufacturing times by using optimization technics on manufacturing scheduling,
- To decrease manufacturing cost caused from purchasing raw materials, operation scheduling, WIP (Work-in-Process), overstock etc.
- To prevent lack of production and lack of operation
- To increase quality levels both of purchased and manufactured product.
- To increase control on manufacturing, scheduling, supplier selection, machine usage, operations and process management etc. capabilities
- To satisfy an ability of measurement for manufacturing goals, processes, operations, machines, products and suppliers for a manufacturing
- To prepare a road map for intelligent agent based manufacturing planning for manufacturers.

1.3. Scope of the study

The study is limited to 4 manufacturing functionalities including; procurement, stock management, production planning and manufacturing system. The proposed model is different from other studies in the literature in terms of applying an integrated and interoperable structure including an opportunity for measurability of suppliers; operations included in the processes, machinery for operation execution, products manufactured using available machinery and supplies and raw materials used for the manufacturing of the products.

The proposed framework can be extended to any function provided that the respective modelling is carried out.

1.4. Scope of the thesis

The thesis is set up in the following manner.

In the Chapter 1, Features of agent-based models are highlighted and reviewed. Agent based system interest for manufacturing systems with an overview of existing literature is stated. Based on manufacturing solutions along with several studies on the agent based modeling and demonstration of their effectiveness since the main focuses of this thesis is designed. Chapter 2 states basic concept of agent-based methodology. This Chapter provides basic information on agent technology including a general framework for agent architecture

Chapter 3 covers surveys of the recent papers and literature on multi agent system with an outlook on the described models in the thesis. Chapter 3 explains multi-agent systems covering fundamental characteristics, architecture, and cooperation and coordination issues.

Chapter 4 introduces the proposed model is so called IMPACT (Intelligent Manufacturing Planning using Agent-based Control Technology) and has a real-time information flow (available at any time to many sides). The main objective of this model is to define a general framework for creating a distributed, efficient and effective dynamic multi-agent-based production planning system. The proposed model is composed of four integrated manufacturing functions. These are Production Planning, Manufacturing, Stock Management, and Procurement Management. In order to satisfy the integration of these manufacturing functions, IMPACT proposes six different but interrelated models including;

- Goal Model,
- Process Model,
- Operation Model,
- Machine Model,
- Product Model,
- Supplier Model.

Each Model is considered to have agent architecture to act as a distributed function (component) of the manufacturing system. Intelligent agent technology is employed for generating and implementing the model. Hierarchical and interrelated model based decomposition approach is used to define agents.

In Chapter 5, application of purposed model is performed. The prototype application is carried out over Cabinet Production System. The prototype is implemented in a real manufacturing company located in İstanbul. For developing the prototype, the software program is coded using java programming language for implementing for the cooperation and coordination between the agents. An agent based architecture is developed in such a way that each process with its integrated information flow is designed depending upon the opinion of a responsible experts who actually works in the manufacturing suits.

The last chapter outlines the result of this study and provides research directions along this line.

2. AGENT BASED SYSTEMS

This Chapter provides a basic introduction to agents and agent technologies. The main characteristics of agents are highlighted. Advantages and disadvantages of using agent based technology are provided. Well known agent architectures and environments are introduced and agent coordination, cooperation and negotiation are discussed as key features of an agent.

2.1. Agents

An agent is hardware or software based computer system [Wooldridge and Jenning, 1995] that provides a means to bridge the gap between humans and machines by means of interaction and intelligence [Tweedale et al, 2007]. A physical agent uses eyes, ears and other sense organs life sensor in order to perceive environment. The agent reacts events perceived from environment through body organs such as hands, feet etc. A robotic agent uses technical components such as camera to perceive environment and it reacts events perceived from environment through various engines.

In this sense, a software agent is a structure in which actions are defined to system through codes in order to enable skill of perceiving and reacting to the environment [Russell and Norvig, 1995].

Mondalto et al (2003) provided the following definition of an agent: "A computational system, which has goals, sensors and effectors, decides autonomously which actions to take into account in accordance with the current situation in order to maximize the progress towards the (time varying) goals." The same authors went further to define a software agent as a "particular type of an agent, inhabiting computers and networks, assisting users with computer based-tasks." On the Internet, for example, software agents are programs that can gather information or perform some services without an immediate user presence.

Computational agents have taken its place within smart manufacturing systems. It is observed especially in researches carried out on distributed artificial intelligence (DAI) [Bond and Gasser, 1988].

Figure 2.1. It demonstrates how a generic software agent communicates with the environment through sensors and effectors and shows self-contained components formed in order to enable an information management structure.

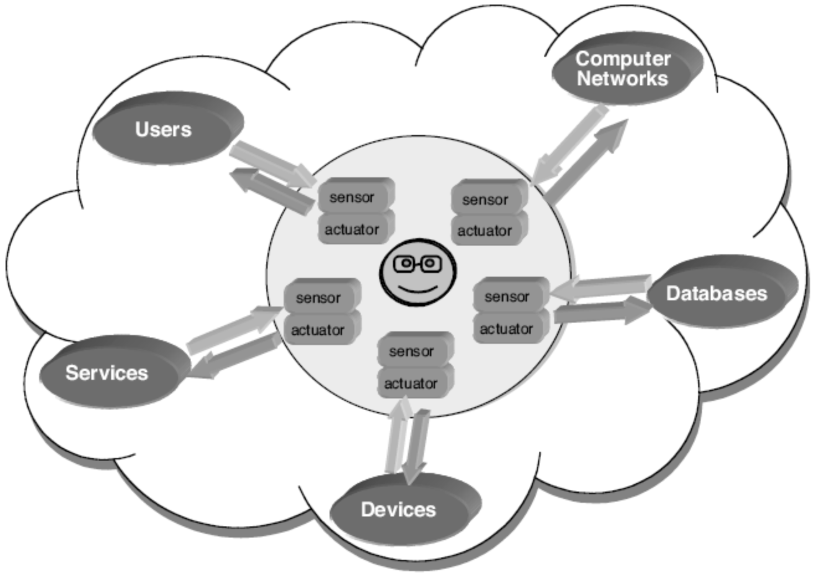

Figure 2.1 A Software Agent and Its Information-Based Environment [Paolucci and Sacile, 2005].

In other words, an **agent** is anything that can be viewed as **perceiving** its environment through **sensors** and **acting** upon that environment through **effectors** [Russell and Norving, 2010]. Illustration of the basic agent architecture in this respect is shown in Figure 2.2.

Wooldridge and Jennings (1995), on the other hand, identified three different classes of agents:

1. Agents fulfill the task brought to it by using predefined terms and assumptions.

2. Agents fulfill a well-defined task according to the demands of user.

3. Agents fulfill information and service operations assumed to be suitable for user without asking directly.

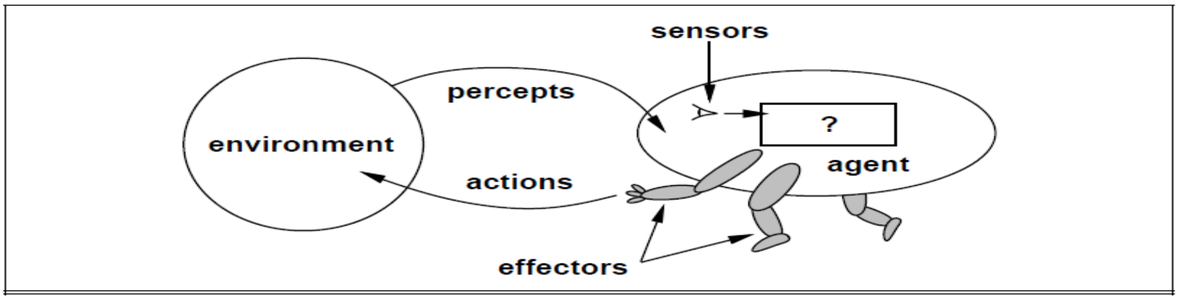

Figure 2.2 Basic components and its environment [Russell and Norving, 2010].

Another software agent representation depends on its attributes is created by Nwana (1996) as depicted in Figure 2.3

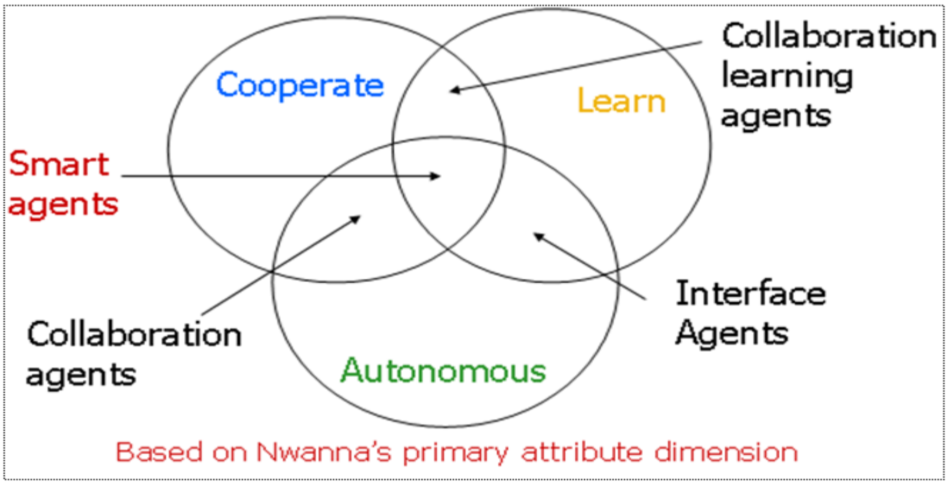

Figure 2.3 Attributes of software agents [Nwana, 1996]

Similarly, Tecuci (1998) defines an action of an agent as a mutual application of internal and external models which were defined within its structure and encoded in knowledge base and inference engine.

Knowledge base is a domain which composes infrastructure of Agent, defines terms defined in agent and objects, classes and relation of classes and objects.

Inference engine includes infrastructure which defines programs formed in order to solve problems and expressed how knowledge base is used in order to solve problem.

When considering successful performance of an agent, right action could be accepted as the most important reason to provide solutions of problems as well as to decide *how* and *when* an agent acts.

2.2. Main Characteristics of the Agents

The main characteristics of the agents are reported to be important in order to distinguish them from traditional systems [Paolucci and Sacile, 2005]. The following characteristics need to be understood clearly.

Autonomy: The agents should be able to perform most of their tasks without the direct intervention of humans and should have a degree of control over their own actions and their own internal state.

Social ability: The agents should be able to interact with other software agents as well as humans.

Responsiveness: The agents should perceive their environment and respond in a timely fashion to changes occurring there.

Proactiveness: When responding to their environment; the agents should exhibit opportunistic, goal-directed behavior and take the initiative when appropriate.

Adaptability: The agents should be able to modify its behavior over time in response to changing environmental conditions and to an enhanced knowledge about its problem-solving role.

Mobility: The agents should possess the ability to change its physical location to improve its problem-solving capacity.

Veracity: The agents should not knowingly communicate false information.

Rationality: The agents should be expected to act in order to achieve its goals and not to prevent its goals from being achieved without good cause

2.3. Intelligent Agents

Similar to the agents described above, an intelligent agent is a kind of knowledge based system that perceives its environment; automatically reasons to interpret perceptions, draw inferences, solve problems and determine actions; and acts upon that environment to realize a set of goals or tasks for which it was designed [Tecuci,1998]

Intelligent agents may be described as autonomous systems creating suitable answers for the events which they perceive from their environments. Intelligent agents have domain knowledge for perceiving events and inferring its consequences. And, they may have sensor and effectors as hardware [Russell and Norvig, 1995]. Figure 2.4 shows general structure of an intelligent agent [Tekez, 2006]. As seen from this Figure, any intelligent agent consists of three main elements:

Inference module: Perceives events from environment.

i. Cognition module: Infers about perceived knowledge.
ii. Action module: Acts suitable behaviors for answers produced by comprehension module.

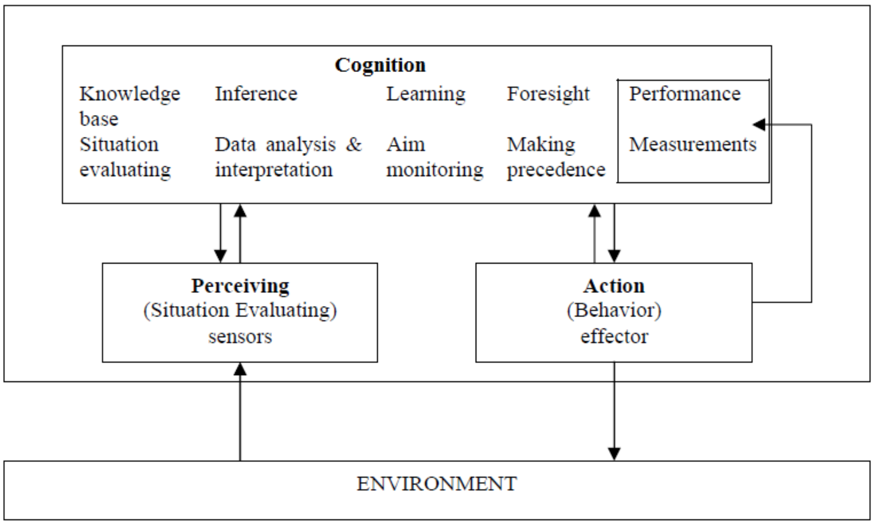

Figure 2.4 General architecture of an intelligent agent [adapted from Tekez, 2006]

An Agent perception module can be defined as computational models that could be considered in some ways analogous to human sensory perception mechanisms. Peters et al. (2011) defines two considerations when designing perceptual systems;

1. Acquiring the input from the real environment, using a laptop-mounted web camera or similar recording device
2. Acquiring input from the virtual environment, by endowing the agent with synthetic senses.
3. The need to integrate synthetic and real perception becomes even more obvious in augmented a reality which combines both virtual reality and real-world objects.

An Agent cognition module is justifiable as operations on knowledge elements

(Planning, Scheduling, Machine Learning, etc.). Bhattacharyya and Ohlsson (2010) explain the knowledge elements as; Knowledge elements are represented as lists of symbols. Symbols generically denote knowledge primitives (concepts), and the lists represent complex knowledge structures (facts, beliefs, principles, rules, schemas, blue prints, etc.) formed by combining the primitives. Huhns and Singh, (1998) defines the cognition as; beliefs, desires and intentions as can be seen in Figure 2.5

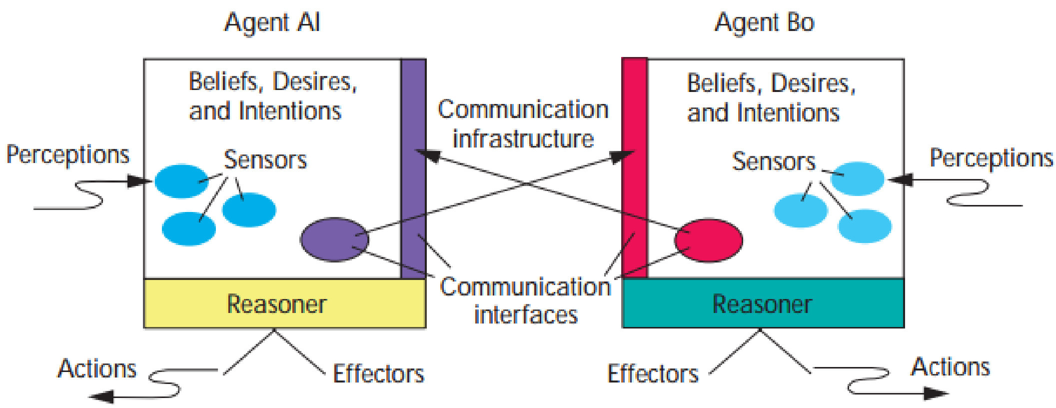

Figure 2.5. An architecture for an agent that captures the beliefs, desires, and intentions, (Huhns and Singh, 1998)

An Agent action module is acts suitable behaviors for answers produced by cognition module by the effector. An Agent "A computational system which is long lived, has goals, sensors and effectors, decides autonomously which actions to take in the current situation to maximize progress toward its (time varying) goals."[Paolucci and Sacile, 2005]. Another definition of Agent action is given by Gheorge (1998) as act upon that environment to realize a set of goals or tasks for which it was designed.

Other Capabilities of an intelligent agent can be list as below;

1. **Focus of attention:** The state of environment of an agent may change at any time so relationship between agent and related domain changes over time
2. **Prioritization:** Defines how tasks are prioritized by agents.

3. **Decentralization:** Each agent has incomplete information or capabilities for solving the problem, there is no system global control and data are decentralized.
4. **Conflict management:** Describes the management of interdependencies between agent activities.
5. **Real time action:** When an input arrives to agents from the system, agents are designed to have a skill to process this immediately.
6. **Performance assessment:** Different recommendation of different decision maker may lead to different assessment of system performance for the decision support system.
7. **Learning:** Agents modify their systems dynamically according to the environmental conditions and provide better operation of the system.
8. **Continuous and asynchronous communication:** Agents can sense dynamic changes in the environmental conditions and have continuous functions to carry out the most appropriate actions for these conditions. Agents have a blackboard structure for data that are in a state erased from the system or not reached from the system. This structure provides a asynchronous local communication for these agents.
9. **Recency:** Agent Recency-based Exploration uses counts of the number of time steps that have passed since an action was taken from a particular state
10. **Coherence:** An Agent coherence means that coherent set of components and subsystems working together to engage in goal-driven activities.
11. **Flexibility:** Agents have flexible behavior: reactive, proactive, social.
12. **Timeliness:** Slowness of the data reaching from the sensors and the controls during the decision making process affect unfavorably controlling the dynamic environmental conditions and right decision making actions in the right time.
13. **Adaptation:** The agents can adapt to changes in the environment, behavior of other agents and learn from experience.

14. **and Graceful degradation**: The graceful degradation means that the multi agent system does not fail drastically at the boundaries.

2.4 Capabilities of Intelligent Agents

Intelligent agents (or intelligent software agents) usually has [Wiki-SA, 2013] the following capabilities.

1. **Adaptation:** It defines sensing the environment and reconfiguring itself according to the changing conditions. Achieving this can be possible by enabling the following; definition of alternative problem solution techniques in the system, improvement of problem defining strategies in order to sense environmental conditions and problems, and expanding algorithmic problem solution techniques within the system.
2. **Learning:** Learning process can be defined in different forms in agent based systems. As it is possible to achieve this by establishing a trial-and-error structure, this characteristic can be defined in the system by the sampling method as well.
3. **Coordination:** Coordinating different events and facts could be easily realized by intelligent agents. Coordination is required to determine organizational structure amongst a group of agents and for task and resource allocation
4. **Planning:** An intelligent agent could create own-plans from system data and information. Planned results and reports could be easily generated by intelligent agent.
5. **Using different knowledge sources:** Especially, an intelligent agent could gather required information and data from different knowledge sources.
6. **Using different AI techniques:** An intelligent agent is also represents an AI technology. An expert system, genetic algorithm or artificial neural network could be easily utilized.

2.5. Advantages and Disadvantages of intelligent agents

Advantages and disadvantages of the intelligent agent based technology will be discussed under this heading

2.5.1. Advantages of using intelligent agents

The main advantages of the intelligent agent technology are listed as follows:

1. **Providing higher productivity:** With agents, the amount of work that can be accomplished is much higher in volume than with a closed system **[**Green et al., 1997].
2. **Enabling distributed computing:** Enabling distributed computing can be expressed as the basic philosophy of agent-based computing. Distributing computing concept is defined as usage of more than one mobile agent in the system by Fuda et al (1999). Das (2013) defines the distributing computing as autonomic movements between nodes within a network.
3. **Sustain economic benefits:** Agent-based systems enable an equivalent computing power by using the method of answering instances that reach separately to the computer situated within a network, instead of answering all instances within a system by a very powerful single computer.
4. **Reducing network traffic:** With today's RPC model commands are shouted across the network from computer to computer. For the reducing network flow of raw data is possible with using mobile agent. For the very large volumes of data are stored at remote hosts, data should be processed in its locality rather than transferred over the network.
5. **Robustness of an agent system:** Instead of a system where more than one error arrives a single computer simultaneously, the ability to manage errors and instances by means of different distributed components within a multi agent based system enables the agents to have a robust structure. MAS is regarded as the system enabling to be robust the best owing to its distributed computing characteristic. Multiple instances of agents remove single points of failure and so improve performance [Bezek and Gams, 2006].

2.5.2. Disadvantages of using intelligent agents

The main disadvantages of the intelligent agent technology are listed as follows:

1. **Security:** The idea of allowing a program that could potentially destroy the contents of someone's computer is very troubling. If agent based computing is to succeed, some concrete measures are to be taken to ensure that agents are unable to harm their host computer **[**Green et al.,, 1997**]**.
2. **No standard:** There is still no agreed upon standard for agent communication. Without a standard, agents cannot talks the same language to would be able to communicate with each other **[**Green et al.,, 1997**]**.
3. **It is impossible to predict exactly the system interaction and execution**: The complexity of an agent society and unpredictability of system resources makes it impossible to predict exactly the agent interaction and activity over time. In addition, there is no guarantee of system correctness [Bezek and Gams, 2006].
4. **Developers must predict failed or multiple responses**: In the agent environment there is no guarantee about the response to agent queries; there may be one, many or even no response at all. Developers must implement additional safety mechanisms to prevent misinterpretations of an unwanted response, which is often a daunting task [Bezek and Gams, 2006].
5. **A MAS introduces new synchronization problems:** This is one of the hardest problems when developing distributed systems. The synchronization of multiple activities with multiple requests, where activities and requests can exclude each other, is difficult. In our approach, each agent prevents concurrent access by locking access to its internal state and by serializing mutually exclusive tasks [Bezek and Gams, 2006].
6. **The behavior of individuals: B**ehavior of individuals cannot clearly be defined through aggregate transition rates [Bonabeau, 2002]
7. **The nature of the system being modelled is another consideration**: For example, a system based on human beings will involve agents with potentially irrational behaviour, subjective choices, and complex psychology. These factors are difficult to quantify, calibrate, and sometimes justify, which complicates the

implementation and development of a model, as well as the interpretation of the simulation outputs [Bonabeau, 2002].

8. **Limitation high computational requirement:** Although computing power is increasing rapidly, the high computational requirement of agent bases systems remains a limitation when modelling large systems [Bonabeau, 2002].

2.6. Architecture of an Agent

An architecture of an agent conceptualizes that the agents are made of 3 main components namely, perception, action, and reasoning. The perception component feeds the reasoning component, which governs the agents' actions, including what to perceive next [Flores-Mendez, 1999]. It is possible to classify agents into five groups based on their degree of perceived intelligence and capability:

1. **Simple reflex agents;** condition-action rules allow the agent to make the connection from percept to action [Russell and Norvig, 1995]
2. **Model-based reflex agents:** Model-based agents can handle partially observable environments. Its current state is stored inside the agent maintaining some kind of structure which describes the part of the world which cannot be seen. This behavior requires information on how the world behaves and works. This additional information completes the "World View" model [Moyaux et al, 2006].
3. **Goal-based agents:** Goal-based agents are model-based agents which store information regarding situations that are desirable. This allows the agent a way to choose among multiple possibilities, selecting the one which reaches a goal state [Moyaux et al, 2006].
4. **Utility-based agents:** Goals alone are not really enough to generate high-quality behavior. Goal just provides a crude distinction between "happy" and "unhappy" states. Utility based agent try to maximize their happiness [Russel and Norvig 1995]
5. **Learning agents:** An agent that is able by itself to acquire and maintain its knowledge is called a learning agent [Tecuci, 1998]. The learning agent could

learn from a variety of information sources in the environment. It could learn its user or from other agents. If an agent could learn, then the process of building and maintaining its knowledge base would be significantly simplified.

2.6.1. Deliberative Architectures of an Agent

Deliberative agent is a kind of a software agent which is use symbolic architecture (Genesereth and Nilsson, 1987). The foundation upon which the symbolic AI paradigm rests is the physical-symbol system hypothesis, formulated by Newell and Simon (1976).

Based on a strong notion of agents, deliberative agent contains an explicitly represented, symbolic model of the world, and decisions reasoning, based on pattern matching and symbolic manipulation.

The agents are modeled through decomposition of the control architecture. In this approach functional information processing model is used. These are "knowledge representation", "learning", "planning", "and qualitative reasoning".Information processor is designed as a symbolic descriptions are used. Combination of these different modules defines the the systems' behavior.

Two important problems to be solved by deliberative agent is listed as:

1. **The transduction problem**: that of translating the real world into an accurate, adequate symbolic description, in time for that description to be useful.
2. **The representation/reasoning problem:** that of how to symbolically represent information about complex real world entities and processes (since computers may simulate reasoning but not the interaction with the environment), and how to get agents to reason with this information in time for the results to be useful.

2.6.2. Reactive Architectures of an Agent

A reactive architecture does not contain any kind of symbolic model, and does not use complex symbolic reasoning.

Possibly the most vocal critic of the symbolic AI notion of agency has been Rodney Brooks, a researcher at MIT who apparently became frustrated by AI approaches to building control mechanisms for autonomous mobile robots. In a 1985 paper, he proposed alternative agent architecture for building agents, the so called subsumption architecture. Layered composition of simple behaviors is described by Brooks (1986), the subsumption architecture provides a powerful means for defining intelligent robot control mechanisms.

Brooks (1986) proposed three main features;

1. Intelligent behavior can be generated without explicit representations of the kind that symbolic AI proposes.
2. Intelligent behavior can be generated without explicit abstract reasoning of the kind that symbolic AI proposes.
3. Intelligence is an emergent property of certain complex systems.

He identifies two key ideas that become the baseline for his research his research:

1. **Situatedness and embodiment:** "Real" intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems.

2. **Intelligence and emergence:** "Intelligent" behavior arises as a result of an agent's interaction with its environment. Also, intelligence manifests "in the eye of the beholder", it is not an innate, isolated property.

2.6.3. Hybrid Architectures

Hybrişd architecture combines reactive and deliberative components and form a hierarchy of interacting layers. Two type of layer is considered in this approach. Firs is horizontal layering and second is vertical layering. These layers are used to reasoning at a different level of abstractions.

Subrahmanian (2000) proposed that neither a completely deliberative nor completely reactive approach is suitable for building agents. Many researchers have argued the case for hybrid systems: which attempt to marry classical and alternative approaches.

An obvious approach is to build an agent out of two (or more) subsystems: a deliberative one, containing a symbolic world model, which develops plans and makes decisions in the way proposed by symbolic AI; and a reactive one, which is capable of reacting to events that occur in the environment without engaging in complex reasoning. Often, the reactive component is given some kind of precedence over the deliberative one, so that it can provide a rapid response to important environmental events. An application of this approach can be seen in Touringmachines (Ferguson, 1992) and Interrap (Müller and Pischel, 1993)

This kind of structuring leads naturally to the idea of a layered architecture. In such architecture, an agent's control subsystems are arranged into a hierarchy, with higher layers dealing with information at increasing levels of abstraction. Thus, for example, the very lowest layer might map raw sensor data directly onto effector outputs, while the uppermost layer deals with long-term goals. A key problem in such architectures is what kind control framework to embed the agent's subsystems in, to manage the interactions between the various layers.

2.7. The Agent Environment

In this section, different types of environments and how they have an effect on the design of agent will be defined. Russel and Norvig (1995) provide different types of environment in which the agents operate;

- **Accessible versus Inaccessible Environments:** If an agent able to obtain complete and accurate state of environment, then the environment is said to be accessible. An environment is effectively accessible if the sensors detect all aspects that are relevant to the choice of action.
- **Deterministic versus Nondeterministic Environments:** If the next state of the environment is completely determined. and there is no uncertainity about the state for the actions selected by agents, then the environment is said to be deterministic.
- **Episodic versus Non-episodic Environments:** If the next event is dependent on previous action environment said to be episodic. In this method the agent's experience is divided into episodes. Each episode consists of the agent perceiving and then acting. Episodic environments is said to be much simpler because the agent does not need to think ahead.
- **Static versus Dynamic Environments:** If the environment can change while an agent is deliberating, the environment is aid to be dynamic otherwise it is static.
- **Discrete versus Continuous Environments:** If there are limited number of distinct, clearly defined percepts and actions then the environment is said to be discrete; otherwise it is continuous.

2.8. Multi Agent Systems

Multi-agent systems (MAS), describes the set of agents with some degree of autonomy and the complexities arising from their interactions. In the agent-oriented approaches, the MAS designer focuses on the individual actions of the agents, so MAS are designed in terms of agent mental states (such as beliefs, intentions, goals, commitments, etc.). Generally, the protocol for multi-agent systems calls for the information exchanging between neighboring agents in a periodic mechanism (Olfati and Murray, 2004).

Several authors have provided different taxonomies for MAS applications. Many early multi-agent systems had closed architectures where the specific interactions were effectively "hard wired" at design time. The FIPA (Foundation for Intelligent Physical Agents') Agent Management Reference model, on the other hand, provides an open architecture, i.e. an architecture to which agents can easily be added and removed (Mcarthur et al., 2007)

Detailed information of Multi Agent Systems is given in Chapter 3

2.9. Coordination, Cooperation and Negotiation of Agents

The key issues of agents in a multi-agent system are coordination, cooperation, negotiation, communication and collaboration. In order to have a coordinated set up of collaborative agents, they may have to negotiate in order to reach mutually acceptable agreements on some matters.

Coordination is central to the successful operation of agent-based manufacturing systems which are very complex and whose stability is essential [Shen et al, 2006]. The agents can generate and execute their plans independently. However, as they operate in the same environment, sometimes conflicts may arise. Therefore, they need to coordinate their course of action in order to avoid harmful interactions [Dimopoulos and Moraitis, 2006]. Without coordination, a group of agents can quickly degenerate into a chaotic collection of individuals. This situation is absolutely unacceptable in real manufacturing enterprises [Shen et al, 2006].

The term coordination has been used in explaining concepts in the areas of psychology, sociology, biology, management, finance and information technology [Consoli et al., 2006]. A working definition is the act of managing interdependencies between activities performed to achieve a goal [Malone and Crowston, 1990].

A universally accepted definition of cooperation is *acting together with a common purpose* [Hua, 2004]. When an autonomous agent enters a relationship with another agent voluntarily, it is said to be cooperating [Consoli et al., 2006].

A successful agent-based systems interoperation initiative is Agent cities [Willmott et al., 2001]. The Agent cities initiative works towards the vision of an ambient proactive environment where heterogeneous, autonomous and increasingly intelligent systems, which represent businesses, services and individuals, are able to interact with each other and enable flexible and dynamic composition of services [Karageorgos et al., 2003].

Agents can use negotiation to reduce the search tree and agree on viable optimized schedules [Karageorgos et al., 2003] .Negotiation is used in most agent-based manufacturing systems for resource allocation[Shen et al., 2006].

3. AGENT BASED MANUFACTURING SYSTEM

3.1. Agent Based Systems and Modelling

Agent-based systems that are composed of simple locally interacting agents demonstrating complex group behavior, offer several advantages to get benefit of of communication and computational science. A well-designed complex agent-based system is so called an efficient, robust, adaptive and stable operating system. It has low communication and computational requirements, meaning that there are virtually no constraints on system size. The simplicity of agent interactions also makes it amenable to quantitative mathematical analysis. In addition to offering predictive power, mathematical analysis enables the system design to optimize system performance [Lerman, 2001].

Based on this understanding an agent-based model consists of:

- A set of agents (part of the user-defined model)
- A set of agent relationship (part of the user-defined model)
- A framework for simulating agent behaviors and interactions provided by an ABMS (Agent Based Manufacturing System) toolkit or other implementation

Unlike other modeling approaches, agent-based modeling begins and ends with the agent's perspective highlighting how agent based models actually work.

Evaluating agent-based system modeling is the examination of agent-based system characteristics as listed below [Sheory, 2000] [Jennings and Wooldridge, 2000] [Jennings, 2000] [Jennings et al., 1996].

- **Autonomy:** unlike other objects, the agents may be active and are responsible for their own activities possessing a self-controlled architecture. An agent in this context has control over both its reactive and proactive behavior. The modeling technique should support the capability of describing an agent's self-control feature.
- **Complexity: A**gent-based systems are basically the set of components (agents) that interact with each other in order to achieve their pre-specified goals. These

systems may consist of decision making, learning, reasoning mechanisms as well as other complex algorithms. Modeling complex algorithms and mechanisms requires strong expressive power and many layers of details. A modeling technique should therefore support such expressiveness in order to model the functionality of agent-based systems. Moreover, the complexity requires that the techniques used for agent modeling should be modular, should support complexity management and describe the complex nature of the agents.

- **Adaptability:** Agent-based systems have to be flexible in order to adjust their activities to the dynamic environmental changes. The adaptability feature may require that the modeling techniques should be modular and that activate each component according to the environmental state.

- **Concurrency:** An agent may need to perform several activities/tasks at the same time. The concurrency feature raises the requirement that some agent-based systems must be designed as parallel processing systems. This requires ability to express parallelism and concurrency in the design and implementation/deployment stages.

- **Distribution** Agent systems are sometimes working on different hosts and should be distributed over a network (multi-agent systems). This requires ability to express distribution in the design and implementation/deployment stages.

- **Communication richness:** Basic definition of an agent consists of its autonomous activities. As such, the agent must establish communication with its environment. The environment may include other agents and information sources. The communication is characterized by its type (either inter-agent or intra-agent communication), its content and architecture (e.g. client-server, peer-to-peer). This requires that a modeling technique should be able to express the communication characterization in order to produce agent communication command or sentences during the implementation stage.

Agent-Based Modeling

Agent based modelling can be considered as creating an agent based information systems designed to perform pre-defined set of activities for achieving a set of goals defined on purpose.

Maca and North (2005), provides the rational for agent based modelling as the following.

- Increasing;
  - the complexity of systems
  - decentralization of decision-making
  - the capability to approach the design limits
  - physical and economic Interdependencies
- Generation and implementation of New tools, toolkits, modeling approaches
  - Some systems have always been complex, but tools did not exist to analyze them
- Controlling the huge Data set
  - Data now organized into databases at finer levels of granularity (micro-data)
  - can now support micro-simulations
- Increasing the Computational Power
  - Computational power advancing

Basic components of an agent-based modelling include [Broersen, 2011];

1. **Environment**, i.e. the place where agents live and interact. The environment can be accessible or non-accessible, static or dynamic, deterministic or non-deterministic and episodic or non-episodic;
2. **Interactions,** which are different perceptions and actions of one or more agents within an environment and probably with the environment itself and amongst themselves as well. Perceptions are usually received from sensors and actions are made with actuators.

An agent-based model consists of [Grim et al, 2006]:

1. **Overview:** Purpose, state variables and scales, process overview and scheduling (flowcharts, etc.),
2. **Design concepts:** Emergence, adaptation, fitness, prediction, sensing, interaction, stochasticity, collectives (groups), observation (data collection), and
3. **Details:** Initialization, input, sub-models (mathematical "skeleton"). Our approach can be used to specify each level.

Agent based modeling includes the simulation of a problem from the real world, the analysis and optimization of the model and the replacement of the solution back to the real world. A simulation model may be considered as a set of rules that defines how the system, being modeled, will change in the future, given its present state. Agent-based model consists of a system of agents and the relationships between them. Even a simple agent-based model can exhibit complex behavior patterns [Reynolds, 1987] and provide valuable information about the dynamics of the real-world system that it emulates. In addition, agents may be capable of evolving, allowing unanticipated behaviors to emerge. Sophisticated ABM sometimes incorporates neural networks, evolutionary algorithms, or other learning techniques to allow realistic learning and adaptation [Bonabeau, 2002].

In agent-based modeling (ABM), a system is modeled as a collection of autonomous decision-making entities called agents. Each agent individually assesses its own situation and makes decisions on the basis of a set of rules. Agents may execute various behaviors appropriate for the system they represent—for example, producing, consuming, or selling. Repetitive competitive interactions between agents are a feature of agent-based modeling, which relies on the power of computers to explore dynamics out of the reach of pure mathematical methods [Epstein&Axtell, 1996 and Axelrod, 1997].

a) Agent Architecture Design which comprises mainly three activities;

1. **Sensors for perception**: the agent receives information from its environment. The environment may be another agent or the general physical system (it could be Camera, Sonar, Laser range-finders etc.),

2. **Cognition**: decision making depends on given filters, conditions, based on its understanding of the context of the information. This component adds to the experience of the agent and if it is necessary changes in the agent that affect future cognition (Planning, Scheduling, Machine Learning etc.).
3. **Detectors or Effectors for action**: as results of the cognition, agent measures of utility of each action and behavior and agent decides a behavior for the change in the environment

b) Behavior Modelling

Each agent has a certain goal and represents an entity (object, person...). Based on this, it should definitely be able to perform the related behavior as in its originality. This consists of the attitudes, believes, desires, intentions and obligations etc. of a certain entity represented by particular agent. This is therefore one of the core component of the agent modelling as the level of intelligence is to be implemented by certain behavior.

c) Business Rules

Agents perform the respective behavior using so called business rules. Business rules as main directors and diverters of the agent baheviours. They identify the limits and conditions for basic activities. The agents receives the knowledge, reasons about it and acts accordingly using these rules. Business rules are those indicating the level olf intelligence that an agent posseses.

d) Environment Design

By being an independent entity, an agent should operate in a common environment. Designing a synthetic or sensor based environment is therefore an important component of agent modelling.

In the light of the above explanations, one of the basic architecture of agent based modelling is described as Belief-Desire-Intention (BDI) Agent [Brathman, 1987] in that the conception of actions is defined by beliefs, desires and intentions. In the BDI approach, the rational balance between mental attitudes is characterized by the pre-defined properties. Figure 3.1 shows that main architecture of the BDI agent.

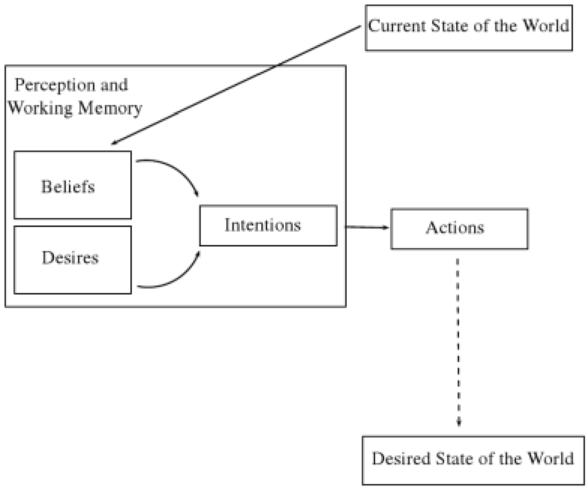

Figure 3.1. The BDI Agent Architecture (Brathman, 1987)

For the conflict tendency of the BDI Agent, Belief-Obligation-Intention-Desire (BOID) is developed. It gives an ability of the set obligation (external motivational attitudes) throw in BID agent performance (Thomason, 2000). BOID interaction is the same with BID agent but its decision alternatives are limited with given obligations. Figure 3.2 displays the BOID architecture.

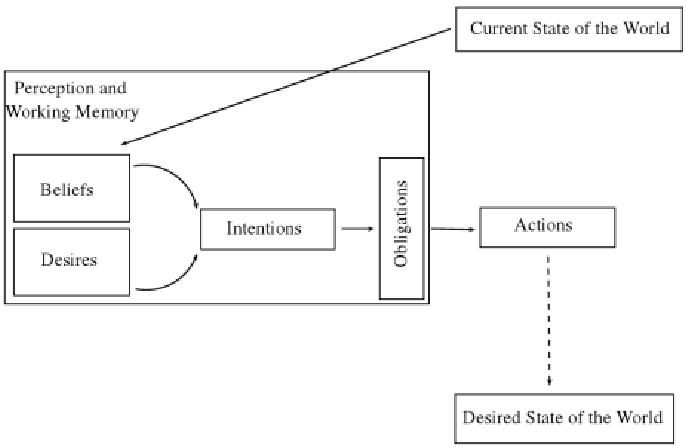

Figure 3.2. The BOID Agent Architecture (Thomason, 2000)

There are certain types of BOID Agents such as;

- **Realistic agents:** beliefs are generated before obligations, intentions or desires Single-minded or stable agents: entertain changes to beliefs than will drop its commitments accordingly. – Open-minded or unstable agents: This allows changes in both desires and beliefs.

- **Selfish agents:** sends a request for cooperation but never accept a cooperation request.
- **Social agents:** obligations override desires

Evaluating the performance of agent based modelling shows that the agent possesses an ability to perform respective dynamics of complex systems and complex adaptive behavior. Also this assessment provides an ability to observe the collective effects of behavior and related interactions.

3.2. Multi-Agent Systems (MAS)

Multi agent systems are considered to be an integrated architecture composed of two or more agent that is designed for a specific purpose.

Existing manufacturing control systems are usually based on deterministic algorithms operating in a predictable and stable environment. However future manufacturing systems will be increasingly more dynamic and flexible as well as mobile. They have to be able to rapidly respond to changing conditions by concurrently balancing and optimizing multiple manufacturing constraints. This will definitely require several agents working together for a specific business purposes. Solutions to manufacturing problems could be assured better utilizing multi-agents. This was realized through numerous examples.

A multi-agent system can be described as following (Wooldridge, 2002):

- A multi-agent system is one that consists of a number of agents, which interact with one-another.
- In the most general case, agents will be acting on behalf of users with different goals and motivations.
- To successfully interact, they will require the ability to cooperate, coordinate, and negotiate with each other, much as people do. Jenning (2000) is proposed an

schematic representation of Multi Agent Systems. An overview of a multi-agent system can be shown as in Figure 3.3.

According to Stone and Veloso (1997) multi-agent systems are best classified along two main dimensions:

- Homogenous vs. heterogeneous agents (in terms of agent architecture: reactive and/or deliberative).
- Non-communicating vs. communicating agents (protocols, human involvement).

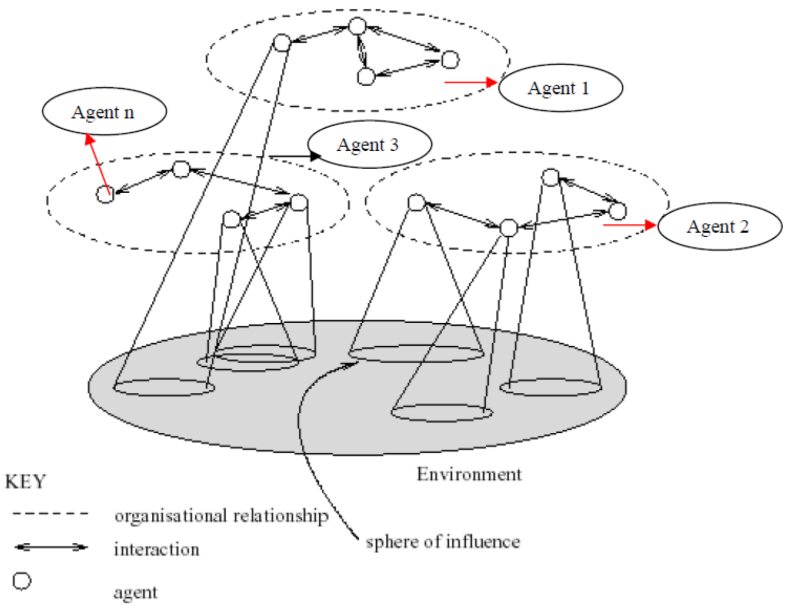

Figure 3.3 schematic representations of Multi Agent Systems (Jenning, 2000)

Multi-agent system is a collection of heterogeneous, encapsulated applications (agents) that participate in the decision making process. Agents communicate, collaborate and negotiate in order to meet its own design objective but also a goal that is shared within the community (Pěchouček et al, 2002).

As in other areas, multi agent systems are also very popular in manufacturing. Some examples of those applications are provided below.

3.2.2. Advantages and Disadvantages of Multi Agent Systems

Because of the large volume of inter-related information, new approaches have to be investigated to achieve the complex multi-objective decisions. The common centralized knowledge base has now been superseded by a distributed base, which appears more advantageous. In a distributed environment, information related to different domains can be separated and better organized. Software agents offer a convenient way of modeling processes that are distributed over space and time. They can be regarded as autonomous and are good candidates for domains that require constant adaptation to changing environment or demands.

Pfeifer (2001) provides the following justification for motivation and benefits of MAS as They can;.

- *s*olve problems that are too large for a centralized single agent to do due to resource limitations or the sheer risk of having one centralized system.
- provide solutions to inherently distributed problems, e.g. air-traffic control.
- provide solutions which draw from distributed information sources, e.g. for distributed on-line information sources, it is natural to adopt a distributed and collaborative agent approach.
- provide solutions where the expertise is distributed, e.g. in health care provisioning.
- exhibit modularity (which reduces complexity).
- provide enough scalability (since they are inherently modular).
- increase response speed (due to parallelism).
- sustain robustness and reliability (due to redundancy).
- provide flexibility (i.e. new tasks are composed more easily from the more modular organization).
- attain a better understanding of interactions among human societies.
- possess enough intelligence in order to make decision and solve related problems.

3.3. Agent Based Manufacturing Systems

Manufacturing processing planning is the process of selecting and sequencing manufacturing processes such that they achieve one or more goals and satisfy a set of domain constraints [Shen et al., 2006]. The intelligent decision support required knowing how information from the data set is produced, measured and interpreted, in order meaningful conclusions to be produced from the agents. Agents are considered an important paradigm for developing industrial distributed systems [Fox et al., 2000].The knowledge repositories given contained the acquired data, while agents could be used for data analysis, integration and understanding of knowledge.

Possible manufacturing agents should possess the architecture given in Figure 3.4.

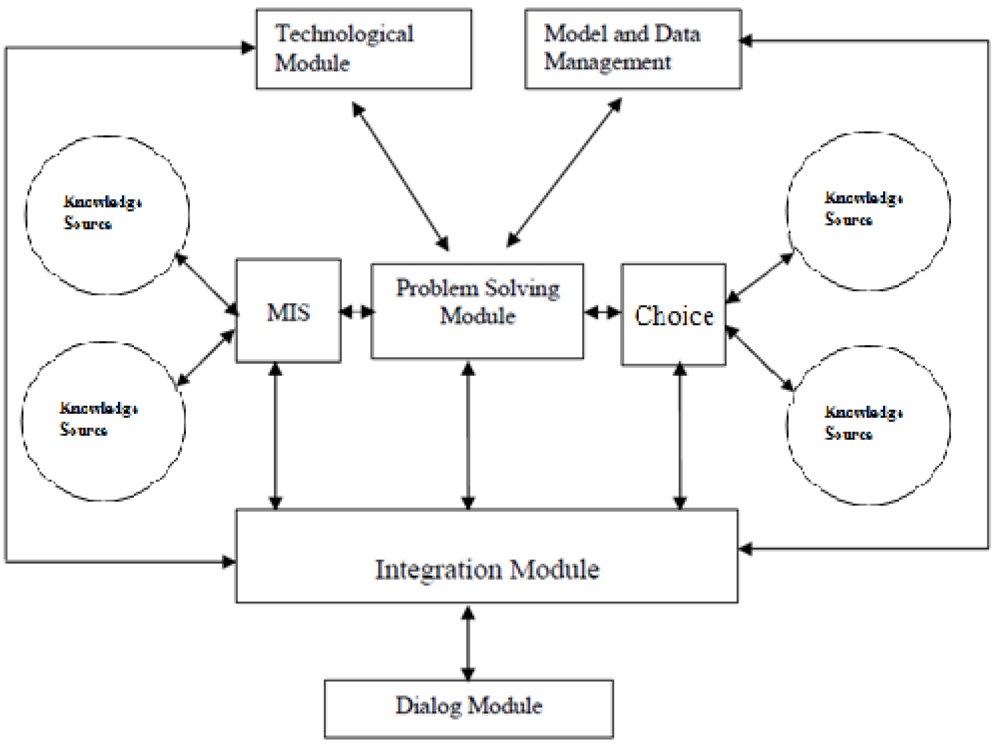

Figure 3.4. A General Framework for an Intelligent Manufacturing System (a manufacturing agent) (Oztemel, 1999)

The manufacturing enterprises of the 21st century are in an environment where markets are frequently shifting, new technologies are continuously emerging, and competition is globally increasing. Rapid changes in product demand, product design, introduction of new products and increasing global competition require manufacturing systems to be highly flexible, adaptable and responsive [Revilla and Cadena,2008] [Petrović et al., 2011].

