

This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

Knowledge discOvery And daTa minING inteGrated (KOATING) Moderators For Collaborative Projects

A.K.Choudhary¹, J.A.Harding^{1*}, H. K. Lin², M. K. Tiwari³ and R. Shankar⁴

ABSTRACT

A major issue in any multidiscipline collaborative project is how to best share and simultaneously exploit different types of expertise, without duplicating efforts or inadvertently causing conflicts or loss of efficiency through misunderstandings of individual or shared goals. Moderators are knowledge based systems designed to support collaborative teams by raising awareness of potential problems or conflicts. However, the functioning of a moderator is limited by the knowledge it has about the team members. Knowledge acquisition, learning and updating of knowledge are the major challenges for a Moderator's implementation. To address these challenges a Knowledge discOvery And daTa minING inteGrated (KOATING) framework is presented for Moderators to enable them to continuously learn from the operational databases of the company and semi-automatically update the their knowledge about team members. This enables the reuse of discovered knowledge from operational databases within collaborative projects. The integration of KDD techniques into the existing knowledge acquisition module of a moderator enable hidden data dependencies and relationships to be utilized to facilitate the moderation process. The architecture for the Universal Knowledge Moderator (UKM) shows how moderators can be extended to incorporate learning element which enables them to provide better support for virtual enterprises. Unified Modelling Language Diagrams were used to specify the ways to design and develop the proposed system. The functioning of a UKM is presented using an illustrative example.

Keywords: Collaborative projects, Learning, Moderators, Knowledge discovery, data mining, Universal knowledge Moderator, virtual enterprise

*Communicating Author: J.A.Harding@lboro.ac.uk

¹: Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough, Leicestershire, United Kingdom

²: Department of Industrial Engineering and Management, I-Shou University, Kaohsiung, Taiwan ROC

³: Department of Industrial Engineering, Indian Institute of Technology, Kharagpur, India.

⁴: Department of Management Studies, Indian institute of Technology, Delhi, India

1. Introduction

Increasingly competitive market trends demand highly customized products with ever shorter production time and these market trends are expected to accelerate. Consequently, modern business entities are challenged to identify effective means of reducing production cost, improving product and service quality, reducing time to market delivery, accelerating responses to customer requirement and bettering flexibility and system's reusability. Industries are striving to meet these challenges by focussing on their core competencies, integrating and collaborating intensively and migrating towards knowledge based manufacturing (Hicks *et al.* 2002, Lastra & Delamer 2006). (Ikujiro & Takeuchi 1995) observed that, as the market shifts, technologies proliferate, competitors multiply, and products and services become rapidly obsolete, successful companies are characterized by their ability to consistently create new knowledge, quickly disseminate it, and embody it in new products and services.

A major issue in multidiscipline collaborative projects is how best to share and simultaneously exploit different types of knowledge, without duplicating efforts or inadvertently causing conflicts or loss of efficiency through misunderstandings of individual or shared goals. The concepts of Moderators to support collaboration and team working have been researched in major research projects (MOSES 1995, MISSION Consortium 2001, Lin 2004). The main function of a moderator is to support collaborative working teams by raising individual members' awareness of the needs and experiences of other team members and the concept has been successfully demonstrated in product design, manufacturing system design, extended enterprise and e-supply chain. Prototype Moderators have been demonstrated in the form of knowledge based software support systems consisting of a moderation module, multiple expert modules and a knowledge acquisition module. Until now, all knowledge acquisition for the prototype moderators has been done manually, based on human expertise and experience.

This research enhances the functionality and capability of Moderators through the integration of a knowledge discovery based semi-automatic knowledge acquisition framework which enables Moderators to "learn" and "update" their relevant expert modules from knowledge discovered in the existing operational databases of companies. To accomplish the task, a knowledge discovery and data mining integrated (KOATING) framework has been developed. The proposed Moderator, equipped with knowledge discovery capability can increase awareness within the project teams by highlighting potential problem areas (from previous experience) and raise awareness of any issues affecting the team members.

2. Knowledge Management Systems to Support Collaborative Projects

Recent developments show that collaboration exists in many forms such as collaborative product design, collaborative product development (Li & Qiu 2006, Harding *et al.* 2007, Popplewell & Harding 1995), collaborative planning forecasting and replenishment (CPFR) (Skjoett-Larsen *et al.* 2003), virtual factory (VF) (Katzy & Dissel 2001, Katzy & Crowston, Katzy & Obozinski 1998), extended enterprises (EE) (Popplewell & Harding 2004), virtual enterprises (VE) (Browne & Zhang 1999), and virtual organizations (VO) (Camarinha-Matos & Afsarmanesh 2007b), virtual breeding environment (Camarinha-Matos & Afsarmanesh 2007b, ECOLEAD 2008, ECOLEAD 2005, Sánchez *et al.* 2005) etc. As discussed below, several types of knowledge management systems have been developed to support these collaborations.

When people from different disciplines, experiences and backgrounds try to work together in a collaborative project, there is potential for misunderstanding or lack of awareness of the needs and interdependencies of each of the individual contributors. The importance of awareness and understanding of other partners' requirements in collaborative projects was highlighted in the mid 1990s. In this context, Mediator was one of the earliest works proposed by (Gaines *et al.* 1995), it is an open architecture-based information and knowledge support system for geographically dispersed manufacturing processes ranging from requirements through design, engineering, production, maintenance and recycling. Also in the mid 1990s, the MOSES project in the UK funded by the Engineering Physical Science Research Council (EPSRC) introduced the concept of a specialist intelligent software system called a "Moderator", to increase understanding and awareness in concurrent engineering teams (Harding & Popplewell 1996).

Frécon & Nöu (1998) developed a distributed virtual environment to support collaborative work in teams that are geographically scattered. They supported synchronous as well as asynchronous group collaboration. Zhou & Nagi (2002) presented a distributed information system architecture using CORBA and STEP standards to overcome the heterogeneity of partners and promote standardization respectively for VE. In a research project partly funded by the European commission, Slade & Bokma (2002) discussed the application of an extensible ontology as a principle for integrated information, knowledge management and knowledge sharing among geographically distributed collaborators of an EE. However, they mainly focussed on sharing, organization, interrelation, and visualization of documents for team members. Shafiei & Sundaram (2004) proposed a multi-enterprise collaborative conceptual Enterprise Resource Planning (ERP)- Decision Support System (DSS) to

maximize the intelligence density, improve the quality and visibility of information and to achieve the foundation for multi-enterprise collaboration. Panteli & Sockalingam (2005) developed a framework for understanding the dynamics of trust and conflict within the context of virtual inter-organizational arrangements. Ahn *et al.* (2005) presented a knowledge context model to facilitate the use of contextual information in virtual collaborative work. A new software system called IntelliTeam was developed based on a web-based collaborative system framework using a multiple perspective approach (Cil *et al.* 2005). It consists of a group decision-making approach, many multiple criteria decision-making techniques