Such manufacturing systems will need to satisfy a number of fundamental requirements [Shen et al., 2006] as listed below:

R1. Full integration of heterogeneous software and hardware systems within an enterprise, a virtual enterprise, or across a supply chain;

R2. Open system architecture to accommodate new subsystems (software or hardware) or dismantle existing subsystems ‘‘on the fly’’;

R3. Efficient and effective communication and cooperation among departments within an enterprise and among enterprises;

R4. Embodiment of human factors into manufacturing systems;

R5. Quick response to external order changes and unexpected disturbances from both internal and external manufacturing environments;

R6. Fault tolerance both at the system level and at the subsystem level so as to detect and recover from system failures and minimize their impacts on the working environment.

The agents can be used to encapsulate manufacturing activities or wrap legacy software systems in an open, distributed intelligent environment using a functional decomposition approach [Shen et al., 2006] Examples of such functional agents include order processing, product design, engineering analysis, process planning, production planning and scheduling, simulation, and execution [Shen et al. (2006); Fox et al., (1993);Peng et al., (1999);Barry et al., (1998);Shen , (2002);Azevedo et al., (2004)].

Agents can be implemented to represent physical manufacturing resources (e.g., machines, robots, tools, fixtures, AGVs, and operators) or aggregations of resources (e.g., cells, production lines, and shop floors) as well as products, parts, and operations (Shen et al, 2006; Butler and Ohtsubo, 1992; Shen and Norrie, 2001; Lu and Yih, 2001; Usher, 2003]. For example, scheduling of product orders on a number of machines is a non-deterministic polynomial (NP) hard problem [Bongaerts, 1998] [Jain and Meeran, 1998].Given this set of assumptions there are (n!) possible solutions for the allocation of orders to machines [Jain and Meeran, 1998].

Agents can be deployed to represent negotiation partners, either physical plants or virtual players, such as master plants, virtual partners, or dynamic consortia [Nigro et al., *2003*] [Sadeh et al., 2003] [Hao et al., 2003] to facilitate enterprise collaboration [Shen et al., 2006].

Qiao and Zhu, (2000) presented conceptual model of the software agent that includes manufacturing agent and proposed an agent based intelligent manufacturing system. In that model; efficient acquisition and utilization of knowledge have been considered as the trump to win the competition in the knowledge economy

Albus (1994) generated a reference model including functions and relationship of each function to a real time control mechanism. Lu and Yih (2001) studied an elevator manufacturing control model including simple index to partial assembly knowledge exchange and adjustment of dynamic precedence. Nassima et al. (2013) create a new agent based structure to solve scheduling of workshops problem. Watanabe et. al. (2001) studied an autonomous decentralized manufacturing system for intelligent driving toward to understanding mutual behavior. Alippi et al. (2002) studied on process quality measuring and generated an autonomous system to measure of quality analysis. Kim et al. (2003) used both genetic algorithm and agent based scheduling model to warehouse management.

Agent based systems are also used for solution of well-known scheduling problems, some of them is listed in Table 3.1;

Various methods have been proposed to deal with the optimal allocation problem. For example, Ren et al. (2011) proposed an effective methodology for a robust global optimization of electromagnetic devices, which is based on the gradient index and multi-objective optimization method. Zhang et al. (2011) presented a new manufacturing resource allocation method, by using an extended genetic algorithm (GA) to support the multi-objective decision-making optimization for the supply chain deployment.

Song et al. (2005) proposed a machine learning approach for determining feasible plans of a remanufacturing system, where the rough set theory was applied to establish the relationship between a plan and its feasibility, and an iterative reinforcement process was used to enhance the confidence.

The dynamic resource allocation based on distributed multiple criteria decisions in computing cloud is explained in Ruiz-Alvarez and Humphrey (2011). In it, author contributes in two ways. First distributed architecture is adopted, in which resource management is divided into independent tasks, each of which is performed by Autonomous Node Agents (NA) in a cycle of three activities: (1) VM Placement, in it suitable physical machine (PM) is found which is capable of running a given VM and then assigning VM to that PM, (2) Monitoring, in which total resources use by hosted VM are monitored by NA, (3) In VM Selection, if local accommodation is not possible, a VM need to migrate at another PM and process loops back to into placement. The second using PROMETHEE method and NA carry out configuration in parallel through multiple criteria decision analysis. This approach is potentially more feasible in large data centers than centralized approaches.

In Alvarez and Humphrey (2011), an automated approach to the selection of cloud storage services that can meet the user requirements is described. In Hill and Humphrey (2011), the authors' goal with CSAL (Cloud Storage Abstraction Layer) is to leverage application portability to explore multi-cloud application deployments and management as well as dynamic resource allocation optimization for cost and performance metrics.

Intelligent multi-agent model is purposed by Kim et al., (2011.Proposes an intelligent multi-agent model based on virtualization rules for resource virtualization (IMAV) to automatically allocate service resources suitable for mobile devices. It infers user demand by analyzing and learning user context information. In addition, it allocates service resources according to use types so that users are able to utilize reliable service resources.

Table 3.1 Agent Based Scheduling Systems (Calis and Bulkan, 2012)

Objective Function	**Explanation of Study**	**Reference**
Flow time and Tardiness	The higher frequency of learning may help an agent to quickly adapt to variations on the shop floor.	Pendharkar. (1999).
Mean Tardiness	The system is composed of the agent and the simulated environment (SE). The agent is able to perform dynamic scheduling based on the available information provided by the SE. It makes decision for selection of the most appropriate dispatching rule in real time. At the end of training, the agent gives better results than the traditional alternatives (SPT, COVERT, and CR rules).	Aydin and Oztemel (2000)
Makespan	A parallel implementation of modular simulated annealing algorithm (MSA) is presented, In order to run the parallel MSA , multi agent systems is used .The empirical results show that the method is quite successful comparing to the sequential version of MSA.	Aydın and Fogarty, (2004)
Number of Tardy job	An experimental approach for performance analysis of a multi-agent system for job routing in job-shop settings. Some simple but practical mechanisms are proposed and implemented	Usher (2003)
Total Tardiness	Neural reinforcement learning method is used and The empirical evaluation on large-scale benchmark problems leads to the conclusion that problems of current standards of difficulty can very well be effectively solved by the learning method. A disadvantage of this reactive scheduling approach is the fact that only non-delay schedules can be produced, which in many cases prohibits finding optimal schedules.	Gabel and Riedmiller (2007),
Makespan	The simulation results show that the principle of the algorithms is simple, their computational quantity is small, and the algorithms can be applied to multi-batch dynamic scheduling with unpredictable entry time due to their favorable potential. It is suitable for multi-objective scheduling problems that need to consider average delay time, delivery cut-off time. The scheduling results of wasp colony algorithms are better than those of static scheduling algorithms for divided scheduling batch by batch.	Cao et al. (2009),

For a supply chain, it is practical for the purchasing company and suppliers to negotiate on the details of products. A series of studies were conducted to automate supplier selection negotiation through the agent-based negotiation models. Cakravastia and Takahashi (2004) proposed a multi-objective model to support the process of supplier selection and negotiation that considers the effect of these decisions on the manufacturing plan. Kim and Cho (2010) used agent negotiation as a way to allocate numerous orders to many participants for supply chain formation. Wong and Fang (2010) presented ECNPro (the Extended Contract-Net- like multilateral Protocol), which is a new multi-agent protocol for handling buyer–seller negotiations in SCM. Huang, Liang, Lai, and Lin (2010) presented a multiple attributes negotiation model for B2C e-commerce, which deploys intelligent agents to facilitate autonomous and automatic on-line buying and selling by intelligent agents while quickly responding to consumers. Wang et al. (2012) established a MAS platform for individual companies to form an ecological virtual enterprise based on ontology theory and intelligent agents. In these agent-based negotiation models, the supplier selection process is abstracted as a buyer–seller relationship, and the MCDM approaches are used in the decision making process of agents to evaluate suppliers

Some of researchers examined the agent based supply chain model. Gao (2008) is purposed an agent-based computational experiment model for four-level reverse logistic network. Zhang et al. (2010) are worked on computational experiments on a heuristic approach with agent negotiation mechanism for the optimization problem of distributed project scheduling in the supply chain network. Long et al. (2011) proposed an agent based simulation which is a multi-agent platform framework with multiple layers for modeling and distributed simulation of complex supply chains. Santa-Eulalia, D'Amours, and Frayret (2012) studied a novel methodological framework called FAMASS (FORAC Architecture for Modeling Agent-based Simulation for Supply chain planning). Li et al. (2012) worked on a single-stage incentive model and a multi-stage incentive model is analyzed by agent-based computational experiments. Meng et al. (2012) presented a controllable and reusable computational experiment model using computational experiment methods and multi-agent technology. Saxena and Jain (2012) also presented a controllable and reusable computational experiment model using computational experiment methods and multi-agent technology. Saxena and Jain (2012) studied computational experiment on an integrated model of dynamic cellular manufacturing and supply chain design with consideration of various issues.

3.3.1. Multi-Agent Manufacturing Systems

In general, manufacturing MAS (Multi Agent System) can accommodate many different agent types; each of them performs specialized functions. In addition to interacting (even partially) with the basic components of their environment, some agents can communicate with each other in order to cooperate or to provide some service or information. For instance, specialized “shop floor watcher” agents are in charge of promptly reacting to unexpected events in the physical world perceived by their sensors and, at the same time, reporting the fact to high-level, knowledge-based agents that elaborate a recovery strategy, ultimately forwarding summary information to a human manager by means of interface agents.

Agent technology also seems to satisfy the fundamental requirements that a modern manufacturing system will need, namely:

- Enterprise integration.
- Distributed organization
- Interoperability.
- Open and dynamic structure; cooperation.
- Integration of humans with software and hardware.
- Agility.
- Scalability.
- Fault tolerance.

Before any attempt can be made to implement agent societies effectively in a manufacturing system, an analysis of the industrial life cycle is pivotal. In fact, this cycle poses restrictions and constraints on the development of an agent-based system that are not present in most research environments

Material, semi-manufactured products, single and standard parts, components etc. are either produced by the company itself or purchased. The operations which are necessary for this have to be provided with resources of manufacturing capacities (manufacturing systems, assembly units, devices etc.) as well as personnel capacities [Shen, 2002].

Figure 3.5 shows the classification and possible functions of the manufacturing management systems in the companies.

Due to technological developments, manufacturing systems has been evolved. However whatever the complexity and uncertainty is main functionalities of manufacturing remains the same but the methods and methodologies to carry out certain manufacturing functionalities improves accordingly that means each manufacturing should and will have design but the wait a design will change in accordance with the product and process characteristics as well as expected functionalities and available technologies.

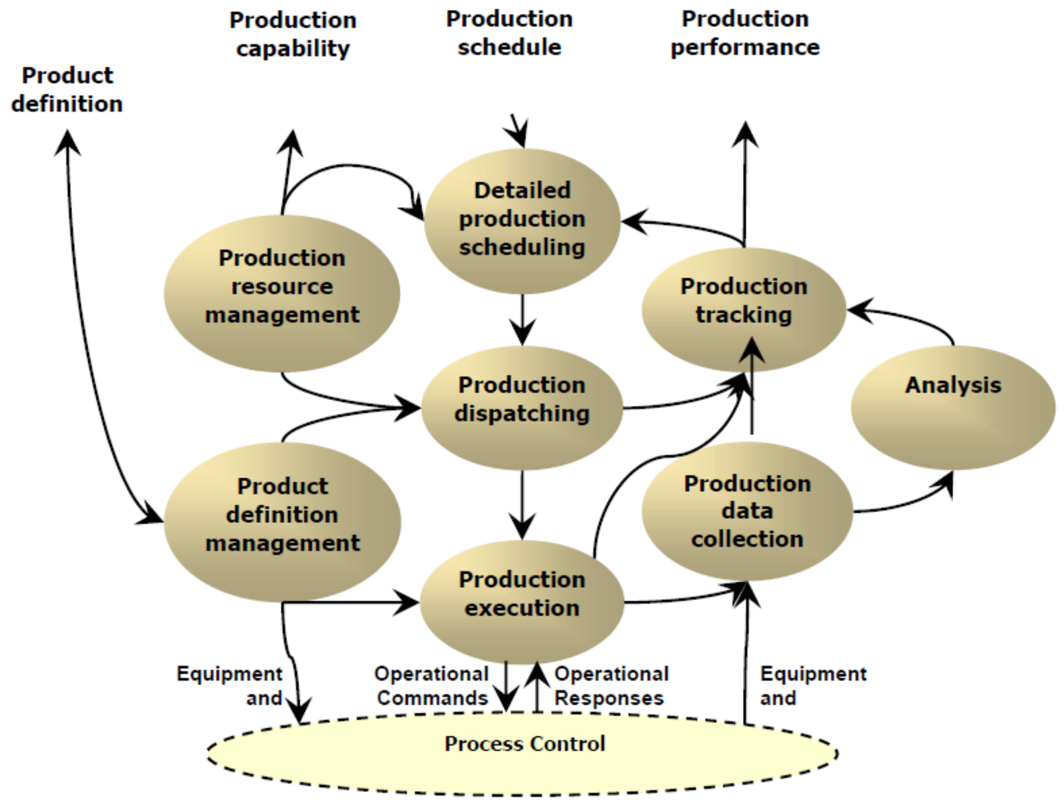

Figure 3.5 A blok diagram of a manufacturing system [Brandl and Owen, 2013]

An observation of a today's manufacturing life-cycle activity shows that it is still difficult to control multi-stage, flexible production processes. Considerations such as assembly, manufacturability, serviceability and dynamic process planning should be incorporated in the early design phases. Ideally, a product designer should be aware of all the technical constraints (materials, processes, tools) and the shop floor resources (routing, scheduling). However, with the very large amount of information involved, a designer often needs the assistance of other domain experts. This process of information exchange can be very time consuming, even if it is done partly concurrently

As a result, there is a strong need for software tools for collecting this kind of knowledge and data and for synthesizing optimal design with balanced overall performance. Such an optimal design could be obtained at runtime by the generation of all feasible designs and the evaluation of their manufacturing and production costs. There is also a need for manufacturing decision support systems having, for example, competences such as planning, temporal projection and hypothetical reasoning. Agent based systems can also be utilized for this purpose which directly creates agile manufacturing information systems.

3.3.2. Standardization of MAS Architectures

Although agent research started more than two decades ago, few efforts have been directed toward a definition of an acceptable MAS architecture. It is possible that one of the reasons for such an absence of consensus might be the common misconception in research circles that MAS architectures and frameworks need to be designed from first principles to match project requirements [Tweedalea et al., 2006]. Although this approach might prove time efficient for individual projects, it certainly creates incompatible systems that are difficult to reuse from project to project. Therefore, one might expect that the widespread adoption of MAS technology will only begin after the formalization and standardization of architectures, mechanisms, and protocols supporting distributed interoperation of agents [Virdhagriswaran et al, 1995].

Recently, several independent industrial and research groups started to pursue the standardization of multi-agent technology such as those of the;

- Object Manager Group (OMG),
- the Foundation for Physical Agents (FIPA),
- the Knowledge-able Agent-oriented System (KAoS) group,
- and the General Magic group
- A Referans Model for Integrated Manufacturing Systems (REMIMS)

These are briefly described below.

3.3.2.1. OMG Model

The OMG group proposes a reference model as a guideline for the development of agent technologies [Virdhagriswaran et al, 1995]. This model outlines the characteristics of an environment composed of agents (i.e., components) and agencies (i.e., places) as entities that collaborate using general patterns and policies of interaction. Under this model, agents are characterized by their capabilities (e.g., inference, planning, and so on), type of interactions (e.g., synchronous, asynchronous), and mobility (e.g., static, movable with or without state). Agencies, on the other hand, support concurrent agent execution, security and agent mobility, among others.

3.3.2.2. FIPA's Model

The Foundation for Intelligent Physical Agents (FIPA) is a multi-disciplinary group pursuing the standardization of agent technology. This organization has made available a series of specifications to direct the development of multi-agent systems.

Of particular importance are their Agent Management [FIPA-a, 1997] and Agent Communication Language [FIPA-b, 1997] specifications. FIPA's approach to MAS development is based on a ``minimal framework for the management of agents in an open environment." This framework is described using a reference model (which specifies the normative environment within which agents exist and operate), and an agent platform (which specifies an infrastructure for the deployment and interaction of agents) [Flores-Mendez, 1999].

3.3.2.3. KAoS Model

Another important standardization effort is pursued by researchers of the Knowledgeable Agent-oriented System [Bradshaw et al., 1997] Architecture. This system, which is also known as KAoS, is described as ``an open distributed architecture for software agents." The KAoS architecture describes agent implementations (starting from the notion of a simple generic agent, to role-oriented agents such as mediators and matchmakers), and elaborates on the interactive dynamics of agent-to-agent messaging communication by using conversation policies [Flores-Mendez, 1999].

3.3.2.4. General Magic Model

General Magic is a commercial endeavor researching mobile agent technology for electronic commerce. Conceptually, this technology models a MAS as an electronic marketplace that lets providers and consumers of goods and services find one another and transact business. This marketplace is modeled as a network of computers supporting a collection of places that offer services to mobile agents.

Mobile agents, which are entities that reside in one particular place at a time, have the following capabilities [White, 1997]:

- They can travel, to move from one place to another
- They can meet other agents, which allows them to call one another agent's procedures
- They can create connections, to allow an agent to communicate with another agent in a different place
- They have authority, which indicates the real-world individual or organization that the agent represents
- They have permits to indicate the capabilities of agents

3.3.2.5. REMIMS Model

REMIMS (A Reference Model for Intelligent Integrated Manufacturing Systems) model is developed to supply necessary information to designer by the using artificial intelligence techniques for integrated manufacturing systems. This model can be used for the development of full automated systems. This model is describes an ideal manufacturing environment. User can design his/her own system depending on own suitable resources and facilities. Main specialty of REMIMS is that the model constructed on multi agent systems and model supports integrated manufacturing systems [Tekez, 2006]

In manufacturing systems developed in accordance with REMIMS standards, every agent is responsible for applying of its duties and transferring required knowledge to the integrated system within a so called Knowledge Network.

The Knowledge Network is a distributed knowledge structure that allows for sharing knowledge of agents with knowledge protocol and knowledge forms. As seen in Figure 3.6, nodes represent main modules; connections represent relations among manufacturing activities and represent double-direction knowledge flow. Also, every module in this Figure 3.6 obtains its inner knowledge flow with the knowledge network.

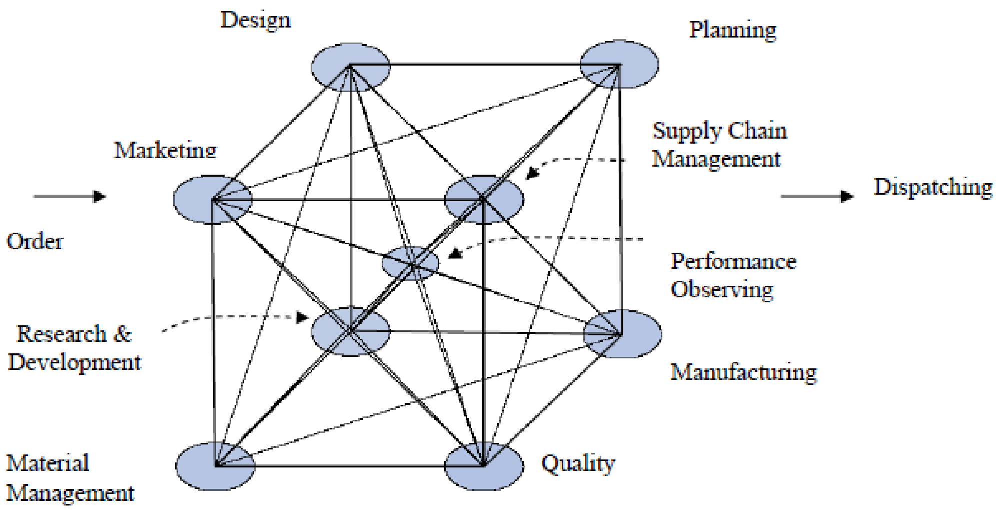

Figure 3.6 A knowledge network of a MAS [Tekez, 2006].

Literature review obviously highlights the possibility of implementing agent based manufacturing systems. There are various advantageous of agent technology to benefit for better manufacturing systems. The literature review clearly indicates that the current approaches are mainly concentrated on using agents for certain manufacturing systems. Possible implementations focus on;

- Traditional agent architectures to be implemented
- Agent communication and concentration
- Agent interactions and integration
- Agent based Simulation
- Agent based supply chain including supplier selection and resource management.
- Agent based process planning
- Agent based real time control mechanism
- Enterprise integration
- Agent negotiation mechanism for the optimization problem
- Standardization agent related activities,

Note that in those applications, the designs of agents are considered to be performed independent of processes and overall manufacturing goals. However, the effectiveness of manufacturing systems is very closely related to achieving overall manufacturing objectives and related process effectiveness which are defined through performance indicators. Creating an integrated goal based agent architecture taking the goals defined, related processes with respective tasks into account is considered to assure remarkable contribution for sustaining manufacturing effectiveness. Due to this fact, such a system design is developed in this study and explained in Chapter 4.

4. PROPOSED MULTI-AGENT BASED MODEL (IMPACT)

Today; business is characterized by an ever increasing pace in production, decreasing product cycle times, and an increasing shift from mass production to mass customization. Companies must be more responsive to market changes and more sensitive to customer needs in order to survive and continue to operate and sustain their business goals. This obviously necessitates systematic monitoring and control over manufacturing systems.

Traditional monitoring and control of manufacturing systems are usually based on deterministic algorithms. However, since the expectations of customer are dynamic in their nature, the manufacturing systems also need to be more responsive to the changing conditions through concurrently balancing and optimizing multiple manufacturing constraints. In order to achieve this, companies must have flexible manufacturing environments to be able to respond technological and service expectations as well

Having being able to control the manufacturing environment strictly requires that the related interdepartmental operation to be executed in optimal condition as a whole. Since, this makes it difficult to control all stages of flexible manufacturing system; there is a need to sustain coordination and cooperation between all manufacturing units. At this point, integration between information and technology becomes the main concern through requiring concrete data collection and analysis in order to achieve the best information flow with a balanced overall performance and optimize respective manufacturing processes.

Similarly, there is a need for machine-to-machine communication systems in order to satisfy the overall control of distributed manufacturing systems with effective and efficient automated decision making systems.

On the other hand, the agent technology is well equipped to perceive environment, reason about it and act accordingly. The computer agents are now proven to be capable of having been able to make better decisions under very complex situations, behaving as flexibly much as possible and preventing misuse of knowledge with excellent communication capabilities.

Based on the explanations given above, the main focus of this thesis is to provide information and a prototype of dynamic, flexible and intelligent manufacturing control model integrating various manufacturing processes at different stages of the planning process. Similarly, a general framework on how to achieve this is provided.

The prototype model proposed is so called IMPACT (Intelligent Manufacturing Planning using Agent-based Control Technology) and has a real-time information flow (available at any time to many sides). Note that the intelligent agent technology is employed for generating and implementing the model. As well known, the agents may work together and share their knowledge related to manufacturing activities. Multi agent systems in manufacturing process provide well-defined integration and automation capability and can ensure several advantages of distributed manufacturing such as;

- They enable restate, scale and adapt their processes.
- New products and services can be marketed to a wide range of potential customers. It provides motivation for innovation and technical specialization.
- They easy swift introduction of new products adjusting the architecture of plant's production system and providing timely response to unforeseen changes in requirements.
- They improve credibility of distributed systems, and sustain unity of shared data.
- They automatically adjust to varying workloads and conditions, harmonizing of a plant's production system architecture.

Based on these advantages, the main objectives of agent based systems can be listed as given below.

- To set-up well-defined integrated manufacturing environment with self-decision making capability and modeling. This certainly increase the efficiency of manufacturing systems using capabilities of agents such as self-organization, pro-activeness and reactiveness with related experience and knowledge as well as inter-agent interaction mechanisms.

- To demonstrate the role of communication and coordination between manufacturing agents with decreasing communication problems between different units;
- To decrease manufacturing span by using optimization methods for scheduling,
- To decrease manufacturing cost caused by purchasing raw materials, operation scheduling, WIP (Work-in-Process), overstock etc.
- To prevent deficiencies of production and operations;
- To increase quality level of both purchased and manufactured products.
- To increase the control on manufacturing, scheduling, supplier selection, machine usage, operations and process management etc.
- To satisfy an ability to measure for the manufacturing goals, processes, operations, machines, products and suppliers for manufacturing
- To prepare a road map for intelligent agent-based manufacturing planning.

To satisfy these advantages, a multi-agent based manufacturing model which is named as IMPACT **(I**ntelligent **M**anufacturing **P**lanning using **A**gent-based **C**ontrol **T**echnology**)** is proposed.

The main objective of this model is to define a general framework for creating a distributed, efficient and effective dynamic multi-agent-based production planning system.

There are countless frameworks that are used to Multi Agent System Architecture. But none of them is designed in such a way that an entire practice from supplier management up to process execution is taken into account.

For the competition of the proposed model, 4 java-based MAS Framework is studied and the differences are listed in Table 1.

Table 4.1. Difference between IMPACT and 4 Java-based Agent Frameworks

Feature	*IMPACT*	*JADE*	*LEAP*	*Agent Factory*	*JACK*
Technology	Java	Java	Java	Java	Java
FIPA Service Components	yes	yes	yes	yes	yes
FIPA Compliance	in preparation	yes	yes	yes	yes
Agent Model	**Goal directed Behavior**	**behavior**	**behavior**	**behavior**	**behavior**
Advance Scheduling	**Yes**	No	No	No	No
Advance Supplier Selection	**Yes**	No	No	No	No
Advance Operations Management	**Yes**	No	No	No	No
Mobility	No	yes	yes	yes	yes

General architecture of the proposed model is outlined in the next section.

4.2. Proposed Model Architecture (IMPACT)

In this section, IMPACT which is a prototype production planning model depends on multi agent system architecture is introduced for the shake of integrated goal satisfaction and an improvement system.

The proposed model is composed of four integrated manufacturing functions. These are Production Planning, Manufacturing, Stock Management, and Procurement Management (see Figure 4.1). In order to satisfy the integration of these manufacturing functions, IMPACT proposes six different but interrelated models including;

- Goal Model,
- Process Model,
- Operation Model,
- Machine Model,
- Product Model,
- Supplier Model.

Each Model is considered to have an agent architecture to act as a distributed function (component) of the manufacturing system. Hierarchical and interrelated model based decomposition approach is used in defining the agents. Although there is a hierarchical relationship between models, all models constructed can work concurrently on an integrated software program. Each model constructed relies upon some business rules. Business rules are intended to assert business structure or to control or influence the behavior of the respective business operations (Bussiness, 2000). Those are the rules that define or constraints several aspects of business and resolves related issues. The rules are intended to assert business structure or to control or influence the behavior of the business for creating the dynamic structure of the agent. Respective performance measurement criteria to assess the effectiveness of each production function have also been defined.

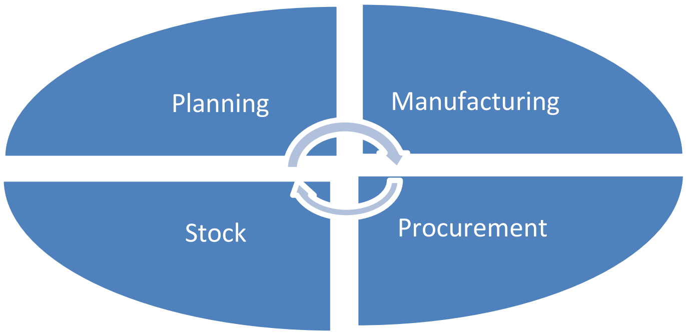

Figure 4.1. Manufacturing functions covered by the proposed Model

IMPACT is structured in a way that the Impact Manager is the main driver of all. It directly updates the related information which includes system dynamics of the manufacturing system, identifies the goals with related processes to run which in turn triggers the respective operations. In the same way, each operation initiates required machines to run in order to produce expected products with the fore mentioned Operations. The model would not be complete if the suppliers are left out of the circle. Therefore the IMPACT is designed to identify and manage the manufacturing goals to be satisfied in such a way that an entire practice from supplier management up to process execution is taken into account.

Each sub-model of IMPACT is represented by an agent where the main architecture of it is illustrated in Figure 4.2. Sub components for operational process execution have been defined under the main agents and business rules.

As shown in Figure 4.2 Intelligent Manufacturing Management System belongs to part of the whole system including User Interface to satisfy the communication 'outlet' between the users and network system, Task Coordinator enables system administrators to update the related information coordinator which includes system dynamics (beliefs, desires, rules, etc.), Scheduler for the assignment of company dynamics to related models, Inference Engine responsible for monitoring the operations carried out within the system using the models of IMPACT to give appropriate action for given task, Knowledge Base is the central data storage system which stores various groups of data for different processes.

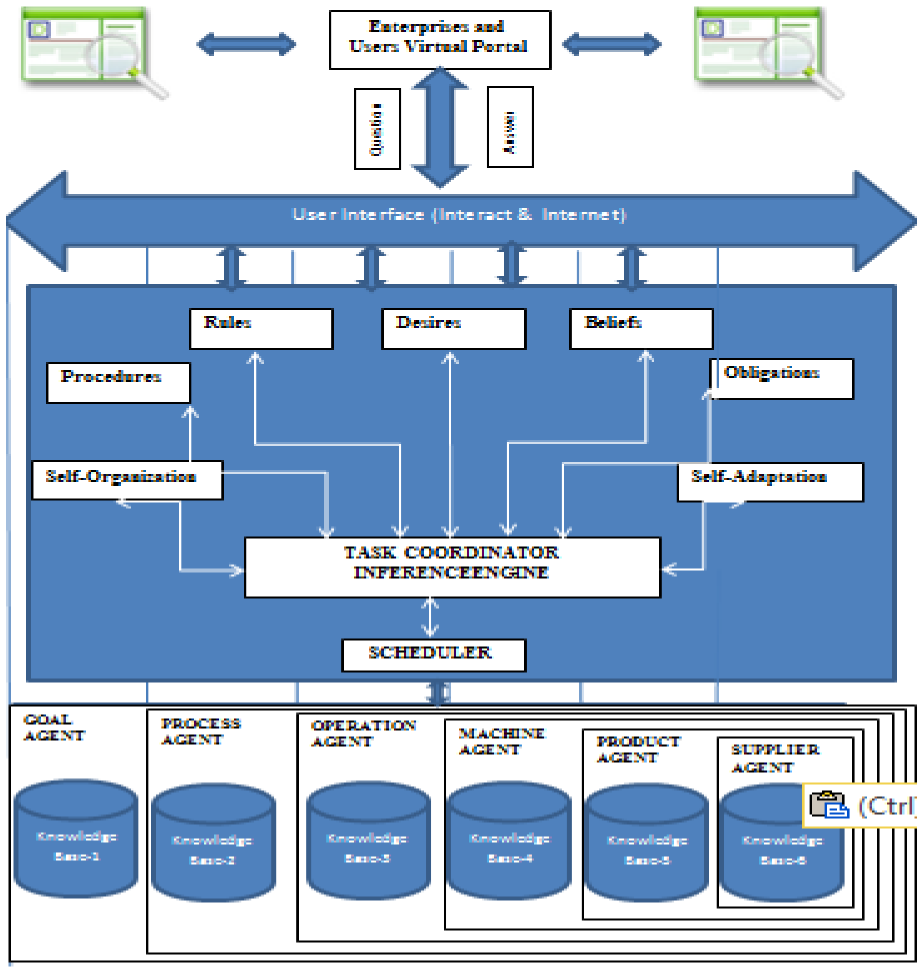

Figure 4.2. Main Architecture of IMPACT and Its Components

Also note that; the IMPACT is designed to create a manufacturing environment in order to operate on a so called ***"success level"*** defined as the following;

- Goal Success; the degree to which the goal is satisfied
- Process Success; the level of performance of implementing the process
- Operation Success; the degree to which the operation is completed on time.
- Machine Success; the level of utilizing a machine for manufacturing as opposed to setup times
- Product Success; the degree to which the product is produced on time.
- And Supplier Success; the degree of selection of appropriate suppliers in terms of cost, quality and payment terms

Proposed model is structured in a manner that each goal is satisfied through execution of a set of processes where each process in turn contributes to the goal success with a significance ratio which is stated in this paper as the "***level of significance***"

Similarly each process is composed a set of operations contributes to the process success with "***operational significance***" when completed.

Similar to those explained above; this approach to define the related success levels is implemented for operations with machines, machines with products and product with suppliers.

Defining the architecture of respective manufacturing agents as well as respective functions and populating their knowledgebase: In this study, a scheduler is proposed to integrate a model-based intelligent design system. The functions of the scheduler in the present implementation of IMPACT consist of 3 main features which are (1) generating model tasks from the functional requirements; (2) satisfying communication of the sub models and (3) representing the interoperability using model agent's features.

IMPACT manager mathematical model parameters are defined as the following.

For ISS (IMPACT system state)

$$ISS = \begin{Bmatrix} 1 \text{ for the system is busy} \\ 0 \text{ for system is available} \end{Bmatrix}$$

Task set (ITS) of the IMPACT

$$\mathrm{ITS} = \{\mathrm{G}_{\mathrm{i,ts}}, \mathrm{Pc}_{\mathrm{j,ts}}, \mathrm{O}_{\mathrm{k,ts}}, \mathrm{M}_{\mathrm{m,ts}}, \mathrm{PR}_{\mathrm{p,ts}}, \mathrm{S}_{\mathrm{q,ts}},\ P_{prc}, A_{act}\ \}$$

Where;

G_i represents the goal i

$G_{i,ts}$ denotes the task ts of goal i

Pc_j presents process j

$Pc_{j,ts}$ shows that task ts of process j

O_k represents the operation k

$O_{k,ts}$ is used the define task ts of operation k

M_m denotes the machine m

$M_{m,ts}$ is shows that task ts of machine m

PR_p defines product p

$PR_{p,ts}$ denotes task ts of the product p

S_q represents the supplier q

S_{qts} denotes task ts of supplier q

P_{prc} represents set of percept of impact manager,

A_{act} denotes action set of impact manager

Goal directed model of the impact is illustrated in Figure 4.3. Main aim of the IMPACT is to minimize differences between planned and actual value of the each model success. Success value calculation of the each model is stated in related model section.

The task coordinator and inference engine is the most innovative feature of the framework. It controls the execution of software agents and it is directly responsible for the performance of the model. The scheduler is responsible from the result of the coordinated activity of 6 models: Goal, Process, Operations, Machine, Product, and Supplier. Scheduler maps each event (each task) to related model.

After receiving the message, the IMPACT manager only need to update the corresponding control input from the message sources. Two type situation are considered which are **synchrony operate the work at any time** and **Asynchrony** allows the work to be performed as soon as applications are ready. This property is the basis of message queuing systems. Channel Message queue (it is the communication infrastructure) infrastructure is used for reliability of communication. Depends on the hierarchical structure of the proposed model, Causal Ordering properties is used which means that; Let Goal Agent, Process Agent and Operations Agents are three agents in our model. Goal Agent sends task event ts_1 to Process agent and Operations Agent.

Process Agent reacts to task event by sending new task event ts_2 to Operations agent. This approach ensures that Operations agents receive task event ts_1 before task event ts_2.

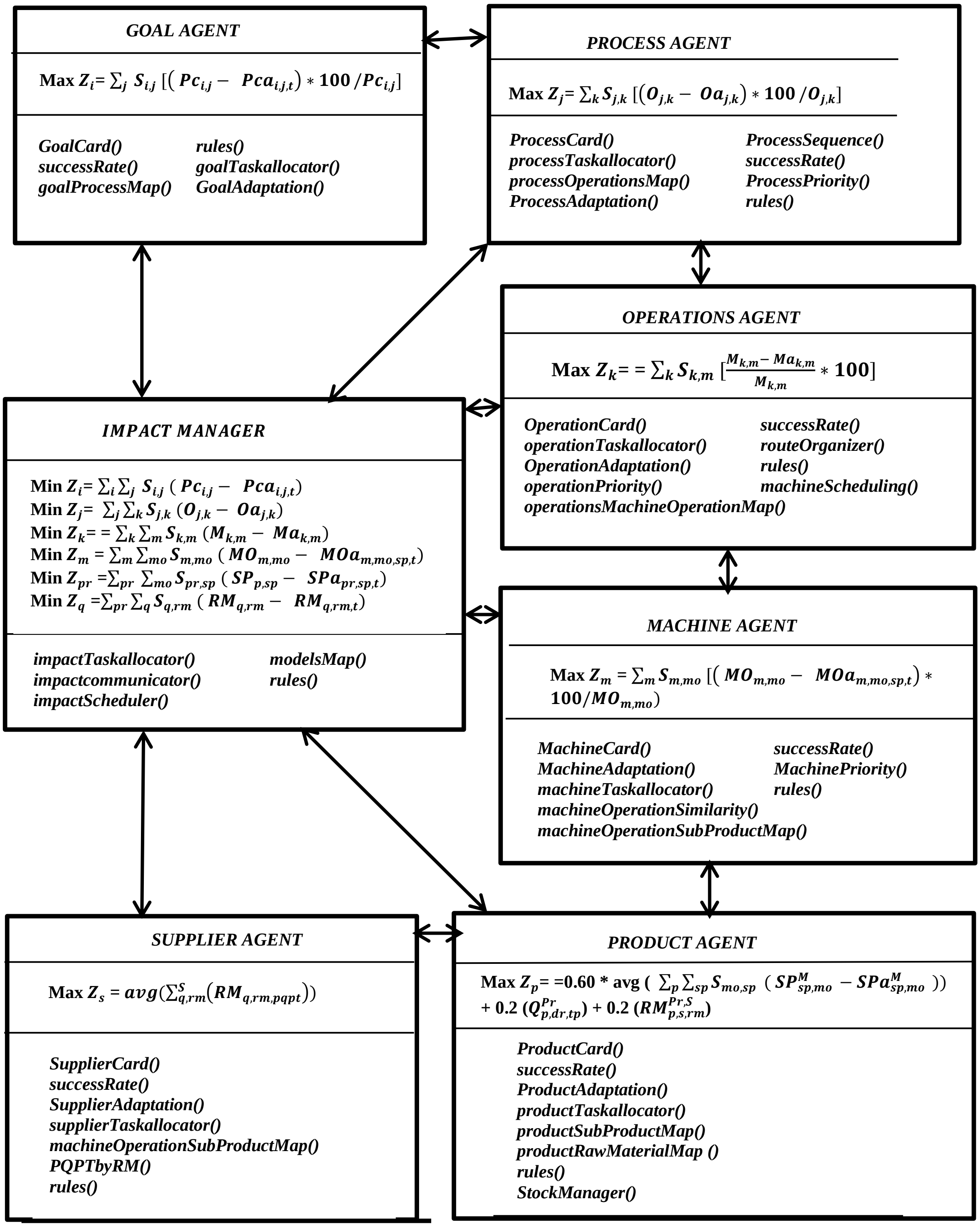

Figure 4.3. Goal directed agent architecture of IMPACT

Perception capabilities of IMPACT manager are implemented by class Perceptor. A meta-object of this class is notified with **messagePerceived** when a message is received by one of its perceived objects. By default, that method notifies to Scheduler object about every perception to satisfy related action from respective sub-models. When a messagePerceived situation occurs, Scheduler can react quickly or analyze carefully by taking into knowledge and goals depends on predefined six sub models.

An agent composed by a Communicator object is able to interact with other agents. This class only defines an abstract interface for communication, so programmers should extend it according to their requirements.

Sharing a common syntax is a major part of Agent based systems. In this study semantics for Knowledge Query Manipulation Language (KQML) is used as a language and a protocol to support communication between software agents. The class ImpactCommunicator is able to send KQML messages and XML is used for ACL (Agent communication Language) representation (as explained later).

In this section, main architecture of the IMPACT is discussed. Structure and components of six sub models of the proposed model will be introduced respectively.

4.2.1. The Goal Model (Goal Agent)

This model is operated by a Goal Agent enabling the measurement of each Goal through handling and performing respective processes. Goal Agent identifies the goals with related processes to run which in turn triggers the respective operations.
In this study, 4 different goals are defined in the domain which is listed below;

G1: Control of manufacturing activities
G2: Increasing manufacturing performance
G3: Decreasing manufacturing idle Times
G4: Increasing manufacturing quality

Goal – Process Relations for the 4 predefined Goals and respective performance measurement KPI (Key Performance Indicator) techniques See Appendix 1

Main motivations behind generating the Goal Agent are listed below;

1. ability to **integrate** (Specific Purpose - integrating the goals to the related processes)
2. ability to **measure** (Specific Purpose - measurement of each Goal through handling and performing respective processes)
3. ability to **adapt** (Social Ability - a**bility** to **adapt** over time in response to changes in the environment)
4. ability to **self-organize** (Reactiveness - ability to establish basic rules for the interactions between the models of the system)
5. ability to **control** (Autonomy - ability to control of manufacturing goals)

Ability to **integrate:** "Goal Card" is created to identify predefined set of goals of a certain manufacturing system and support the assessment with respect to whether those are satisfied or not.

An example of Goal Card is shown in Table 4.2. The main aim of this card is to highlight key performance indicators with respect to the goal definition as well as to indicate the respective processes together. Each goal card is defined in terms of the following attributes.

Goal Name**:** Short Cut Explanation of Goal

Card Code**:** Each goal has to be a unique card code to accessibility

Goal Definition**:** Detail information about the goal of the Company

Related processes' codes and names**:** To Identify the relations of the Goal and respective Processes

Related goals' codes and names To Identify the relations of the Goal and respective Goals

Planned Process Success Value**:** Target value of the related Goals and Processes of the new Goal success

Actual Process Success Value: This value comes of completed process and Goal measurement which is total value of determined key performance ındicators (KPI) in other model cards

***% Level of significance*:** The contribution of the related Goals and Processes of the new Goal success

***% Success Level*:** To calculate the new goal success level by making calculation of (Planned Process Success value-Actual Process Success Value)* % Level of significance

Table 4.2. The structure of the "Goal Card"

GOAL CARD						
Goal Name:					No:	00001-G1
Card Code:					Date:	
Goal Definition:						
No	Related Process Code	Process Name	Process Success Value	Actual Process Success Value	% Level of significance	% Success Level
Success value comes from the related processes:						
No	Related Goal Code	Goal Name	Planned Goal Success Value	Actual Goal Success Value	% Level of significance	% Success Level
Success value comes from the related goals:						
Total Success						

***Success value comes from the related process*:** To calculate success Level in terms of processes of the new Goal.

***Success value comes from the related goals*:** This space is to calculate success Level in terms of based on goals of the new Goal

Total Success: This space is to measures the new goal success considering both related processes and goals.

Goal Agent stores the main goals of the manufacturing company by using Goal cards within Database 1 (DB1). A different goal card is designed for each goal. For aiding goal-reaching in production processes are given under the Process Name heading as illustrated in Table 4.2. Specialists determine aimed performance of the processes and these are included under the heading “Planned process success value”. The concerned process cards provide success rates of the completed processes. For reaching manufacturing goal each process is listed as percentages under the heading Level of Significance (total process effects should be 100%).

Structural frame code of Goal card definition function is as follows;

```
public class Goal {
	public void addGoalCard(String name, String code, String definitions,   List<ProcessCard> 
processes,
				List<GoalCard> goals, String activationRule, HashMap<ProcessCard, Float> 
processLOS,
				HashMap<GoalCard, Float> goalLOS) {
			GoalCard goalcard = new GoalCard(name, code, definitions, processes, goals, 
activationRule, processLOS, goalLOS);
			this.saveGoalCard(goalcard);
	}
	private void saveGoalCard(GoalCard goalcard) {

	}
}
public class GoalCard {
	public String Name;
	public String Code;
	public String Definitions;
	public List<ProcessCard> Processes;
	public HashMap<ProcessCard, Float> ProcessLOS;
	public List<GoalCard> Goals;
	public HashMap<GoalCard, Float> GoalLOS;
	public String ActivationRule;
	public GoalCard(String name, String code, String definitions, List<ProcessCard> processes, 
List<GoalCard> goals,
				String activationRule, HashMap<ProcessCard, Float> processLOS, 
HashMap<GoalCard, Float> goalLOS) {
			for (Process process : processes) {
				if (process.notExist()) {
					process.addProcessCard();
				}
			}
	}

}
```

A significant part of goal definition function has been carried out on process definition function.

Ability to measure; In order to provide Goal model to have measurement ability, primarily, Goal Agent structure was developed. The business rules were defined so as to provide it to have self-organizing property .The mathematical model, which will measure the value of goal directed success of the model, was defined. Afterwards, it was provided to have measurement ability on the software environment through coding these business rules to the model. Now, these studies will be explained respectively.

Goal Model Agent Architecture: The Goal model is composed of goal definition, goal assessment, goal processing, performance management as well as the respective knowledge base. The “Goal Scheduler” is in the heart of the model. It works in two ways. One is to report the user about the level of goal satisfaction. The second is to respond internally to other modules request. Agent architecture for the proposed Goal model is given in Figure 4.4; as seen in the figure, the double-headed arrow in the figure indicates two-way communication between various agents.

When an input reaches by impact manager (calculate success level of G1) to Goal Agent, Goal Task Manager reaches the relevant goal card by means of DB1 (database belonging to Goal Model). It receives DB2 (Database belonging to the Process Model) over the information system by reaching current success rates of the processes that are to be used for measuring the success of the goal by reading the relevant Goal Card. Measurements defined on the goal card reflects the result of the measurements.

Tasks are defined by a function as $\mathrm{GTS} \rightarrow \mathrm{t}$. After results are received, Scheduler changes the system state. This operation continues until executions of all tasks are completed and no more tasks are available to run.

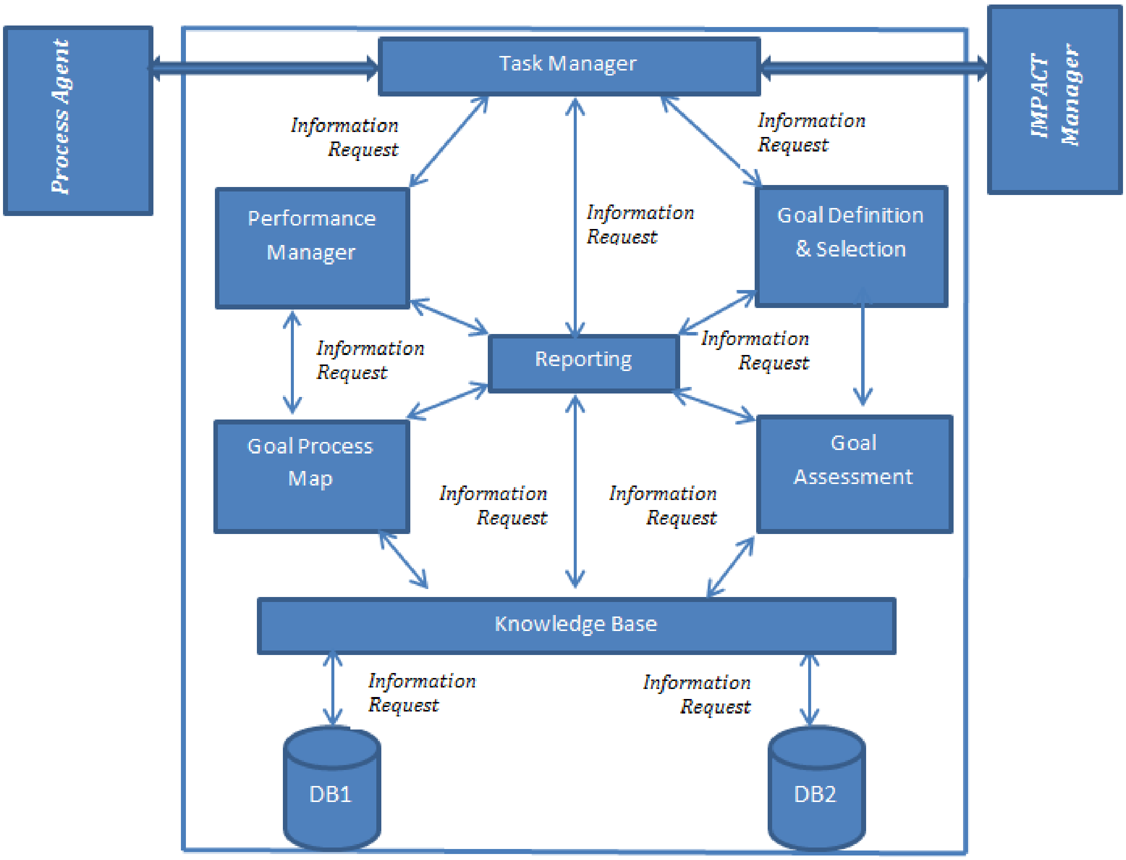

Figure 4.4. Goal Model Agent Architecture,

Definition of the Goal Mathematical Model; Goal model parameters are defined as the following;

$f(GA) = \{Pt_i^G, A_i^G, \mathrm{f}_{\mathrm{i,ts,t}}^{\mathrm{GD}}, \mathrm{f}_{\mathrm{i,ts,t}}^{\mathrm{GPm}}, \mathrm{f}_{\mathrm{i,ts,t}}^{\mathrm{G}}\ \mathrm{SSG_i}, \in_{\mathrm{i}}\}$

GA represents the function set of the Goal Model.

Pt_i^G represents set of percept of goal i,

A_i^G set of actions of goal i,

$\mathrm{f}_{\mathrm{i,ts,t}}^{\mathrm{GD}}$ denotes the Goal card definition function task ts of the Goal i at time t,

$\mathrm{f}_{\mathrm{i,ts,t}}^{\mathrm{GPm}}$ is the performance measurement function for task ts of the Goal i at time t,

$\mathrm{f}_{\mathrm{i,ts,ad}}^{\mathrm{G}}$ denotes the adaptation function ad for task ts of Goal i.

$\mathrm{SSG_i}$ denotes the system states of goals and

$\in_{\mathrm{i}}$ defines set of environmental states of goals

Mathematical model of Goal performance measurement function is given in Equation 1.

$$\textbf{Max } \boldsymbol{Z_i} = \sum_j \boldsymbol{S_{i,j}}\left[\left(\boldsymbol{Pc_{i,j}} - \boldsymbol{Pca_{i,j,t}}\right) * \mathbf{100} / \boldsymbol{Pc_{i,j}}\right] \qquad \textbf{(1)}$$

According to Equation 1, $\boldsymbol{Z_i}$ is the total success value of the Goal i. $\boldsymbol{S_{i,j}}$ is level of the significance value of process j in goal i (This parameter is designed as an expert decision) , $\boldsymbol{Pc_{i,j}}$ is the planned process success rate in the goal i, $\boldsymbol{Pca_{i,j,t}}$ actual process success rate at time t.

$$\sum_j S_{i,j} = 100 \tag{2}$$

According to Equation 2, sum of level of significance values must be equal to 100

$$Pc_{i,j} = max\left(Pc_{i,j,tp}\right) + \beta_{Pc_{i,j}} \tag{3}$$

According to Equation 3, attributes of the planned process value are defined. Where $\boldsymbol{max\left(Pc_{i,j,tp}\right)}$ maximum success rate of the process j in goal i at time period tp. $\boldsymbol{\beta_{Pc_{i,j}}}$parameter gives dynamic nature to the model. This value can be between 0 and 100$\boldsymbol{-max\left(Pc_{i,j,tp}\right).}$ Based on the experience this value is taken as 5 during the running process

$$Pca_{i,j,t} = \sum_k S_{j,k}\left(O_{j,k} - Oac_{j,k,t} \right) \tag{4}$$

According to Equation 4, calculation of the actual process time is defined. Where $\boldsymbol{S_{j,k}}$ is the level of the significance value of operation k in process j. $\boldsymbol{O_{j,k}}$ is the planned success rate of operation k in process j and $\boldsymbol{Oac_{j,k,t}}$ actual operation success rate of operation k in process j at time t.

Structural frame code of Goal success measurement function; Structural frame code of Goal measurement function $f^{GPm}_{i,ts,t}$ is designed as follows;

```
public class GoalCard {
	public float successRate() {
		float rate = 0;
		for (ProcessCard pc : this.Processes) {
			rate += ProcessLOS.get(pc) * (plannedValue - pc.successRate());
		}
		for (GoalCard gc : Goals) {
			rate += GoalLOS.get(gc) * (plannedValue - gc.successRate());
		}
		return rate;
	}
}
```

This function gives an ability to calculate each goal success level depends on related processes.

Ability to adapt: Ability to adapt is one of the most important features of agent based systems. In this study, "Inductive learning technique" is utilized for information retrieval and data retrieval.

For the classification of the data, Let, S is the instance (Goal, process, operation, etc.), each of these instance are pre-classified into different classes by n attributes At_1, …At_n. For any attribute At_j value is defined in a domain such as;

$At_j = \{D_{ju} | U = 1, \ldots, U\}$ and D_{ju} is the string or numeric value

[x, y] is the pre-defined interval range for the numeric values and { "a", "b" } is the pre-defined string set.

$At_j \;\; x \leq D_{ju} \leq y$ is identified for numerical attributes

Atj = a or Atj = b … etc is identified for the string values.

An example pre-defined adaptation rule for Goal G2 is as follows;

```
//G2 Adaptaion rule
public String G2_Adaptation =
"time|setup|schedule|manufacturing|efficiency|planning";
    public void checkAdaptability(String processName) {
    if(Pattern.compile(G2_Adaptation).matcher(processName).find()){
   //Save Process Card under G2 Goal Card
  }
```

If a new Process Card is created in the Model, Goal Model creates a new relation between new Process and Pre-defined Goals. The example adaptation rule given works as follows: If any of these words of "time", “setup”, "schedule”, “manufacturing”, "efficiency” and “planning” take place in the name of the newly created processes card, the goal agent establishes a relationship between the newly created process and G2 goal. G2 goal inserts the newly created process into the card.

Ability to self-organize: Proposed Model also considered three type business rules. First type is for defining the manufacturing objects, second type of rules are created to constraints the manufacturing system and the third is defined in order to create decision alternatives for manufacturing operations. Each agent reacts to the message created by executing a business rule. For the interaction KQML message standard is used and XML is selected for Agent communication Language. One of the applied XML structure is illustrated for the Goal measurement task event of Goal G1 (see Figure 4.5)


```
<envelop>
        <from>
         G
        </from>
        <to>
        P
        </to>
        <Processes>
                <Process>
                 P1
                </Process>
                <Process>
                 P5
                </Process>
                <Process>
                 P6
                </Process>
                <Process>
                 P8
                </Process>
                <Process>
                 P9
                <Process>
                 P16
                </Process>
        </Processes>
        <requestType>
         successValue
        </requestType>
</envelop>
```


Figure 4.5. An example Request G→ P (from Goal Agent to Process Agent)

Some of pre-defined business rules listed as below;

1. It gathers Company goals information from Goal Cards
2. The goal card should identify the relations of the Goal and respective Processes
3. It gathers process times from Process Agent
4. It measures the new goal success level by taking difference;
5. It controls completion of manufacturing process and gives an approval for manufacturing.
6. It satisfies observation and intervention of the decrease in performance of Processes.

The control interaction provides the **ability to control** the progress of the agent while executing a task.

An abstract data flow scheme belonging to the Goal Agent information flow is shown in Figure 4.6.

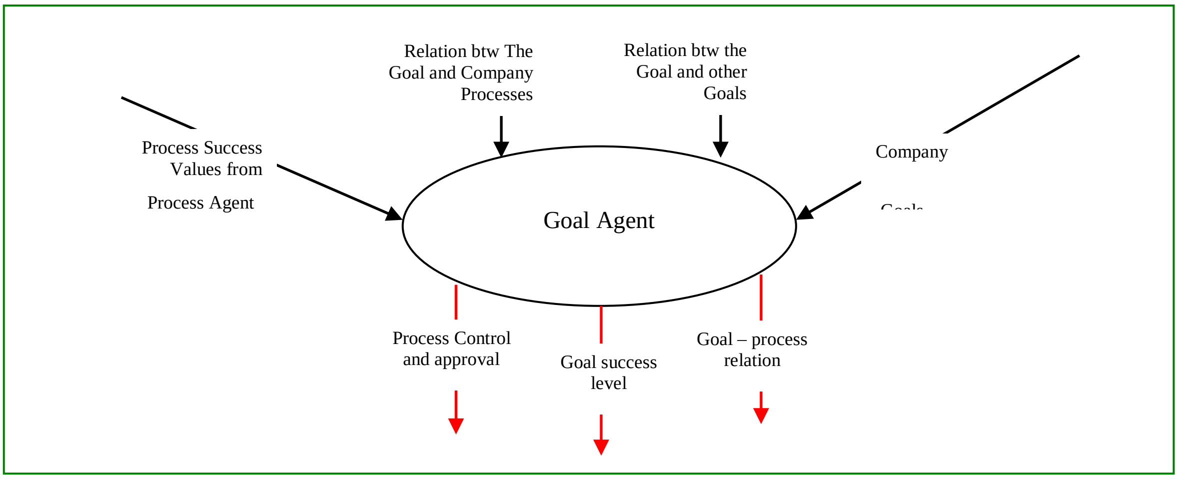

Figure 4.6. Information flow of the Goal Model

4.2.2. Process Model (Process Agent)

This model is operated by the Process Agent enabling the measurement of each process through handling and performing respective operations. The primary task of this model is to identify predefined processes of a manufacturing system to generate assessment with respect to whether those are satisfied or not

In this study, 17 different processes are pre-defined as follows;

P1: Planning Process

P2: National Logistic Process

P3: International Logistic Process

P4: Dispatching-Shipping Process

P5: Storage Management Process

P6: Purchasing Process

P7: Supplier Selection-Management Process

P8: Mechanic Manufacturing Process

P9: Assembly Manufacturing Process

P10: Chemistry Lab Process

P11: Maintenance and Repair Process

P12: Product development Process

P13: Patent Process

P14: Engineering Process

P15: Quality Control Process

P16: Sales and Marketing Process

P17: Training Process

(For the Process – Operation Relations for the 17 predefined Processes and respective performance measurement KPI (Key Performance Indicator) See Appendix 2)

Main motivations of the Process Model are listed below;

1. ability to **integrate** (Specific Purpose - integrating the processes to the related operations)
2. ability to **measure** (Specific Purpose - measurement of each process through handling and performing respective operations)

3. ability to **adapt** (Social Ability - a**bility** to **adapt** over time in response to changes in the environment)
4. ability to **self-organize** (Reactiveness - ability to establish basic rules for the interactions between the process model and other models)
5. ability to **control** (Autonomy - ability to control of manufacturing processes)

Ability to **integrate:** Similar to Goal Model, the process model is also represented by a so called *"Process Card"* identifying the respective operations as well as related business rules. Note that the Process Card is also defined separately for each process. An example of a Process Card is shown in Table 4.3.

Each Process is defined in terms of the following attributes;

Process Name: Short Cut Explanation of a process

Card Code: Each Process has to be a unique card code to accessibility

Process Definition: Detail information about the Process

Process Sequence Level: It defines a sequence the new Process (Each process)

Related Operations' codes and names To Identify the relations of the new Process and respective operations

Related Processes' codes and names: To Identify the relations of the new Process and other Processes

Operation Parameters

- ***Activation Rule Code :*** is used to define activation rule of the operation
- ***Operation Activation Rule Explanation :*** Detail information about the activation rule

Operation M & E Parameter / TMTH

- ***KPI Code:*** It defines key performance indicator of the operation (for the 17 pre-defined processes see Appendix 3)

- ***Planned KPI Value :*** Target value of the related operation
- ***Actual KPI Value:*** This value comes from the completed operations and processes
- **% Level of significance:** The contribution of the related Operation and other related Processes of the new Process
- **% Success Level:** To calculate the new Process success level by making calculation of (Planned KPI value-Actual KPI Value)* % Level of significance

Table 4.3. The Structure of “Process Card”

<table>
<tr><td colspan="9">PROCESS CARD</td></tr>
<tr><td colspan="2">Process Name:</td><td colspan="3"></td><td>No:</td><td colspan="3"></td></tr>
<tr><td colspan="2">Card Code:</td><td colspan="3"></td><td>Date:</td><td colspan="3"></td></tr>
<tr><td>Process Definition:</td><td colspan="3"></td><td colspan="3">Process Sequence Level</td><td colspan="2"></td></tr>
<tr><td rowspan="2">Related Operation Code</td><td rowspan="2">Operation Name</td><td colspan="2">Operation Parameters</td><td colspan="3">Operation M & E Parameter / TMTH</td><td rowspan="2">% Level of significance</td><td rowspan="2">% Success Value</td></tr>
<tr><td>Operation Activation Rule Code</td><td>Operation Activation Rule Explanation</td><td>KPI Code</td><td>Planned KPI Value</td><td>Actual KPI Value</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="3">Success value of P</td><td></td><td colspan="3">% Level of significance</td><td colspan="2"></td></tr>
<tr><td>Related Processes Code</td><td colspan="5">Process Name</td><td>Actual KPI Value</td><td>% Level of significance</td><td>% Success Value</td></tr>
<tr><td></td><td colspan="5"></td><td></td><td></td><td></td></tr>
<tr><td></td><td colspan="5"></td><td></td><td></td><td></td></tr>
<tr><td colspan="3">Success value comes from the related processes:</td><td></td><td colspan="3">% Level of significance</td><td colspan="2"></td></tr>
<tr><td colspan="2">Total Process Success</td><td colspan="6"></td><td></td></tr>
</table>

Total Success value comes from the related process: To calculate success level depends on related operations.

Success value comes from the related process: To calculate success level depends on related processes.

Total Success: This space is to measures the new Process success considering both related processes and operations.

Process Card is made of the first main field where fundamental information including Process name and card number are contained, and the second field where including detailed information about operations to enable defining the process successfully, and the third main field enabling inclusion of success level of other processes (if there are any). For measuring success of the process, this approach has enabled inclusion of both success ratio of the operations and success ratios of the relevant operations in the estimation. Effect of the second and third main field is divided as percentage and added at the end of both fields.

In the first field, fundamental information about the process is given and Process Sequence Level, where priority level of the process is defined, has been formed. This field is for checking whether the process before it is completed or not for starting the Process when activation is achieved reaching the Process Model.

In the second field; the columns designated for code and names of the operations related to the first column and the column where activation of the type of input reaching the system in its continuation is defined and activation rules are defined, and the code belonging to KPI (key performance indicator) where performance measurement criteria are defined for usage in the operation performance which is the most important field, and measurement field where company planning value and current value will be contained is added. In the second to last column, effect of each operation process on the Level of Significance (so that the entire operation effects will be 100%) will be included and % success value column is added where percentage of success will be recorded for the measurement results of the operations. Right under these columns, success value of P line is added enabling success of the process operationally and average of all operations included within is estimated.

Success percentages of the processes related with the third field are obtained from the process cards of the relevant processes by means of the information system. Effect of each process is defined so that their sum will be 100% under the heading of Level of significance. Average of the emergent success levels gives the success ratio of the effects of the processes.

After the above given estimations are completed, total success rate of the Process can now be calculated. Total success level is found by adding multiplication of average success ratios and Level of Significance values, which define the effect of the second and third main field designed at the beginning after this point.

The Structural frame code of the Process Card definition function is given below;

```
public class Process {
	public void addProcessCard(String name, String code, String definition, List<ProcessCard>
processes,
				HashMap<ProcessCard, Float> processLOS, List<OperationCard> operations,
				HashMap<OperationCard, Float> operationLOS) {
		ProcessCard processcard = new ProcessCard(name, code, definition, processes,
processLOS, operations,
					operationLOS);
		this.saveProcessCard(processcard);
	}

	private void saveProcessCard(ProcessCard processcard) {

	}
}

public class ProcessCard {
	public String Name;
	public String Code;
	public String Definition;
	public List<ProcessCard> Processes;
	public List<OperationCard> Operations;
	public HashMap<ProcessCard, Float> ProcessLOS;
	public HashMap<OperationCard, Float> OperationLOS;

	public ProcessCard(String name, String code, String definition, List<ProcessCard> processes,
				HashMap<ProcessCard, Float> processLOS, List<OperationCard> operations,
				HashMap<OperationCard, Float> operationLOS) {
	}
}
```

Ability to measure:

Similarly to the Goal model, in order to provide the Process model to have measurement ability, primarily, the agent structure of the process model was developed. The business rules were defined so as to provide it to have self-organize property. The mathematical model was defined to measure the value of the success rate of the processes. Afterwards, it was provided to have measurement ability on the software environment through coding these business rules to the model. Now, these studies will be explained respectively.

Process Model Agent Architecture: As shown in Figure 4.6, to enable the relationship of the targets, which are defined within the Goal Model, and the processes, and to enable their relationship with the operations included within for measuring the performances of the processes, and while doing all of these, for protecting the manufacturing specifications the process model is related with these three separate models. R&D (Research and Development) data in this study scope are accepted as recorded ready in the system.

Agent architecture of the proposed Process model is given in Figure 4.7;

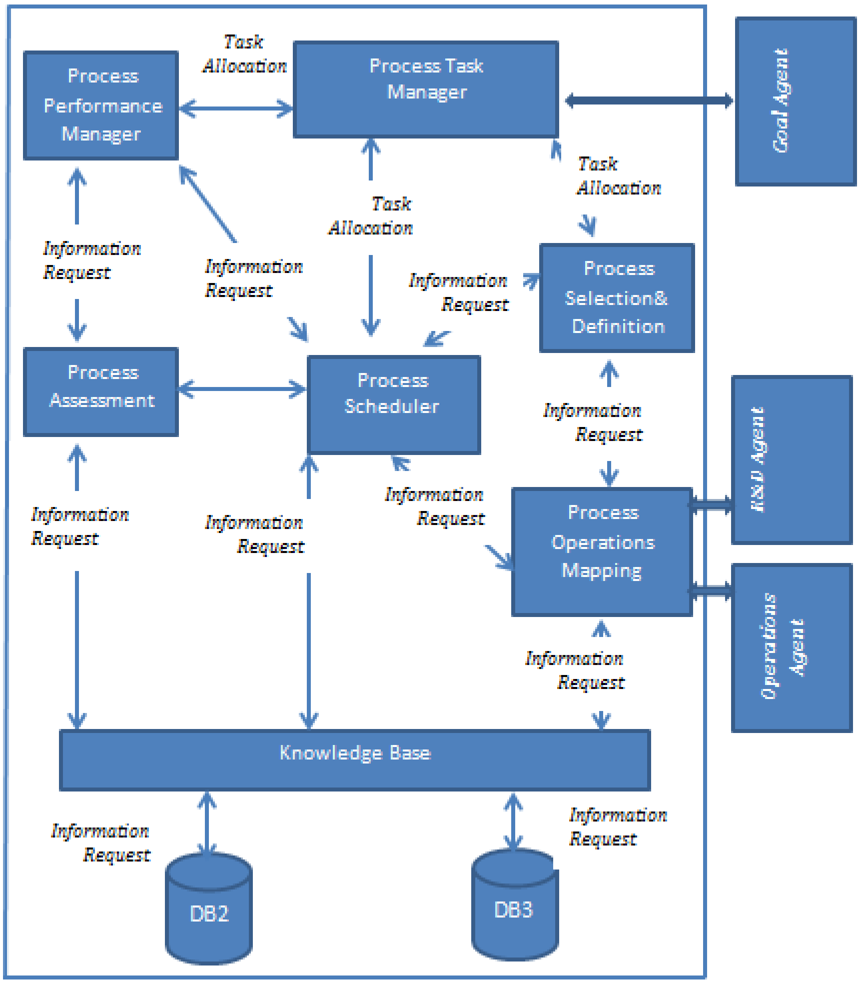

Figure 4.7. Process Model Agent Architecture

Process Task Manager is responsible to task allocation in the Process Model. The process performance manager is responsible for the calculation of the performances of the processes. He performs this work through 3 ways;

1. When any success rate is wanted through goal agent to any process,
2. In performing any operation in the process again
3. When average success level task has been taken for any process that had been received from impact manager. (In this case, it calculates the average success level for the time period specified through process assessment).

Process selection and definition is activated in two ways: These are as follows;

1. A newly created operation card is associated directly with any process by the system. (In this case, the values of the level of significance will be updated again)

2. When a new card creation has been requested by impact manager (in this case, a new process card is created through Process Operations Mapping.) DB1 was created for the process' records and DB2 was created for maintaining the records of the operations included in the processes.

Definition of the Process Mathematical Model is explained below

$\mathrm{f(PA)} = \left\{Pt_j^{Pc}, A_j^{Pc}, \mathrm{f}_{\mathrm{j,ts,t}}^{\mathrm{PcD}}, \mathrm{f}_{\mathrm{j,ts,t}}^{\mathrm{PcP}}, \mathrm{f}_{\mathrm{j,ts,ad}}^{\mathrm{Pc}} \mathrm{SSPc_j}\ , \in_j^{Pc}\right\}$ denotes the function set of the Process Model

Where;

- Pt_j^{Pc} set of percept of Process j,
- A_j^{Pc} set of actions of Process j,
- $\mathrm{f}_{\mathrm{j,ts,t}}^{\mathrm{PcD}}$ denotes the Process card definition function task of process j at time t,
- $\mathrm{f}_{\mathrm{j,ts,t}}^{\mathrm{PcP}}$ is performance measurement function task of process j at time t,
- $\mathrm{f}_{\mathrm{j,ts,ad}}^{\mathrm{Pc}}$ denotes the adaptation function ad for task ts of process j.
- $\mathrm{SSPc_j}$ denotes the system states of Process Agent and
- $\in_j^{Pc}$defines set of environmental states of processes

Mathematical model of a process success rate function is given in Equation 5.

Max $Z_j = \sum_k S_{j,k} \left[\left(O_{j,k} - Oa_{j,k}\right) * 100 / O_{j,k}\right]$ **(5)**

According to Equation 5, Z_j is the total success value of the process j. $S_{j,k}$ is level of the significance value of operation k in process j (This parameter is designed as an expert decision) , $O_{j,k}$is the planned operation value of operation k in the process j, $Oa_{j,k}$ actual operation value at time t.

$\sum_k S_{j,k} = 100$ **(6)**

According to Equation 6, Sum of level of significance values must be equal to 100 in each process card.

$O_{j,k} = max\left(O_{j,k,tp}\right) + \beta_{O_{j,k}}$ **(7)**

According to Equation 7, attributes of the planned process value are defined. $max\left(O_{j,k,tp}\right)$ maximum success rate of the operation k in process j at time period tp. This $\beta_{O_{j,k}}$ parameter gives dynamic nature to the model. This value can be between 0 and $100 - max\left(O_{j,k,tp}\right)$. Based on the experience this value is taken as 5 during the running process

$Oa_{j,k} = \sum_k S_{k,m}\left(M_{k,m} - Mac_{k,m,t}\right)$ **(8)**

According to Equation 8, calculation of the actual operation value is defined. Where $S_{k,m}$ is the level of the significance value of machine m in operation k. $M_{k,m}$ is the planned success rate of machine m in operation k and $Mac_{k,m,t}$ actual machine success rate of machine m in operation k at time t.

Structural frame code of Process success measurement; Structural frame code of Process measurement function $f_{i,ts,t}^{PPm}$ is designed as follows;

```
public class Process {
        public float successRate() {
                float rate = 0;
                for (OperationCard oc : Operations) {
                        rate += OperationLOS.get(oc) * (plannedValue - oc.successRate());
                }
                for (ProcessCard pc : Processes) {
                        rate += ProcessLOS.get(pc) * (plannedValue - pc.successRate());
                }
                return rate;
         }
}
```

Ability to adapt: An example adaptation rule of the process agent is pre-defined as follows;

Structural frame code of the considered adaptation rule is as follows;

```
public void onOperationRemoved(OperationCard operationCard) {
      // Iterate Saved Process cards
         // Remove operationCard if exists
     for (ProcessCard pc : ProcessCardList) {
         if (pc.Operations.contains(operationCard)) {
            pc.Operations.remove(operationCard);
            pc.ProcessLOS.remove(operationCard);
         }
      }
}
```

This rule is as follows; When any operation included in the system has been deleted, it deletes this operation from the processes, in which this operation has been included.)It calculates $\boldsymbol{S_{j,k}}$ values again for the process cards, at which deletion process has been performed.

Ability to Self-organize: Some of pre-determined business rules for self-organization of the model are listed below;

- It gathers Company Process information from Process Cards
- It gathers operation success levels from Operation Cards
- It gives an ability of measure the completion of manufacturing operations and gives an approval for manufacturing.
- It satisfy the new Process integration to related operations
- It satisfy the new Process integration to other processes
- It defines critical operation in the new Process(maximum completion time)
- It satisfies observation and intervention of the decrease in performance of new Processes.
- It gives the sequence the new Process

Abstract Information flow of the Process model is also figure out in Figure 4.8;

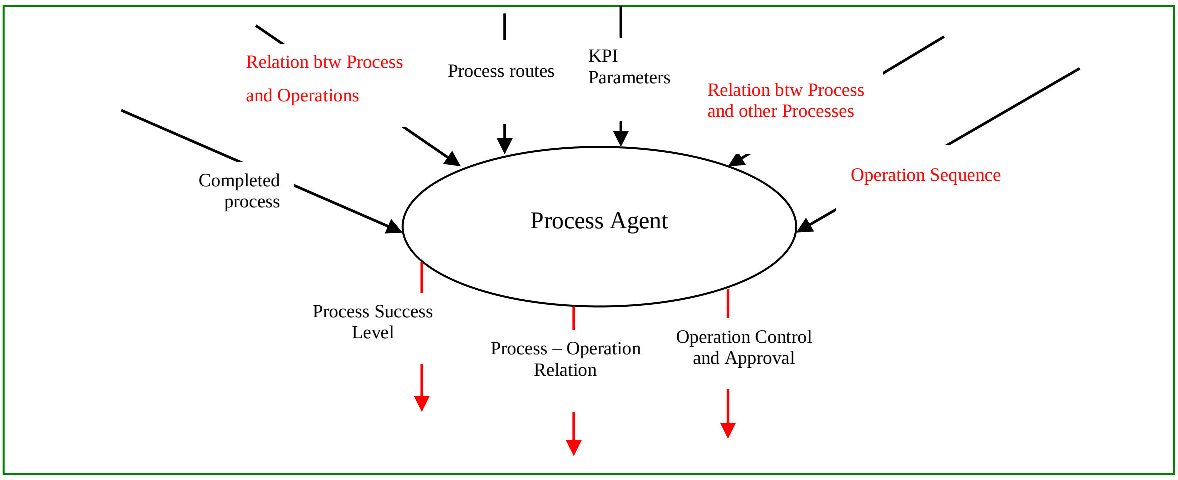

Figure 4.8. Process Model Information Flow

4.2.3. Operations Model (Operations Agent)

Primary task of this model is to identify well-structured definition of production operations for the best assignment of the operations to appropriate machine. To accomplish this some set of operations must be determined. Operation Card must be generated with each operation and machine relations.

Main motivations of the Operations Model are listed below;

1. ability to **integrate** (Specific Purpose - integrating the operation to the related machines)
2. ability to **measure** (Specific Purpose - measurement of machinery for operation execution)
3. ability to **adapt** (Social Ability - ability to adapt over time in response to changes in the environment)
4. ability to **self-organize** (Reactiveness - ability to establish basic rules for the interactions between the operations model and other models)
5. ability to **control** (Autonomy - ability to control of manufacturing operations)

In this study, 135 different operations are pre-defined. (For the Operation – Machine Relations for the 135 predefined Operation and respective measurement KPI (Key Performance Indicator) See Appendix 2)

Ability to integrate: This model is designed to identify well-structured definition of production operations for the best assignment of the operations to appropriate machines.

To accomplish this, a predefined set of operations must be determined. An "Operation Card" (see Table 4.4.) must be generated with each operation with respective machine relations

Each operation is defined in terms of the following attributes.

Operation Name: to quick identify opera

Card Code: to give unique number to operation

Operation Definition: to define detail information of the new operation

Related Machine Codes: To describe operation- machine relation

Machine Name: To identify related machine

Table 4.4 The structure of the "Operation Card"

OPERATIONS CARD																		
Operation Name:															**No:**	00001-O1		
Card Code:															Date:			
Operation Definition:											Operation Sequence Level							
Related Machine Codes	Machine Name	Machine Availability				Basic Information of Machine		Maintenance Information of Machines				Machine-Labor Relation		KPI Measurement			% Level of significance	Success Value
		Machine State	State Duration / hour.	Available Time	Available Duration	Capacity hour / TD	Priority Level	Maintenance Period	Maintenance Code	Needed maintenance Time	Next maintenance Time	Appropriate Lobar set	Needed Unit /hour	KPI Code	Planned KPI Value	Actual KPI Value		
Total Success Value of O1																		

Machine Availability: to update machine availability on software

Basic Information of Machine: to control machine shelf-life

Maintenance Information of Machines: to keep under control of the regular maintenance

Machine-Labor Relation: to identify set of operator could work on related machine

KPI Measurement: to measure machine capability and success values

% Level of significance: The contribution of the related Machines for the new Operation success

% Success Level: To calculate the new Operation success level by making calculation

Operation card is made of two main fields. These are the first field, which contains the fundamental information about the operation, and the second field, which is designed for defining and measuring of the machinery where the operations are carried out.

The first field contains operation name and explanation information and the operation sequence field for the control of operation activation sequence. When an operation order reaches, the Model does not activate this operation if the previous levels are not completed.

The first column in the second main field is the areas for machinery where the operations are defined. Content of the columns making up the second area is defined as follows; "Machine Availability" column providing the machine state information is generated and this column is divided into 4 fundamental columns enabling observation of machinery access. Machine state (if it is empty 0, if it is full 1) column indicating the load state of each machine and State Duration column, where active period of this state is indicated, are added. If the machine is empty, Available Time column is added for recording when it will be available, and then if there is an assignment is made for a future date, Available Duration column is added for seeing when the machine will be available.

"Basic Information of Machine" column is opened for enabling access of machinery fundamental information and this column is divided within itself into 2 columns. These are "Machine Capacity" column for enabling operation capacity measurement (daily capacity of the machine for the relevant operation) and machinery priority level column (if there is a more effective machine for the relevant operation, it is aimed that it is selected first).

"Maintenance Information of Machines" column is generated as a maintenance control area for the machine operation not to fail and this column is divided into 4 columns within itself. These are "Maintenance Period" showing the frequency for maintenance of the machinery, and Needed Maintenance Time column for defining operation stop time when the machine is taken for maintenance (for example 2 hours) and "Next Maintenance Time" column which will prevent work assignment to the following maintenance time.

"Machine-Labor Relation" column is generated for defining the operator list needed in the relevant machine operations and for controlling the operator capacity and this column is divided into 2 columns within itself. These are the column with the title "Appropriate Labor Set" to define the operator set (for example PNO; Punch operator list) and "Needed unit/hour" column for defining how much labor force is needed for the relevant operation.

"KPI Measurement" column is generated for enabling measurement of sub operations connected to the operation. This column is divided into three columns within itself. These are "KPI Code" defining the measurement criteria of the related sub operation, and "Planned KPI Value" column for defining the success level that the company aims, and "Actual KPI Value" column which is the result of the relevant KPI estimations (see Appendix 5 for the pre-defined KPI "Key Performance Measurement" techniques in Operations Agent).

Lastly, in order to calculate operation success percentage, "Level of Significance" column defining the effect of each machine on total success (again for the total effect of all machinery to be 100%) and "% Success Value" column is generated for entering success levels that are to be obtained by the multiplication of the Actual KPI value and level of Significance values.

"Total Success Value of 0" is added to the bottom part of the operation card and average of the success percentages of all relevant machinery is taken and total operation success level area that is to be calculated is formed.

The Structural frame code of the Operation Card definition function f_{kt}^{OD} is given below;

```
public class Operation {
	public void addOperationCard(String name, String code, String definition, List<MachineCard>
machines,
				HashMap<MachineCard, Float> machineLOS, List<KPIMeasurement>
measurement) {
		OperationCard operationcard = new OperationCard(name, code, definition, machines,
machineLOS, measurement);
		this.saveOperationCard(operationcard);
	}

	private void saveOperationCard(OperationCard operationcard) {

	}
}

public class OperationCard {
	public String Name;
	public String Code;
	public String Definition;
	public List<MachineCard> Machines;
	public HashMap<MachineCard, Float> MachineLOS;
	public List<KPIMeasurement> Measurement;

	public OperationCard(String name, String code, String definition, List<MachineCard> machines,
				HashMap<MachineCard, Float> machineLOS, List<KPIMeasurement>
measurement) {
	}
}
```

This code is used to create new process card in the software

Ability to Measure; In order to provide the operations model to have measurement ability, primarily, Operations Agent architecture was designed. The business rules were developed so as to provide it to have adaptation ability. The mathematical model, which will be used to measure the performance of the operation, was designed. Afterwards, it was provided to have measurement ability on the software environment through coding these business rules and the mathematical model. Now, these studies will be explained respectively.

Operations Model is generated for measuring operation performances included under the processes fundamentally and for defining the relationships of the operations with the process and machinery. Operations model agent architecture is design to satisfy this main aim (see Figure 4.9). Operations model is related with three models; these are Machine Model, Process Model and R&D Model. R&D Model transmits specifications belonging to operations present in the system to the information system. Machine

Model represents the machinery where the operations are realized, and transmits this information to the operation model once an operation is completed in the machinery.

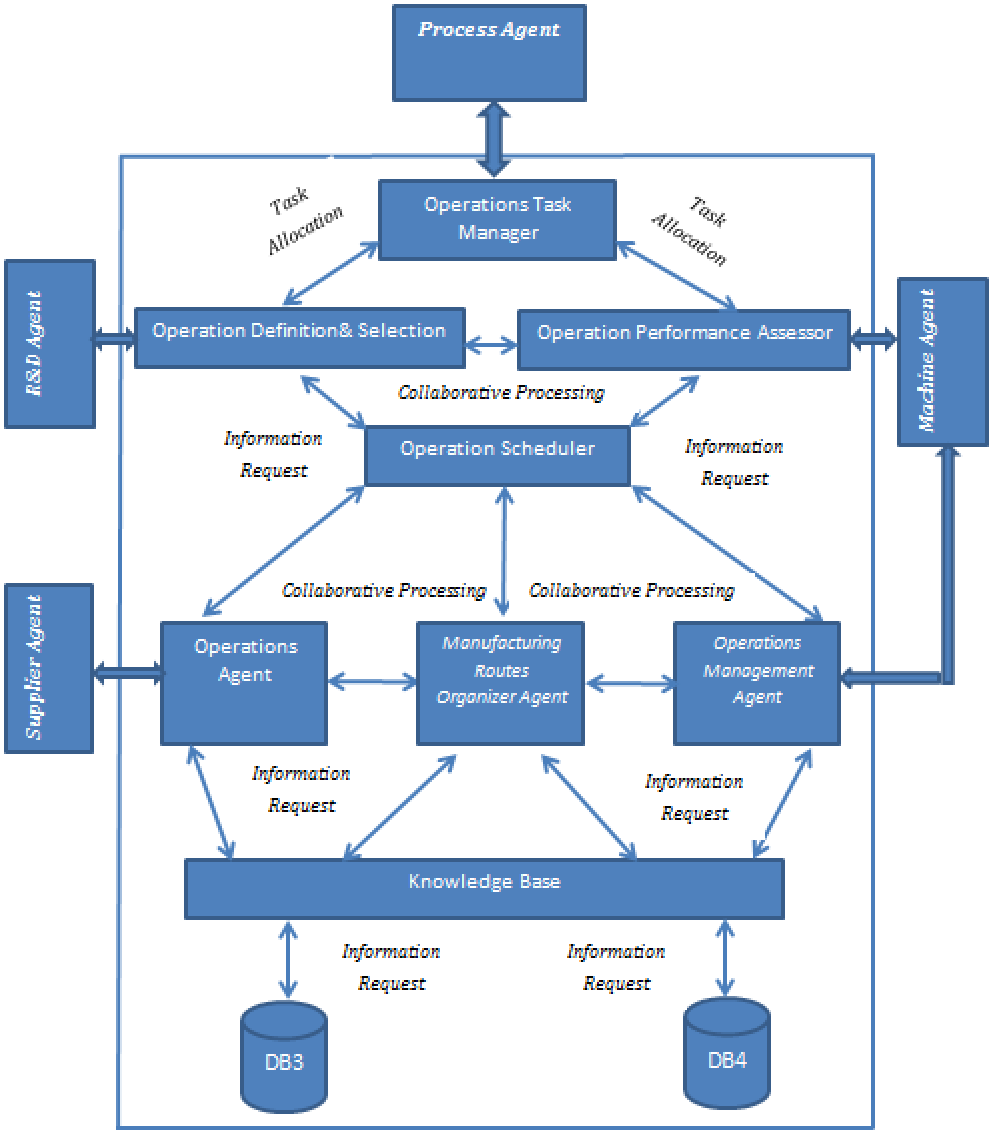

Figure 4.9. Operations Model Agent Architecture

Operation Model answers to two different inputs by using this information. If a performance measurement request is submitted from the system; Scheduler gives the data belonging to the operation to the user as an output by using the relevant performance measurement criteria. If an operation order reaches the system, the operation model sends the work order about the machinery where operation model operations will be realized, and when the process is completed, the scheduler records these data within DB3 (data base belonging the Operation Model).

New data generated following each completed operation are updated as well within DB2 belonging to the operation model simultaneously.

Operation model consists of 3 agents called Operations Agent, Manufacturing Route Organizer Agent, and Operations Management Agent. For each agent the following working principles are to be implemented;

1. **Operations Agent (OA):** This agent receives operation order information from the order organizer agent, raw material information from the supplier agent and prepares operations route order to send Manufacturing Route Organizer Agent.
2. **Manufacturing Route Organizer Agent (MROA):** This agent receives product operations route order from operations agent, operations list from database, engineering drawings from R&D agent and order information from order organizer agent and prepares an operation technique and sequence of each product.
3. **Operations Management Agent (OMA):** Operation management agent receives product operations list and route sequence and processing times then calculates the total manufacturing operations times of each operation to determine bottleneck operation and prepares a machine schedule order to send it Machine Operations manager agent.

Definition of the Operations Mathematical Model is explained below

$\mathrm{f(OA)} = \left\{Pt_k^O, A_k^O, \mathrm{f_{k,ts,t}^{OD}}, \mathrm{f_{k,ts,t}^{OPm}}, \mathrm{f_{k,ts,ad}^{O}}, \mathrm{SSO_k}\ , \in_k^O\right\}$ denotes the function set of the Operations Model where

- Pt_k^O set of percept of Operation k,
- A_k^O set of actions of Operation k,
- $\mathrm{f_{k,ts,t}^{OD}}$ denotes the Operation card definition function task ts of Operation k at time t,
- $\mathrm{f_{k,ts,t}^{OPm}}$ is performance measurement function task ts of Operation k at time t,
- $\mathrm{f_{k,ts,ad}^{O}}$ is adaptation function ad for task ts of Operation k

- $\mathrm{SSO_k}$ denotes the system states of Operations Agent and
- $\in_k^O$ defines set of environmental states of Operations

Success rate measurement function of the Operations model is given in Equation 9.

$$\textbf{Max } \boldsymbol{Z_k} = = \sum_k \boldsymbol{S_{k,m}} \left[\frac{\boldsymbol{M_{k,m}} - \boldsymbol{Ma_{k,m}}}{\boldsymbol{M_{k,m}}} * \mathbf{100}\right] \quad \textbf{(9)}$$

According to Equation 9, $\boldsymbol{Z_k}$ is the total success value of the operation k. $\boldsymbol{S_{j,k}}$ is the level of the significance value of machine m in operation k (The value of this parameter is based on expert decision) , $\boldsymbol{M_{k,m}}$is the planned machine value of machine m in the operation k, $\boldsymbol{Oa_{j,k}}$ actual machine value at time t.

$$\sum_k \boldsymbol{S_{k,m}} = \mathbf{100} \quad \textbf{(10)}$$

According to Equation 10, Total level of significance value must be equal to 100 for each operation.

$$\boldsymbol{M_{k,m}} = \boldsymbol{max}\left(\boldsymbol{M_{k,m,tp}}\right) + \boldsymbol{\beta_{M_{k,m}}}$$

(11)

According to Equation 11, attributes of the planned process value are defined. $\boldsymbol{max}\left(\boldsymbol{M_{k,m,tp}}\right)$ maximum success rate of the machine m in operation k at time period tp. $\boldsymbol{\beta_{M_{k,m}}}$ parameter gives dynamic nature to the model. This value can be between 0 and $100 \boldsymbol{-max}\left(\boldsymbol{M_{k,m,tp}}\right)$**.** Based on the experience this value is taken as 5 during the running process

$$\boldsymbol{Ma_{k,m}} = \sum_k \boldsymbol{S_{m,p}} \left(\boldsymbol{Pr_{m,p}} - \boldsymbol{Pra_{m,p,t}} \right) \quad \textbf{(12)}$$

According to Equation 12, calculation of the actual operation value is defined. The level of the significance value of product p is in machine m is defined as $\boldsymbol{S_{m,p}}$. planned success rate of product p in operation k is defined as $\boldsymbol{Pr_{m,p}}$ and $\boldsymbol{Mac_{k,m,t}}$ actual machine success rate of machine m in operation k at time t.

Structural frame code of Operation success measurement; Structural frame code of Operation measurement function $\mathrm{f_{k,ts,t}^{OPm}}$ is designed as follows;

```
public class Operation {
        public float successRate() {
                float rate = 0;
                for (MachineCard mc : Machines) {
                        rate += MachineLOS.get(mc) * (plannedValue - mc.successRate());
                }
                return rate;
        }
}
```

Ability to adapt: An example adaptation rule of the Operations Agent is pre-defined as follows;

Structural frame code of the considered adaptation rule is as follows;

```
//Operation machine error

public void onMachineError(MachineCard mc){
      for (OperationCard oc : OperationCardList) {
            if(oc.Machines.contains(mc)){
               oc.Reschedule(mc);
            }
      }
}
```

This rule is as follows; when one of the machines in the system has broken down; assign the operation to a new machine.

Ability to Self-organize: Basic rules are determined for the interactions between the operations model and other models. The business rules used to describe the information transfers, that will be performed during running of the model, (Which factor (agent) will perform how kind of information transfer with which agent) have been defined briefly in Figure 4.8

Some of pre-defined busiess rules listed as below;

- It gathers Company Operation information from Process Cards
- It gathers machine’s operations capacity from Machine Cards
- It gives an ability to measure the machine performance and selection of the most appropriate machine for the predefined operation
- It satisfies observation and intervention of the decrease in performance of Operations.

- It satisfy the new Operation integration to related machines
- It defines machine sequence
- It controls machine availability during manufacturing timeline
- It control machine maintenance periods
- It gathers machine-labor relations to assign
- For the each new order takes product information from the Product Card
- It reviews the operation tasks and appropriate operator name and validate based on their login credentials,
- It retrieves skill information for that operator to confirm their skills are within compliance guidelines, and
- It continue to guide to the next required operation or take alternate action

Abstract information flow of Operation model is given in Figure 4.10;

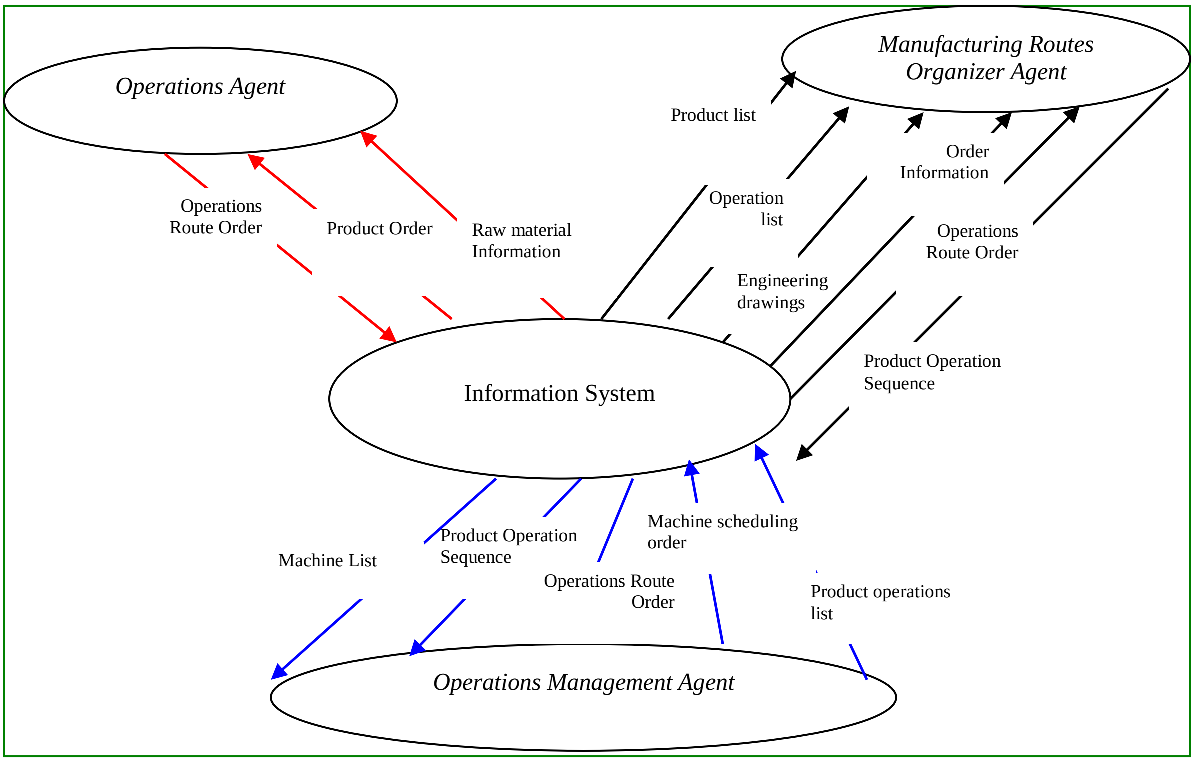

Figure 4.10. Operation Model Information Flow

When the Operation starts model agents use precedence rule to activate related agents. This Precedence rule is illustrated in Figure 4.11;

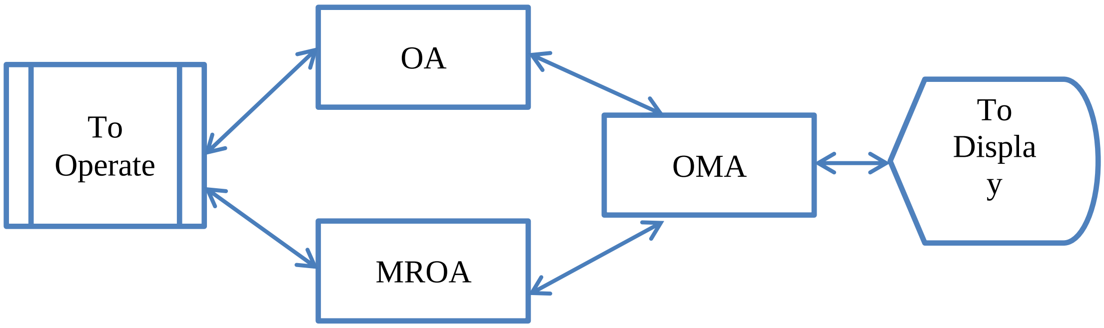

Figure 4.11. Precedence Structure of the proposed Operation Agents

4.2.4. Machine Model (Machine Agent)

Main task of this model is to create relationship between machines and products to be operated. To obtain this, a "Machine Card" is generated indicating respective machine operations which are needed to be used for manufacturing of the product on the machine. These machine operations are associated with the Product through Product Card of the Product Model.

Main motivations of the Machine Model are listed below;

1. ability to **integrate** (Specific Purpose - integrating the machines to the related products)
2. ability to **measure** (Specific Purpose - measurement of products manufactured using available machinery)
3. ability to **adapt** (Social Ability - ability to adapt over time in response to changes in the environment)
4. ability to **self-organize** (Reactiveness - ability to establish basic rules for the interactions between the Machine model and the system)
5. ability to **control** (Autonomy - ability to control of manufacturing machines)

Ability to integrate: In order to ensure the machines to be associated with the products manufactured, the machine operations used in manufacturing these products were taken into consideration. For example, for a product manufactured in a machine some operations, such as square cutting, circular cutting, and etc. were defined. Which product will be manufactured at which machine is decided considering the machine's operations which have been specified in its design. If the machine is convenient for all operations required for the manufacturing of the product (more than one machine operation may be required at the same time), that machine is considered as the manufacturing tool for that product.

In other words, main task of this model is to create relationship between machines and products to be operated. To obtain this, a "Machine Card" is generated indicating respective machine operations which are needed to be used for manufacturing of the product on the machine. These machine operations are associated with the Product through Product Card of the Product Model. The structure of the Machine Card is provided in Table 4.5.

Each machine is defined in terms of the following attributes;

Machine Name: to quick identify machine

Card Code: to give unique number to machine

Machine Tolerance Information: To identify machine tolerance information (X-Axis, Y-Axis and Z-Thickness)

Machine Operations Definition: To identify machine operations

Machine Operation Code: To quick identification of manufacturing operation

Machine operation Name: To identify area of usage of the machine operation

Operation Capacity: To structure equipment operation capacity

Operation Availability: To satisfy control of the usage of the equipment

KPI Measurement of the equipment: to measure equipment usage efficiency and measure success values

% Level of significance: The contribution of the related Equipment for the new machine success

Table 4.5. The structure of “Machine Card”

<table>
<tr><td colspan="10">MACHINE CARD</td><td>No</td><td>00001-PR1</td></tr>
<tr><td colspan="2">Machine Name</td><td colspan="8"></td><td>Date</td><td>01.01.2014</td></tr>
<tr><td colspan="2">Card Code:</td><td colspan="10"></td></tr>
<tr><td colspan="2">Machine Tolerance Information
(x,y,z)</td><td colspan="3">X:</td><td colspan="4">Y:</td><td colspan="3">Z:</td></tr>
<tr><td colspan="2">Operation Types:</td><td colspan="10"></td></tr>
<tr><td colspan="12"></td></tr>
<tr><td rowspan="2">Machine Operation Code</td><td rowspan="2">Machine Operation Name</td><td colspan="2">Operation Capacity</td><td colspan="4">Operation Availability</td><td colspan="4">KPI Measurement</td></tr>
<tr><td># of Unit produced / one operation</td><td>Operation Cycle Times / TSD</td><td>Unit 1 State</td><td>Unit 1 State Duration</td><td>Purchase Date / TYR</td><td>Unit 1 Shelf Life</td><td>KPI Code</td><td>Planned KPI Value</td><td>Actual KPI Value</td><td>Success Value</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="11">Total Machine Success</td><td></td></tr>
</table>

% Success Level: To calculate the Equipment success level by making calculation

Total Success Value From Equipment: Total Contribution of the equipment’s level success to the new Machine

Total Success: To calculate new Machine Success Value

Machine card is made of the first field containing fundamental information on machinery, and the second field defining manufacturing operations that are executed within the Machines. When we want to measure the machine efficiency, efficiency of all manufacturing operations that are executed in the machine is measured. Average of manufacturing operations, whose success measurements are estimated as percentage separately, gives the amount of usage efficiency of the machinery.

The first field contains machinery name and card code information and machine tolerance information (X-Axis, Y-Axis and Z-Thickness). Tolerance range is important for determining the kind of products for which the machinery is appropriate. Machine operation types are listed under a field is generated under the title “Operation Types”. This field is significant for part manufacturing that contains multi processes.

In the second field, defining and measuring of manufacturing operations that could be carried out in the Machinery is planned. The first column is the code of the manufacturing operation and the second column is a field designated for the name of the manufacturing operation. In the third column, under the title “Equipment Capacity”, capacity information field of the equipment that are used in the relevant machine operations is generated (these equipment represent machine manufacturing operations at the same time) and it is divided into two separate sub columns. These are; “# of Unit produced / one operation” column defining how many products could be processed in one time when the relevant manufacturing operation is executed in the machine (what is meant here is that the cutting apparatus used for square cutting process can cut 4 pieces of the same dimension in a single strike, and this is very important to estimate manufacturing numbers) and “Operation Cycle Times” column where cycle time of the machine operation is defined.

The next column is “Equipment Availability” column generated for defining accessibility of Machine manufacturing operations (if this is a mold, its change date, if it is a software, its usage time, etc.) and this column is divided into 4 sections within itself. These are the column showing the current state of the machine operation under the title “Unit 1 State” (“0” for available, “1” for busy); the column defining the duration of the current state “Unit 1 State Duration”; the column defining purchase duration of the equipment or software under the heading “Unit 1 Shelf Life”. The significance of this field is to provide checking of limited number of manufacturing equipment and/or software while job assignment is made in the machines in the information management system. Since it is possible for a software and/or equipment to be present more than one, the headings about the properness is opened as “Unit 1”, and when necessary the table can be extended rightward.

The last column is "KPI Measurement" column to define methods that are to be used to measure the efficiency of manufacturing operations and this column is divided into 4 columns within itself. The first column is "KPI Code" column where performance measurement criteria code is written. The second column is "Planned KPI Value" column where success percentage expected from the manufacturing operation is entered. The third column is "Actual KPI Value" column where current success percentage coming up as a result of KPI calculation is entered, and the last column is "% Success Value" where success level of machine operations is calculated. Since effect of each operation that is executed in the machinery on machine effectiveness is accepted to be equal "Level of Significance" title is not included in this card. Percentage rate is distributed to all machinery operations equally and multiplied by "Actual KPI Value" and success percentage of each machine operation is calculated.

Undermost, average of machine operation success percentages estimated separately above in "Total Machine Success Value" line is taken, and average success percentage of the machine (effective usage percentage) is obtained.

The Structural frame code of the Machine Card definition function $f^{MD}_{m,ts,t}$ is given below;

```
public class Machine {
	public void addMachineCard(String name, String code, String definition, String type, MachineTolerance tolerance,
			List<MachineOperation> machineOperations, List<KPIMeasurement> measurements, List<Equipment> equipments,
			List<RequiredUnit> requiredEquipment) {
		MachineCard machinecard = new MachineCard(name, code, definition, type, tolerance, machineOperations,
				measurements, equipments, requiredEquipment);
		this.saveMachineCard(machinecard);
	}
	private void saveMachineCard(MachineCard machinecard) {

	}
}
```

```
public class MachineCard {
	public String Name;
	public String Code;
	public String Definition;
	public MachineTolerance Tolerance;
	public String Type;
	public List<MachineOperation> MachineOperations;
	public List<KPIMeasurement> MachineOperationKPIMeasurements;
	public List<Equipment> Equipments;
	public List<RequiredUnit> requiredEquipment;

	public MachineCard(String name2, String code2, String definition2, String type2,
MachineTolerance tolerance2,
			List<MachineOperation> machineOperations2, List<KPIMeasurement>
measurements, List<Equipment> equipments2,
			List<RequiredUnit> requiredEquipment2) {
	}
}
```

Ability to Measure: In order to provide the Machine model to have measurement ability, primarily, the machine's operations were defined. So as to perform these operations, Machine Agent Architecture was designed. The business rules were defined so as to provide it to have adaptation ability. The machine model, which will be used in measuring the performance of the machine model, was designed. Afterwards, it was provided to have measurement ability on the software environment through coding these business rules and the mathematical model. Now, these studies will be explained respectively.

Main agent architecture of the Machine model are illustrated in Figure 4.12. As can be seen in the figure the Machine model consists of 5 agents called Master Production Planner, Capacity Planner, Machine Operations Manager, Equipment Organizer, and Machine Scheduler.

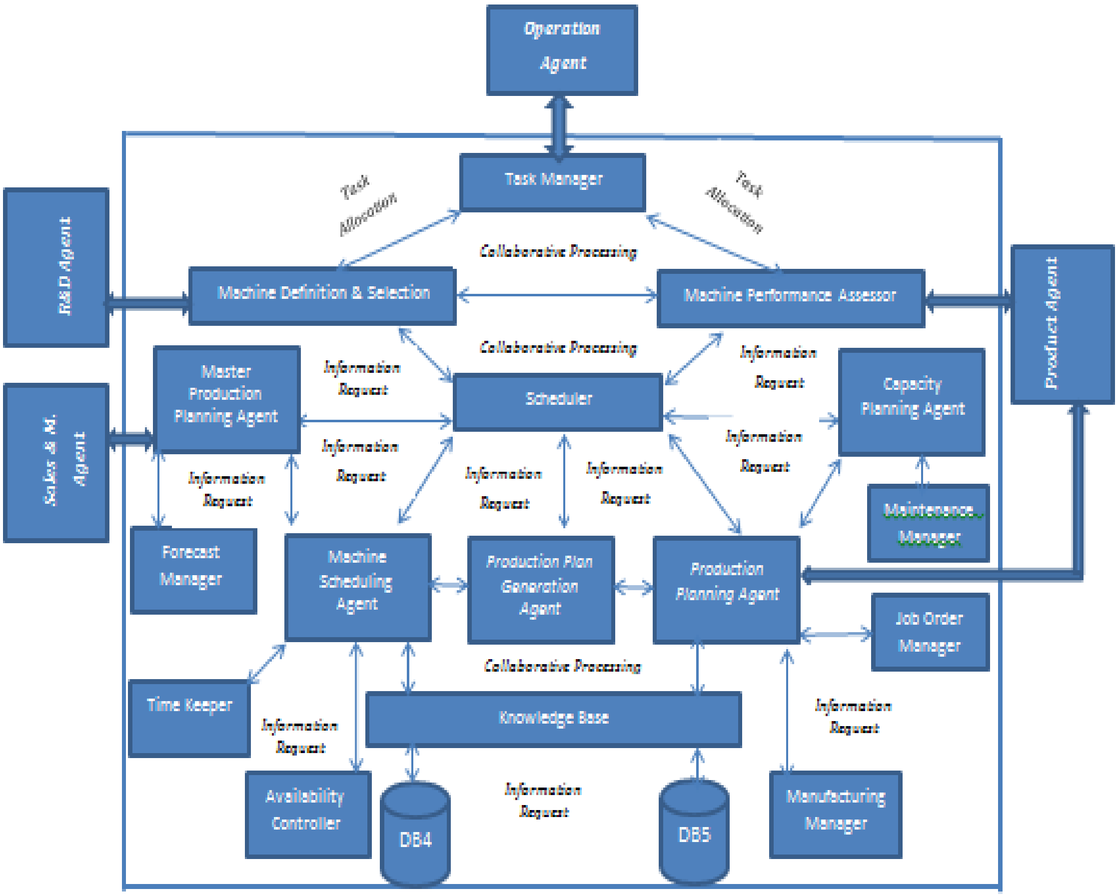

Figure 4.12. Main components of the proposed Machine Agents

Similarly, the model is also utilizing the following task; job order, availability controller, forecast manager, and maintenance manager.

1. **Master production planning agent;** This agent receives production evaluation Reports from the sales management agent, machine schedule order from sales and marketing agent, process information from process planning agent, machine information from supply agent, loaded job information from manufacturing agent, status information from capacity planning agent, products technical information from design agent, information from stock level control agent, and provides master production schedule accordingly. This agent's primary task is to create master production planning annually and seasonally in accordance with order estimation and capacity information. This plan plays a guiding role for production planning based on long term sales targets of the company.

2. **Production planning agent;** This agent receives master production schedules from master production planning agent, products information from design agent, machines information from supply agent, loaded job information from manufacturing agent and prepares a production plan for production plan generation agent and order delivery information for customer relationship agent. Primary task of this agent is to form and transfer current production orders in most appropriate manner without violating master production plan. This agent also determines annual production planning form depending on master production planning and sends it to the production planning generation agent.
3. **Production plan generation agent**; This agent receives process flow Information from process planning agent, standard times of the part that is to be produced from time study agent, loaded job information from manufacturing agent, capacity status information from capacity planning agent, production plan from production planning agent, and prepares production programming for scheduling agent. Primary task of this agent is to form and transfer current production orders according to production plans without violating them. This agent also determines a delivery date for each order according to designed plan and provides a production programming form to determine date of delivery to customer by informing sales management agent.
4. **Machine Scheduling agent;** This agent receives operation orders from process planning agent, production programming from production plan generating agent, loaded job information from manufacturing agent, machine information from supply agent, maintenance time scheduling from capacity planning agent, and prepares job orders for manufacturing and transmitting agent. Primary objective of this agent is to form and transfer current production orders properly without leaving production program. To do this, this agent prepares job order.
5. **Capacity planning agent**; This agent receives machine information from supply agent, person information from human research management agent, sub product standard times from time study agent, maintenance time scheduling information from scheduling agent, and prepares capacity status information for master production planning agent and production planning generation agent and

capacity requirements for factory management agent. Primary task is to form current status information and capacity requirements depending on company assets capacities and order estimations to anticipate the future.

Definition of the Machine Mathematical Model is explained below

$f(\mathrm{MA}) = \{Pt_m^M, A_m^M, \mathrm{f}_{\mathrm{m,ts,t}}^{\mathrm{MD}}, \mathrm{f}_{\mathrm{m,ts,t}}^{\mathrm{MPm}}, \mathrm{SSM_m}\ , \in_m^M\}$ denotes the function set of the Machine Model

where

- Pt_m^M set of percept of Machine m,
- A_m^M set of actions of Machine m,
- $\mathrm{f}_{\mathrm{m,ts,t}}^{\mathrm{MD}}$ denotes the Machine card definition function task ts of Machine m at time t,
- $\mathrm{f}_{\mathrm{m,ts,t}}^{\mathrm{MPm}}$ is performance measurement function task ts of Machine m at time t,
- $\mathrm{SSM_m}$ denotes the system states of Machine Agent and
- $\in_m^M$defines set of environmental states of Machines

Success rate of the Machine model is given in Equation 13.

$$\textbf{Max } \boldsymbol{Z_m} = \sum_{m} \boldsymbol{S_{m,mo}}\ [\left(\boldsymbol{MO_{m,mo}} - \ \boldsymbol{MOa_{m,mo,sp,t}}\right) * \mathbf{100}/\boldsymbol{MO_{m,mo}}) \qquad \textbf{(13)}$$

According to Equation 13; **Max** $\boldsymbol{Z_m}$ is the objective function of the machine model. $\boldsymbol{S_{m,mo}}$ level of significance value of machine operation mo in machine m. $\boldsymbol{MO_{m,mo}}$ planned value of operation mo in machine m and $\boldsymbol{MOa_{m,p,mo,t}}$ is the actual value of machine operation mo, in machine m which is occurred production of sub product sp.

$$\sum_{k} \boldsymbol{S_{m,mo}} = \mathbf{100} \qquad \textbf{(14)}$$

According to Equation 14, Total level of significance value must be equal to 100 for each operation.

$$\boldsymbol{Pr_{m,p}} = \boldsymbol{max}\ \left(\boldsymbol{MO_{m,mo,tp}}\right) + \boldsymbol{\beta_{Pr_{m,p}}}$$

(15)

According to Equation 15, attributes of the planned product success rate are defined.

$\boldsymbol{max}\left(\boldsymbol{MO}_{\boldsymbol{m,mo,tp}}\right)$ maximum success rate of the machine operation mo in machine m, at time period tp. $\boldsymbol{\beta}_{\boldsymbol{Pr}_{\boldsymbol{m,p}}}$ parameter gives dynamic nature to the model. This value can be between 0 and $100\boldsymbol{-max}\left(\boldsymbol{MO}_{\boldsymbol{m,mo,tp}}\right)$**.** Based on the experience this value is taken as 5 during the running process

$$\boldsymbol{MOa}_{\boldsymbol{m,mo,sp,t}} = \sum_{sp} \boldsymbol{S}_{\boldsymbol{mo,sp}} \left(\boldsymbol{Pr}_{\boldsymbol{mo,p,sp,}} - \boldsymbol{Pr}_{\boldsymbol{mo,p,sp,t}} \right) \tag{16}$$

According to Equation 16, calculation of the actual production success rate is defined. The level of the significance value of machine operation mo. $\boldsymbol{Pr}_{\boldsymbol{mo,p,sp,}}$ is the planned success rate of machine operation mo, for sub-pruduct sp in product p. $\boldsymbol{Pr}_{\boldsymbol{mo,p,sp,t}}$ is actual success rate of machine operation mo at time t.

Structural frame code of Machine success measurement; Structural frame code of Machine measurement function $f^{MPm}_{m,ts,t}$ is designed as follows;

```
public class Machine {
// Machine Success Rate
	public float successRate() {
		float rate = 0;
		for (MachineOperation mo : MachineOperations) {
			rate += mo.LevelOfSignificance * (mo.PlannedKPIValue -
mo.ActualKPIValue);
		}
		return rate;
	}
}
```

Ability to adapt: An example adaptation rule of the Machine Agent is pre-defined as follows;

```
public class Machine {
	// New Machine operation
	public void onMachineOperationAdded(MachineOperation mo) {
		for (MachineCard mc : MachineCardList) {
			if ((mc.Tolerance.minWidth <= mo.Width && mo.Width <=
		mc.Tolerance.maxWidth)
&& (mc.Tolerance.minLength <= mo.Length && mo.Length <= mc.Tolerance.maxLength)
&& (mc.Tolerance.minThickness <= mo.Thickness && mo.Thickness <=
mc.Tolerance.maxThickness)) {
				mc.MachineOperations.add(mo);
			}
		}
	}
}
```

This rule is as follows: When a new machine operation has been created in the system, (new product manufacturing case), it compares the tolerance value (x: weight y: length and y: thickness) of each machine in the system with the tolerance values of the operations of the new machine. This new operation is included in the machine's cards satisfying the tolerance values.

Ability to Self-organize So as to provide the model to have self-organizing property, the precedence rule primarily for the agents was included in the model. Afterwards, it was designed that which information the system would get through the model and the other agents inside it (Perceive set), and which information it will provide to the system (action set). This summary information flow map is shown in Figure 4.13.

Some of the business rules of Machine Model are listed below;

- It gathers Company Machine information from Machine Cards
- It gathers machine's operations specifications from Product Cards
- It measures the new Machine success level by the using defined KPI calculations
- It gives an ability of measure the equipment and software performance and selection of available software and equipment
- For the new manufacturing job order controls the availability of the respected software and equipment
- It satisfies observation and intervention of the decrease in performance of Machine capacity.
- It satisfy the new Machine integration to related products
- It defines needs of the labor force for machine's operation
- It controls equipment shelf life and prepares a report for upcoming out of date equipment list
- It control software shelf life and prepares a report for upcoming out of date software list

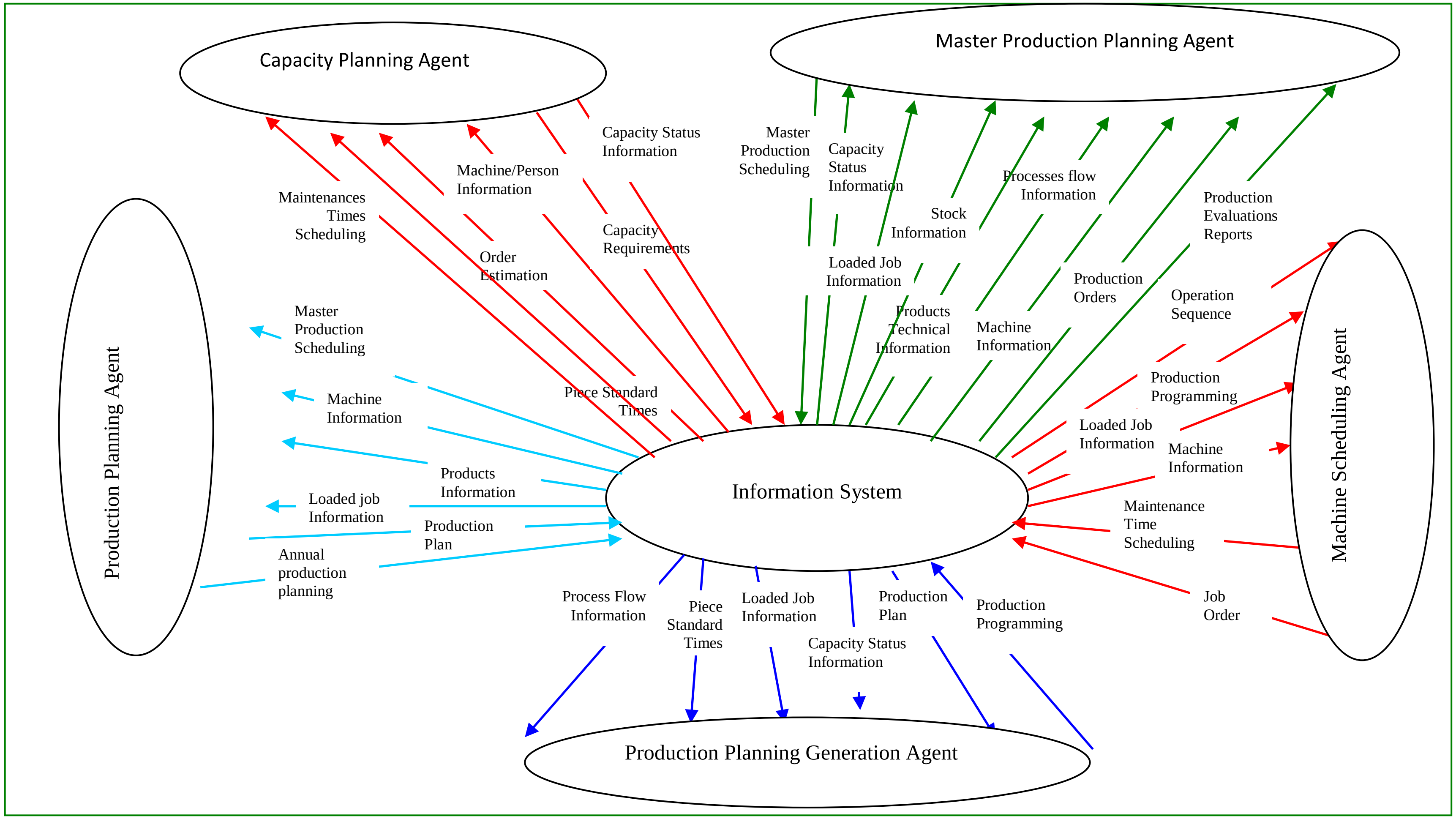

Figure 4.13. Machine Model Agent Information exchange architecture

When the Machine Model starts model agents use precedence rule given in Figure 4.14;

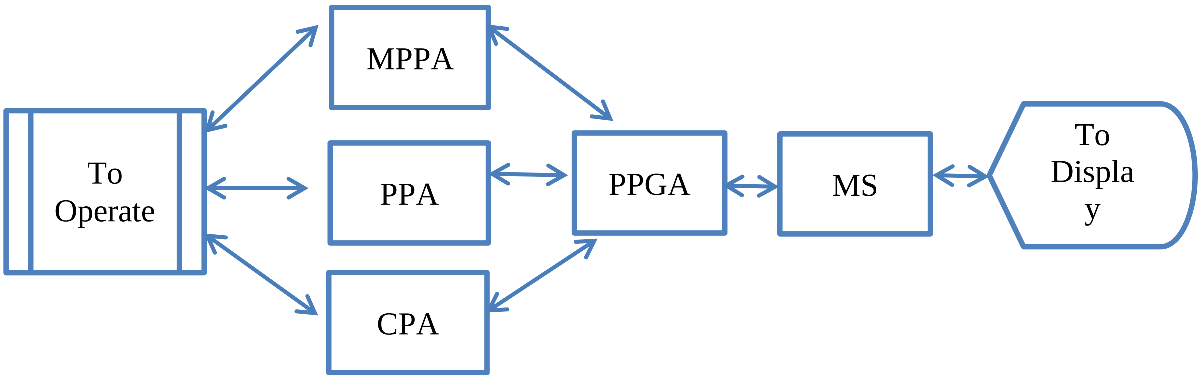

Figure 4.14. Precedence Structure of proposed Machine Agents

4.2.5. Product Model (Product Agent)

Primary task of this model is to create relationship between products and the raw materials in a certain manufacturing system and support the assessment with respect to whether those manufacturing specifications are satisfied or not.

Main motivations of the Product Model are listed below;

1. ability to **integrate** (Specific Purpose - integrating the products to the related raw materials)
2. ability to **measure** (Specific Purpose – machine operations and raw materials used for the manufacturing of the products)
3. ability to **adapt** (Social Ability - ability to adapt over time in response to changes in the environment)
4. ability to **self-organize** (Reactiveness - ability to establish basic rules for the interactions between the Product model and the system)
5. ability to **control** (Autonomy - ability to control of manufacturing of the products)

Ability to integrate: In this model, the raw materials used in manufacturing the products were associated with the products and the integration of the product and the supplier models was created. The sub products were associated with the machine operations, and the integration of Machine and Product models was ensured. In addition, after the exchange of information between the model and the agents inside it was defined, the integration structure of the model was completed.

To achieve this, a “Product Card” is created including the respective information in Table 4.6.

Each Product is defined in terms of the following attributes;

Product Name: to quick identify Product

Card Code: to give unique number to Product

Product Family Code: To prepare Product family groups

Product Color Code: To identify new Product Color

Weight, Height, Thicknesses: To identify new Product Specifications

Machine Definition: To define detail information of the new machine

Shelf Life: To control new Product shelf life

Product Description: To give detail information of the new Product

Machine Set Code: To identify manufacturing root depends on Machine sets

Machine Operations and Specifications: To identify each route machine operation and respective specifications (bill of material units, cycle times, manufacturing setup times and sequence of the machine operations)

Sub product Code: To determine sub products in new Product

Sub Product Name: To identify name of the sub products

N. R. M.T: denotes the needed Raw Material Type

N. M. Volume /Unit: denotes the needed Raw Material volume for per product

BOM Unit: indicates that needed number of parts that are used in one product

KPI Measurement: to measure new Product efficiency and measure success values

% Level of significance: The contribution of the related Sub product for the new Product success

% Success Level: To calculate the sub product success level by making calculation

Total Success: To calculate new Product Success Value

Product Cards store all information on the products that are to be manufactured in the machines. First three lines of this card that is generated for each Main Product are assigned for Product Name, Card Code and Product Description (Product Detail) which are the main information on the product. The fourth line is a field assigned for Route Sequence, sub monetary production process route, and the fifth line is assigned for the names of the Processes. Columns are generated for detailed product information.

Table 4.6. The Structure of “Product Card”

<table>
<tr><td colspan="31">PRODUCT CARD</td></tr>
<tr><td colspan="2">Product Name</td><td colspan="25"></td><td colspan="2">No</td><td colspan="2">00001-PR1</td></tr>
<tr><td colspan="2">Card Code:</td><td colspan="25"></td><td colspan="2">Date</td><td colspan="2">01.01.2014</td></tr>
<tr><td colspan="2">Product Description</td><td colspan="18"></td><td colspan="3">Shelf life:</td><td colspan="4"></td><td colspan="2">Stock Quantity:</td><td colspan="2"></td></tr>
<tr><td colspan="31"></td></tr>
<tr><td colspan="2">Route Sequence</td><td colspan="3">0</td><td colspan="10">1</td><td colspan="5">2</td><td colspan="2">3</td><td colspan="4">4</td><td colspan="5">5</td></tr>
<tr><td colspan="2">Machine
Operation Type</td><td colspan="3">PREPARING</td><td colspan="10">Type 1</td><td colspan="5">Type 2</td><td colspan="2">Type 3</td><td colspan="4">Type 4</td><td colspan="5">Type 5</td></tr>
<tr><td>Sub Product Code</td><td>Sub Product Name</td><td>N. R. M.T</td><td>N. M. Volume /Unit</td><td>BOM Unit</td><td>N. PCH.O.1</td><td>N. PCH.O.2</td><td>N. PCH.O.3</td><td>N. PCH.O.4</td><td>N. PCH.O.5</td><td>N. PCH.O.6</td><td>N. PCH.O.7</td><td>N. PCH.O.8</td><td>Unit Cycle Time / TMT</td><td>Unit Setup Time / TMT</td><td>BOM Unit</td><td>N. BND.O.1</td><td>N. BND.O.2</td><td>Unit Cycle Time / TMT</td><td>Unit Setup Time / TMT</td><td>N. JT. O.1</td><td>Unit Cycle Time / TMT</td><td>BOM Unit</td><td>N. TM. O.1</td><td>Unit Setup Time / TMT</td><td>Unit Cycle Time / TMT</td><td>BOM Unit</td><td>N. ASLO. O.1</td><td>Assembly Level</td><td>Unit Setup Time / TMT</td><td>Unit Cycle Time / TMT</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="5">KPI Code</td><td colspan="10"></td><td colspan="5"></td><td colspan="2"></td><td colspan="4"></td><td colspan="5"></td></tr>
<tr><td colspan="5">Planned KPI Value</td><td colspan="10"></td><td colspan="5"></td><td colspan="2"></td><td colspan="4"></td><td colspan="5"></td></tr>
<tr><td colspan="5">Actual KPI Value</td><td colspan="10"></td><td colspan="5"></td><td colspan="2"></td><td colspan="4"></td><td colspan="5"></td></tr>
<tr><td colspan="5">Level of Significance</td><td colspan="10"></td><td colspan="5"></td><td colspan="2"></td><td colspan="4"></td><td colspan="5"></td></tr>
<tr><td colspan="26">Total Success value of PR</td><td colspan="5"></td></tr>
</table>

The first column is assigned for code of sub pieces making up the product, and the second column is assigned for the names of the sub products. In the third column, type of raw material to be used in sub piece manufacturing and in the fourth column the amount of the used raw materials is given under the title “N.M. Volume/Unit”. In the fifth column, unit number showing how many sub pieces are included within the product is given under the title “BOM (Bill of Material) Unit”. After these first 5 column data containing preparatory information of the product are formed, codes of the operations to be applied in machinery within the manufacturing processes are written under the relevant process and at the end “Unit Cycle Time” for cycle time of these machine operations, and “Unit Setup Time” fields are generated for preparatory time.

Owing to this structure that is generated, general information on a product is formed on a single card, and relevant machine operation codes are related with Machine model, and raw material codes are related with supplier model.

When a request is generated on the system about a product, Product Agent will transmit necessity information located on the card to Supplier Agent, and for the assignment of machine planning, it will transmit machine operations list needed to Machine Agent by means of this Card.

A KPI (key performance indicator) field is designed at the low part of the Card for determining performance measurement criteria on the product and for measuring the difference between the executed performances. In this field, percentage success rates are obtained by comparing in percentage time spent in each manufacturing process currently of Product sub pieces to the planned time. Product effectiveness is measured as an average of percentage success obtained from manufacture in all processes.

The Structural frame code of the Product Card definition function $f_{p,ts,t}^{PrD}$ is given below;

```
public class Product {
	public void addProductCard(String name, String code, String desciption,
List<SubProduct> subProducts,
				List<ProductOperation> productOperations) {
		ProductCard productcard = new ProductCard(name, code, desciption,
subProducts, productOperations);
		this.saveProductCard(productcard);
	}

	private void saveProductCard(ProductCard productcard) {

	}
}

public class ProductCard {
	public String Name;
	public String Code;
	public String Description;
	public List<SubProduct> SubProducts;
	public List<ProductOperation> ProductOperations;

	public ProductCard(String name, String code, String desciption, List<SubProduct>
subProducts,
				List<ProductOperation> productOperations) {
	}

}
```

Ability to Measure: In order to provide the Product model to have measurement ability, primarily, sample products and raw materials were designed. In order to define the working principles of the model, the Product Agent Architecture was designed, The business rules were defined so as to provide it to have adaptation ability. A mathematical model, which will be used in measuring the performance of the product model, was designed. Afterwards, it was provided to have measurement ability on the software environment through coding these business rules and the mathematical model. Now, these studies will be explained respectively.

The Product Model is made of only a single main Agent. This agent is called the Product Agent. Main components of the Product Agent are illustrated in Figure 4.15. It is assumed that input will reach the Product Agent from the system in two ways. These are inquiry of performance measurement data on a product or manufacturing job order coming from the Machine agent on a product. When a manufacturing job order reaches, the Product Agent reaches product stock information by means of Material requirement planner located within, and reaches product manufacturing technical operation information by means of Product Technical Manager. In case there is sufficient product stock, it sends information on the readiness of product transportation to the system, and if the product stock is not sufficient, it sends a Procurement Order to the Supplier Agent for execution of the relevant purchasing.

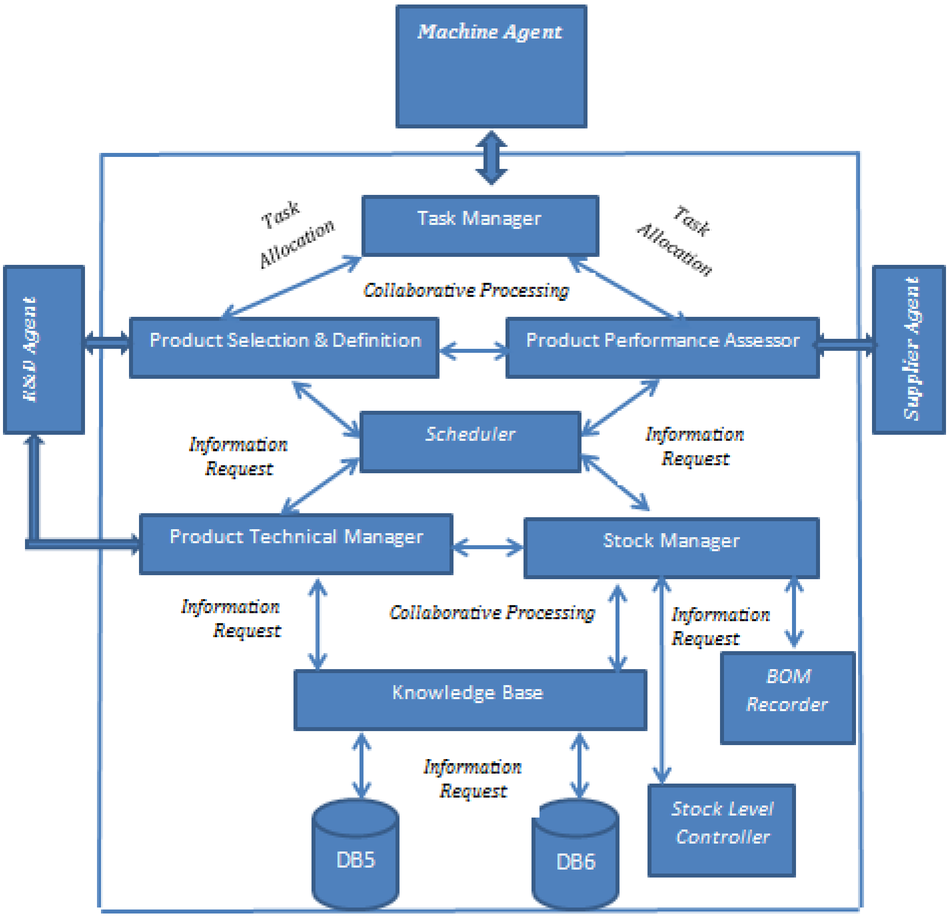

Figure 4.15. Product Model Agent Architecture

During Model production is carried out, it receives manufactured number information from the Machine Agent and checks whether it is appropriate to the numbers of Bill of Material and when there is insufficient production, it transmits this information to the Machine Agent. Also this agent controls shelf life of the manufactured product and prepares a report for out of date inventory.

Definition of the Product Mathematical Model is explained below

$f(\mathrm{Pr\ A}) = \left\{Pt_p^{Pr}, A_p^{Pr}, Q_{p,dr}^{Pr}, RM_{p,rm,s}^{Pr,S} \mathrm{f}_{\mathrm{p,ts,t}}^{\mathrm{PrD}}, \mathrm{f}_{\mathrm{p,ts,t}}^{\mathrm{PrPm}}, \mathrm{SSPr_p}\ , \in_m^{Pr}\right\}$ denotes the function set of the Product Model where

- Pt_p^{Pr} set of percept of product p,
- A_p^{Pr} set of actions of product p
- $Q_{p,dr}^{Pr}$ set of defect rate dr of product p
- $RM_{p,rm,q}^{Pr,S}$ is the success rate sr of raw material rm which is used in product p and purchased from supplier q
- $\mathrm{f}_{\mathrm{p,ts,t}}^{\mathrm{PrD}}$ denotes the Product card definition function task ts of Product p, at time t,
- $\mathrm{f}_{\mathrm{p,ts,t}}^{\mathrm{PrPm}}$ is performance measurement function task ts of Product p, at time t,
- $, \mathrm{SSPr}_{\mathrm{p}}$ denotes the system states of Product Agent and
- $\in_m^{Pr}$ defines set of environmental states of Products

Mathematical formulation of the Product model success rate measurement function is pre-defined in the model as given in Equation 17.

$$\textbf{Max } \boldsymbol{Z_p} = =\alpha^* \textbf{ avg}(\ \textstyle\sum_p \sum_{sp} \boldsymbol{S_{mo,sp}}\ (\boldsymbol{SP_{sp,mo}^{M}} - \boldsymbol{SPa_{sp,mo}^{M}})) + \gamma\ (\boldsymbol{Q_{p,dr,tp}^{Pr}}) + \theta\ (\boldsymbol{RM_{p,s,rm}^{Pr,S}}) \qquad \textbf{(17)}$$

According to Equation 17, $\boldsymbol{Z_p}$ is the total success value of the machine m. $\boldsymbol{S_{mo,sp}}$ level of significance value of machine operation mo which is used manufacture sub product sp. $\boldsymbol{SP_{sp,mo}^{M}}$ is planned success rate of the machine operation mo for the manufacturing sub product sp, $\boldsymbol{SPa_{sp,mo}^{M}}$ actual success rate of the machine operation mo at time t. $\boldsymbol{Q_{p,spdr,tp}^{Pr}}$ is defect rate dr of sub product sp at time period tp. $\boldsymbol{RM_{p,rm,q}^{Pr,S}}$ is the success rate of raw material rm used to manufacturing sub product sp in product p and which is purchased from supplier q. $\boldsymbol{\alpha}$**,** $\boldsymbol{\gamma}$ and $\boldsymbol{\theta}$ stand for impact factor of operational success, quality defect rate and raw material success rate respectively Where the summation of these factors is unity. In this study $\boldsymbol{\alpha}$ is assumed as 0.60, $\boldsymbol{\gamma}$ is assumed as 0.20 and $\boldsymbol{\theta}$ is assumed as 0.20.

$$\textstyle\sum_p \sum_{sp} \boldsymbol{S_{mo,sp}} = \mathbf{100} \qquad \textbf{(18)}$$

According to Equation 18, Total level of significance value must be equal to 100 for each sub product in product p.

$$\boldsymbol{SP^{M}_{sp,mo} = max\left(SP^{M}_{sp,mo,tp}\right) + \beta_{SP^{M}_{sp,mo}}} \quad \textbf{(19)}$$

According to Equation 19, attributes of the planned product success rate are defined. $\boldsymbol{max\left(SP^{M}_{sp,mo,tp}\right)}$ maximum success rate of the machine operation mo for sub product sp, at time period tp. $\boldsymbol{\beta_{SP^{M}_{sp,mo}}}$ parameter gives dynamic nature to the model. This value can be between 0 and 100$\boldsymbol{-max\left(SP^{M}_{sp,mo,tp}\right).}$ Based on the experience this value is taken as 5 during the running process.

$$\boldsymbol{SPa^{M}_{sp,mo} = \sum_{sp} SP_{mo,sp}\left(M_{m,mo} - Pr_{m,p,sp,mo,t}\right)} \quad \textbf{(20)}$$

According to Equation 20, calculation of the actual sub product machine operation success rate is defined. $\boldsymbol{SP_{mo,sp}}$ **is the** level of the significance value of machine operation mo for sub product sp. $\boldsymbol{Pr_{mo,p,sp,}}$ is the actual success rate of machine operation mo, for sub-product sp in product p.

$$\boldsymbol{Q^{Pr}_{p,dr,tp} = \sum_{sp}\sum_{mo}\left(NSPa_{mo,sp} / NSP_{mo,sp}\right) * 100} \quad \textbf{(21)}$$

According to Equation 21, calculation of defect rate dr of product p, at time period t is defined as $\boldsymbol{NSPa_{mo,sp}}$ is the actual number of manufactured unit for sub product sp after operation mo. $\boldsymbol{NSP_{mo,sp}}$ is the number of planned manufactured unit for sub product sp in operation mo.

Structural frame code of Product success measurement function; Structural frame code of Product success rate measurement function is given below;

```
public class Product {
	public float successRate() {
		float rate = 0;
		float avg = 0;
		// Machine operations success rate  * 0.6
		for (ProductOperation po : this.ProductOperations) {
			avg += po.successRate();
		}
		rate = avg / this.ProductOperations.size();
		// Hammaddelerden gelen başarı yüzdesi * 0.2
		HashMap<RawMaterial, Float> neededRM = this.calculateNeededRM();
		float _total = 0;
		float _val = 0;
		for (Entry<RawMaterial, Float> entry : neededRM.entrySet()) {
			RawMaterial rm = entry.getKey();
			Float needed = entry.getValue();
			_val += (needed * rm.getSupplier().calculatePQPTbyRM(rm));
			_total += needed;
		}
		rate += (_val / _total * 0.2);

		// Qualitydefect rate * 0.2
		rate += (this.getNonDefectRate() * 0.2);

		return rate;
	}

	public HashMap<RawMaterial, Float> calculateNeededRM() {
		HashMap<RawMaterial, Float> rmList = new HashMap<RawMaterial, Float>();

		for (SubProduct sp : this.SubProducts) {
			float val = sp.BillOfMaterial * sp.RawMaterialVolume;

			if (rmList.containsKey(sp.RawMaterial)) {
				rmList.replace(sp.RawMaterial, rmList.get(sp.RawMaterial) + val);
			} else {
				rmList.put(sp.RawMaterial, val);
			}
		}

		return rmList;
	}
	public float getNonDefectRate() {
		float rate = 96;
		// calculate rate
		return rate;
	}
}
public class SubProduct {
	public String Code;
	public String Name;
	public RawMaterial RawMaterial;
	public float RawMaterialVolume;
	public int BillOfMaterial;
	public List<MachineOperation> Operations;
}
public class ProductOperation {
	public KPIMeasurement KPIMeasurement;
	public List<SubProduct> SubProducts;
	public float calculateUCT(String SubProductCode) {
	}
	public float claculateUST(String SubProductCode) {

	}
```

```
// Machine operations success rate calculations
public float successRate() {
        float rate = 0;
        float _totalPlanned = 0;
        float _totalActual = 0;
        List<MachineOperation> iterated = new ArrayList<MachineOperation>();
        for (SubProduct sp : SubProducts) {
                for (MachineOperation mo : sp.Operations) {
                        _totalPlanned += mo.CycleTime;

                        if (!iterated.contains(mo)) {
                                iterated.add(mo);
                                _totalActual += mo.ActualKPIValue;
                        }
                }
        }
        rate = _totalActual / _totalPlanned * 100;
        return rate;
}
}
```

Ability to adapt: An example adaptation rule of the Product Agent is pre-defined as follows;

```
//raw material usage
private void onProductManufactured(ProductCard pc) {
        HashMap<RawMaterial, Float> usedRM = pc.calculateNeededRM();
        for (Entry<RawMaterial, Float> entry : usedRM.entrySet()) {
                RawMaterial rm = entry.getKey();
                Float used = entry.getValue();
                rm.useRawMaterial(used);
        }
}
```

This rule is as follows: After each production process, it reduces the inventory of the raw materials used in the product in the consumed amount.

The most basic difference of this model is that it’s association with machine model was performed not directly with the product card, but with machine operations used in the product card. The main reason for this is that a product can contain more than one sub part. Each sub product may contain different operations. Therefore, a special area was prepared in the product card so that each sub part could be associated directly with machines. Machine card calculates an average machine operation time for every manufactured product and transmits it to the machine card.

Ability to Self-organize: In order to provide the model to have self-organizing property; primarily, the business rules, which will ensure information flow for the model, were created. In order to ensure information flow, an information flow map between product agent and system was designed (see Figure 4.16).

Product agent uses the following business rules to operate.

- It gathers Company’s products information from Product Cards
- It gathers raw material specifications from Supplier Cards
- It measures the new Product success level by the using defined KPI calculations
- It gives an ability of measure the manufacturing times and selection of available appropriate raw material by the using Product Card
- It controls manufacturing routs of the new Product by using Product Card
- It composes bill of materials of the new Product
- It satisfies observation and intervention of the decrease in performance of manufacturing cycle times
- It satisfy the new Product integration to related Machines
- It satisfy the new Product integration to related materials and/or raw materials
- It defines manufacturing specifications (weight, height, color etc.)
- It defines needs of raw materials/materials for new Product
- It controls sub product operation routes and quantities
- For each new order, model controls inventory, calculates needs of Raw material and send this information to Supplier Model for the purchasing order, also send this information to machine model to prepare capacity planning
- For Each manufactured order control manufactured units and gives manufacturing approval
- For each product, Model controls shelf life and prepares a report for the upcoming out of date manufacturing Product list

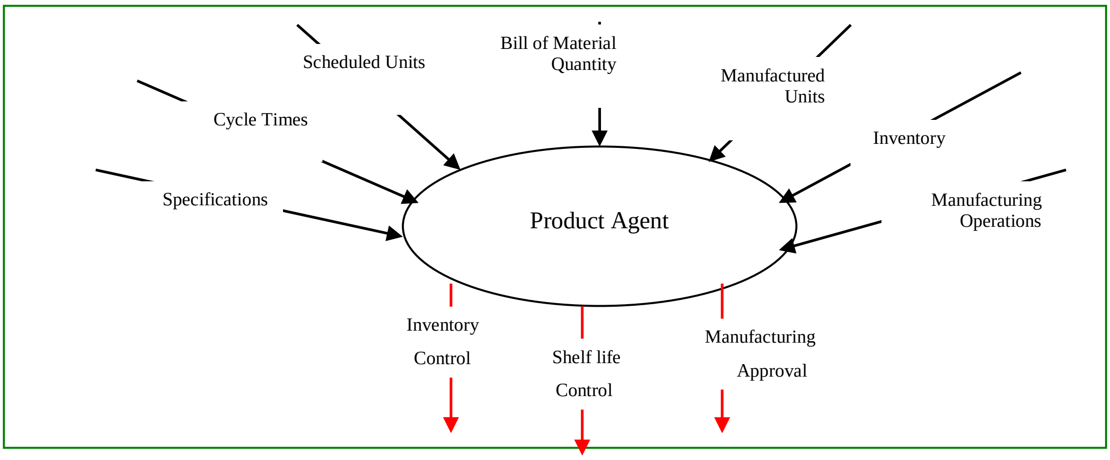

Figure 4.16. Information flow of Product Model Agent

4.2.6. Supplier Model (Supplier Agent)

Primary task of this model is to create relationship between products and the raw materials in a certain manufacturing system and support the assessment with respect to whether those manufacturing specifications are satisfied or not.

Main motivations of the Supplier Model are listed below;

1. ability to **integrate** (Specific Purpose - integrating the raw materials to the related suppliers)
2. ability to **measure** (Specific Purpose – raw materials used for the manufacturing of the products)
3. ability to **adapt** (Social Ability - ability to adapt over time in response to changes in the environment of the Supplier Model)
4. ability to **self-organize** (Reactiveness - ability to establish basic rules for the interactions between the Supplier model and the system)
5. ability to **control** (Autonomy - ability to control of supplier selection mechanism of the system to purchase needed raw materials)

Ability to integrate: In this model, the raw materials used for manufacturing the products were associated with suppliers, and thus, the integration between the raw material and the supplier was supplied. In order to ensure this integration, a “Supplier

Card” was designed for each supplier (see Table 4.7).

Each Supplier is defined in terms of the following attributes;

Supplier Name: to quick identify Supplier

Card Code: to give unique number to Supplier

Basic Information (Adress, City, County.. etc): To record contact information of the new Supplier

Material Code: To identify quick identify of Supplier product range

Material Name: To give an explanation of the Supplier products

Measurement Parameters: To state supplier price and payment term. Also depends on previous purchasing determine a quality value

Technical parameter: To define detail information of supplier product range (width, height, etc)

Shipment Type: To selection appropriate purchasing order

Delivery Time: To prevent purchasing delay

Total Capacity: To identify new Supplier capacity

Used Capacity: To control available purchasing capacity of the new Supplier

KPI Measurement: to measure new Supplier efficiency and measure success values

% Level of significance: The contribution of the related material and or raw material for the new Supplier success

% Success Level: To calculate the material based success level by making calculation

Total Success: To calculate new Supplier Success Value

Table 4.7. The Structure “Supplier Card”

<table>
<tr><td colspan="20">SUPPLIER CARD</td></tr>
<tr><td>Supplier Name</td><td colspan="14"></td><td>No :</td><td colspan="4">00001-S1</td></tr>
<tr><td>Card Code:</td><td colspan="14"></td><td>Dat e:</td><td colspan="4"></td></tr>
<tr><td>Address</td><td colspan="5"></td><td colspan="4">City:</td><td colspan="3"></td><td colspan="3">Country</td><td colspan="4"></td></tr>
<tr><td>Contact Person</td><td colspan="5"></td><td colspan="4">Contact Phone:</td><td colspan="3"></td><td colspan="3">Email</td><td colspan="4"></td></tr>
<tr><td colspan="20"></td></tr>
<tr><td rowspan="2">Material Code</td><td rowspan="2">Material Name</td><td rowspan="2">Quality Test Code</td><td colspan="3">Measurement Parameters</td><td colspan="4">Technical Parameters</td><td rowspan="2">Shipment Type</td><td rowspan="2">Delivery Time / Lot</td><td rowspan="2">Total Capacity / TMTH</td><td rowspan="2">Used Capacity / TMTH</td><td rowspan="2">Available Capacity / TMTH</td><td colspan="3"></td><td rowspan="2">% Level of significance</td><td rowspan="2">Success Value</td></tr>
<tr><td>Price / Lot (TL)</td><td>Quality Value</td><td>Payment Term</td><td>Width</td><td>Height</td><td>Thickness</td><td>Color</td><td>KPI Code</td><td>Planned KPI Value</td><td>Actual KPI Value</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="2">Total Success Value</td><td colspan="17"></td><td></td></tr>
<tr><td>Prepared by</td><td colspan="3"></td><td colspan="5">Sign</td><td colspan="11"></td></tr>
<tr><td>Approved by</td><td colspan="3"></td><td colspan="5">Sign</td><td colspan="11"></td></tr>
</table>

Supplier Card is made of the first main field including main information on the supplier (name, communication, location, etc.), and underneath this field, the second main field containing information on raw material and materials that could be purchased from the Supplier. After information on the supplier is entered, the first column opened in the second field contains raw material and material that could be purchased under the title “Material Code”, and an explanation on the product is included in the second column which is the “Material Name” column. Third column is generated under the title “Test Code” for recording quality control test that is defined for each purchased product (T1: Stretch break resistance, etc.)

The fourth column is opened under the title “Measurement Parameter” and is divided into three sections within itself. These are the most fundamental information on the products namely Price, Quality Value and Payment Term data. Price and Payment Term are data that are determined directly by the supplier, and the Quality Value is a result of the average measurement coming up as a result of the purchases made form the company previously. This measurement is discussed in detail within the “Procurement Agent” working principles below. The fifth column is made under the title “Technical Parameters” for providing recording of technical information on raw material and equipment and is divided into 4 different fields. These are; Width, Height, Thickness, and Color.

The sixth column is generated under the title “Shipment Type” for defining product delivery type (for example; ST1 shipment by truck). The seventh column is generated under the title “Delivery Date” for recording information regarding the number of days to pass for the product to reach the company after placing the order. The eighth column is added under the title “Total Capacity” for recording the capacity of the supplier about the relevant raw material and supplies and if this capacity is used previously, “Used Capacity” field is formed in the ninth column to keep this amount. The 10^{th} column “Available Capacity” is for taking the difference of the current capacity and its usage and keeping it.

The 11^{th} column has the title “KPI Measurement” providing recording of the measurement and evaluation of the supplier. Calculations made in this field are explained in detail below under the title “Procurement Agent”. Average of the sequencing of PQPT values made according to other suppliers for each raw material/supplies in the portfolio of the supplier (selection criteria are explained in detail in the Business Rules section) made up the average success percentage of the supplier.

The Structural frame code of the Supplier Card definition function $f^{PrD}_{p,ts,t}$ is given below;

```
public class Supplier {
	public void addSupplierCard(String name, String code, ContactInformation supplier,
List<RawMaterial> materials,
				List<PQPT> pqpt, List<SupplierParameter> supplierParameters) {
		SupplierCard suppliercard = new SupplierCard(name, code, supplier, materials,
pqpt, supplierParameters);
		this.saveSupplierCard(suppliercard);
	}

	private void saveSupplierCard(SupplierCard suppliercard) {

	}
}
public class SupplierCard {
	public String Name;
	public String Code;
	public ContactInformation Supplier;
	public List<RawMaterial> Materials;
	public List<PQPT> PQPT;
	public List<SupplierParameter> supplierParameters;
	public SupplierCard(String name2, String code2, ContactInformation supplier2,
List<RawMaterial> materials2,
				List<PQPT> pqpt2, List<SupplierParameter> supplierParameters2) {
	}

	public float calculatePQPTbyRM(RawMaterial rawMaterial) {
		float pqpt;
		// calculate pqpt
		return pqpt;
	}
}
public class SupplierParameter {
	public RawMaterial RawMaterial;
	public String Shipment;
	public float DeliveryTime;
	public float TotalCapacity;
	public float UsedCapacity;
	public float calculateAvailable() {

	}
}
```

Ability to Measure:

In order to provide the Supplier model to have measurement ability, primarily Supplier Agent Architecture was designed. In order to define the working principles of the model, the business rules were defined so as to provide it to have adaptation ability. A mathematical model, which will be used in measuring the performance of the Supplier model, was designed. Afterwards, it was provided to have measurement ability on the software environment through coding these business rules and the mathematical model. Now, these studies will be explained respectively.

Supplier Model Agent Architecture: Supplier model consist of four agents, namely; Material Requirement Agent, Procurement Agent, Buyer Agent, and Order Organizer Agent as defined below (See Figure 4.17).

1. **Order Organizer Agent:** This agent receives list of out of stock raw materials and or materials for each ordered product from Product Model and groups them depends on similarities and sends this information to material requirement agent for material requirement planning.
2. **Material requirement agent:** This agent calculate material requirement planning for each material need comes from order organizer agent and master production planning agent and prepares a purchase order for to send procurement agent
3. **Procurement agent:** This agent searches possible suppliers for each non-exist units of raw materials and select one of them to start; eliminates the suppliers which could not deliver on time. These available suppliers has positive idle time value (this time is calculate as; (Order date- (processing time + delivery time). The main objective of this step is avoiding the manufacturing delay come from shortcoming goods. If none of supplier is not able to supply the agent then sends a message to user to search for a new supplier

   If there is an available supplier then the agent starts a selection process by;

   i. Calculating Quality value of each available supplier depend on 3 main criteria; Correct Raw material percentage, Correct Received unit percentage and correct shipment percentage. Take these percentage values of supplier from Quality agent and calculate average quality value of each supplier

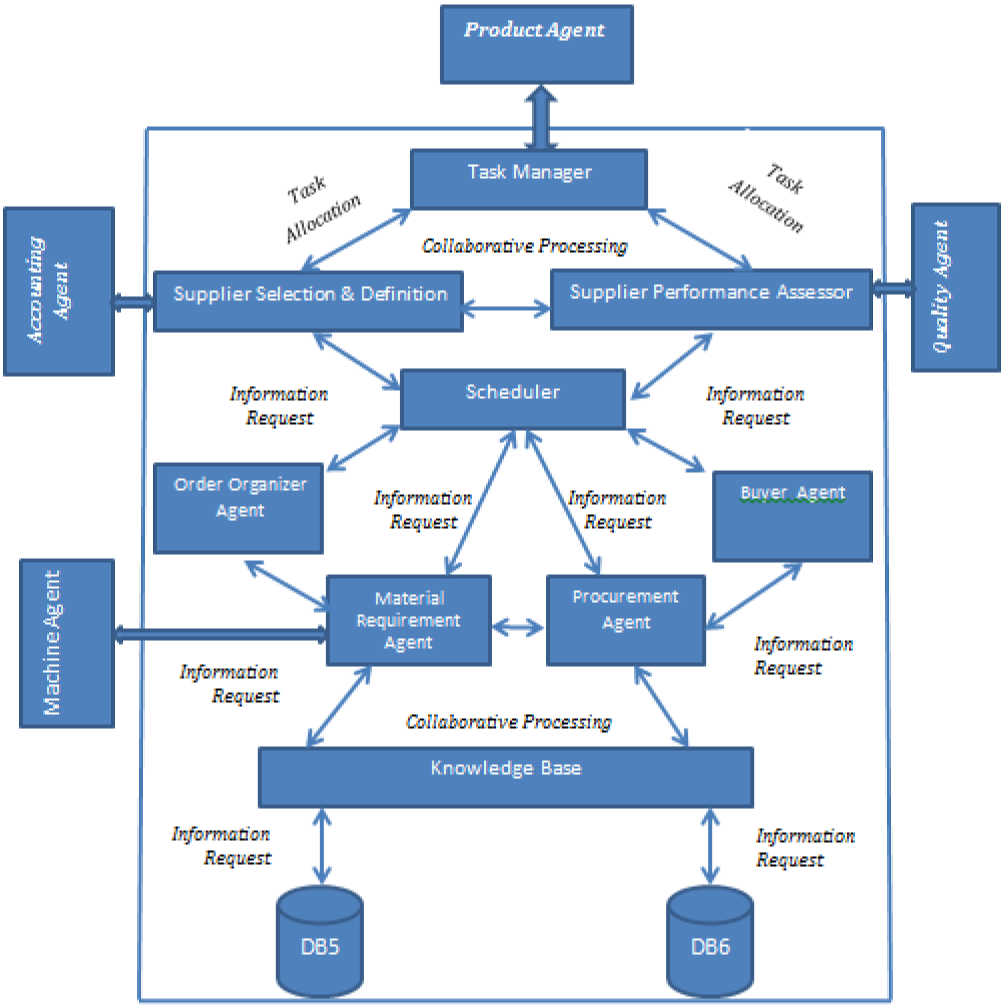

Figure 4.17. Agent architecture of proposed Supplier Model

ii. Gathering supplier price and supplier payment term information from Accounting agent

iii. Calculating each supplier utility value use AHP (analytical Hierarchy Process) decision making method (AHP is the most appropriate method for Companies to determination of priority and grading of suppliers depends on their price, quality and payment term values) and record them in information system (see Table 4.8). Give an importance value for price, quality and payment value then normalize the matrices (depends on AHP rules, see Taha, 2007) to get utility value of the each supplier for needed material.

iv. For the each available suppliers use these values to calculate supplier PQPT value and select the Supplier which has maximum utility value

An example calculation is as follows:

P: Payment Q: Quality and PT: Payment Term

Table 4.8. AHP matrix of price (p), quality (q) and payment term (pt)

	P	Q	PT
P	1	4	3
Q	0,25	1,00	3,00
PT	0,33	0,33	1,00

Table values are assumed. Expert can change these values. Then calculate normalized matrix and gathered weights;

Table 4.9. AHP normalized matrix

	P	q	Pt	W
p	0,63	0,75	0,43	0,60
q	0,16	0,19	0,43	0,26
pt	0,21	0,06	0,14	0,14

For the calculation tecqnique of normalized matrix see (Taha, 2007)

Supplier Utulity function is as given;

$SUF_q=1/P_q^r * w_1 + Q_q^r *w_2 + PT_q *w_3$

Where,

SUF denotes the utility function of supplier q

P_q^r represents the unit price of raw material r of supplier q

Q_q^r denotes the quality value for raw material r of supplier q

PT_q is payment term of supplier q

The agent also calculate opportunity cost of each available supplier

a. If all supplier Payment terms are equal no need for holding cost calculation
b. For the different payment terms then the agent calculate opportunity costs as;

$$\text{Opportunity cost of rawmaterial i of supplier j}:$$

Unit Price * Total needs* $(1+\frac{\text{Interes rate}}{100})^{\text{t}})$

ii. Select most appropriate supplier for the Purchasing

Once all costs are calculated then the agent sends the cost report and supplier performance report to Procurement Agent

4. **Buyer agent;** This agent receives the supplier cost report and supplier performance report from procurement agent than combines them as a list prepares an purchase order form to send most available suppliers then selects them depends on maximum utility value which is described above.

Definition of the Operations Mathematical Model is explained below

f(SA) $=\{Pt_q^S, A_q^S, Q_{q,dr}^S, \mathrm{f}_{\mathrm{q,ts,t}}^{\mathrm{SD}}, \mathrm{f}_{\mathrm{q,ts,t}}^{\mathrm{SPm}}, \mathrm{SSS_q}\ , \in_q^S\}$ denotes the function set of the Supplier Model

where

- Pt_q^S set of percept of supplier q,
- A_q^S set of actions of supplier q
- $Q_{q,dr}^S$ set of defect rate dr of supplier q
- $\mathrm{f}_{\mathrm{q,ts,t}}^{\mathrm{SD}}$ denotes the Supplier card definition function task ts of supplier q, at time t,
- $\mathrm{f}_{\mathrm{q,ts,t}}^{\mathrm{SPm}}$ is performance measurement function task ts of supplier q , at time t,
- $, \mathrm{SSS_q}$ denotes the system states of the Supplier Agent and
- $\in_q^S$defines set of environmental states of the Supplier Agent

Mathematical formulation of the Supplier model success rate is pre-defined in the model as given in Equation 22.

Max $\boldsymbol{Z_s}$ **=** $\boldsymbol{avg}(\sum_{\boldsymbol{q,rm}}^{\boldsymbol{S}}(\boldsymbol{RM_{q,rm,pqpt}}))$ **(22)**

According to Equation 22, $\boldsymbol{Z_s}$ is the total success value of the supplier q. $\boldsymbol{RM_{rm,pqpt}}$ is the PQPT vale of raw material rm. And average of them is assumed as success rate of in Supplier q

Structural frame code of Supplier success measurement; Structural frame code of Machine measurement function $f^{SPm}_{q,ts,t}$ is designed as follows;

```
public float calculatePQPTbyRM(RawMaterial rawMaterial) {
                float pqpt;
                // calculate pqpt
                return pqpt;

}

public class PQPT {
        public RawMaterial RawMaterial;
        public float Price;
        public float QualityValue;
        public float PaymentTerm;

        public float calculate() {

        }
}
```

Ability to adapt: An example adaptation rule of the Supplier Agent is pre-defined as follows;

```
//Supplier selection
public SupplierCard supplierSelection(RawMaterial rm){
        SupplierCard supplier;
        float pqpt = 0;
        for(SupplierCard sc : SupplierCardList){
                if(sc.Materials.contains(rm)){
                        if(sc.calculatePQPTbyRM(rm) > pqpt){
                        supplier = sc;
                        }
                }
        }
        return supplier;
}
```

After every production process, when the stock level of any raw material being used gets below the minimum stock level, the process of supplier selection is started. In this case, firstly, it is investigated that from which supplier that raw material could be supplied. Then, it lists the supplier card of the supplier, of which PQPT value is maximum among all of them.

Ability to Self-organize: In order to provide the model to have self-organizing property; primarily, the business rules, which will ensure information flow for the model, were created. Agent precedence order in the model was determined (see Figure 4.15). These information flows were described in detail under the headline of ability to measurement in the section of Agent architecture. In order to ensure information flow, an information flow was designed (see Figure 4.16).

Some of Supplier Agent business rules are listed below.

- It gathers Supplier's products information from Supplier Cards
- It gathers quality rate of the supplier from the Quality Agent
- It gives an ability of measure the Shipment times and selection of available appropriate Supplier by the using Supplier Card for the each new order
- It prevents delay from purchasing order
- It composes material ranges of the new Supplier
- It satisfies observation and intervention of the decrease in performance of the supplier
- It satisfy raw material and or material integration with Supplier
- It calculates Supplier PQPT (price, quality and payment) value of each Supplier (read Procurement agent working principle)
- It calculates opportunity cost for each purchasing order (read Procurement agent working principle)
- It selects most appropriate supplier depends on PQPT or opportunity cost calculation

When the Operation starts model agents use precedence rule given in Figure 4.18

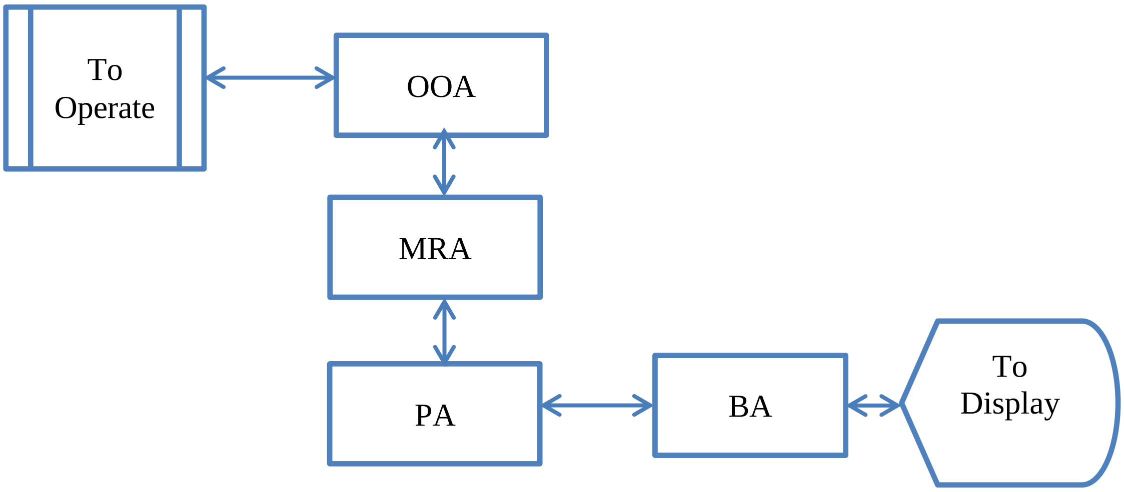

Figure 4.18. Supplier Model Agents Precedence Structure

Similarly information flow map for supplier agent is shown in Figure 4.19

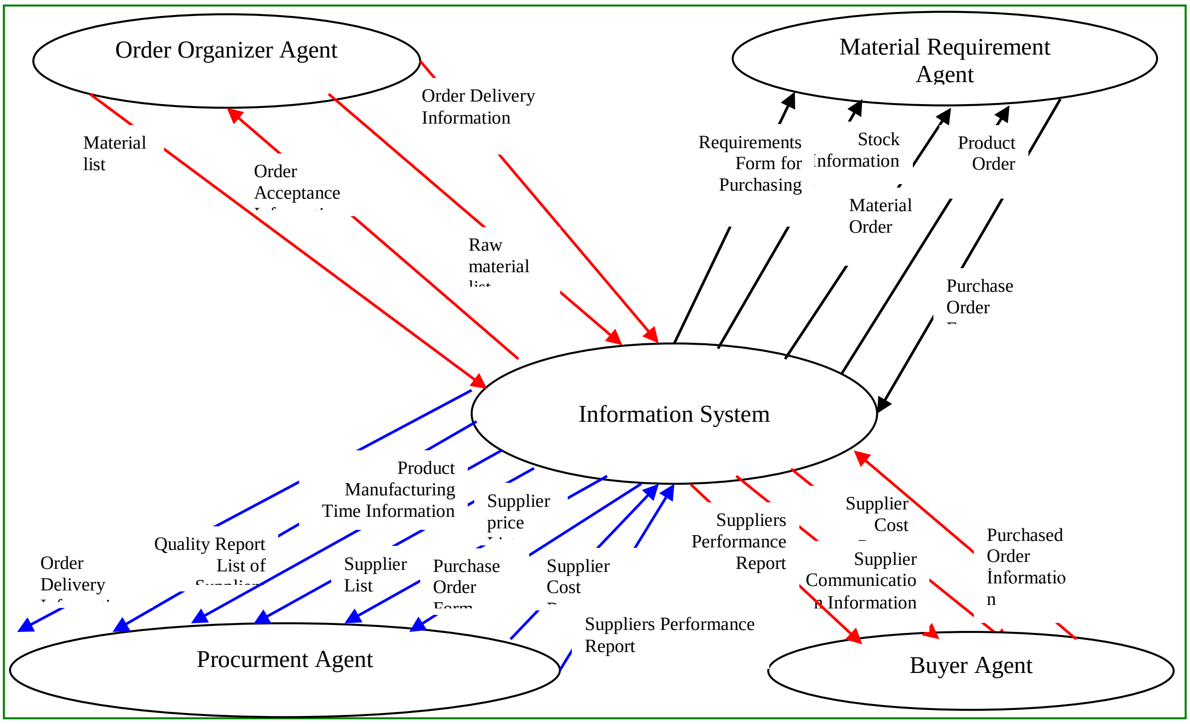

Figure 4.19 Supply Model Agents Information exchange architecture

In this section, the detailed information relating to the proposed model was realized. The application results of the model will be explained in the next section.

5. IMPLEMENTATION OF PROPOSED MODEL (IMPACT)

In this chapter, a prototype production planning model using the proposed agent based system architecture is introduced for the sake of a proof of concept in MAS (multi agent systems) applications of manufacturing systems. Implementation program consist of five main stages:

1. Defining the application domain for prototyping.
2. Composing the respective knowledge and collecting the related data
3. Defining the architecture of respective manufacturing agents as well as respective functions and populating their knowledgebase.
4. Setting up the model.
5. Develop the respective software and implement the proposed model.
6. Doing the tests by real production data and reporting the results
7. Testing the model in the simulation setting and reporting the results

A dynamic agent based structure is proposed in order to assure that the change in any input data affects all layers of model simultaneously aiming to eliminate the production loss, increase productivity, enable more efficient schedule management, assure the control of supplier selection, increase efficiency of machines as well as coordination of the departments etc.

5.1. Real Life Application Domain and the Production Suit

The prototype application is carried out over Cabinet Production System. The prototype is implemented in a real manufacturing company located in İstanbul. Due to the nature of the information provided and the competitive clause, the name of the company is intentionally not provided and named as XYZ.

Company XYZ was established in 1965 in Istanbul to produce cabinet and technical installation equipment. IT and Telecom Infrastructure technology offerings that span from complete data centers, data center green cooling technologies to fiber optic connectivity based FTTX and fiber optic transmission and access solutions. 80% of

Company production is exported to Europe, Russia, Middle East, North & Sub-Saharan Africa, USA, CIS and Central America.

Company product range consists of 4 basic production types which are;

1. **Fiber Optical Products**
    - High density optical distribution frame (ODF)
    - Fiber termination modules
    - Fiber outdoor cabinet
    - Optical distribution boxes and FTTH boxes
    - 19” fiber patch panels, pigtails & adaptors
2. **Fiber Optic Raceway/Ducting**
    - Cooling Solutions (Water & DX based)
    - IP PDU
    - Modular UPS
    - Environmental Monitoring System
    - Aisle Containment System
3. **Mobile Container Datacenter**

    Indoor 19” Racks & Cabinets
    - Aluminum frame floor standing racks
    - Steel frame floor standing racks
    - Wall-mount cabinets
    - ETSI cabinets
    - Open & Relay racks
4. **Outdoor Cabinets**
    - Double wall aluminum based IP65 DSLAM outdoor
    - Double wall galvanized steel based IP65 DSLAM outdoor cabinet

- IP65 galvanized steel based fiber optic outdoor Cabinets

In this study **Outdoor Cabinets are selected for the proposed model implementation** An example of the outdoor Cabinet is given in Figure 5.1.

Figure 5.1. Street cabinet

In order to generate the proposed agent based model, manufacturing processes and operations are inspected and business rules are gathered from the Company experts.

5.1.1. Company Departments for the Case Study

The manufacturing functions that are included in the model scope in this section of the study are examined in detail. The responsibilities of the production functions are defined below.

Planning, manufacturing, stock management and Procurement departments and activities in each department selected to implement proposed model. Each department main responsibilities are explained below.

Planning Department is mainly responsible to management of operations which is start with doing master production scheduling and ends with job order to manufacturing. Planning manager must supervise procedures for making work efficient. Considered planning processes are listed below;

Manufacturing department is responsible for turning raw materials into finished products through a series of production operations. Production manager is responsible for making sure that raw materials are provided and made into finished goods effectively

Considered Company Manufacturing Operations are listed below;

1. ***Punching operation:*** it is a kind of sharing operation. Punching operation can be defined as desired parts left out after making a punch hole. (See Figure 5.2.)

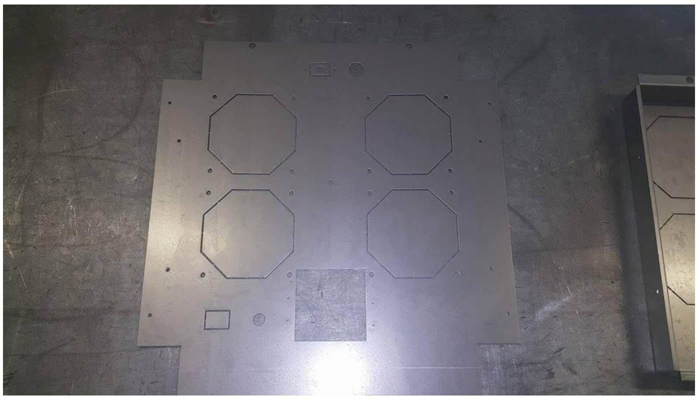

Figure 5.2. An example sub-product after punching operation

2. ***Bending operation:*** Bending changes the shapes of the product while the volume of material will remain the same. (see Figure 5.3.)

Figure 5.3. An example sub-product after Bending operation

3. ***Metal Burr operation:*** A burr is a raised edge or small piece of material remaining attached to a part after punching or bending operation see (Figure 5.4.)

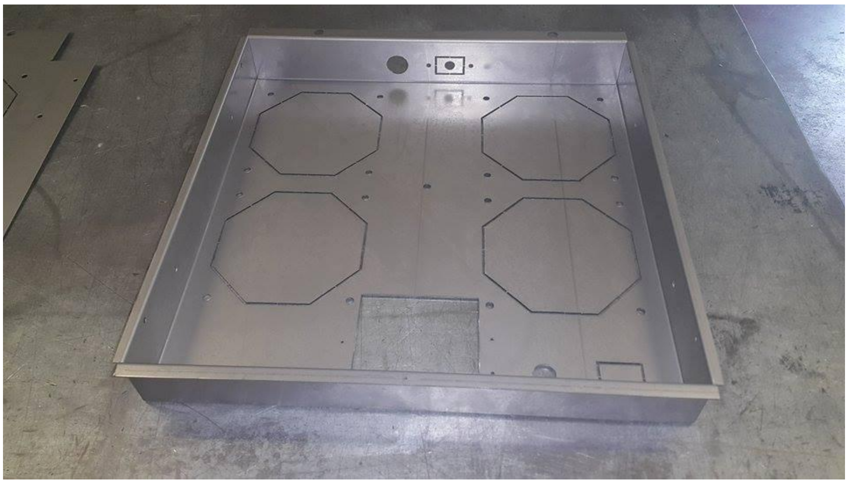

Figure 5.4. An example sub-product after the Metal burr operation

4. ***Tig welding operation:*** TIG (Tungsten Inert Gas) operation is used to weld two work pieces together (see Figure 5.5.).

Figure 5.5. An example sub-product after TIG welding operation

Stock department mainly responsible to management of operations that start with storage materials ends shipment materials to customer. Stock manager responsible to control on hand inventory levels to satisfy continuous manufacturing. Classification of the Company materials is listed below;

a. ***Raw Materials:*** These are the resources required in the production or processing activity of the firm.

b. ***Components:*** Components correspond to items that have not yet reached completion in the production process. Components are sometimes referred to as subassemblies.

c. ***Work-in-Process:*** Work in Process (WIP) is inventorying either waiting in the system for processing or being processed.

d. ***Finished Goods:*** Also known as end items, these are the final products of the production process.

Procurement department mainly responsible to management of procurement operations that start with material requirements and ends with purchasing necessary materials. Procurement manager responsible for buying the best quality raw materials, market analysis for the low price and satisfy on time procurement for manufacturing.

After collecting information about the responsibilities of production and production functions, all stages of the production process and business rules that are used in these stages have been formed. All operations from the arrival of the order until the product delivery are shown in Figure 5.6.

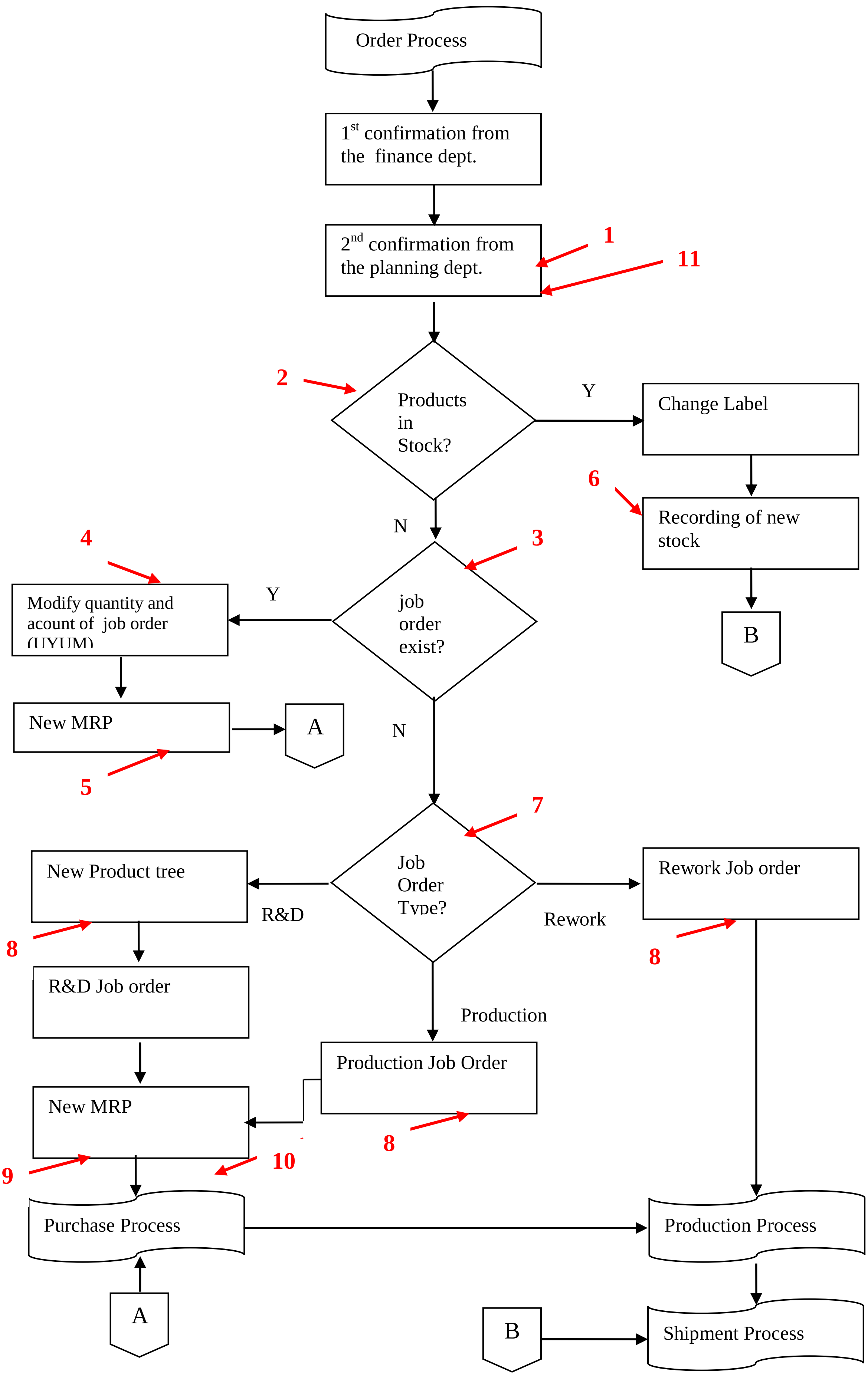

Figure 5.6. Current workflow of planning department

Sales Representative receives and records the orders into MRP program, and then first confirmation is granted by financial management department. Which checks out the financial record of the customer putting the order? Following this, the planning department receives the order receipt and checks the delivery time and quantity of the order. If it is possible to deliver the required number of products in a given time, the planner confirms the order and the date of delivery. The planner, finalize scheduled dates depending on her previous experiences. Currently, she doesn't use any systematic model to calculate the planning time horizon but relies only her experiences.

The meaning of the numbers provided in red in Figure 5.5 indicates various activities as listed in Table 5.1.

Table 5.1. Production planning activities and respective methods in XYZ.

	Planning Activities	**Methods Employed**
1	Order confirmation	Manual confirmation through MRP system
2	Stock control	Manual control through MRP system
3	Open job order control	Manual checking of open job orders through MRP system
4	Order quantity change	Manual change through MRP system
5	MRP run	Manual parameter supply to machine performing MRP activities
6	Recording of stocks	Manual recording
7	Order type decision	Human decision making
8	Preparing Job Order	Manual preparation through MRP system
9	Purchase List	Automated List generation but manual control
10	MPS	Manual Planning using Excel
11	Delivery Date	Manual planning based on self-experience

Main aim of IMPACT is to eliminate uncontrolled flow of raw materials, unscheduled manufacturing activities, and unscheduled delivery date caused by unreliable data, as well as the lack of methods and undefined procedures to be implemented. In order to be able to control production operations, the data related to the operations have to be reliable and accessible. One of the acceptable ways of increasing and controlling manufacturing efficiency is to well-define the system components with respective functions and to monitor those on predefined time frame. In this study it is intended to provide such control and monitoring functions through implementing intelligent agent-based technology. As indicated in the literature review, this technology is proven to be capable of satisfying manufacturing functionalities in variety of ways.

The proposed model is prototyped for highlighting the difference between current working performance and agent based approach.

For developing the prototype, the software program is coded using java programming language for implementing for the cooperation and coordination between the agents. An agent based architecture is developed in such a way that each process with its integrated information flow is designed depending upon the opinion of a responsible experts who actually works in the manufacturing suits.

5.2. Data Collection

To implement the proposed model, manufacturing system is monitored and respective data is collected throughout six month time period. 2 main manufacturing goals, related processes, operations, machines, sample products and suppliers were defined and integrated.

At this stage, first of all, sample products were selected and detailed information about these products were collected. Subsequently, based on the obtained data, model cards were developed. Now, these studies will be discussed in order.

5.2.1. Time and Motion Study

2 different types of products are selected to implementation of the purposed model which are codded as respectively 069007Q00 and 48417250GGB-24, Original names of the given products are not used due to commercial privacy. For these 2 products bill of material list is gathered and time-and-motion studies were conducted for each machine operation. Detail information of time study is given in Table 5.2.

- ✓ First Column provides the product id
- ✓ Second Column indicates part id (lower segment unit) of main product,

- ✓ Third Column shows required number of unit of Part in one product.
- ✓ Fourth Column depicts raw material type of Part.(for the list of Raw material see Table 5.9)
- ✓ Fifth Column to Ninth Column shows needed Mold Operations type of sub-product
- ✓ Tenth Column shows needed Laser operations of sub-product
- ✓ Eleventh Column to Fifteenth Column displays needed manufacturing details of sub-product for PNC operations respectively Number of production of each time, Cycle time (min.), Process time of one product, Lost Time and Setup Time (min.)
- ✓ Sixteenth Column indicates Total needed time for PNC manufacturing of sub-product.
- ✓ Seventeenth Column and Eighteenth Column includes BND operation types
- ✓ Nineteenth Column and Twentieth Column displays needed Cycle time (min) and Setup Time (min) of sub-product for BND operations
- ✓ Twenty first Column indicates Total needed time for BND manufacturing of sub-product.
- ✓ Twenty-second Column shows JET operation code
- ✓ Twenty-third Column includes needed Cycle time (min) of sub-product for JET operations

Table 5.2. A Part of Time study table

				PUNCH OPERATION												BND Operations					Welding Operations					
				PNC Operations					PLM	Manufacturing Times						BND Operations		Manufacturing Times			JET operations		TM Operations			
Product ID	Lower Segment Product ID	BOM UNIT	Rw Material Type	Mould ID					LASER Operation	Number of production of each time	Cycle Time (min.)	Process time of one product	Lost Time	Setup Time (min.)	Total Time (min.)	Underframe	Upperframe	Cycle Time (min.)	Setup Time (min.)	Total Time (min.)	Operation Code	Cycle Time (min.)	Operaiton Code	Cycle Time (min.)	Total Time (min.)
069007Q00	060400P00	1	RSMG6	MD5						11	0,7	0,7	0,8	8,0	17,5	BND10	BND31	0,4	20	20,4	J1	0,5	K1	1,33	1,8
	060401P00	1	RSMG6	MD104	MD109	MD113	MD65	MD72		2	2,8	2,8	0,8	3,3	7,7	BND14	BND29	0,8	20	20,8	J1	2,0	K1	1,33	3,3
	060402P00	2	RSMG6	MD65	MD38					8	0,2	0,5	0,8	1,3	8,2	BND5	BND25	0,4	20	20,4			K1	1,33	1,3
	060403P00	2	RSMG6	MD37	MD38					3	0,3	0,5	0,8	1,3	4,2	BND10	BND27	0,4	20	20,4			K1	1,33	1,3
	060405P00	1	RSMG4	MD39						54	0,1	0,1	0,8	0,7	44,0	BND6	BND26	0,4	20	20,4					
	060407P00	1	RSMG4	MD104	MD73	MD34	MD38			3	2,4	2,4	0,8	2,7	7,5	BND1	BND28	1,6	20	21,6			K1	3,00	3,0
	060408P00	2	RSMG4	MD88	MD100	MD103	MD30	MD34		36	0,3	0,6	0,8	3,3	32,7	BND2	BND24	0,8	20	20,8					
	060410P00	9	RSMG6	MD97	MD117	MD36	MD37			125	0,1	0,5	0,8	10,0	110,5						J1	0,2	K1	1,33	1,5
	060412P00	1	RSMG6	MD10						2	1,3	1,3	0,8	8,0	10,9	BND16	BND29	1,8	20	21,8	J1	2,0	K1	1,33	3,3
	060523P00	5	RSMG2						YES		0,1	0,3	0,8	5,0	5,3										
	060525P00	1	RSMG2						YES		0,1	0,1	0,8	5,0	5,1										
	060525P00	1	RSMG2						YES		0,1	0,1	0,8	5,0	5,1										

- ✓ Twenty fourth Column displays TM operation code
- ✓ Twenty fifth Column respect to Cycle time (min) of sub-product for TM operations
- ✓ And Last Column is show that Total needed time for Welding manufacturing of sub-product.

When the table above is examined; the first lower segment of the product 069007Q00 is 0604P00. This lower segment is present in the main product as 1 unit (BOM unit). The raw material that is used for its production is RSMG 6 (the amount of the raw material that is used in the production of this lower segment is estimated separately). The sub product production needs only a punching operation. The code of this machine operation is MD5. A punch machine can produce 11 units from this lower segment in a single time. The unit production time is 0.7 minutes. The time that is formed in the production preparation time is 0.8 minutes. The preparation time that is necessary for product production is 8 minutes. The total time needed for the completion of the production of this product in Punch operation is 17.5 minutes. After the completion of product punching operation, bending operation starts.

For this purpose, BND10 and BND31 operations are needed. A bending operation for a sub product lasts 0.4 minutes. The preparation period for bending operation is 20 minutes. The company manufactures by molds, therefore the preparation times are very high. The total operation time for bending operation of the lower segment is obtained as 20.4 minutes. The sub product is subjected to metal burr operation after the bending operation. This operation is a manual process. There is no preparation time and the total operation time is 0.5 minutes. After the metal burr operation, K1 machine operation is applied to the product finally in the scope of Tig operation. The unit period of this operation is 1.33 minutes. The set up time for this operation is constant and 0.5 minutes. The total time that is needed for Tig operation of the product is estimated as 1.83 minutes.

In the given table, the data belonging to some operations for some low segments are not included. The reason for this is that there is no need for the usage of those operations in the production of the lower segment. After this point, a list of the machines used in production was formed according to the operation types. This list is given in Table 5.3.

Table 5.3. Machine list of the Mechanic Manufacturing

Machine Code	Machine Name
PNC1	Punch Machine 1
PNC2	Punch Machine 2
PNC3	Punch Machine 3
PNC4	Punch Machine 4
PNC5	Punch Laser Machine 5
BND1	Bend Machine 1
BND2	Bend Machine 2
BND3	Bend Machine 3
BND4	Bend Machine 4
BND5	Bend Machine 5
TIG1	Tig Machine 1

After the machine data are obtained, a list of the machine operations was formed. The lower segments used in the product go through various machine operations in the machines. Maximum 8 machine operations can be carried out simultaneously in a cutting machine (for PNC1 to PNC4). There is not such a limitation in a laser machine. A desired number of machine operations among machine tolerances can be applied to a lower segment. 128 different machine operations are defined in the company. 50 of these are shown in the below given Table 5.4. Punching machine operations are defined by MD code in the data base.

Bending operation can be carried out by one lower and one upper mold. The lower segments can either use the same bending mold or they may need different molds. These molds require large and detailed adjustment therefore their preparation times last a long time namely 20 minutes. A list of bending machine operations that could be applied in the Company structure is shown in the below given Table 5.5. The reason for the low number of operations in the list is that an operation mold is appropriate for usage for more than one product. Bending machine operations are coded by BND code in software.

Table 5.4. Mold list of Punch Machines

Machine Code	Machine Name	Related Machine operation Code	Machine Operation Name	Operation setup Time
PNC1-PNC2-PNC3-PNC4	All Punch Machines	MD1	3*15 cross over	480
		MD2	DD205 20.5x17.5	480
		MD3	F_079	480
		MD4	F_087	480
		MD5	F078	480
		MD6	F089	480
		MD7	F090	480
		MD8	Formi 041	480
		MD9	Formi 064	480
		MD10	Formi 56.0x5.0	480
		MD11	Langloch 10x7	40
		MD12	Langloch 12x6	40
		MD13	Langloch 20x5	40
		MD14	Oblon 10.0x5.0	40
		MD15	Oblon 11.2x6.2	40
		MD16	Oblon 14.5x6.5	40
		MD17	Oblon 15.0x4.0	40
		MD18	Oblon 15.0x7.5	40
		MD19	Oblon 15.0x8.5	40
		MD20	Oblon 15.5x8.5	40
		MD21	Oblon 16.5x4.5	40
		MD22	Oblon 8.0x5.5	40
		MD23	Oblon 8.3x4.3	40
		MD24	Quadrat 10	40
		MD25	Quadrat 11	40
		MD26	Quadrat 20	40
		MD27	Quadrat 30	40
		MD28	Quadrat 4	40
		MD29	Quadrat 40	40
		MD30	Quadrat 5	40
		MD31	Quadrat 50	40
		MD32	Quadrat 6	40
		MD33	Quadrat 7	40
		MD34	Quadrat 9.5	40
		MD35	Rechteck 10x2	40
		MD36	Rechteck 15x5	40
		MD37	Rechteck 20x5	40

Table 5.5. A part of Bend Mechanism List

Machine Family Code	Operation Code	Operation Name (CNC BEND)	Operation Time (min.)
BND	BND1	BIU-011; 805 mm bending	20
	BND2	BIU-011;100 mm bending	20
	BND3	BIU-011;1300 mm bending	20
	BND4	BIU-011;1410 mm bending	20
	BND5	BIU-011;200 mm bending	20
	BND6	BIU-011;400 mm bending	20
	BND7	BIU-011;425 mm bending	20
	BND8	BIU-011;460 mm bending	20
	BND9	BIU-011;485 mm bending	20
	BND10	BIU-011;500 mm bending	20
	BND11	BIU-011;600 mm bending	20
	BND12	BIU-011;800 mm bending	20
	BND13	BIU-011;900 mm bending	20
	BND14	BIU-011;955 mm bending	20
	BND15	BIU-013;100 mm bending	20
	BND16	BIU-013;1000 mm bending	20
	BND17	BIU-013;1315 mm bending	20

After the data belonging to bending machine operations are obtained, the number of different ways that metal burr operations are carried out was examined. It was observed that this operation has a structure in two different hand looms and non-standard. The reason for this is that some of the byproducts that come out of the machines require little correction and some of them require more correction than the others. This operation is coded to the software by JO code (see Table 5.6.).

Table 5.6. Jet Operation

Machine	Operation Code	Operation Name
J1- J2	JO1	Trimming

When TIG Welding operation, which is the final stage of the mechanical production, is examined, it is observed that this operation is applied to the lower segments in 2 different ways (see Table 5.7). TIG machine is an autonomous machine therefore its preparation period is constant and is 0.5 minutes in the average. Tig operations is defined as software K code.

Table 5.7. Welding Machine Operations

Machine Family Code	**Operation Code**	**Operation Name**
TM	K1	TİG Welding
	K2	Spot welding

After the examination of the machine and machine operations, a list of raw materials used in the selected products was formed. The raw materials were listed according to their types and then as the model cards were generated, they were related with the usage amounts (see Table 5.8.).

Table 5.8. Raw Material List of the Selected Products

Raw Material Family Code	**Raw Material Code**	**Raw Material Name**
RSMG	RSMG1	0.8 mm Sheet Metal Gage
	RSMG2	1 mm Sheet Metal Gage
	RSMG3	1.2 mm Sheet Metal Gage
	RSMG4	1.5 mm Sheet Metal Gage
	RSMG5	2 mm Sheet Metal Gage
	RSMG6	3 mm Sheet Metal Gage
RSCM	RSCM1	0.8 mm Sheet Copper Gage
	RSCM2	1 mm Sheet Copper Gage
	RSCM3	1,5 mm Sheet Copper Gage
	RSCM4	2 mm Sheet Copper Gage

The Model Cards that are suggested after this stage were formed by defining in Excel and measurements were obtained based on real data. In the next section, model cards will be discussed in the application scope.

5.2.2. Implementation of the IMPACT Model Cards

In this section, model cards were generated as the second stage of database formation. A model card that is developed for each model is discussed in detail respectively. KPI methods that are required for card measurements are determined by company experts and discussed in detail. These studies from Goal Card to Supplier Card will be discussed below respectively.

1. Goal Card

Among the production goals that are suggested in the model "**Satisfy control of manufacturing activities by a schedule**" is selected as the first company purpose. In order to measure whether this goal is achieved or not, the relevant processes have been defined. These are respectively Planning, Storage Management, Purchasing, Mechanic Manufacturing, Assembly Manufacturing, and Sales and Marketing Process.

The company experts did not relate another goal with this goal, and for each process equally, defined the "level of significance" value that exerts its effect for all processes to achieve the goal. The (goal) success rate value that is planned for each process has been defined as 100%. After this point, a Process Card is generated for each process that takes place within the Goal Card. The success rate values of the process cards that are measured are transferred to the Goal Card and "total success rate" value belonging to G1 Goal is estimated.

Table 5.9. The structure of the G1 Goal

GOAL CARD							
Goal Name:		**Satisfy control of manufacturing activities by a schedule**			No:		00001-G1
Card Code:		**G1**			Date:		
Goal Definition:							
No	Related Process Code	Process Name	Planned Process Success Value	Actual Process Success Value	% Level of significance		% Success Level
1	P1	Planning Processes	100%	67,84%	16,67%		12,99%
2	P5	Storage Management Process	100%	66,00%	16,67%		11,33%
3	P6	Purchasing Process	100%	85,00%	16,67%		14,17%
4	P8	Mechanic Manufacturing	100%	77,95%	16,67%		12,99%
5	P9	Assembly Manufacturing	100%	68,00%	16,67%		11,33%
6	P16	Sales and Marketing Process	100%	77,00%	16,67%		12,83%
Succuss value of G1							75,64%
No	Related Goal Code	Goal Name	Planned Goal Success Value	Actual Goal Success Value	% Level of significance		% Success Level
Succuss value comes from the related goals:							
Total Success							**75,64%**

In the examination of G1 Card, the following findings have been obtained;

The “actual success rate” value of P1 Process has been obtained as 67,84% from P1 Card.

The “actual success rate” value of P5 Process has been obtained as 66,00% from P5 Card.

The “actual success rate” value of P6 Process has been obtained as 85,00% from P6 Card.

The “actual success rate” value of P8 Process has been obtained as 77,95% from P8 Card.

The “actual success rate” value of P9 Process has been obtained as 68,00% from P9 Card.

The “actual success rate” value of P16 Process has been obtained as 77,00% from P9 Card.

As a work principle that is defined for the Model;

If the obtained success value is (0,40), it is accepted that “Goal is not satisfied”, (40, 60) “partially satisfied”, (60, 80) “satisfied” and (80,100) “perfectly Satisfied”.

Based on the obtained result, G1 Goal is accepted as “satisfied”.

In a similar fashion, Goal Cards have been prepared for G2 and G3 Goals (see Chapter 4) included in the suggested model, and total success values have been estimated (see Appendix 3).

Goal success rates are calculated as;

G1: Satisfy control of manufacturing activities by a schedule	75.64%
G2: Increasing manufacturing performance	75.08%
G3: Decreasing manufacturing idle Times	73.36%
G4: Increasing manufacturing quality	78.21%
Average success rate of the Goal Model	**75.57%**

2. Process Card

In this section process cards belonging to the processes carried out in the company in question were prepared. The relations between company processes and operations were defined. The operations that are needed to measure each process were determined. In this scope, at first "**Planning Process Cards**" were prepared. First selected implementation processes is “Planning Process”

9 different operations related with planning process is identified. These operations are; process planning calculation, Material requirement planning calculation, Capacity Planning Calculation, Line Planning Calculation, Stock Control operation, Job Order operation, Manufacturing Scheduling calculation, Task Scheduling calculation and Manufacturing Control operation. Also 2 different Processes are identified as related with planning processes. Also 2 different processes is defined as related with planning process which are storage management process and supplier selection management process.

Company expert is decided that 70% of the process success comes from the related operations and remaining 30% of the process success comes from the related processes for the Planning processes.

From this point, operations activation rules are identified. Operations measurement techniques are stated. Calculation of the operations success rate is done by considering related KPI code. These calculations are listed below respectively.

KPI1 is used to measure process plan operation and actual success rate calculation of the operation is given in Table 11.

Table 5.10. Measurement Technique of the KPI1

Number of customer complaint / number of delivered order	Grade of KPI1
0	100%
0-0,2	80%
0,3-0,5	50%
0,5-0,7	30%
0,7>	0%

Table 5.11. The structure of P1 Process

PROCESS CARD								
Process Name:		**Planning Process**			No:	00001-P1		
Card Code:		**P1**			Date:	01.04.2014		
Process Definition:	**Planning Process Attributes**			Process Sequence Level			**1**	
		Operation Parameters		Operation M & E Parameter / TMTH				
Related Operation Code	Operation Name	Operation Activation Rule Code	Operation Activation Rule Explanation	KPI Code	Planned KPI Value	Actual KPI Value	% Level of significance	% Success Value
O1	Process planning calculation	OARC1	New Operation Card	KPI1	0	0,3	11,11%	0,05555
O2	Material Requirement Planning Calculation	OARC2	New Customer Order	KPI2	0	0,8	11,11%	0,08888
O3	Capacity Planning Calculation	OARC3	New Machine Card	KPI3	0-1	0,74	11,11%	0,1111
O4	Line Planning Calculation	OARC4	New Job Order	KPI4	1,0	1,5	11,11%	0,06666
O5	Stock Control operation	OARC6	New Procurement	KPI5	<1	4,5	11,11%	0,03333
O6	Job Order operation	OARC2	New Customer Order	KPI6	0	6	11,11%	0,08888
O7	Manufacturing Scheduling calculation	OARC4	New Job Order	KPI7	0	0,36	11,11%	0,05555
O8	Task Scheduling calculation	OARC4	New Job Order	KPI8	0	0,8	11,11%	0,08888
O9	Manufacturing Control operation	OARC7	Accomplished job order	KPI9	0	0,03	11,11%	0,08888
Success value of P1			67,77%	% Level of significance			70%	
Related Processes Code	Process Name					Actual KPI Value	% Level of significance	% Success Value
P5	Storage Management					66%	50,00%	33,00%
P7	Supplier Selection-Management					70%	50,00%	35,00%
Success value comes from the related processes:			68,00%	% Level of significance			30%	
Total Process Success								67,84%

This KP2 is used to measure material requirement planning operation and actual success rate calculation technique of the operation is given in Table 12.

Table 5.12. Measurement Technique of the KPI2

Nonproduction time caused by shortcoming goods / manufacturing time period	Grade of KPI2
0	100%
0-2	80%
3-4	50%
4-6	30%
6>	0%

This KP3 is used to measure capacity planning operation and actual success rate calculation technique of the operation is given in Table 5.13.

Table 5.13. Measurement Technique of the KPI3

Number of units output per period / the number of unit could be served per time period	Grade of KPI3
<0	100%
0-1	80%
1-2	50%
2-4	30%
4>	0%

This KP4 is used to measure line planning operation and actual success rate calculation technique of the operation is given in Table 5.14.

Table 5.14. Measurement Technique of the KPI4

Maximum number of time spent in each line/ Minimum time spent in each line	Grade of KPI4
0-1	100%
1-1,2	80%
1,2-1,5	60%
1,5-2	40%
2-2,5	20%
2,5>	0%

This KP5 is used to measure Stock Control operation and actual success rate calculation technique of the operation is given in Table 5.15.

Table 5.15. Measurement Technique of the KPI5

Cost of goods sold / Average İnventory	Grade of KPI5
<1	100%
1-2	80%
2-3	50%
4-5	30%
5>	0%

This KP6 is used to measure Job Order operation and actual success rate calculation technique of the operation is given in Table 5.16.

Table 5.16. Measurement Technique of the KPI6

Number of change in list of job orders-time period	Grade of KPI6
1-3	100%
4-6	80%
6-10	50%
12-20	30%
20>	0%

This KP7 is used to measure Manufacturing Scheduling operation and actual success rate calculation technique of the operation is given in Table 5.17.

Table 5.17. Measurement Technique of the KPI7.

Number of late manufacturing / Number of on time production	Grade of KPI7
0	100%
0-0,1	85%
0,2-0,3	75%
0,3-0,5	50%
0,5-0,7	30%
0,7>	0%

This KP8 is used to measure Task Scheduling operation and actual success rate calculation technique of the operation is given in Table 5.18.

Table 5.18. Measurement Technique of the KPI8

Number of change in task order / Number of total task order	Grade of KPI8
1-3	100%
4-6	80%
6-10	50%
12-20	30%
20>	0%

This KP9 is used to measure Manufacturing control operation and actual success rate calculation technique of the operation is given in Table 5.19.

Table 5.19. Measurement Technique of the KPI9

Total non-production time / Total manufacturing time	Grade of KPI9
0-0,01	100%
0,01-0,02	90%
0,02-0,05	80%
0,05-0,08	70%
0,08-0,1	50%
0,1-0,15	30%
0,15-0,2	10%
0,2>	0%

For the 2 other pre-defined Process Card See Appendix 4.

Table 5.20. Success Rate values of the Company Processes

Process Code	Process Name	Respective success Level
P1	Planning Process	67,84%
P5	Storage Management Process	66.00%
P8	Mechanic Manufacturing Process	77,95%
Avg Success Level of Process Model		**70,59%**

3. Operations Card

In this section, the relations between company operations and the machines used in production were defined. The first operation to be defined was "**Punch Mold Setup**" (see Table 5.22). 5 different Machines are related with Punch Mold Setup operation. These machines are; PNC1, PNC2, PNC3, PNC4 and PNC5. Machine activation rule is identified as new job order to related machine.

Basic machine information is explained below;

Maintenance period of the machine is defined as given Table 5.20.

Table 5.20. Maintenance Periods

Maintenance Period	Explanation
1	Every month
3	Every four months
6	Six time a year
12	Annually

The workers working on production in the company were grouped on the basis of production groups. This separation is important because each production operation is carried out by means of a different machine. Accordingly, the identification made for defining the Labor set from which an assignment can be made to the Machine cards is presented in Table 5.21.

Table 5.21. Labor Set definition

Labor set	Explanation
PNO	Punch Operator Set
BNO	Bend Operator Set
JO	Jet Operator Set
TMO	Tig Operator Set
ASMO	Assembly Operator Set

Table 5.22. The information to structure operation O30 .

OPERATIONS CARD					
Operation Name:	**Punch Mold Setup**			**No:**	00001-O1
Card Code:	**O30**			Date:	
Operation Definition:		Operation Sequence	**1**		

Related Machine Codes	Machine Name	Machine Availability				Basic Information of Machine		Maintenance Information of Machines				Machine-Labor Relation		KPI Measurement			% Level of significance	Success Value
		Machine State	State Duration / hour.	Available Time	Available Duration	Capacity hour / TD	Priority Level	Maintenance Period	Maintenance Code	Needed maintenance Time	Machine Activation Rule	Appropriate Lobar set	Needed Unit /hour	KPI Code	Planned KPI Value	Actual KPI Value		
PNC1	Punch Machine 1	1	250	-	-	16	2	3	PM	2	New job order	PNC	8	KPI101	24600	19560	20,00%	**120,48 %**
PNC2	Punch Machine 2	1	250	-	-	16	1	3	PM	2	New job order	PNC	8	KPI102	22400	17280	20,00%	**129,75%**
PNC3	Punch Machine 3	1	250	-	-	16	2	3	PM	2	New job order	PNC	8	KPI103	22400	20440	20,00%	**108,75%**
PNC4	Punch Machine 4	1	250	-	-	16	2	3	PM	2	New job order	PNO	8	KPI104	13600	8440	20,00%	**137,94%**
PNC5	Punch Laser Machine 5	1	250	-	-	16	2	3	PM	2	New job order	PNC	8	KPI105	24600	19560	20,00%	**120,48 %**
Total Success Value of O1																	**123,48%**	

The plant works in two shifts a day. Therefore, the daily operation time capacity "Capacity hour/TD" was defined with the fixed period of 16 hours for every machine.

Priority Level was established by classifying the machines on the basis of their operation capabilities. Since the performances (speed and capacity) of newer machines are better, in cases when all machines are available production supervisor prefers newer machines. This rule was defined to the model cart as shown in Table 5.23.

Table 5.23. Machine Prioritization Table

Priority Level	Explanation
0	unique machine
1	Prioritized Machine
2	Standard Machine
3	In emergencies

Machine Card satisfies observation and intervention of the decrease in performance of Operations. For the Punch Mold Setup Operation uses predefined KPI106.

KPI106 =$1-\left(\frac{total\ actual\ manufacturing\ times\ in\ PNC1}{Total\ needed\ manufacturing\ \ time\ for\ PNC1}\right)$

If this calculation gives a positive result it means that there is a downturn for manufacturing times, else there is an increment for manufacturing times

After these calculations are done for each punch machine, model takes average of each result to find setup operation success value. Proposed model sets for many KPI in order to measure operation success rate values. In addition, new KPI calculation techniques were defined in the implementation.

The company was observed to have too much intermediate stock. In order to prevent this, it was determined that bottleneck operation has to be defined for the company. Before starting the production process, planned production times in every operation were compared and the one with the highest value was set as the bottleneck operation. While making assignments to the machines, the products that require the longest production period in the bottleneck operation were assigned first and thus waiting times were shortened.

One of the established samples Operation card are explained in details above. The other 3 Operation Cars are presented in Appendix 5. The "Total success rate" values calculated for the other 4 operations are as shown in Table 5.24.

Table 5.24. Operation Model Success Level

Operation Code	Operation Name	Respective success Percentage
O30	Punch Mold Setup	123.48%
O31	Punch Manufacturing	158.00%
O33	Bend Mold Setup	116.55%
O33	Bend Manufacturing	130.08%
Avg. Success Level of Operation Model		**132.02%**

Operations Model Success rate is calculated as 131% . This value indicates that with the assignments made through the use of the model, the operations turned out to be at much better levels than the expected. The main reason for this is considered to be the fact that the company decides on the operator-origin data. In consequence of the work and time studies conducted during the work, it was clearly demonstrated that these values are far away from the actual values.

4. Machine Card

In this section of the work, machine cards were prepared in order to establish the relation between the machines and machine cards. Each type of operation is defined with different types of machines. First selected Machin is Punch Machine 1. Structure of the Punch Machine 1 is given in Table 5.25. During the preparation of the machine cards, the machines were defined as machine operation types, while machine operations were also defined on the basis of machine types. In Data Collection, sample lists of the machine operations are provided were assigned to machines in machine cards.

In the measurement of machine operation performances, the priory defined KPI139 was utilized. The related calculation is as presented below.

$$KPI139 = \frac{planned - actual\ setup\ time}{planned} * 100$$

Table 5.25. The structure of a machine card Punch Machine 1

MACHINE CARD											**No**	00001-PR1
Machine Name		**Punch Machine 1**									**Date**	**01.01.2014**
Card Code:		**PNC1**										
# of Mold space in one :		Width :11050 KG		X-Axis :50"			Y-Axis : 100"				Thickness: 0.31"	
Operation Types:		**Screw press, punching, countersink**										
PNC Equipment Code	Equipment Name	Equipment Capacity		Equipment Availability				KPI Measurement				
		# of Unit produced / one operation	Operation Cycle Times / TSD	Unit 1 State	Unit 1 State Duration	Purchase Date / TYR	Unit 1 Shelf Life	KPI Code	Planned KPI Value	Actual KPI Value	Success Value	
MD1	3*15 Köprü	3	480	0	0	2012	2016	KPI139	2400	1850	23%	
MD2	DD205 20.5x17.5	3	480	0	0	2012	2016	KPI139	2400	1850	23%	
MD3	F_079	2	480	0	0	2012	2016	KPI139	2400	1750	27%	
MD4	F_087	2	480	0	0	2012	2016	KPI139	2400	1800	25%	
MD5	F078	2	480	0	0	2012	2016	KPI139	2400	1850	23%	
MD6	F089	1	480	0	0	2012	2016	KPI139	2400	1950	19%	
MD7	F090	1	480	0	0	2012	2016	KPI139	2400	1995	17%	
MD8	Formi 041	2	480	0	0	2012	2016	KPI139	2400	2050	15%	
MD9	Formi 064	6	480	0	0	2012	2016	KPI139	2400	2125	11%	
MD10	Formi 56.0x5.0	8	480	0	0	2012	2016	KPI139	2400	1850	23%	
MD11	Langloch 10x7	1	40	0	0	2012	2016	KPI139	200	180	10%	
MD12	Langloch 12x6	10	40	0	0	2012	2016	KPI139	200	165	18%	
MD13	Langloch 20x5	12	40	0	0	2012	2016	KPI139	200	145	28%	
Total Machine 1 Success Value											**120,03%**	

While a positive result obtained from the calculation indicates that the actual value is better than the planned value, a negative value indicates the opposite.

For all predefined machine operation list in this study, see Appendix 7.

The machine selected as the example is the machine PNC1. Examining the "success rate" calculations pertaining to this machine shows that it produced a positive success rate for all machine operations. Its average success rate was calculated to be 120%.

At this point it was understood that the company assumptions were quite below the capacity.

Since the company is engaged in producing with molds, machine operations were set as molds. Producing with molds is a type of production where production setup period is very long and order-dependent. Accordingly, it was considered that for products to be produced consecutively the less change of molds would mean the faster the production. The Punch and Bend machines used in the company are as shown in Figure 5.7 and Figure 5.8, respectively.

Figure 5.7. PNC1- Punch Machine 1

PNC1 is used to Metal cutting, forming, tap & die, and counter sink. This machine is CNC Computer Numeric Controlled Punch Press

Figure 5.8. BND1- Bend Machine 1

This machine is a CNC machine which is a six axis press brake with programmable crowning for precise and contestant bends.

Similarity matrix is design for the Punch and Bend operations, this matrix gives an ability to decrease setup times caused by equipment replacement. Table 5.27 shows an example of a similarity matrix

By using this similarity, it controls same operations for the respective orders on the same machine and gives a priority to similar products on that particular machine.

One of the pre-defined Operation card are explained in details above. The other 3 Machine Cars are presented in Appendix 6. The "Avg. success rate" values calculated for the other 4 Machines are as shown in Table 5.26.

Table 5.26. Machine Model Success Level

Machine Code	Machine Name	Respective success Level
PNC1	Punch Machine 1	120,03%
PNC2	Punch Machine 2	121,52%
PNC3	Punch Machine 3	106,99%
PNC4	Punch Machine 4	131,88%
Avg. Success rate of Machine Model		**120,35%**

Results indicate that machine model estimations of the Company are highly low than performance of the machine model.

Table 5.27. Similarity Matrices of sub products

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Operation Name	060402P00	060407P00	060435P00	060445P00	060438P00	060430P00	060439P00	060405P00	060400P00	060401P00	060410P00	060412P00	060420P00	060414P00	060416P00	060505P00	060426P00	060506P00	060453P00	060418P00	060415P00	060423P00	060414P00	060437P00	060438P00	060431P00	060439P00	060434P00	060433P00	060442P00	060436P00	060483P00	060456P00	060457P00	060464P00	060408P00	060471P00	060462P00	060463P00	060465P00
060402P00	5	1	0	2	0	0	2	1	0	2	1	0	2	1	1	1	1	1	0	1	1	2	1	2	0	0	2	0	0	0	0	2	2	1	1	0	3	2	2	3
060407P00	1	7	0	2	1	0	2	2	0	4	0	0	1	2	1	1	1	1	0	1	1	1	2	2	1	0	1	0	0	1	0	1	0	4	3	2	2	2	3	2
060435P00	0	0	8	2	1	4	1	0	3	2	1	1	1	1	2	2	2	0	3	2	1	0	1	3	1	6	0	4	4	0	4	0	2	4	1	1	1	0	0	2
060445P00	2	2	2	7	0	1	2	1	0	4	0	0	2	1	2	3	3	1	1	2	1	2	1	5	0	1	2	1	1	0	1	2	2	3	2	0	2	1	2	2
060438P00	0	1	1	0	4	1	1	2	1	2	2	0	1	1	1	0	0	0	2	1	1	1	1	1	4	1	0	1	1	3	1	1	2	2	2	1	2	2	0	1
060430P00	0	0	4	1	1	6	0	0	3	1	1	1	1	1	2	1	1	0	2	2	1	0	1	1	1	6	0	4	5	0	5	0	2	1	2	0	1	0	0	1
060439P00	2	2	1	2	1	0	4	2	1	3	0	0	2	1	1	1	1	1	1	1	1	2	1	3	1	1	2	0	0	1	0	2	0	3	2	1	3	2	2	2
060405P00	1	2	0	1	2	0	2	4	0	2	0	0	1	1	2	1	1	1	1	2	1	1	1	1	2	0	1	0	0	3	0	1	2	3	3	1	2	3	1	1
060400P00	0	0	3	0	1	3	1	0	4	1	1	1	1	1	1	0	0	0	2	1	1	0	1	2	1	4	0	2	3	0	3	0	1	2	1	1	1	0	0	2
060401P00	2	4	2	4	2	1	3	2	1	11	1	0	3	3	2	2	2	2	1	2	2	2	3	6	2	2	2	1	1	1	1	2	3	6	3	1	4	2	3	3
060410P00	1	0	1	0	2	1	0	0	1	1	4	0	1	1	1	0	0	0	1	1	1	1	1	1	2	1	0	1	1	1	1	1	2	1	1	0	2	1	0	2
060412P00	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0
060420P00	2	1	1	2	1	1	2	1	1	3	1	0	5	4	3	2	2	1	1	3	4	2	4	3	1	1	2	1	1	0	1	2	2	2	4	0	3	1	2	2
060414P00	1	2	1	1	1	1	1	1	1	3	1	0	4	7	3	2	2	1	1	3	4	1	7	3	1	1	1	1	1	0	1	1	2	3	4	0	2	1	1	3
060416P00	1	1	2	2	1	2	1	2	1	2	1	0	3	3	8	3	3	1	2	8	3	1	3	2	1	2	2	3	3	1	3	1	4	3	4	1	2	1	1	2
060505P00	1	1	2	3	0	1	1	1	0	2	0	0	2	2	3	4	4	1	1	3	2	1	2	2	0	1	1	1	1	0	1	1	2	2	2	0	1	1	1	1
060426P00	1	1	2	3	0	1	1	1	0	2	0	0	2	2	3	4	4	1	1	3	2	1	2	2	0	1	1	1	1	0	1	1	2	2	2	0	1	1	1	1
060506P00	1	1	0	1	0	0	1	1	0	2	0	0	1	1	1	1	1	3	0	1	1	1	1	1	0	0	1	0	0	0	0	1	1	1	1	0	1	1	1	1
060453P00	0	0	3	1	2	2	1	1	2	1	1	0	1	1	2	1	1	0	6	2	1	0	1	2	2	3	0	2	2	1	2	0	3	2	1	3	1	1	1	2
060418P00	1	1	2	2	1	2	1	2	1	2	1	0	3	3	8	3	3	1	2	9	3	1	3	2	1	2	2	3	3	1	3	1	4	3	4	1	2	1	1	2
060415P00	1	1	1	1	1	1	1	1	1	2	1	0	4	4	3	2	2	1	1	3	6	1	4	2	1	1	1	1	1	0	1	1	2	2	4	0	2	1	1	2
060423P00	2	1	0	2	1	0	2	1	0	2	1	0	2	1	1	1	1	1	0	1	1	3	1	2	1	0	2	0	0	1	0	3	0	1	1	0	2	1	2	1
060414P00	1	2	1	1	1	1	1	1	1	3	1	0	4	7	3	2	2	1	1	3	4	1	7	3	1	1	1	1	1	0	1	1	2	3	4	0	2	1	1	3
060437P00	2	2	3	5	1	1	3	1	2	6	1	0	3	3	2	2	2	1	2	2	2	2	3	9	1	2	2	1	1	0	1	2	2	6	2	1	3	1	2	5
060438P00	0	1	1	0	4	1	1	2	1	2	2	0	1	1	1	0	0	0	2	1	1	1	1	1	4	1	0	1	1	3	1	1	2	2	2	1	2	2	0	1
060431P00	0	0	6	1	1	6	1	0	4	2	1	1	1	1	2	1	1	0	3	2	1	0	1	2	1	9	0	4	5	0	5	0	2	2	2	1	1	0	0	2
060439P00	2	1	0	2	0	0	2	1	0	2	0	0	2	1	2	1	1	1	0	2	1	2	1	2	0	0	4	0	0	0	0	2	0	1	1	1	2	1	2	1

5. Product Card

In order to define product - raw material relations and associate them with the machines two selected sample Product Cards were prepared. Product cards contain highly detailed information. For the first selected sample product 069007Q00, the card PR1 was created (see Table 5.28). The machine operations the product needs in 4 basic operation types, the type of required raw material, the amount of raw material needed for a sub product and production times were calculated separately.

As stipulated in the proposed model, at first the operational success rate of the product was obtained. By adding the success rate value of supplier B (see Table 5.29), from where the raw material was obtained, and the rate of defect that occurred during the production to the obtained operational success rate in proportion of their respective coefficients (see chapter 4 product success measurement calculation)

Calculation of the operational success of the product: KPI 138 as described below.

$$\frac{\text{Actual Manufacturing time of product } i}{Planned\ manufacturing\ time\ of\ product\ i} * 100$$

The agent calculates KPI138 for all manufactured products. It than takes an average of results as the success value of the Product

For selected sample product PR1: 069007Q00;

$$\frac{\frac{17}{42}*100+\frac{29}{42}*100+\frac{0,7}{0,7}*100++\frac{37}{37}*100}{4} = 77,37\% \text{ it means that}$$

100-77,37 =22,63% it means that operational performance of PR1 is increased 22,63%

Calculation of the raw material success of the product: KPI137; After each manufacturing operation, It checks the sequence of purchased materials in PQPT list than grade each purchased material depends on specified below;

1. If used raw material PQPT sequence is 1 its success level is 100%
2. If used raw material PQPT sequence is 2 its success level is 90%
3. If used raw material PQPT sequence is 3 its success level is 80%
4. If p used raw material PQPT sequence is 4 its success level is 70%

Table 5.28. Product Card for the product 069007Q00

PRODUCT CARD					
Product Name	**069007Q00**			**No**	00001-PR1
Card Code:	**PR1**			**Date**	**01.01.2014**
Product Description		**Shelf life:5 year**		**Stock Quantity :0**	

Route Sequence		0			**1**										**2**					**3**		**4**				**5**				
Route Name		**PREPARING**			**PUNCH**										**BEND**					**JET**		**TIG**				**ASSEMBLY**				
Sub Product Code	Sub Product Name	N. R. M.1	N. M. Volume /Unit	BOM Unit	N. PCH.O.1	N. PCH.O.2	N. PCH.O.3	N. PCH.O.4	N. PCH.O.5	N. PCH.O.6	N. PCH.O.7	N. PCH.O.8	Unit Cycle Time / TMT	Unit Setup Time / TMT	BOM Unit	N. BND.O.1	N. BND.O.2	Unit Cycle Time / TMT	Unit Setup Time / TMT	N. JT. O.1	Unit Cycle Time / TMT	BOM Unit	N. TM. O.1	Unit Setup Time / TMT	Unit Cycle Time / TMT	BOM Unit	N. ASLO. O.1	Assembly Level	Unit Setup Time / TMT	Unit Cycle Time / TMT
SPR1	060400P00	RSMG1	0,1	2	MD5								0,7	8	2	BND21	BND31	0,4	20	J1	0,5	2	TM	1	1,3	2	L3	1	0,5	1
SPR2	060401P00	RSMG1	0,1	1	MD104	MD109	MD113	MD65	MD72	MD121	MD36	MD38	2,8	12,6	1	BND14	BND29	0,4	20	J1	2	1	TM	1	1,3	1	L3	1	1,0	1
SPR3	060402P00	RSMG1	0,1	1	MD65	MD38							0,2	1,3	1	BND5	BND25	1,6	20			1	TM	1	1,3	1	L3	1	0,4	3
SPR4	060403P00	RSMG1	0,1	2	MD37	MD38							0,2	1,3	2	BND10	BND27	0,8	20			2	TM	1	1,3	2	L3	2	0,4	1
SPR5	060405P00	RSMG1	0,1	9	MD39								0,1	0,6	9							9	TM	1		9	L3	2	0,4	1
SPR6	060407P00	RSMG1	0,1	1	MD104	MD73	MD34	MD38					2,4	2,6	1	BND12	BND28	1,8	20			1	TM	1	3	1	L3	2	1,0	1
SPR7	060408P00	RSMG1	0,1	1	MD88	MD100	MD103	MD30	MD34	MD39			0,3	4	1	BND2	BND24	1,1	20			1	TM	1		1	L3	2	0,4	1
SPR8	060410P00	RSMG1	0,1	1	MD97	MD117	MD36	MD37					0,1	10	1	BND16	BND29	1,1	20	J1	0,2	1	TM	1	1,3	1	L3	2	0,4	5
KPI Code					KPI138										KPI138					KPI138		KPI138				KPI138				
Planned KPI Value					42 min										42,0					9,7		37				76				
Actual KPI Value					**17,96**										29,94					9,7		37				76				

Total Success value of PR1	**111.08%**

5. If used raw material PQPT sequence is 5 its success level is 60%

6. If used raw material PQPT sequence is 6 its success level is 50%

7. If used raw material PQPT sequence is 7 its success level is 40%

8. If used raw material PQPT sequence is 8 its success level is 30%

9. If p used raw material PQPT sequence is 9 its success level is 20%

10. If used raw material PQPT sequence is 10 its success level is 10%

11. used raw material PQPT sequence is greater or equal 11, its success level is 0%

Calculation of the Quality defect rate of the product

(Number of scrap i /Number of product)*100 values are calculated for each operation than average of them is calculated as; The operational defect rates are gathered from quality department for a specific time period. 10, 15, 5 and 4 defective units from 350 products for punch, bend, metal burr and tig operations are observed respectively.

$$Q^{Pr}_{pr1,dr,tp} = 1 - \frac{\frac{10}{350}*100 + \frac{15}{350}*100 + \frac{5}{350}*100 + + \frac{4}{350}*100}{4} = 97.55$$

Total success value of the PR1 is calculated as;

0.60 *122.63 + 0.20*90+0.20*97.55 = 73.57 + 18+19.51 =111.08%

One of the predefined product Card calculation is explained in detail. Second product card which is designed for PR2: **48417250GGB-24** is given in Appendix 8.

Success rate comes from predefined products are calculated as;

PR1: 069007Q00	111.08%
PR2: 48417250GGB-24	108.22%
Avg. success rate of the Product Model	**109.65 %**

Result of the Product Model shows that Products can be produced more efficiently by using proposed Model

6. Supplier Card

Supplier cards were created in order to enable the raw materials to be integrated with the suppliers. Basic information concerning the suppliers and the raw materials that can be procured from the suppliers were defined. On the basis of past-performances of the suppliers, at first supplier quality levels were obtained. Later on, the calculation of the PQPT values on the basis of price and payment terms was explained. Finally, total success rate for the selected example supplier 1 was calculated. Now these will be explained respectively.

A sample supplier card is prepared for Company A (see Table 5.30). Raw material range, unit price and payment terms values are assumed as given. Other values are calculated as explained below;

Supplier Quality Value is calculated as given below;

$$\frac{\frac{\textbf{Number of fault Raw Material Type}}{\textbf{Number of Raw Material Type}} * 100 + \frac{\textbf{Fault Raw Material}}{\textbf{Received Raw Material}} * 100 + \frac{\textbf{Number of Fault Shipment}}{\textbf{Number of Shipment}} * 100}{\mathbf{3}}$$

Table 5.29. Measurement and Evaluation of Supplier Quality

SUPPLIERS	Number of Raw Material Type	Number of fault Raw Material Type	Correct Raw material percentage	Received Raw Material	Fault Raw Material	Correct Received unit percentage	Number of Shipment	Number of Fault Shipment	Correct shipment percentage	Average quality value of each supplier
S1	21	2	**90,48**	169000	27402	**83,79**	43	2	**95,35**	**89,87**
S2	22	9	**59,09**	2169	520	**76,03**	13	5	**61,54**	**65,55**
S3	11	1	**90,91**	12834	540	**95,79**	7	1	**85,71**	**90,81**
S4	28	5	**82,14**	167033	14980	**91,03**	40	6	**85,00**	**86,06**
S5	71	7	**90,14**	2203477	242724	**88,98**	79	7	**91,14**	**90,09**
S6	1	1	**0,00**	176000	16000	**90,91**	8	1	**87,50**	**59,47**

Table 5.30. Supplier Card for Company A

SUPPLIER CARD					
Supplier Name	**Company A**			No:	00001-S1
Card Code:	**S1**			Date:	01.01.2014
Address	**Üsküdar**	**City:**	**Istanbul**	**Country**	**Turkey**
Contact Person		**Contact Phone:**		**Email**	

Material Code	Name	Quality Test Code	Measurement Parameters-1			Technical Parameters				Shipment Type	Delivery Time / Lot	Total Capacity / TMTH	Used Capacity / TMTH	Available Capacity / TMTH				
			Price / Lot (TL)	Quality Value	Payment Term	Width	Height	Thickness	Color						PQPT Value	Sequence in PQPT list	Opportunity Cost	Sequence in Opportunity list
RSMG1	0.8 mm Sheet Metal Gage	T1	1,1	60	3	-	-	0,8	-	ST1	5	250	-	250	17,5	3	-	2
RSMG2	1 mm Sheet Metal Gage	T1	0,5	60	3	-	-	1	-	ST1	5	250	-	250	19,3	4	1330	1
RSMG3	1.2 mm Sheet Metal Gage	T1	1	60	3	-	-	1,2	-	ST1	5	250	-	250	17,6	6	-	4
Total Success Value of S1																	**73.33%**	

After Quality levels are calculated as shown in Table 5.32

Calculation of PQPT value of the Supplier for each raw material is as follows

An example calculation for RSMG4;

The Supplier code is; S4

Price / Lot (TL)	: 1.3
Quality Value	: 59
Payment Term	: 3
Total need (from production order)	: 350
Interest Rate	: 6

Utility Function of raw material type RSMG4;

(RSMG4) = $\frac{\mathbf{1}}{\mathbf{0{,}60*1{,}3}}$ **+ 0,26*59+0,14*3 =17,04**

For a given orders, the agent calculates opportunity cost for the raw materials required. Than selects the most appropriate supplier for the given order.

$$\text{Opportunity cost of rawmaterial i of supplier j}:$$

$$\text{Unit Price * Total needs* } (1 + (\frac{\text{Interes rate}}{100})^{t})$$

Opportunity Cost of RSMG4 **=1,3*350*** $\mathbf{(1 + (\frac{6}{100})^{3})}$) =**1001**

For a given order, if the opportunity cost is greater than or equal to order profit, the agent sends a message to expert including "don't produce"

It satisfies a supplier selection method; For a given order, supplier selection mechanism;

Supplier success rate for each raw material calculated as given below;

At first a ranking is made separately for every raw material included in the supplier card. Then, the average success level of the supplier is calculated by means of a weighted rating.

For each raw material, the PQPT value for the raw material offered by the supplier is compared to the PQPT values of other suppliers that offer the same raw material. The ranking of the PQPT value of the supplier for the particular raw material shows the success rate of the supplier for that raw material. Ranking and the corresponding success rate values are as presented below;

a. If solution of the equation is 1 supplier success level is 100%
b. If solution of the equation is 2 supplier success level is 80%
c. If solution of the equation is 3 supplier success level is 60%
d. If solution of the equation is 4 supplier success level is 40%
e. If solution of the equation is 5 supplier success level is 20%
f. If solution of the equation is greater than 5 supplier success level is 0%

The Model Success Level is calculated then by average of all.

Total success value of the supplier S1 is calculated as ;RMSG 1 success rate is 80%, RMSG 2 success rate is 100%, and RMSG 3 success rate is 40%, and

Total success rate of S1 is **73.33%**

One of the supplier card is explained in detail above. Other 3 pre-defined supplier Cards is given in Appendix 9.

Success rate comes from suppliers is calculated as

S1: Company A	73.33%
S1: Company B	60.00%
S1: Company C	70.00%
S1: Company D	45.00%
Avg. Success rate of the model	**62.08%**

Test results shows that company must be improved their Suppliers list.

5.2. Integrated Production planning and scheduling results of Proposed Model

In this section, proposed integrated process planning and scheduling algorithm is prepared for the selected Company. Algorithm steps are given below;

Algorithm:

Step1. List Received order

Step2. Check the stock quantity of each product of each order and take difference from order quantity to reveal actual needed quantity

1. Calculate daily mechanical production capacity of product
    a. If order quantity greater than daily capacity;
        i. Divide order quantity to daily capacity then create job order as result of arithmetical equation.
    b. If order quantity less than production capacity of product
        i. Start scheduling of first order. Go to step 3

Step3. Create an order list which including product quantities and due dates of open orders.

Step4. Crate two temporary list; first is unscheduled products, second is scheduled products

Step5. Take first rank product of unscheduled product list.

Step6. List lower segment of product,

1. Create and lower segment list of product including stock quantity, needed quantity, actual need, unit production time, unit setup time, unit lacking time
2. Create an common mould matrix of lower segment units
3. Sequence new list depends on upper total welding time to lower of actual needed quantities.
4. Fractionate the list in two part, first is punch machine, second is laser machine
5. Start schedule of punching machines with first list
    a. Crate two temporary list for punch machines; first is unscheduled parts, second is scheduled parts
    b. If machines are empty assign first five punch unit to five machine and go to step 6.5.e
    c. If machines not empty go to step6.5.d
    d. To select new element, check the mould matrix to reveal minimum change of moulds. Delete added parts from unscheduled list and add to scheduled list
    e. if scheduled list is not empty go to step 6.5.d

f. if scheduled list is empty go to step 6.6,

6. Create an time table for five punch machine for all parts are scheduled, including sequence of parts and production times
7. Start schedule of laser machine
    a. Crate two temporary list for laser machine; first is unscheduled laser parts, second is scheduled laser parts
    b. Assign first part to laser machine
    c. Delete first lower segment part from unscheduled laser list and add to scheduled laser list
    d. And assign new part to laser machine depends on list rank
    e. if scheduled list is not empty go to step 6.7.d
    f. if scheduled list is empty go to step 6.8,
8. Create an time table for laser machine for all parts , including sequence of parts and production times
9. Record grater last operation ending time of punch machine and laser machine, select maximum of these dates to accept.
10. Delete products from unscheduled product list, add to scheduled product list
11. If unscheduled product list is not empty go to step5.
12. If unscheduled product list is empty go to stop.

Step7. Start schedule of twisting machines

1. Create an twisting list which including expected parts available times, needed units, unit production times, unit lacking times
2. Sequence parts depend on upper total welding time to lower of actual needed quantities.
3. Create two temporary list; first is un scheduled twisting list which is gathered from scheduled part list and second is scheduled twisting list
4. Create an common twisting matrix for parts in list
5. Start schedule
    a. If machines are empty, assign first five part to five machine and go to step7.5.d

   b. If machines are not empty, check producing part (BND mechanism) and do the selection in first five element of list to use same (BND mechanism)
   c. Assign selected part to selected machine
   d. Delete part from unscheduled list, add part to schedule list
   e. If unscheduled list is not empty go to step7.5.b
   f. Is unscheduled list is empty go to step7.6
6. Create a time table for twisting machine for all parts, including sequence of parts and production times.

Setting up the model;

Definition of the model assumptions; assumptions of the proposed model are defined as given below;

1. Jobs are independent. Job preemption is not allowed and each machine can handle only one job at a time
2. The different operations of one job cannot be processed simultaneously.
3. Only one operation is allowed to be a bottleneck operation.
4. The prototype is designed to handle only 4 products. Test and reliability of the Database therefore includes only those four types of product information.
5. It is assumed that there is no stock for all products.
6. All jobs and machines are available at time zero simultaneously.
7. After a job is processed on a machine, it is immediately transported to the next machine on its process, and the transmission time is assumed to be negligible.
8. The sales department is left out of the scope of the study. *Master production planning part* of model is therefore not executed. Respective forecasts are therefore assumed to be done beforehand.

To differentiate between current working performance and agent based system IMPACT was implemented. Implementation model had been design as a multi-agent system capable of performing all operations required. Agents developed increase the manufacturing performance make faster and effect process orders and delivery. Models of IMPACT were employed in JAVAbased software environment

Generated java framework consists of 22 class and 296 methods (see Appendix 10.). By the using this software Company is gained a comparative performance as explained below

Figure 5.9. Shows that a part of simulation results are made on manufacturing completion time

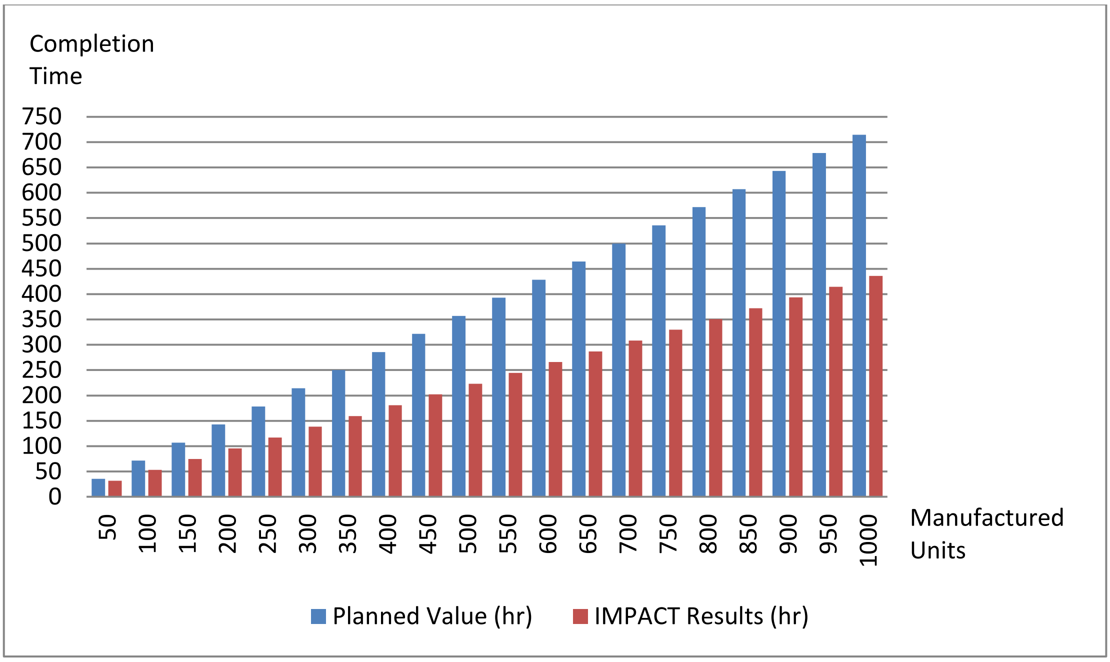

Figure 5.9. Some of the simulation results are made on manufacturing completion time (in terms of hr)

As can be seen from the Figure 13 proposed model has provided a significant decrease on manufacturing completion time.

Model is simulated for the supplier selection function. Opportunity cost is calculated for the suppliers which are have the same quality and price range. Some of results are shown in Figure 5.10.

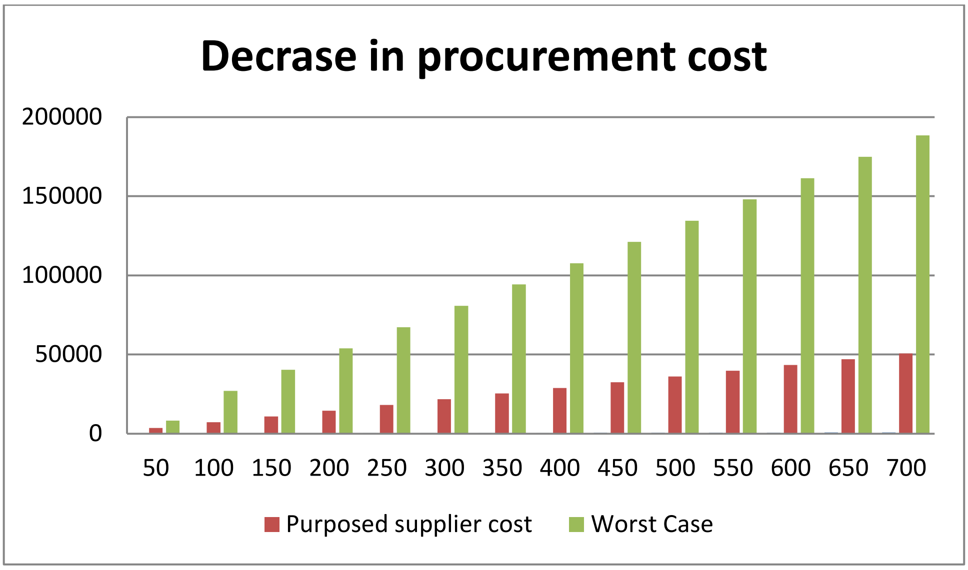

Figure 5.10. Some of the simulation results conducted on procurement cost (in terms of $)

As can be seen from the Figure 5.9.proposed model has provided a significant decrease in procurement cost. Another finding of the simulation result is the decrease in the procurement cost is increased for high level order quantities. When the purchased quantity increase the decrease rate of the model on purchasing cost is increasing.

By the using similarity matrix which is explained in machine model, setup times are decreased. Some of the simulation results that show total reduction on machine setup time is given in Figure 5.11.

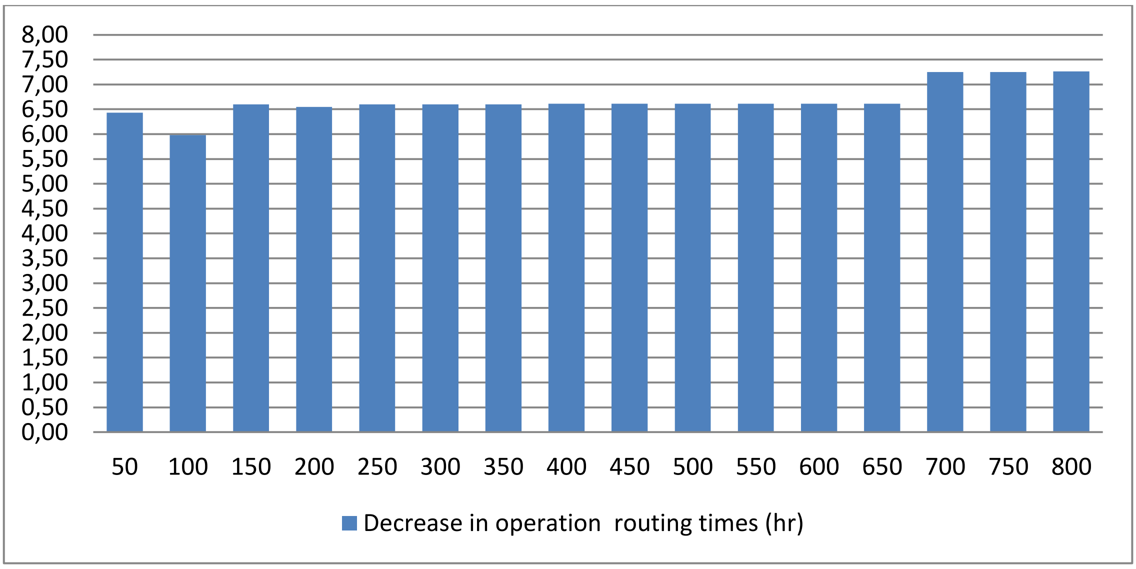

Figure 5.11. Simulation results on reduction in the setup times for different production units

Manufacturing scheduling is prepared as Figure 5.12

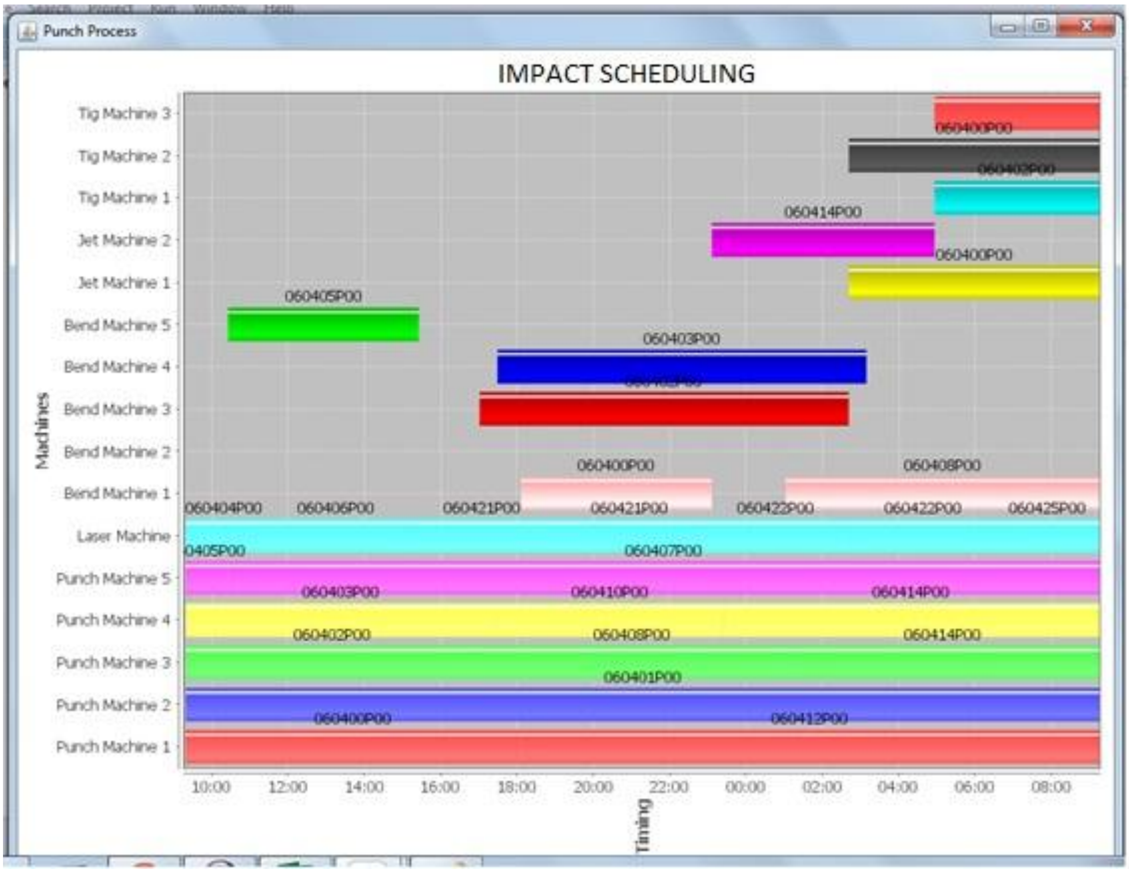

Figure 5.12. Manufacturing schedule for a specific order

Other Result od Study

1. 7 of 11 parameter is upgraded by using these model and given in Table 5.31

Table 5.31. Upgraded planning activities

	Planning Activities	**Employed Methods**
1	Order confirmation	**Manually**
2	Stock control	Automatically
3	Open job order control	Automatically
4	Order quantity change	Manual changes through MRP system
5	MRP run	Automatically for each order
6	Recording of stocks	Automatically
7	Order type decision	Human decision making
8	Preparing Job Order	Automatically
9	Purchase List	Automated List generation but manual controls
10	MPS	Automatically
11	Delivery Date	Automatically

2. Due to appear high setup time as a result of study, Company decided to decrease their product range.

In this section, proposed multi agent framework is applied on cabinet manufacturing. Computational results show that IMPACT gives an efficient way to manage of the manufacturing functions.

6. CONCLUSIONS

In this thesis, multi agent systems (MAS) were extensively studied and elaborated on the possibility of using MAS for manufacturing systems in order to increase automation and autonomy of manufacturing functionalities for the sake of productivity as well as efficient utilization of resources.

Integrated manufacturing environment and possible set of manufacturing agents were introduced. A concept of knowledge network was introduced in order to point out the importance of knowledge sharing within the integrated environment.

Proposed Multi Agent Systems is built on using as six integrated models mainly named as Goal Model, Process Model, Machine Model, Operations Model, Product Model and Supplier Model (**IMPACT).** These models are integrated using agent technology. A prototype application of the proposed model was also developed for the sake of the proof of concept.

The implementation is carried out in Company XYZ, using the actual manufacturing data available. Current implementation and the result of the proposed model implemented are compared. As described earlier, the implementation model was designed as a multi-agent system capable of performing all operations needed. It is experienced that the manufacturing performance is increased in terms of production speed, effectiveness of the processes as well as respective orders and delivery.

The result of implementation of the proposed technology in an actual manufacturing environment was provided. The results of this implementation clearly indicated that the makespan is minimized, manufacturing control capability is increased, setup times are minimized, control of the manufacturing is satisfied and supplier selection capability is increased.

Relation between propose of the thesis and proposed algorithm (IMPACT) is figure out in Table 6.1.

Table 6.1. Realization of the Goals

Thesis Goals	Purposed Model Goals	Agent İmplementation
Demonstrate the role of communication and coordination between agents	Design a Multi Agent Based Communication and Coordination Model	This Goals are considered as keystone of building Model
Design a method that combines manufacturing units.	Define business rules for each step of operations	This Goals are considered as another keystone of building Model
Model each proposed function using agent interaction mechanisms	Identify manufacturing functions accurately	For this Goal, a manufacturing Company is selected and all manufacturing functions are investigated for generation of the IMPACT model
Increase efficiency of manufacturing systems using agents' self-organization, machine learning, pro-activeness and reactiveness properties.	Goal 1: Satisfy control of manufacturing activities by a schedule Goal 2: Increasing manufacturing performance Goal 3: Decreasing manufacturing idle Times Goal 4: Increasing manufacturing quality	These Goals are satisfied using numerous software agents Which are broadly succinct in Chapter 4. For The Minimizing WIP, Operation Model is generated (giving priority to critical jobs) For optimizing manufacturing schedule, Increase manufacturing performance and Decrease operation and set up times, Machine Model is created, business rule are developed For Decrease cost of inventory and Selection of appropriate supplier, Supplier Model is established.
Increase control on manufacturing, scheduling, supplier selection, machine usage, operations and process management etc. Capabilities	Goal 1 Satisfy control of manufacturing activities by a schedule Goal 2: Increasing manufacturing performance Goal 3: Decreasing manufacturing idle Times	For the Goal I, a poster session of scheduled operations are obtained by implementation of the model
Decrease communication problem between different units of manufacturing	Avoid missing information by using appropriate software	A Java Software program is developed with an information system. FIPA standards is used to satisfy message standard and XML is used for the satisfy communication between agents.
Decrease manufacturing times	Goal 3: Decreasing manufacturing idle Times	For this goal, numerous business rules are determined in each model. Integrated process planning results shows that proposed algorithm gives very efficient results.

An introduction and brief information about MASs was explained to the thesis in Chapter 1. Basic information on agent based technology including agent architecture is provided. in Chapter 2, agent architecture is explained in detail. An agent is hardware or software based computer system providing a means to bridge the gap between humans and machines by means of interaction and intelligence. Multi-agent systems covering fundamental characteristics, architecture, cooperation and coordination issues were highlighted in Chapter 3.

A general framework for multi-agent manufacturing systems was proposed. Possible set of agents were introduced. XML is used to satisfy communication and coordination between agents. Some adaptation and self-organizing rules are stated. A concept of knowledge network was introduced in order to point out the importance of knowledge sharing within the integrated environment.

Chapter 5 includes a real life implementation of the proposed model IMPACT. Results of implementation are compared with company estimations and IMPACT outperforms the company estimations.

As stated in Chapter 1 the main aim of the thesis is set up as;

- ***Aim 1: To set-up well defined integrated manufacturing environment with self-decision making capability and modelling agents Increasing efficiency of manufacturing systems using agents' self-organization, pro-activeness and reactiveness properties with related experience and knowledge as well as inter-agent interaction mechanisms***

In this study, IMPACT composed of set of agents to integrate diffetrent functions of the manufacturing systems. ***self-decision making capability*** are realized by adaptation and self organizing rules where proactiveness is realized with the action set respect to perceived task set. With these properties of IMPACT model efficiency of the system is observed to increase.

- ***Aim 2: To demonstrate the role of communication and coordination between manufacturing agents with decreasing communication problems between different units of manufacturing***

In this study, IMPACT model communicates via KQML standard for message transfer between agents. XML is used as ACL. Coordination between manufacturing agents is also satisfied by business rules of the agents and precedence rules of the agents.

- ***Aim 3: To decrease manufacturing times by using optimization technics on manufacturing scheduling***

In this study, IMPACT has these abilities to decrease manufacturing lead time; priority of botttleneck operation, similarity based sequencing of operations, a supplier selection model that minimizes waiting times of operations, a thorough reordering policy, machine load balancing.

- ***Aim 4: To decrease manufacturing cost caused from purchasing raw materials, operation scheduling, WIP (Work-in-Process), overstock etc.***

In this study, IMPACT model defines a supplier selection technique named as opportunity cost to decrease procurement cost. Indirect cost reductions are satisfied via defined machine scheduling business rules which are defined in Machine Model. By the determination of the bottleneck operation, WIP inventory is decreased in the system.

- ***Aim 5: To prevent lack of production and lack of operation***

In this study, IMPACT useses Supplier Model to prevent lack of raw material that causes production stoppages and rescheduling rule inside the Machine model that dynamically schedules operations in case of machine failures. …………

- ***Aim 6: To increase quality level both of purchased and manufactured product.***

In this study, IMPACT defines a supplier quality rate in the Supplier Model that guaranties a desired quality level for raw materials. A quality control process which controls quality for determined periods to guarantee operational quality for every machine.

- ***Aim 7: To increase control on manufacturing, scheduling, supplier selection, machine usage, operations and process management etc. capabilities***

In this study, IMPACT increases control of the manufacturing via a well defined communication mechanism across with a hierarchial approach starts from Goals to Suppliers.

- ***Aim 8: To satisfy an ability of measurement for manufacturing goals, processes, operations, machines, products and suppliers for a manufacturing***

In this study, IMPACT model useses defined success rate functions to ***satisfy an ability to measurement for each model.***

- ***Aim 9: To prepare a road map for intelligent agent based manufacturing planning for manufacturers.***

In this study, IMPACT includes four manufacturing functions i.e. planning, manufacturing, stock management and procurement functions. Six models are generated to integrate four functions. Comprehensive step by step definitions of all models are explained for goal directed and integrated agent based manufacturing planning model. For both researchers and practioners can utilize this road map for different applications in manufacturing.

Future Work: The study will continue with the following objectives

- Basic aim of this thesis is to prepare a framework for the goal directed multi agent system model for the manufacturing systems. This study brings four different manufacturing function and the extended current model to all manufacturing functions should be developed.
- Although agent adaptation and agent learning is stated in the model to to increase integration multi-agent learning should be developed
- Proposed model is tested on a cabinet manufacturing system. To satisfy more standardization of the model, proposed model should be tested and resolved in different domains
- A negotiation mechanism is developed for the set of task, an extended negotiation mechanism should be developed to automate the procedure of finding the best available for each target task.
- Mobility is the more important feature of the agent based systems. it could be enhanced with mobile agent in further studies.

REFERENCES

Alippi, C., Bono, V., Piuri, V., & Scotti, F. **(2002).** Toward real-time quality analysis measurement of metal laser cutting. In *Virtual and Intelligent Measurement Systems, 2002. VIMS'02. 2002 IEEE International Symposium on,* 39-44.

Albus, J.S**.(1994),** "RCS: a reference model architecture for intelligent control" Robot Syst. Div., Nat. Inst. of Stand. & Technol., Gaithersburg, MD, USA ; Rippey, W.G.

Alvarez A. & Marty H.. **(2011),** "An automated approach to cloud storage service selection." In: Proceedings of the 2nd international workshop on scientific cloud computing, vol. 8, no. 3; 2011. p. 39–48.

Axelrod R., **(1997)** The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration (Princeton Univ. Press, Princeton, NJ).

Aydın E. & Fogarty T. C., (**2004**). A simulated annealing algorithm for multi-agent systems: a job shop scheduling application. Journal of Intelligent Manufacturing, Vol. 15, 805-814

Aydin M. E. & Öztemel E. (**2000),** □Dynamic job-shop scheduling using reinforcement learning agents□, Robotics and Autonomous Systems, 33, 169-178

Azevedo A. L., Toscano C., Sousa J.P. & Soares A. L**., (2004),** "An advanced agent-based order planning system fordynamic networked enterprises" Production Planning & Control,Vol. 15, 133–144

Barry J., Aparicio M., Durniak T., Herman P., Karuturi J., Woods C., Gilman C., Lam H., Ramnath R., (1998), "NIIIP-SMART: an investigation of distributed object approaches to support MES development and deployment in a virtual enterprise" Proc. of EDOC'98, La Jolla, CA,. 366–377.

Bezek A. & Gams M., **(2006),** "Comparing a traditional and a multi-agent load-balancing system" Computing and Informatics, Vol. 25, 2006, 17–42

Bhattacharyya S. & Ohlsson S., **(2010),** "Social creativity as a function of agent cognition and network properties: A computer model", Social Networks 32, 263–278

Bond A.H. & Gasser L., **(1988)**, "An analysis of problems and research in DAI" A.M. Bond, L. Gasser (Eds.), Readings in Distributed Artificial Intelligence, Morgan Kaufmann, San Mateo, CA, pp. 3–35

Bonabeau, E. **(2002).** "Agent-based modelling: Methods and techniques for simulating human systems". Proceedings of the National Academy of Sciences of the United States of America (PNAS), 99 (3), 7280–7287.

Bongaerts, L., (1998). Integration of scheduling and control in Holonicmanufacturing systems. Ph.D. Thesis, KU Leuven

Bradshaw, J.M., Dutfield, S., Benoit, P. and Woolley, J.D. (1997), "KAoS: Toward An Industrial-Strength Open Agent Architecture." In: "Software Agents," J.M. Bradshaw (Ed.), Menlo Park, Calif., AAAI Press,

Brandl D.Owen P.(2014)"Manufacturing Operations Management" availablity http://www.brlconsulting.com/Files/2003-09%20IEE%20Cambridge-V03.ppt (29.05.13)

Bratman B.E., **(1987),** Intentions, Plans, and Practical Reason. Harvard University Press, Cambridge MA

Broersen, C., **(2011).** A complex model for generating sustainable land use plans: Determining the spatial location allocations of energy sources in urban planning. Graduation program - Construction Management and Urban Development 2011-2012 Process Engineering

Brooks, R. A. **(1986)**. “A robust layered control system for a mobile robot.”, IEEE Journal of Robotics and Automation, 2(1):14–23.

Burke P.&Prosser P. (**1991**), A distributed asynchronous system for predictive and reactive scheduling , Artificial Intelligence in Engineering, 6, 106124

Butler, J. & Ohtsubo, H., **(1992).** “ADDYMS: architecture for distributed dynamic manufacturing scheduling. In: Famili, A., Nan, D.S., Kim, S.H. (Eds.), Artificial Intelligence Applications in Manufacturing. The AAAI Press, 199–214

Cakravastia, A., & Takahashi, K. **(2004).** Integrated model for supplier selection and negotiation in a make-to-order environment. International Journal of Production Research, 42(21), 4457–4474.

Calis B. & Bulkan S. **(2012).** “A Research Survey: Application Of AI Strategies For Optimal Production Scheduling” IMS2012. Antalya, TURKEY

Cao Y., Yang Y., Wang H. & Yang L. (**2009**), Intelligent Job Shop Scheduling Based on MAS and Integrated Routing Wasp Algorithm and Scheduling Wasp Algorithm, Journal of Software, 4, 487-494

Chan F. T. S. & Chan H. K. (**2004**), A new model for manufacturing supply chain networks: a multiagent approach, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 218, 443-454

Chan F. T. S. & Chan H. K. (**2005**), □The future trend on system-wide modelling in supply chain studies□**,** The International Journal of Advanced Manufacturing Technology, 25, 820-832

Choy K. L., Lee W. B., Lau H. C. W., Lu D. & Lo V. (**2004**), □Design of an intelligent supplier relationship management system for new product development□, International Journal of Computer Integrated Manufacturing, 17, 692-715

Colombo A.W., Schoop R. & Neubert R. (**2006). **□An Agent-Based Intelligent Control Platform for Industrial Holonic Manufacturing Systems□, IEEE Transactions On Industrial Electronics, 53, 322-337

Consoli A., Tweedale J. & Jain L. **(2006).** “The Link Between Agent Coordination and Cooperation”, Intelligent Information Processing III

Green S., Hurst L., Nangle B., Cunningham P., Somers F. & Evans R. **(1997)**, “Software agent: A review”, https://www.cs.tcd.ie/publications/tech-reports/reports.97/TCD-CS-1997-06.pdf

Das S. **(2013),** “Mobile Agents in Distributed Computing: Network Exploration”, Bulletin of the EATCS no 109, 54-69

Deshpande S. & Cagan J. (**2004**), □An Agent Based Optimization Approach to Manufacturing Process Planning□, Journal of Mechanical Design , 126, 46-55

Dimopoulos Y. & Moraitis P., **(2006). “**Multi-agent coordination and cooperation through classical planning”, Intelligent Agent Technology, IAT '06. IEEE/WIC/ACM International Conference on

Epstein J. M. & Axtell, R. L., **(1996).** Growing Artificial Societies: Social Science from the Bottom Up(MIT Press, Cambridge, MA).

Ferguson, A. **(1992),** Touring Machines: Autonomous Agents with Attitudes, Vol. 25, No. 5, May 1992, IEEE Computer Society Press Los Alamitos, CA, USA

Findler N. V., & Elder G. D., **(1995).** "Multi-agent coordination and cooperation in a distributed dynamic environment with limited resources." Artificial Intelligence in Engineering, 9, 229–238.

FIPA-a **(1997)** Specification, Part 1: "Agent Management. Foundation for Intelligent Physical Agents", Version 1.2, October 10,

FIPA-b **(1997)** Specification, Part 2: "Agent Communication Language. Foundation for Intelligent Physical Agents", Version 1.2, October 10, (1997)

Flores-Mendez, R. A. **(1999).** Towards a standardization of multi-agent system framework. *Crossroads*, *5*(4), 18-24.

Fox, M. S., Chionglo, J. F., & Barbuceanu, M., **(1993).** The integrated supply chain management system. URL: http://www.eil.utoronto.ca/public/icsm-intro.ps.

Fox, M.S., Barbuceanau, M. & Teigen, R., **(2000).** "Agent-oriented supplychain management.", The International Journal of Flexible Manufacturing Systems 12, 165–188.

Fukuda M., Bic L.F., Dillencourt M.B., & J.M. Cahill. **(1999)** Messages versus messengers in distributed programming. Journal of Parallel and Distributed Computing, 57(2):188–211,

Gabel T. & Riedmiller M. (**2007**), Scaling Adaptive Agent-Based Reactive Job-Shop Scheduling to Large-Scale Problems, Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Scheduling , 259 - 266

Gao J., **(2008)**, "Research on reverse logistic network optimization based on computational experiment method” East China Economic Management, 22 (12) pp. 139–143

Genesereth M. R.& Nilsson N., **(1987)**, “Logical Foundations of Artificial Intelligence”, Morgan Kaufmann Publishers: San Mateo, CA. ISBN 0-434613-31-1

Giannakis M. & Louis M. (**2011**), □A multi-agent based framework for supply chain risk management□, Journal of Purchasing & Supply Management, 17, 23-31

Grimm, V., U. Berger, F. Bastiansen, S. Eliassen, V. Ginot, J. Giske, J. Goss-Custard, T. Grand, S.K. Heinz, G. Huse, A. Huth, J.U. Jepsen, C. Jørgensen, W.M. Mooij, B. Müller, G. Pe’er, C. Piou, S.F. Railsback, A.M. Robbins, M.M. Robbins, E. Rossmanith, N. Rüger, E. Strand, S. Souissi, R.A. Stillman, R. Vabø, U. Visser, and D.L. DeAngelis, **(2006).** “A standard protocol for describing individual-based and agent-based models.” Ecological modelling, 198(1-2): p. 115-126.

Hao Q., Shen W., Stecca G.,& Wang L., **(2003).** “Towards an Internet enabled cooperative manufacturing management framework”, Proc. ofPRO-VE’03, Lugano, Switzerland,. 191–200.

Heyne D. & Monch L. (**2011**), □An agent-based planning approach within the framework of distributed hierarchical enterprise management□, Journal of Management Control, 22, 205-236

Hua, Z., **(2004)**, “Study of Multi-Agent Cooperation”, In: Proceedings of Third International Conference on Machine Learning and Cybernetics, vol. 5, IEEE, Shanghai, China, 3014–3017

Huang, C. C., Liang, W. Y., Lai, Y. H., & Lin, Y. C.**, (2010).** “The agent-based negotiation process for B2C e-commerce.” Expert Systems with Applications, 37(1), 348–359.

Huhns M. N. & Singh M. P., **(1998), "**Cognitive Agents", *IEEE Internet Computing*, vol. 2, no. 6, November/December 1998, pp. 87-89

Jain, A.S.& Meeran, S., **(1998).** A state-of-the-art review of job-shop scheduling techniques. Technical Report, Department of Applied Physics, Electronic and Mechanical Engineering, University of Dundee, Dundee, Scotland, 1998.

Jennings N.R. & Wooldridge M., **(2000).** "Agent-oriented software engineering", in: J. Bradshaw (Ed.), Handbook of Agent Technology, AAAI/MIT Press,

Jennings N.R., **(2000).** "On agent-based software engineering" Artificial Intelligence 117 (2000) 277–296

Jennings N.R., Faratin P., Johnson M.J., Norman T.J., O'Brien P.& M.E. Wiegand, **(1996)**. "Agent-based business process management", Internat. J. Cooperative Information Systems 5 (2–3) 105–130.

Karageorgosa A., Mehandjieva N., Weichhartb G. & Hammerle A. **(2003)**, "Agent-based optimisation of logistics and production planning", Engineering Applications of Artificial Intelligence 16, 335–348

Kim B.I., Heragu S.S., Graves R.J., & Onge A.St., **(2003).** "A hybrid scheduling and control system architecture for warehouse management" IEEE TR&A 19 (6) 991–1001

Kim, H. S., & Cho, J. H. **(2010).** Supply chain formation using agent negotiation. Decision Support Systems, 49(1), 77–90.

Kim M, Hanku L., Hyogun Y., Jee-In K., & HyungSeok K., (2011), "IMAV: an intelligent multi-agent model based on cloud computing for resource virtualization." In: Proceedings of the international conference on information and electronics engineering, vol. 6, no. 5; 2011. p. 30–5.

Lerman K. **(2001).** “Design and Mathematical Analysis of Agent-based Systems” Lecture Notes in Computer Science Volume 1871, 2001, pp 222-234

Leung C. W., Wong T. N., Mak K. L. & Funk R. Y.K. (**2010**), □Integrated process planning and scheduling by an agent-based ant colony optimization□, Computers & Industrial Engineering, 59, 166 - 180

Li X., Zhang C., Gao L., Li W. & Shao X. (**2010**), □An agent-based approach for integrated process planning and scheduling□, Expert Systems with Applications, 37, 1256-1264

Li, Z., Meng, Q.F., Sheng, Z.H.& Li, Q., **(2012)** "Analysis on performance and evolution of mass simulation under project quality optimization", Chinese Journal of Management Science, 20 (3), 112–121

Liu J. S.& Sycara K. P**. (1997),** □Coordination of multiple agents for production management□**,** Annals of Operations Research, 75, 235 - 289

Long Q.Q., Lin J., & Sun Z.X., **(2011).** “Modeling and distributed simulation of supply chain with a multi-agent platform”International Journal of Advanced Manufacturing Technology, 55 (9–12) pp. 1241–1252

Lu T.P.&Yih Y. (**2001**), □An agent-based production control framework for multiple-line collaborative manufacturing□**,** International Journal of Production Research, 39, 2155 - 2176

Macal, C.M. & North, M.J. **(2005).** 'Tutorial on Agent-Based Modelling and Simulation', in Euhl, M.E., Steiger, N.M., Armstrong, F.B. and Joines, J.A. (eds.), Proceedings of the 2005 Winter Simulation Conference.

Malone, T. W. & Crowston, K., **(1990)**. What is coordination theory and how can it help design cooperative work systems”, In Proceeding of the Third Conference on Computer-Supported Co- operative Work. ACM Press, New York, 357-310.

McArthur S. D. J, Davidson, E. M., Catterson, V. M., Dimeas, A. L. , Hatziargyriou, N. D., Ponci, F., & Funabashi, T. **(2007)** Multi-agent systems for power engineering applications - part 2: technologies, standards and tools for building multi-agent systems. IEEE Transactions on Power Systems, 22 (4). pp. 1753-1759. ISSN 0885-8950

Meng Q.F., Sheng Z.H., & Li Z., **(2012).** “Efficiency evolution of quality incentive in supply chain based on fairness preference”, Systems Engineering Theory and Practice, 32 (11) , pp. 2394–2403

Montaldo E., Sacile R., & Boccalatte A., **(2003),** “Enhancing Workflow Management in the Manufacturing Information System of a Small-Medium Enterprise: An Agent-Based Approach”, Information Systems Frontiers 5:2, pp.195–205,

Moyaux T., Chaib-Draa B., D'Amours. B., **(2006)**, “Supply Chain Management and Multiagent Systems: An Overview:,pp 1-27 Berlin, Springer

Müller, J.P., Pischel, M. **(1993),** The agent architecture inteRRaP: Concept and application, German Research Center for Artificial Intelligence

Nassima, A., Bouziane, B., & Amine, M. **(2013).** “Heurestics and reinforcement learning in manufacturing control: Optimization by phases.” Control, Decision and Information Technologies (CoDIT), 2013 International Conference on (pp. 798-803). IEEE.

Nigro G. L., Diega S. N. L., Perrone G.,& Renna P, **(2003).** “Coordination policies to support decision making in distributed production planning”, RCIM 19 (6) 521–531.

Nwana H. S., **(1996)**, “Software Agents: An Overview”, Knowledge Engineering Review, Vol. 11, No 3, pp. 205-244

Newell, A. & Simon, H. A., **(1976)**, “Computer science as empirical enquiry”, Communications of the ACM, 19,113–126.

Ottaway T. A. & Burns J. R. (**2000**), An adaptive production control system utilizing agent technology, International Journal of Production Research, 38, 721- 737

Oztemel E., **(1999).** “Entegre zeki imalat sistemleri geliştirme için bir model”, *Teknoloji, Kalite ve Üretim Sempozyumu*, Sakarya Kalite Derneği, Sapanca Oteli, 29 May, pp. 178-86.

Oztemel E. & Tekez E. **(2009)** “A general framework of a Reference Model for Intelligent Integrated Manufacturing Systems (REMIMS)” , Engineering Applications of Artificial Intelligence, Volume 22, Issue 6, September 2009, Pages 855–864

Paolucci M. & Sacile R. **(2005)**, Agent-Based Manufacturing and Control Systems, New Agile Manufacturing Solutions for Achieving Peak Performance, Florida CRC Press ISBN 1-57444-336-4

Pěchouček, M., Marık, V., & Barta, J. **(2002).** “A knowledge-based approach to coalition formation. IEEE Intelligent Systems”, 17(3), 17–25

Pendharkar P.C. (**1999),** A computational study on design and performance issues of multi-agent intelligent systems for dynamic scheduling environments, Expert Systems with Applications, 16, 121-133

Peng Y., Finin T., Labrou Y., & Cost R.S. **(1999).** “An agent-based approach for manufacturing integration --the CIIMPLEX experience” Applied Artificial Intelligence, 13, pp. 39–63

Peters C., Castellano G., Rehm M., André E., Raouzaiou A., Rapantzikos K., Karpouzis K., Volpe G., Camurri A., & Vasalou A. **(2011)** "Fundamentals of Agent Perception and Attention Modelling" Emotion-Oriented Systems, Cognitive Technologies, 293 DOI 10.1007/978-3-642-15184-2_16, C Springer-Verlag Berlin Heidelberg

Petrović M., Miljković Z., Babić B., Vuković N. & Čović N. **(2011).** "Towards A Conceptual Design Of An Intelligent Material Transport Based On Machine Learning And Axiomatic Design Theory" 34th International Conference On Production Engineering Niš, Serbia University Of Niš, Faculty Of Mechanical Engineering

Pfeifer, R. **(2001).** Agent Life. In Exploring Spatial Intelligence. Seminar Zurich: Artificial Intelligence Laboratory at the University of Zurich.

Qiao, B., & Zhu, J. **(2000).** "Agent-based intelligent manufacturing system for the 21st century". In Proceedings of the international forum for graduates and young researchers at EYPO, Hannover. The world exposition in Germany

Ren Z., Pham M.-T., Song M., Kim D.-H., & Koh C. S., **(2011).** "A robust global optimization algorithm of electromagnetic device utilizing gradient index and multi-objective optimization method," IEEE Trans. Magn., vol. 47, no. 5, pp. 1254–1257.

Renna P. & Argoneto P. (**2010**), □Production planning and automated negotiation for SMEs: An agent based e-procurement application□, International journal of Production Economics, 127, 73-84

Revilla J. G.& Cadena M. R. **(2008).** "Intelligent Manufacturing Systems: a methodology for technological migration" Proceedings of the World Congress on Engineering 2, London, U.K.

Ruiz-Alvarez A & Humphrey M., **(2011).** "An automated approach to cloud storage service selection.", In: Proceedings of the 2nd workshop on scientific cloud computing, vol. 10, no. 2; 2011. p. 44–56.

Russell S. J. & Norvig P., **(1995)**.: “Artificial Intelligence: A Modern Approach”, by Prentice-Hall inc Englewood Cliffs, NJ

Russell, S. & Norvig, P. **(2010),** “Artificial Intelligence: A modern approach” 3rd Edition. P 34 New Jersey: Prentice Hall

Sadeh N., Hildum D.W.,& Kjenstad D., **(2003).** “Agent-based e-supply chain decision support”, Journal of Organizational Computing and Electronic Commerce 33 (3–4) 225–241.

Santa-Eulalia L.A., D’Amours S., & Frayret J.M., **(2012).** “Agent-based simulations for advanced supply chain planning and scheduling: The FAMASS methodological framework for requirements analysis” International Journal of Computer Integrated Manufacturing, 25 (10), pp. 963–980

Saxena L.K. & Jain P.K., **(2012).** “An integrated model of dynamic cellular manufacturing and supply chain system design.” International Journal of Advanced Manufacturing Technology, 62 (1–4) (2012), pp. 385–404

Shen, W., & Norrie, D. H. **(2001).** Dynamic manufacturing scheduling using both functional and resource related agents. Integrated Computer- Aided Engineering, 8(1), 17–30.

Shen, W., **(2002).** “Distributed manufacturing scheduling using intelligent agents”, Intelligent Syst., 17, 88.

Shen W.*, Hao Q., Yoon H. J. & Norrie D. H., **(2006)**, “Applications of agent-based systems in intelligent manufacturing: An updated review”, Advanced Engineering Informatics 20, 415–431

Sheory O.M. **(2000).** “A scalable Agent Location Mechanism.” In: Wooldridge M. Lesperance Y. (eds.)1757, 162-173

Shukla S. K., Tiwari M. K. & Son Y. J. (**2008),** Bidding-based multi-agent system for integrated process planning and scheduling: a data-mining and hybrid tabu-SA algorithm-oriented approach, The International Journal of Advanced Manufacturing Technology, 38,163-175

Song C., Guan X., Zhao Q., & Ho Y., **(2005).** "Machine learning approach for determining feasible plans of a remanufacturing system," IEEE Trans. Autom. Sci. Eng., vol. 2, no. 3, pp. 262–275

Stone P. & Veloso M. **(1997).** "Towards Collaborative and adversarial learning: A case study in robotic soccer." International Journal of Human-Computer Systems (IJHCS),

Subrahmanian, V. S. **(2000).** "Heterogeneous agent systems*",* Cambridge, MIT Press.

Tecuci G., **(1998).** Building Intelligent Agents, Academic Press ISBN 10: 0126851255

Tekez E. **(2006).** "A reference model for ıntellıgent ıntegrated manufacturıng systems.",study of doctorate thesis in Sakarya University, June,

Thomason, R**., (2000)."** Desires and Defaults: A Framework for Planning with Inferred Goals." Proceedings of the 7th International Conference on the Principles of Knowledge Representation and Reasoning (KR'00). 702-713

Tweedale J., Ichalkaranje N., Sioutisb C., Jarvis B., Consoli A. &Philips-Wren G. **(2007),** "Innovations in multi-agent systems" Journal of Network and Computer Applications Volume 30, Issue 3, 1089–1115

Ulieru, M. and Cobzaru, M. () Building Holonic Supply Chain Management Systems: An e-Logistics Application for the Telephone Manufacturing Industry, IEEE Transactions on Industrial Informatics, 1, 18-31

Usher J. M. (**2003**), Negotiation-based routing in job shops via collaborative agent**,** Journal of Intelligent Manufacturing, 14, 485-499

Virdhagriswaran, S., Osisek, D., & O'Connor, P. **(1995).** “Standardizing agent technology”, *StandardView, 3*(3), 96-101.

Wang L. C. & Lin S. K. (**2009**), A multi-agent based agile manufacturing planning and control system, Computers & Industrial Engineering 57, 620-640

Wang, X., Wong, T. N., & Wang, G. **(2012). “**An ontological intelligent agent platform to establish an ecological virtual enterprise”. Expert Systems with Applications, 39(8), 7050–7061

Watanabe, M., Furukawa, M., & Kakazu, Y. (2001). “Intelligent AGV driving toward an autonomous decentralized manufacturing system.” Robotics and computer-integrated manufacturing, 17(1), 57-64.

White, J.E. **(1997).** “Mobile Agents.” In: “Software Agents”, J.M. Bradshaw (Ed.), Menlo Park, Calif., AAAI Press, 437-472

Willmott S., Constantinescu I. & Calisti M., **(2001),** “Multilingual Agents: Ontologies, Languages and”, Proceedings of the Workshop on Ontologies in Agent.

Wiki-SA **(2013)**: Software agent, Wikipedia free Encyclopedia http://en.wikipedia.org/wiki/Software_agent#Intelligent_agents available on **20.11.2013**

Wong T. N., Leung C. W., Mak K. L. & Fung R. Y. K. (**2006**), Integrated process planning and scheduling/rescheduling-an agent-based approach, International Journal of Production Research, 44, 3627-3655, Wong, T. N., & Fang, F. **(2010).** “A multi-agent protocol for multilateral negotiations in supply chain management.” International Journal of Production Research, 48(1), 271–299.

Wooldridge M. and Jenning Nicholas R., **(1995).** “Intelligent agents: theory and practice.” The Knowledge Engineering Review, 10, pp 115-152.

Wooldridge, M. **(2002).** An Introduction to Multiagent Systems. Wiley, New York.

Zhang, H. L., Jiang, G. R., Huang, T. Y. **(2010).** “An agent-based optimization approach for distributed project scheduling in supply chain with partial information sharing”. InProceedings of the 2nd international conference on agents and artificial intelligence, artificial intelligence (pp. 603–606).

Zhang W. Y., Zhang S., Cai M., &Huang J. X., **(2011).** “A new manufacturing resource allocation method for supply chain optimization using extended genetic algorithm,” International J.

a

APPENDIX

APPENDİX 1. Pre-defined Goals and related processes

Goal Code	Related Process Code	Process Name	KPI Code	KPI Explanation	KPI Technique
G1	P1	Planning Processes	KPI79	Measurement of P1 process performance	Success percentage of P1 Processes
	P5	Storage Management Process	KPI80	Measurement of P5 process performance	Success percentage of P5 Processes
	P6	Purchasing Process	KPI81	Measurement of P6 process performance	Success percentage of P6 Processes
	P8	Mechanic Manufacturing	KPI82	Measurement of P8 process performance	Success percentage of P8 Processes
	P9	Assembly Manufacturing	KPI83	Measurement of P9 process performance	Success percentage of P9 Processes
	P16	Sales and Marketing Process	KPI84	Measurement of P16 process performance	Success percentage of P16 Processes
G2	P1	Planning Processes	KPI85	Measurement of P1 process performance	Success percentage of P1 Processes
	P5	Storage Management Process	KPI91	Measurement of P5 process performance	Success percentage of P5 Processes
	P6	Purchasing Process	KPI81	Measurement of P6 process performance	Success percentage of P6 Processes
	P7	Supplier Selection Management Process	KPI92	Measurement of P7 process performance	Success percentage of P7 Processes
	P8	Mechanic Manufacturing	KPI86	Measurement of P8 process performance	Success percentage of P8 Processes
G3	P9	Assembly Manufacturing	KPI83	Measurement of P9 process performance	Success percentage of P9 Processes
	P12	Product Development	KPI87	Measurement of P12 process performance	Success percentage of P12 Processes
	P15	Quality Control Process	KPI195	Measurement of P15 process performance	Success percentage of P12 Processes
	P1	Planning Processes	KPI90	Measurement of P1 process performance	Success percentage of P1 Processes
	P5	Storage Management Process	KPI91	Measurement of P5 process performance	Success percentage of P5 Processes
G3	P9	Assembly Manufacturing	KPI83	Measurement of P9 process performance	Success percentage of P9 Processes
	P12	Product Development	KPI87	Measurement of P12 process performance	Success percentage of P12 Processes
	P15	Quality Control Process	KPI195	Measurement of P15 process performance	Success percentage of P12 Processes
	P1	Planning Processes	KPI90	Measurement of P1 process performance	Success percentage of P1 Processes
	P5	Storage Management Process	KPI91	Measurement of P5 process performance	Success percentage of P5 Processes
G4	P1	Planning Processes	KPI85	Measurement of P1 process performance	Success percentage of P1 Processes
	P6	Purchasing Process	KPI81	Measurement of P6 process performance	Success percentage of P6 Processes
	P7	Supplier Selection Management Process	KPI92	Measurement of P7 process performance	Success percentage of P7 Processes
	P15	Quality Control Process	KPI195	Measurement of P15 process performance	Success percentage of P12 Processes

APPENDIX 2 Pre-defined Processes and related operations (Part 1)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P1	Planning Process	O1	Process planning	KPI1	Degree to which the process output (work product) conforms to requirements	Number of customer complaint / number of delivered order
		O2	Material Requirement Planning	KPI2	Degree to which a product or service meets manufacturing requirements and expectations.	Nonproduction time caused by shortcoming goods / manufacturing time period
		O3	Capacity Planning	KPI3	Degree to Performance of the Capacity Plan	number of units output per period / the number of unit could be served per time period
		O4	Line Planning	KPI4	Degree to Equal shares of job loading	Maximum number of time spent in each line/ Minimum time spent in each line
		O5	Stock Control	KPI5	Degree to Inventory turnover rate	Cost of goods sold / Average İnventory
		O6	Job Order	KPI6	Degree to changes of Job order list	Number of change in list of job orders-time period
		O7	Manufacturing Scheduling	KPI7	Degree to on time manufacturing	Number of late manufacturing / Number of on time production
		O8	Task Scheduling	KPI8	Degree to indirect manufacturing activities	Number of change in task order / Number of total task order
		O9	Manufacturing Control	KPI9	Degree to control of manufacturing	Lose time caused by unrealized productive maintenance / Total manufacturing time

APPENDIX 2 Pre-defined Processes and related operations (Part 2)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P2	National Logistic	O10	Dispatching	KPI10	Degre to minimizing cost of tax penalty caused by miss dispatching	Total amount of tax penalty
		O11	Invoice	KPI11	Degree to minimizing over-due receivables	Total amount of over-due receivables
		O12	Transportation Operations	KPI12	Degree to minimizing transportation cost	Total amount of transportation cost
		O13	Packaging Order	KPI13	Degree to minimizing lose time caused by packaging process	Number of late order caused by packaging processes
P3	International Logistic	O10	Dispatching	KPI10	Degree to minimizing cost of tax penalty caused by miss dispatching	Total amount of tax penalty
		O11	Invoice	KPI11	Degree to minimizing over-due receivables	Total amount of over-due receivables
		O12	Transportation Operations	KPI12	Degree to minimizing transportation cost	Total amount of transportation cost
		O13	Packaging Order	KPI13	Degree to minimizing lose time caused by packaging process	Number of late order caused by packaging processes
		O14	Customs procedures	KPI14	Degree to minimizing lose time caused by custom procedures	Number of late order caused by custom procedures

APPENDIX 2 Pre-defined Processes and related operations (Part 3)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P4	Dispatching-Shipping	O15	Loading Trucks	KPI15	Degree to minimizing lose time caused by loading Trucks	Total over amount of differences between planned and actual loading times
		O16	Shipment order	KPI16	Degree to minimize misdelivery	Number of customer complaint caused by misdelivery
P5	Storage Management	O17	Stock Layout	KPI17	Degree to increase of the usage capacity of storage	Total number of rejected order caused by lead time
		O18	Outgoing Stocks	KPI18	Degree to minimize misdelivery of the order	Number of customer complaint caused by missing in order
		O19	İncoming Raw Material	KPI19	Degree to minimizing cost of manufacturing caused by incoming raw materials or WIP (work in process) material	Total nonproduction time caused by incoming raw material or WIP material
		O20	Updating of Inventory	KPI20	Degree to manage of inventory	Difference between recorded inventory and actual inventory
		O21	Stock Level Control	KPI21	Degree to control of safety stock level of stocks	Total number of stock of current period / Total number of stock of previous period
P6	Purchasing	O22	Procurement Order	KPI22	Degree to increase on time procurement	Total nonproduction time caused by procurement
		O23	Purchasing cost management	KPI23	Degree to minimizing cost of procurement	Difference between recorded cost of procurement and new order cost
		O24	Market Survey	KPI24	Degree to increase supplier list	Number of new supplier

APPENDIX 2 Pre-defined Processes and related operations (Part 4)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P7	Supplier Selection-Management	O25	Supplier Selection	KPI25	Degree to increase appropriate supplier selection	Total lose amount depends on AHP (Analytical hierarchy Technique) technique
		O26	Supplier management	KPI26	Degree to manage supplier capacity	Total lose amount caused by wrong capacity information
		O27	Current Supplier Performance Measurement and Evaluation	KPI27	Degree to control of changes in procurement orders	Number of selected supplier which has under 70% success value depends on quality report
P8	Mechanic Manufacturing	O28	Punch Software Setup	KPI28	Degree to minimizing idle times caused by punch software setup	Actual setup time / Planned setup time
		O29	Punch Mold Selection	KPI29	Degree to minimizing idle time caused by haven't access to necessary mold	Total amount of loss time
		O30	Punch Mold Setup	KPI30	Degree to minimizing punch mold setup time	Actual mold setup time / Planned mold setup time
		O31	Punch Manufacturing	KPI31	Degree to increase mold manufacturing performance	Actual manufacturing capacity / Planned manufacturing capacity
		O32	Bend Software Setup	KPI32	Degree to minimizing idle times caused by bend software setup	Actual setup time / Planned setup time
		O33	Bend Mold Setup	KPI33	Degree to minimizing Bend mold setup time	Actual bend setup time / Planned bend setup time
		O34	Bend Manufacturing	KPI34	Degree to increase bend manufacturing performance	Actual bend manufacturing capacity / Planned bend manufacturing capacity
		O35	Jet Manufacturing	KPI35	Degree to increase Jet manufacturing performance	Actual jet manufacturing capacity / Planned jet manufacturing capacity
		O36	Tig Setup	KPI36	Degree to minimizing Tig setup time	Actual Tig setup time / Planned Tig setup time
		O37	Tig Manufacturing	KPI37	Degree to increase Tig manufacturing performance	Actual Tig manufacturing capacity / Planned Tig manufacturing capacity
		O38	Painting Setup	KPI38	Degree to minimizing Painting setup time	Actual Painting setup time / Planned Painting setup time
		O39	Painting	KPI39	Degree to increase Painting manufacturing performance	Actual Painting manufacturing capacity / Planned Painting manufacturing capacity
		O40	Transfer	KPI40	Degree to minimizing Transfer time	Actual Transfer time / Planned Transfer time

APPENDIX 2 Pre-defined Processes and related operations (Part 5)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P9	Assembly Manufacturing	O41	Assembly Setup	KPI41	Degree to minimizing assembly setup time	Actual Assembly setup time / Planned Assembly setup time
		O42	Assembly	KPI42	Degre to increase Assembly performance	Actual Assembly capacity / Planned Assembly capacity
		O43	Transfer	KPI43	Degree to minimizing loss time caused by the transfer of fully-manufactured goods	Actual Transfer time / Planned Transfer time
P10	Chemistry Lab	O44	Testing	KPI44	Degree to minimizing total amount of the clin ship materials	Total number of clin ship materials / Total number of material
		O45	Waste Management	KPI45	Degree to increase of the usage of best material	Total amount of recycled material / Total amount of waste material
		O46	Raw Material Measurement and Evaluation	KPI46	Degree to minimizing non production time caused by tested materials	Total non-production time caused by tested material month over month
P11	Maintenance and Repair	O47	Productive Maintenance	KPI47	Degree to increasing of the r&d studies	Number of new activity that is developed to prevent manufacturing defects month over month
		O48	Periodic Maintenance	KPI48	Degree to increasing of the periodic maintenance plan	Difference between planned periodic maintenance unit and actual productive maintenance unit
		O49	Maintenance material List	KPI49	Degree to suitability of the list of material and actual material list	Total amount of idle time caused by not found material is already exist in list
		O51	Failure Logging	KPI51	Degree to minimizing machine failures caused by irregular proceedings of the maintenance	Number of failures month over month

APPENDIX 2 Pre-defined Processes and related operations (Part 6)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P12	Product development	O52	New Engineering Drawing	KPI52	Degree to increase of the new engineering design to satisfy customer satisfaction	number of new engineering drawing
		O53	Sample Manufacturing	KPI53	Degree to increase performance of the sample manufacturing	Number of break downs during sample manufacturing
		O54	Updating of Engineering Drawing	KPI54	Degree to minimizing updating of the engineering drawings caused by customer complaints	number of updating of engineering drawing product over product
		O55	Manufacturing Specifications	KPI55	Degree to minimizing of the non-production time caused by ill-defined manufacturing specifications	Total nonproduction time caused by ill-defined manufacturing specifications
		O56	Material Specification	KPI56	Degree to minimizing of loss time caused by ill-defined material specifications	Total nonproduction time caused by ill-defined material specifications
		O57	Patent Order	KPI57	Degree to increase number of patent order	Total number of patent order year over year
P13	Patent	O58	Patent Application	KPI58	Degree to increase the number of registered pattern	Total number of registered pattern year over year
P14	Engineering	O59	Codding	KPI59	Degree to increase usage of the codding system	Total number of undefined material in the system
		O60	Bill of Material List (BOM)	KPI60	Degree to minimizing non production time caused by ill-defined BOM list	Total loss time caused by ill-defined BOM list
		O61	Operations route of product	KPI61	Degree to minimizing number of faulty product caused by ill-defined operation route	Total loss caused by ill-defined operation route
		O62	Operations Sequence	KPI62	Degree to minimizing number of faulty product caused by ill-defined operation sequence	Total number of waste product caused by ill-defined operation sequence
		O63	Costing of Labor	KPI63	Degree to manage labor cost	Total amount of labor cost moth over month
		O64	Costing of Manufacturing	KPI64	Degree to manage manufacturing cost	Total amount of actual manufacturing cost over previous manufacturing cost
		O65	Mold Design	KPI65	Degree to manage mold necessity	Number of waste product caused by mold design

APPENDIX 2 Pre-defined Processes and related operations (Part 7)

Process Code	Process Name	Related Operation Code	Operation Name	KPI Code	KPI Explanation	KPI Technique
P15	Quality Control	O66	Incoming material Quality Control	KPI66	Degree to minimizing manufacturing waste caused by incoming materials	Total amount of waste material / Total amount of purchased order
		O67	WIP Quality Control	KPI67	Degree to minimizing scrap cost caused by ill-defined quality process management	Total individual production time of the scrap material * cost time/unit
		O68	Assembly Quality Control	KPI68	Degree to minimizing faults in final assembled product	Number of products which is put in quarantine for a given time period
		O69	Packaging Quality Control	KPI69	Degree to minimizing wrongful delivery of manufactured product	Number of return shipping / Total number of shipping
		O70	Supplier Quality Control Report	KPI70	Degree to managing changes in quality level of suppliers	Number of supplier which are withdrawn from supplier list caused by their quality scale/Number of contracted suppliers
P16	Sales and Marketing Process	O71	Submit offer	KPI71	Degree to increase number of offer	Number of given offers / target number of offers
		O72	Receipt order	KPI72	Degree to increase number of receipt order	Number of receipt order / target value
		O73	Prepare Contract-listing the term of a contract	KPI73	Degree to minimizing losses caused by incorrectly prepared documents	Number of reject offer caused by prepared contract
		O74	Received Order List	KPI74	Degree to increasing order management	Number of late order caused by wrong information in order report
		O75	Customer Orders Report	KPI75	Degree to increase sales return of company	Sales return of actual period / sales return of last period
		O76	Searching New Customer	KPI76	Degree to increase availability of potential buyer	Number of new customer of actual period /number of new customer of last period
P17	Education Process	O77	Education participation report	KPI77	Degree to increase educated blue-collar	Number of educated blue-collar / Total blue-collar
		O78	Education programs	KPI78	Degree to increase training ground related manufacturing	Number of training types of actual year / Number of training types of last year

APPENDİX 3.Pre-defined Goal Cards (Part 1-G2)

GOAL CARD							
Goal Name:		**Increase Manufacturing Performance**			No:		00001-G2
Card Code:		**G2**			Date:		01.01.2014
Goal Definition:							
No	Related Process Code	Process Name	Planned Process Success Value	Actual Process Success Value	% Level of significance		% Success Level
1	P1	Planning Processes	100%	67,84%	11,11%		7,54%
2	P5	Storage Management Process	100%	66,00%	11,11%		7,33%
3	P6	Purchasing Process	100%	85,00%	11,11%		9,44%
4	P7	Supplier Selection Management Process	100%	75,00%	11,11%		8,33%
5	P8	Mechanic Manufacturing	100%	77,95%	11,11%		8,66%
6	P9	Assembly Manufacturing	100%	68,00%	11,11%		7,55%
7	P12	Product Development	100%	80,00%	11,11%		8,89%
8	P15	Quality Control Process	100%	85,00%	11,11%		9,44%
9	P14	Engineering	100%	71%	11,11%		7,89%
Succuss value of G2							75,08%
No	Related Goal Code	Goal Name	Planned Goal Success Value	Actual Goal Success Value	% Level of significance		% Success Level
Succuss value comes from the related goals:							
Total Success							**75,08%**

APPENDİX 3.Pre-defined Goal Cards (Part 2-G3)

GOAL CARD							
Goal Name:		**Decreasing manufacturing idle Times**			No:		00001-G3
Card Code:		**G3**			Date:		01.01.2014
Goal Definition:							
No	Related Process Code	Process Name	Planned Process Success Value	Actual Process Success Value	% Level of significance	% Success Level	
1	P9	Assembly Manufacturing	100%	68,00%	20%	13.6%	
2	P12	Product Development	100%	80,00%	20%	16%	
3	P15	Quality Control Process	100%	85,00%	20%	17%	
4	P1	Planning Processes	100%	67,84%	20%	13.56%	
5	P5	Storage Management Process	100%	66,00%	20%	13.2%	
Succuss value of G2						73,36%	
No	Related Goal Code	Goal Name	Planned Goal Success Value	Actual Goal Success Value	% Level of significance	% Success Level	
Succuss value comes from the related goals:							
Total Success						**73,36%**	

APPENDİX 3.Pre-defined Goal Cards (Part 3-G4)

GOAL CARD							
Goal Name:		**Increasing manufacturing quality**				No:	00001-G4
Card Code:		**G4**				Date:	01.01.2014
Goal Definition:							
No	Related Process Code	Process Name	Planned Process Success Value	Actual Process Success Value	% Level of significance		% Success Level
1	P1	Planning Processes	100%	67,84%	25%		16.96%
2	P6	Purchasing Process	100%	85,00%	25%		21.25%
3	P7	Supplier Selection Management Process	100%	75,00%	25%		18.75%
4	P15	Quality Control Process	100%	85,00%	25%		21.25%
5							
6							
7							
Success value comes from the related processes							78.21%
No	Related Goal Code	Goal Name	Planned Goal Success Value	Actual Goal Success Value	% Level of significance		% Success Level
Succuss value comes from the related goals:							
Total Success							**78.21%**

APPENDİX 4.Pre-defined Process Cards (Part 1-P5)

PROCESS CARD									
Process Name:		**Storage Management**			No:	00001-P5			
Card Code:		**P5**			Date:	01.04.2014			
Process Definition:	**Storage Management Process Attributes**			Process Sequence Level					
Related Operation Code	Operation Name	Operation Parameters		Operation M & E Parameter / TMTH				% Level of significance	% Success Value
		Operation Activation Rule Code	Operation Activation Rule	KPI Code	Planned KPI Value	Actual Value			
O17	Defining Safety stock	OARC6	New Procurement	KPI17	0	3		20,00%	14,00%
O18	Calculation of the Outgoing Stocks	OARC8	New Shipment	KPI18	0	0		20,00%	20,00%
O19	Recording İncoming Raw Material	OARC6	New Procurement	KPI19	0	0		20,00%	20,00%
O20	Updating of Inventory	OARC5-OARC6-OARC8	New Procurement - New Shipment	KPI20	0	0,1		20,00%	2,00%
O21	Stock level control	OARC9	Each Month	KPI21	<1	2,7		20,00%	10,00%
Success value comes fof P5:			66,00%	% Level of significance					**100%**
Related Processes Code	Process Name						Actual KPI Value	% Level of significance	% Success Value
Success value comes from the related processes:									
Total Operations Success									**66%**

APPENDİX 4.Pre-defined Process Cards (Part 2-P8)

PROCESS CARD									
Process Name:	**Mechanic Manufacturing**				No:	00001-P8			
Card Code:	**P8**				Date:	01.04.2014			
Process Definition:	**Mechanic Manufacturing Process Attributes**			Process Sequence Level					
Related Operation Code	Operation Name	Operation Parameters		Operation M & E Parameter / TMTH			% Level of significance	% Success Value	
		Operation Activation Rule Code	Operation Activation Rule	KPI Code	Planned KPI Value	Actual KPI Value			
O31	Punch Manufacturing	OARC12	After Mold Setup	KPI31	100%	42,00%	20,00%	8,40%	
O34	Bend Manufacturing	OARC14	After Bend Mold Setup	KPI34	100%	69,92%	20,00%	13,98%	
O35	Jet Manufacturing	OARC10	Manufacturing Scheduling	KPI35	100%	78%	10,00%	7,8%	
O37	Tig Manufacturing	OARC15	After Tig Setup	KPI37	100%	96%	20,00%	19.2%	
O39	Painting operation	OARC16	After Painting Setup	KPI39	100%	100%	20,00%	20,00%	
O40	Transfer operation	OARC17	After each WIP Manufacturing	KPI40	100%	100%	10,00%	10,00%	
Success value comes fof P8:			77,95%	% Level of significance			**100%**		
Related Processes Code	Process Name					Actual KPI Value	% Level of significance	% Success Value	
Success value comes from the related processes:									
Total Operations Success								**79,38%**	

APPENDİX 5.Pre-defined Operation Cards (Part 1-O31)

OPERATIONS CARD			
Operation Name:	**Punch Manufacturing**	**No:**	00001-O1
Card Code:	**O31**	Date:	
Operation Definition:		Operation Sequence Level	**2**

		Machine Availability				Basic Information of Machine		Maintenance Information of Machines				Machine-Labor Relation		KPI Measurement				
Related Machine Codes	Machine Name	Machine State	State Duration	Available Time	Available Duration	Capacity hour / TD	Priority Level	Maintenance Period	Maintenance Code	Needed maintenance Time	Next maintenance Time	Appropriate Lobar set	Needed Unit /hour	KPI Code	Planned KPI Value	Actual KPI Value	% Level of significance	Success Value
PNC1	Punch Machine 1	1	250	-	-	16	2	3	PM	2	01.09.14	PNO	8	KPI106	250	105	20,00%	0,116
PNC2	Punch Machine 2	1	250	-	-	16	1	3	PM	2	01.09.14	PNO	8	KPI107	250	108	20,00%	0,1136
PNC3	Punch Machine 3	1	250	-	-	16	2	3	PM	2	01.09.14	PNO	8	KPI108	250	107	20,00%	0,1144
PNC4	Punch Machine 4	1	250	-	-	16	2	3	PM	2	01.09.14	PNO	8	KPI109	250	103	20,00%	0,1176
PNC5	Punch Machine 5	1	250	-	-	16	2	3	PM	2	01.09.14	PNO	8	KPI110	250	102	20,00%	0,1184
Total Success Value of O1																	**158,00%**	

APPENDİX 5.Pre-defined Operation Cards (Part 2-O33)

OPERATIONS CARD

Operation Name:	**Bend Mold Setup**	**No:**	00001-O1
Card Code:	**O33**	Date:	
Operation Definition:		Operation Sequence Level	**1**

		Machine Availability				Basic Information of Machine		Maintenance Information of Machines				Machine-Labor Relation		KPI Measurement				
Related Machine Codes	Machine Name	Machine State	State Duration	Available Time	Available Duration	Capacity hour / TD	Priority Level	Maintenance Period	Maintenance Code	Needed maintenance Time	Next maintenance Time	Appropriate Lobar set	Needed Unit /hour	KPI Code	Planned KPI Value	Actual KPI Value	% Level of significance	Success Value
BND1	Bend Machine 1	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	16	KPI117	6000	5270	20,00%	0,0243
BND2	Bend Machine 2	1	250	-	-	16	1	3	PM	3	01.09.14	BNO	8	KPI118	6000	5410	20,00%	0,0197
BND3	Bend Machine 3	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	8	KPI119	6000	4850	20,00%	0,0383
BND4	Bend Machine 4	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	16	KPI120	6000	4930	20,00%	0,0357
BND5	Bend Machine 5	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	8	KPI121	6000	4575	20,00%	0,0475
Total Success Value of O1																	**116,55%**	

APPENDİX 5.Pre-defined Operation Cards (Part 3-O34)

OPERATIONS CARD			
Operation Name:	**Bend Manufacturing**	**No:**	00001-O1
Card Code:	**O34**	Date:	
Operation Definition:		Operation Sequence Level	

Related Machine Codes	Machine Name	Machine Availability				Basic Information of Machine		Maintenance Information of Machines				Machine-Labor Relation		KPI Measurement			% Level of significance	Success Value
		Machine State	State Duration	Available Time	Available Duration	Capacity hour / TD	Priority Level	Maintenance Period	Maintenance Code	Needed maintenance Time	Next maintenance Time	Appropriate Lobar set	Needed Unit /hour	KPI Code	Planned KPI Value	Actual KPI Value		
BND1	Bend Machine 1	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	16	KPI122	250	176	20,00%	0,0592
BND2	Bend Machine 2	1	250	-	-	16	1	3	PM	3	01.09.14	BNO	8	KPI123	250	173	20,00%	0,0616
BND3	Bend Machine 3	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	8	KPI124	250	183	20,00%	0,0536
BND4	Bend Machine 4	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	16	KPI125	250	179	20,00%	0,0568
BND5	Bend Machine 5	1	250	-	-	16	2	3	PM	3	01.09.14	BNO	8	KPI126	250	163	20,00%	0,0696
Total Success Value of O1																	**130,08%**	

APPENDİX 6.Pre-defined Machine Cards (Part 1-PNC2)

MACHINE CARD										**No**	00001-PNC2
Machine Name		**Punch Machine 2**								**Date**	**01.01.2014**
Card Code:		**PNC2**									
# of Mold space in one :		Width :11050 KG			X-Axis :50"		Y-Axis : 100"	Sheet Thickness: 0.31"			
Operation Types:		**Screw press, punching, countersink**									
		Equipment Capacity		Equipment Availability				KPI Measurement			
PNC Equipment Code	Equipment Name	# of Unit produced / one operation	Operation Cycle Times / TSD	Unit 1 State	Unit 1 State Duration	Purchase Date / TYR	Unit 1 Shelf Life	KPI Code	Planned KPI Value	Actual KPI Value	Success Value
MD114	S135	3	480	0	0	2012	2016	KPI139	2400	1850	23%
MD115	S189 11.1x23.0	3	480	0	0	2012	2016	KPI139	2400	1750	27%
MD116	S249 R2	3	480	0	0	2012	2016	KPI139	2400	1740	28%
MD117	S250 R3	3	480	0	0	2012	2016	KPI139	2400	1780	26%
MD118	S285 23.0x8.1	3	480	0	0	2012	2016	KPI139	2400	1815	24%
MD119	S286 27.7x8.4	3	480	0	0	2012	2016	KPI139	2400	1965	18%
MD120	S287 32.4x8.7	3	480	0	0	2012	2016	KPI139	2400	1840	23%
MD121	S291	3	480	0	0	2012	2016	KPI139	2400	2050	15%
MD122	S292	3	480	0	0	2012	2016	KPI139	2400	1835	24%
MD23	Oblon 8.3x4.3	3	40	0	0	2012	2016	KPI139	200	165	18%
MD24	Quadrat 10	5	40	0	0	2012	2016	KPI139	200	180	10%
MD25	Quadrat 11	4	40	0	0	2012	2016	KPI139	200	165	18%
MD26	Quadrat 20	8	40	0	0	2012	2016	KPI139	200	145	28%
Total Machine 1 Success Value											**121,52%**

APPENDİX 6.Pre-defined Machine Cards (Part 2-PNC3)

MACHINE CARD				**No**	00001-PNC3
Machine Name	**Punch Machine 3**			**Date**	**01.01.2014**
Card Code:	**PNC 3**				
# of Mold space in one :	Width :11050 KG	X-Axis :50"	Y-Axis : 100"	Sheet Thickness: 0.31"	
Operation Types:	**Screw press, punching, countersink**				

PNC Equipment Code	Equipment Name	Equipment Capacity		Equipment Availability				KPI Measurement			
		# of Unit produced / one operation	Operation Cycle Times / TSD	Unit 1 State	Unit 1 State Duration	Purchase Date / TYR	Unit 1 Shelf Life	KPI Code	Planned KPI Value	Actual KPI Value	Success Value
MD1	3*15 Köprü	3	480	0	0	2012	2016	KPI139	2400	2255	6%
MD2	DD205 20.5x17.5	3	480	0	0	2012	2016	KPI139	2400	2300	4%
MD3	F_079	2	480	0	0	2012	2016	KPI139	2400	2150	10%
MD4	F_087	2	480	0	0	2012	2016	KPI139	2400	2350	2%
MD5	F078	2	480	0	0	2012	2016	KPI139	2400	2200	8%
MD6	F089	1	480	0	0	2012	2016	KPI139	2400	2140	11%
MD110	Rund 8.4	4	40	0	0	2012	2016	KPI139	200	190	5%
MD111	Rund 8.5	8	40	0	0	2012	2016	KPI139	200	185	8%
MD112	Rund 9	10	40	0	0	2012	2016	KPI139	200	195	3%
MD113	Rund 9.2	12	40	0	0	2012	2016	KPI139	200	210	-5%
MD114	S135	5	480	0	0	2012	2016	KPI139	2400	2140	11%
MD115	S189 11.1x23.0	6	480	0	0	2012	2016	KPI139	2400	2145	11%
MD116	S249 R2	9	480	0	0	2012	2016	KPI139	2400	1980	18%
Total Machine 1 Success Value											**106,99%**

APPENDİX 6.Pre-defined Machine Cards (Part 3-PNC4)

MACHINE CARD										**No**	00001-PNC4	
Machine Name		**Punch Machine 4**								**Date**	**01.01.2014**	
Card Code:		**PNC4**										
# of Mold space in one :		Width :11050 KG		X-Axis :50"		Y-Axis : 100"		Sheet Thickness: 0.31"				
Operation Types:		**Screw press, punching, countersink**										
		Equipment Capacity		Equipment Availability				KPI Measurement				
PNC Equipment Code	Equipment Name	# of Unit produced / one operation	Operation Cycle Times / TSD	Unit 1 State	Unit 1 State Duration	Purchase Date / TYR	Unit 1 Shelf Life	KPI Code	Planned KPI Value	Actual KPI Value	Success Value	
MD62	Round 37.5	3	40	0	0	2012	2016	KPI139	200	140	30%	
MD63	Round 7.95	2	40	0	0	2012	2016	KPI139	200	170	15%	
MD64	Rund 10	5	40	0	0	2012	2016	KPI139	200	140	30%	
MD8	Formi 041	4	480	0	0	2012	2016	KPI139	2400	1415	41%	
MD9	Formi 064	6	480	0	0	2012	2016	KPI139	2400	1510	37%	
MD67	Rund 11	12	40	0	0	2012	2016	KPI139	200	145	28%	
MD68	Rund 12	10	40	0	0	2012	2016	KPI139	200	115	43%	
MD8	Formi 041	8	480	0	0	2012	2016	KPI139	2400	1490	38%	
MD9	Formi 064	6	480	0	0	2012	2016	KPI139	2400	1430	40%	
MD10	Formi 56.0x5.0	8	480	0	0	2012	2016	KPI139	2400	1430	40%	
MD11	Langloch 10x7	1	40	0	0	2012	2016	KPI139	200	180	10%	
MD12	Langloch 12x6	10	40	0	0	2012	2016	KPI139	200	130	35%	
MD13	Langloch 20x5	12	40	0	0	2012	2016	KPI139	200	145	28%	
Total Machine 1 Success Value											**131,88%**	

APPENDİX 7.Pre-defined Machine operations (Part 1- Punch Machine Operations)

Machine Code	Machine Name	Related Machine operation Code	Machine Operation Name	Operation setup Time
PNC1-PNC2-PNC3-PNC4-PNC5	All Punch Machines	MD1	3*15 cross over	480
		MD2	DD205 20.5x17.5	480
		MD3	F_079	480
		MD4	F_087	480
		MD5	F078	480
		MD6	F089	480
		MD7	F090	480
		MD8	Formi 041	480
		MD9	Formi 064	480
		MD10	Formi 56.0x5.0	480
		MD11	Langloch 10x7	40
		MD12	Langloch 12x6	40
		MD13	Langloch 20x5	40
		MD14	Oblon 10.0x5.0	40
		MD15	Oblon 11.2x6.2	40
		MD16	Oblon 14.5x6.5	40
		MD17	Oblon 15.0x4.0	40
		MD18	Oblon 15.0x7.5	40
		MD19	Oblon 15.0x8.5	40
		MD20	Oblon 15.5x8.5	40
		MD21	Oblon 16.5x4.5	40
		MD22	Oblon 8.0x5.5	40
		MD23	Oblon 8.3x4.3	40
		MD24	Quadrat 10	40
		MD25	Quadrat 11	40
		MD26	Quadrat 20	40
		MD27	Quadrat 30	40
		MD28	Quadrat 4	40
		MD29	Quadrat 40	40
		MD30	Quadrat 5	40
		MD31	Quadrat 50	40
		MD32	Quadrat 6	40
		MD33	Quadrat 7	40
		MD34	Quadrat 9.5	40
		MD35	Rechteck 10x2	40
		MD36	Rechteck 15x5	40
		MD37	Rechteck 20x5	40

APPENDİX 7.Pre-defined Machine operations (Part 2- Punch Machine Operations)

Machine Code	**Machine Name**	**Related Machine operation Code**	**Machine Operation Name**	**Machine Operation setup Time**
PNC1-PNC2-PNC3-PNC4-PNC5	All Punch Machines	MD38	Rechteck 30x5	40
		MD39	Rechteck 50x5	40
		MD40	Recta 12.0x3.0	40
		MD41	Recta 12.0x8.3	40
		MD42	Recta 14.0x2.0	40
		MD43	Recta 15.0x1.5	40
		MD44	Recta 15.0x10.0	40
		MD45	Recta 15.0x4.0	40
		MD46	Recta 15.5x2.5	40
		MD47	Recta 18.0x4.0	40
		MD48	Recta 25.0x2.0	40
		MD49	Recta 26.0x9.0	40
		MD50	Recta 3.0x2.0	40
		MD51	Recta 3.3x3.1	40
		MD52	Recta 30.0x25.0	40
		MD53	Recta 30.0x3.0	40
		MD54	Recta 5.0x2.5	40
		MD55	Recta 8.0x2.0	40
		MD56	Recta 8.0x3.0	40
		MD57	Recta 8.4x1.5	40
		MD58	Round 10.2	40
		MD59	Round 15.0	40
		MD60	Round 22.5	40
		MD61	Round 3.0	40
		MD62	Round 37.5	40
		MD63	Round 7.95	40
		MD64	Rund 10	40
		MD65	Rund 10.2	40
		MD66	Rund 10.5	40
		MD67	Rund 11	40
		MD68	Rund 12	40
		MD69	Rund 12.5	40
		MD70	Rund 13	40
		MD71	Rund 13.1	40
		MD72	Rund 13.5	40
		MD73	Rund 14	40
		MD74	Rund 15	40
		MD75	Rund 16	40

APPENDİX 7.Pre-defined Machine operations (Part 3- Punch Machine Operations)

Machine Code	Machine Name	Related Machine operation Code	Machine Operation Name	Operation setup Time
PNC1-PNC2-PNC3-PNC4-PNC5	All Punch Machines	MD76	Rund 2.0	40
		MD77	Rund 2.5	40
		MD78	Rund 2.6	40
		MD79	Rund 2.7	40
		MD80	Rund 20	40
		MD81	Rund 23	40
		MD82	Rund 24	40
		MD83	Rund 25	40
		MD84	Rund 29	40
		MD85	Rund 3.0	40
		MD86	Rund 3.2	40
		MD87	Rund 3.5	40
		MD88	Rund 4	40
		MD89	Rund 4.2	40
		MD90	Rund 4.3	40
		MD91	Rund 4.4	40
		MD92	Rund 4.5	40
		MD93	Rund 5.0	40
		MD94	Rund 5.2	40
		MD95	Rund 5.4	40
		MD96	Rund 5.5	40
		MD97	Rund 5.6	40
		MD98	Rund 6	40
		MD99	Rund 6.1	40
		MD100	Rund 6.2	40
		MD101	Rund 6.3	40
		MD102	Rund 6.5	40
		MD103	Rund 7	40
		MD104	Rund 7.2	40
		MD105	Rund 7.4	40
		MD106	Rund 7.5	40
		MD107	Rund 8	40
		MD108	Rund 8.1	40
		MD109	Rund 8.2	40
		MD110	Rund 8.4	40
		MD111	Rund 8.5	40
		MD112	Rund 9	40
		MD113	Rund 9.2	40

APPENDİX 7.Pre-defined Machine operations (Part 4- Punch Machine Operations)

Machine Code	Machine Name	Related Machine operation Code	Machine Operation Name	Operation setup Time
PNC1-PNC2-PNC3-PNC4-PNC5	All Punch Machines	MD114	S135	480
		MD115	S189 11.1x23.0	480
		MD116	S249 R2	480
		MD117	S250 R3	480
		MD118	S285 23.0x8.1	480
		MD119	S286 27.7x8.4	480
		MD120	S287 32.4x8.7	480
		MD121	S291	480
		MD122	S292	480
		MD123	Squar 8.5	40
		MD124	Squar 9.5	40
		MD125	Oblon 10.5x7.5	40
		MD126	Recta 14.0x6.0	40
		MD127	Round 11.1	40
		MD128	Langloch 10x6	40

APPENDİX 7.Pre-defined Machine operations (Part 5- Bend Machine Operations)

Machine Family Code	Operation Code	Operation Name (CNC BEND)	Operation Time (min.)
BND	BND1	BIU-011; 805 mm bending	20
	BND2	BIU-011;100 mm bending	20
	BND3	BIU-011;1300 mm bending	20
	BND4	BIU-011;1410 mm bending	20
	BND5	BIU-011;200 mm bending	20
	BND6	BIU-011;400 mm bending	20
	BND7	BIU-011;425 mm bending	20
	BND8	BIU-011;460 mm bending	20
	BND9	BIU-011;485 mm bending	20
	BND10	BIU-011;500 mm bending	20
	BND11	BIU-011;600 mm bending	20
	BND12	BIU-011;800 mm bending	20
	BND13	BIU-011;900 mm bending	20
	BND14	BIU-011;955 mm bending	20
	BND15	BIU-013;100 mm bending	20
	BND16	BIU-013;1000 mm bending	20
	BND17	BIU-013;1315 mm bending	20

APPENDİX 7.Pre-defined Machine operations (Part 6- Jet Machine Operations)

Machine	Operation Code	Operation Name
J1- J2	JO1	Trimming

APPENDİX 7.Pre-defined Machine operations (Part 7- Tig Machine Operations)

Machine Family Code	Operation Code	Operation Name
TM	K1	TİG Welding
	K2	Spot welding

APPENDİX 8.Pre-defined Product Card – PR2

PRODUCT CARD				
Prodct Name	**48417250GGB-24**		**No**	00001-PR2
Card Code:	**PR2**		**Date**	**01.01.2014**
Product Description		**Shelf life:5 years**		**Stock Quantity :0**

Route Sequence		0			1										2					3		4				5				
Route Name		**PREPARING**			**PUNCH**										**BEND**					**JET**		**TIG**				**ASSEMBLY**				
Sub Product Code	Sub Product Name	N. R. M.1	N. M. Volume /Unit	BOM Unit	N. PCH.O.1	N. PCH.O.2	N. PCH.O.3	N. PCH.O.4	N. PCH.O.5	N. PCH.O.6	N. PCH.O.7	N. PCH.O.8	Unit Cycle Time / TMT	Unit Setup Time / TMT	BOM Unit	N. BND.O.1	N. BND.O.2	Unit Cycle Time / TMT	Unit Setup Time / TMT	N. J. O.1	Unit Cycle Time / TMT	BOM Unit	N. TM. O.1	Unit Setup Time / TMT	Unit Cycle Time / TMT	BOM Unit	N. ASLO. O.1	Assembly Level	Unit Setup Time / TMT	Unit Cycle Time / TMT
SPR1	48417250GHCA	RSM G4	0,2	1	MD36	MD85	MD88	MD89	MD92	MD105	MD33	MD48	0,5	5,3	1	BND33	BND35	0,98	20	J2	0,5	1	TM1	0,5	1,4	1	L2	1	0,5	1
SPR2	48417250GHDA	RSM G2	0,1	1	MD88	MD30	MD23	MD8	MD36	MD39			0,4	4,0	1	BND5	BND36	0,3	45	J2	0,6	1	TM1	0,5	1,4	1	L2	1	1,0	1
SPR3	48417250GHBB	RSM G4	3	1	MD77	MD91	MD92	MD95	MD108	MD28	MD32	MD34	0,8	5,3	1	BND34		0,76	5	J2	0,5	1	TM1	0,5	1,5	1	L2	1	0,4	3
SPR4	48417222GJAA	RSM G5	0,5	1	MD98	MD102	MD116	MD30	MD11	MD21	MD36	MD39	0,1	12	1	BND33	BND35	0,24	25	J2	0,8	1	TM1	0,5	0,2	1	L2	2	0,4	1
SPR5	48417222GFAC_200	RSM G4	0,7	2	MD95	MD82	MD30	MD37	MD39				0,07	3,3	2	BND5	BND36	0,09	25	J2	0,6	2	TM1	0,5	0,1	2	L2	2	0,4	1
KPI Code					KPI138										KPI138					KPI138		KPI138				KPI138				
Planned KPI Value					7										5,0					4		5,93				19				
Actual KPI Value					**5,4**										4,5					4		5,93				19				
Total Success value of PR1																										**108,22%**				

APPENDİX 9.Pre-defined Supllier Cards – (Part 1- S2)

SUPPLIER CARD

Supplier Name	**Company B**		No:		00001-S2
Card Code:	**S2**		Date:		01.01.2014
Address	**Bostancı**	**City:**	**Istanbul**	**Country:**	**Turkey**
Contact Person		**Contact Phone:**		**Email:**	

Material Code	Name	Quality Test Code	Measurement Parameters-1			Technical Parameters				Shipment Type	Delivery Time / Lot	Total Capacity / TMTH	Used Capacity / TMTH	Available Capacity / TMTH	PQPT Measurement			
			Price / Lot (TL)	Quality Value	Payment Term	Width	Height	Thickness	Color						PQPT Value	Sequence in PQPT list	Opportunity Cost	Sequence in Price list
RSMG1	0.8 mm Sheet Metal Gage	T1	1,3	89	2	-	-	0,8	-	ST1	12	500	-	500	24,7	2	-	4
RSMG3	1.2 mm Sheet Metal Gage	T1	0,7	89	2	-	-	1,2	-	ST1	12	500	-	500	25,8	1	-	2
RSMG4	1.5 mm Sheet Metal Gage	T1	1,2	89	2	-	-	1,5	-	ST1	5	500	-	500	24,8	2	10920	3

Total Success Value of S2	**60%**

APPENDİX 9.Pre-defined Supllier Cards – (Part 2- S3)

SUPPLIER CARD

Supplier Name	**Company C**			No:		00001-S3
Card Code:	**S3**			Date:		01.01.2014
Address	**Ümraniye**	**City:**	**Istanbul**	**Country:**		**Turkey**
Contact Person		**Contact Phone:**		**Email:**		

			Measurement Parameters-1			Technical Parameters									PQPT Measurement			
Material Code	Name	Quality Test Code	Price / Lot (TL)	Quality Value	Payment Term	Width	Height	Thickness	Color	Shipment Type	Delivery Time / Lot	Total Capacity / TMTH	Used Capacity / TMTH	Available Capacity / TMTH	PQPT Value	Sequence in PQPT list	Opportunity Cost	Sequence in Price list
RSMG1	0.8 mm Sheet Metal Gage	T1	1	90	0	-	-	0,8	-	ST1	11	600	-	600	25	1	-	1
RSMG2	1 mm Sheet Metal Gage	T1	0,8	90	0	-	-	1	-	ST1	3	600	-	600	25,4	1	-	2
RSMG3	1.2 mm Sheet Metal Gage	T1	1,2	90	0	-	-	1,2	-	ST1	5	600	-	600	24,7	4	-	5
RSMG6	3 mm Sheet Metal Gage	T1	1,5	90	0	-	-	3	-	ST1	8	600	-	600	24,5	1	-	2
Total Success Value of S3																	**70%**	

APPENDİX 9.Pre-defined Supllier Cards – (Part 3- S4)

SUPPLIER CARD																		
Supplier Name	**Company D**														No:		00001-S4	
Card Code:	**S4**														Date:		01.01.2014	
Address	**Kadıköy**					**City:**				**Istanbul**			**Country**				**Turkey**	
Contact Person						**Contact Phone:**							**Email**					
			Measurement Parameters-1			Technical Parameters									PQPT Measurement			
Material Code	Name	Quality Test Code	Price / Lot (TL)	Quality Value	Payment Term	Width	Height	Thickness	Color	Shipment Type	Delivery Time / Lot	Total Capacity / TMTH	Used Capacity / TMTH	Available Capacity / TMTH	PQPT Value	Sequence in PQPT list	Opportunity Cost	Sequence in Price list
RSMG1	0.8 mm Sheet Metal Gage	T1	1,1	90	1	-	-	0,8	-	ST1	7	300	-	300	25	1	-	2
RSMG3	1.2 mm Sheet Metal Gage	T1	9	90	1	-	-	1,2	-	ST1	13	300	-	300	25,3	2	-	5
RSMG2	1 mm Sheet Metal Gage	T1	1,3	90	1	-	-	1	-	ST1	5	300	-	300	24,8	2	1152	5
RSMG6	3 mm Sheet Metal Gage	T1	1,6	90	1	-	-	3	-	ST1	8	300	-	300	24,5	1	-	3
Total Success Value of S4																	**45%**	

APPENDİX 10. Java Framework

Java framework is constructed to execution. Framework consists of 22 class and 296 methods.

1.public class Initalizer

File file = new File("docs/Database10.08.12-4.xls");

public void getBenderMachines(){

public void getJetMachines()

public void getTigMachines(){

public void getMachines(){

public ArrayList<Tool> sortTools(ArrayList<Tool> tools){

public void getProducts(){

public double formatNumber(String s){

public boolean isEmpty(Object o){

public void getTools(){

public ArrayList<Tool> getMaxX(ArrayList<Tool> tools,int size){

2.public class Puncher

public Puncher()

public void produce()

public void initalizeMachines(double sTime)

public void print()

public Machine bestMachine()

public BenderMachine bestBenderMachine()

public KenarJetMachine bestJetMachine(){

public TigMachine bestTigMachine()

public static void main(String[] args)

public Product getSingleProduct(String productName)

public void getOrders(File file)

public void upDateProduct(Product product,double qu)

3.public class MyLabelGenerator implements XYItemLabelGenerator

public MyLabelGenerator(ArrayList<ArrayList<String>> labels)

public String generateLabel(XYDataset data, int x, int y)

4.public class <u>XYTaskDatasetDemo1</u> extends ApplicationFrame

public XYTaskDatasetDemo1(String s,ArrayList<ProcessItem> items)

public JFreeChart createChart(IntervalXYDataset intervalxydataset) public JPanel createDemoPanel()

public IntervalXYDataset createDataset()

public TaskSeriesCollection createTasks()

public ArrayList<ProcessItem> getProcessItems() {

public void setProcessItems(ArrayList<ProcessItem> processItems)

5.public class BenderMachine

public BenderMachine(int id)

public void bend(Part part,Product product)

public int getId()

public void setId(int id)

public double getTotalTime() {

public void setTotalTime(double totalTime)

public Part getCurrentPart()

public void setCurrentPart(Part currentPart)

public double getCurrentTime()

public void setCurrentTime(double currentTime)

public double getTotalSetUpTime()

public void setTotalSetUpTime(double totalSetUpTime)

public double getSetUpTime()

public void setSetUpTime(double setUpTime)

public BenderTool getAboveBendTool()

public void setAboveBendTool(BenderTool aboveBendTool)

public BenderTool getBottomBendTool()

public void setBottomBendTool(BenderTool bottomBendTool)

public ArrayList<Part> getParts()

public void setParts(ArrayList<Part> parts)

public double getLastProcessTime()

public void setLastProcessTime(double lastProcessTime)

6.public class BenderTool

public BenderTool(String s,double size,boolean a)

public boolean isAbove()

public void setAbove(boolean above)

public String getName()

public void setName(String name)

public double getSheetSize()

public void setSheetSize(double sheetSize)

7.public class KenarJetMachine

public KenarJetMachine(int id)

public void weld(Part part,Product product)

public int getId()

public double getTotalTime()

public void setTotalTime(double totalTime)

public Part getCurrentPart()

public void setCurrentPart(Part currentPart)

public double getCurrentTime()

public void setCurrentTime(double currentTime)

public ArrayList<Part> getParts()

public void setParts(ArrayList<Part> parts)

public double getLastProcessTime()

public void setLastProcessTime(double lastProcessTime)

8.public class LaserMachine

public LaserMachine(int id)

public void punch(Part part,Product product)

public int getId()

public void setId(int id)

public double getTotalTime()

public void setTotalTime(double totalTime)

public Part getCurrentPart()

public void setCurrentPart(Part currentPart)

public double getCurrentTime()

public void setCurrentTime(double currentTime)

public double getTotalSetUpTime()

public void setTotalSetUpTime(double totalSetUpTime)

public double getLastProcessTime()

public void setLastProcessTime(double lastProcessTime)

9.public class Machine

public Machine(int id)

public void punch(Part part,Product product)

public double getToolsSetupTime(Part part)

public boolean toolExist(Tool tool)

public int getId()

public void setId(int id)

public double getTotalTime()

public void setTotalTime(double totalTime)

public Part getCurrentPart()

public void setCurrentPart(Part currentPart)

public double getCurrentTime()

public void setCurrentTime(double currentTime)

public Part getPreviousPart()

public void setPreviousPart(Part previousPart)

public void addTool(Tool tool)

public Tool getTool(int id)

public void setTools(ArrayList<Tool> tools)

public double getTotalSetUpTime()

public void setTotalSetUpTime(double totalSetUpTime)

public double getTotalSavedTime()

public void setTotalSavedTime(double totalSavedTime)

public double getLastProcessTime()

public void setLastProcessTime(double lastProcessTime)

10.public class Part

public Part(String name,int id)

public int getToolsCount()

public void sortTools()

public int getId()

public void setId(int id)

public String getName()

public void setName(String name)

public ArrayList<Tool> getTools()

public void setTools(ArrayList<Tool> tools)

public Part(int id,String name)

public void addTool(Tool tool)

public Tool getTool(int id)

public int getRecipeNum()

public void setRecipeNum(int recipeNum)

public boolean isLaser()

public void setLaser(boolean isLaser)

public double getUnitProduceTime()

public void setUnitProduceTime(double unitProduceTime)

public double getSetUpTime()

public void setSetUpTime(double setUpTime)

public double getBendProcessTime()

public void setBendProcessTime(double bendProcessTime)

public double getBendSetUpTime()

public void setBendSetUpTime(double bendSetUpTime)

public double getKenar()

public void setKenar(double kenar)

public double getJet()

public void setJet(double jet)

public double getTig()

public void setTig(double tig)

public double getWeldTime()

public double getSheetSize()

public void setSheetSize(double sheetSize)

public int getOneTimeProduce()

public void setOneTimeProduce(int oneTimeProduce)

public double getAdditionalLostTime()

public double getOneTimeProduceTime()

public double getLaserSetUpTime()

public void setLaserSetUpTime(double laserSetUpTime)

public double getStatus()

public void setStatus(double status)

public int getTotalQuantity()

public void setTotalQuantity(int totalQuantity)

public BenderTool getAboveBendTool()

public void setAboveBendTool(BenderTool aboveBendTool)

public BenderTool getBottomBendTool()

public void setBottomBendTool(BenderTool bottomBendTool)

public double getStartTime()

public void setStartTime(double startTime)

public double getFinishTime()

public void setFinishTime(double finishTime)

}

11.public class ProcessItem

public ProcessItem(String pro,Product prod,Part par,int ma,double s,double f,double pTime)

public String getProcess()

public void setProcess(String process)

public Product getProduct()

public void setProduct(Product product)

public Part getPart()

public void setPart(Part part)

public double getStart()
public void setStart(double start)
public double getFinish()
public void setFinish(double finish)
public double getProcessTime()
public void setProcessTime(double processTime)
public int getMac()
public void setMac(int mac)

12.public class Product

public Product(String name,int id,int prodNum)
public void sortParts()
public Part giveMax()
public Part giveMaxBend()
public Part giveMaxJet()
public Part giveMaxTig()
public void addPart(Part part)
public Part getPart(int id)
public ArrayList<Part> getParts()
public void setParts(ArrayList<Part> parts)
public String getName()
public void setName(String name)
public int getId()
public void setId(int id)
public double getProdNum()
public void setProdNum(double prodNum)

13.public class TigMachine

public TigMachine(int id)
public void weld(Part part,Product product)
public int getId()
public void setId(int id)
public double getTotalTime()
public void setTotalTime(double totalTime)
public Part getCurrentPart()
public void setCurrentPart(Part currentPart)
public double getCurrentTime()
public void setCurrentTime(double currentTime)
public ArrayList<Part> getParts()

public void setParts(ArrayList<Part> parts)

public double getLastProcessTime()

public void setLastProcessTime(double lastProcessTime)

14.public class Tool

public Tool(String name,int id,double setUpTime)

public int getId()

public void setId(int id)

public String getName()

public void setName(String name)

public double getSetUpTime()

public void setSetUpTime(double setUpTime)

public int getCount()

public void decCount()

public void incCount()

public void setCount(int count)

15.public class Main

getSuppliers(new File("docs/tedarikçi.xls");

public void getSuppliers(File file)

public SearchItem findBestSupplier(Product p,OrderedProduct op,Order order)

public static void main(String[] args)

public Supplier getSuppllier(String name)

public ArrayList<SearchItem> findSearchItems(Order order)

public ArrayList<Supplier> findSuppliersWithMaterail(String materialName)

16.public class Material {

public Material(String n,double s,double t,double u)

public String getName()

public void setName(String name)

public double getSupplyTime()

public void setSupplyTime(double supplyTime)

public Property getProperty(String name)

public double getSheetSize()

public void setSheetSize(double sheetSize)

public double getUnitPrice()

public void setUnitPrice(double unitPrice)

17.public class MaterialSearch

public MaterialSearch(ArrayList<SearchItem> searchItems)

public void printOrders(ArrayList<SearchItem> items)

public void printOrdersByFKV(ArrayList<SearchItem> items)

public void printOrdersByOpCost(ArrayList<SearchItem> items)

public ArrayList<SearchItem> eliminateForProcessTime()

public ArrayList<SearchItem> eliminateForFKV(ArrayList<SearchItem> items,Product product)

public ArrayList<SearchItem> groupByMaterial(String materialName,ArrayList<SearchItem> items){

public ArrayList<SearchItem> getBestMaterialsForFKV(ArrayList<SearchItem> items)

public ArrayList<SearchItem> getBestMaterialsForTotalCost(ArrayList<SearchItem> items)

18.public class Order {

public ArrayList<OrderedProduct> products=new ArrayList<OrderedProduct>()

public double getQuantity()

public void setQuantity(double quantity)

public double getDueDate()

public void setDueDate(double time)

public String getName()

public void setName(String name)

19.public class OrderedProduct

public String getProductName()

public void setProductName(String productName)

public double getProcessTime()

public void setProcessTime(double processTime)

public double getNeed()

public void setNeed(double need)

20.public class Property

public String getName()

public void setName(String name)

public String getValue()

public int getIntValue()

public double getDoubleValue()

public void setValue(String value)

21.public class Supplier {

public double getCanStart()

public void setCanStart(double canStart)

public double getK()

public void setK(double k)

public double getV()

public void setV(double v)

public String getName()

public void setName(String name)

public ArrayList<OrderedProduct> getProducts()

public void setProducts(ArrayList<OrderedProduct> products)

public Material getMaterial(String m)

22.public class SearchItem {

public SearchItem(Order order,OrderedProduct product,Material material,Supplier supplier)

public Order getOrder()

public void setOrder(Order order)

public OrderedProduct getProduct()

public void setProduct(OrderedProduct product)

public Material getMaterial()

public void setMaterial(Material material)

public Supplier getSupplier()

public void setSupplier(Supplier supplier)

public double getCost()

public void setCost(double cost)

public double getDif()

public void setDif(double dif)

public double getRawQuantity()

public void setRawQuantity(double rawQuantity)

public double getTotalCost()

public void setTotalCost(double totalCost)

public double getPurchasingCost()

public void setPurchasingCost(double purchasingCost)

public double getOpCost()

public void setOpCost(double opCost)

public double getFKV()

public void setFKV(double fKV)

BANU ÇALIŞ

EDUCATION

Marmara Üniversitesi
Master of Science (MSc), Engineering Management
2005 – 2008
Thesis Subject "A General Framework for Multi-Agent Manufacturing Systems"

Sakarya Üniversitesi
Bachelor of Science (BSc), Industrial Engineering
1996 – 2000
Graduation Project "Total Quality Management & Business Excellence (EFQM)"

PUBLICATIONS

1. Articles in Refereed Journals

Calis B., Bulkan S. & Tuncer F. (2015). “A HEURISTIC APPROACH FOR SINGLE MACHINE SCHEDULING PROBLEM TO MINIMIZE MAXIMUM LATENESS SUBJECT TO RELEASE DATES” South African Journal of Industrial Engineering accepted at 27 July 2015

Calis B.& Bulkan S. (2015). “A research survey: review of AI solution strategies of job shop scheduling problem” Journal of Intelligent Manufacturing Volume 26, Issue 5 (2015), pp 961-973. DOI 10.1007/s10845-013-0837-8 Accepted: 17 September 2013

Calis B., Gokmen N & Tozan H (2010). “Efficiency Measures in the Health Service with Data Envelopment Analysis - An Overview” Journal of Naval Science and Engineering 2010, Vol. 6 , No.1, pp. 1-14

2. **Conference Proceedings**

Calis B. (2015). “PROFIT BASED SCHEDULING USING AGENT BASED ARCHITECTURE: A Single Machine Problem” International Conference on Value Chain Sustainability 12 March- Istanbul

Calis B.& Bulkan S. (2012) “A Research Survey: Application of AI Strategies for Optimal Production Scheduling” 8th International Symposium on Intelligent and Manufacturing Systems 27 September - Antalya

Calis B & Er M. (2011), "Future Engineering Education" The International Higher Education Congress: New Trends and Issues (UYK-2011) 27 May- Istanbul

Calis B., Senvar Ö. Cigdem B.T. & Oktay Fırat S. Ü. (2010), "Applicability of Process Control Techniques in PET Strapping Production" Process 30. National Conference YA/EM2010 30 June- Istanbul

Oztemel E., & Calis B. (2008). "Role of Information Systems in Multi Agent Architecture" 6th International Symposium on Intelligent and Manufacturing Systems October 14 - Istanbul

Oztemel E. & Calis B. (2007). "Gelecekte Üretim Sistemleri ve Manufuture" 27. National Conference YA/EM2007 July 2- İzmir

EXPERIENCE IN BUSINESS

Research and Teaching Assistant
Marmara University
Aralık 2005 – cont.

Teaching Assistants;
IE 364 Introduction To Simulation
IE 333 Operations Research
IE 431 Decision Analysis
IE 497 Systems Analysis and Design

Administrative Duties;

1. **Coordinator;**

Course of Training	2010-2014 (IE300 and IE400)
Alumni association	2012-cont.
Faculty Orientation	2011-cont.
Timetable for IE	2006-2007 and 2007-2008

2. **Commission member ;**

MÜDEK	2015-
Member of Strategy Development Commission - Marmara University	2015-cont.
Commission of Summer Practice	2014- cont.

3. **International Organizing Committee**;

ICOVACS 2015 : International Conference on Value Chain Sustainability 12-13 March 2015 **Web: http://www.icovacs.com/en/COMMITTEES.html**

Production Planner

Polipak Plastik Ambalaj San.Ltd.Şti
2004 – 2005) Kocaeli, Turkey
Responsibilities: Production planning and Control, Material Requirement, Performance measurement and Time Study

Project Engineer

Tekseb-İltek Ltd Şti
2003 – 2004 Kocaeli, Turkey
Responsibilities: Project Planning, Coordination and Configuration

Sales and Planning Engineer

Tempa Pano San. ve Tic. A.S.
2002 – 2003 Kocaeli, Turkey
Responsibilities: Sales and Operations Planning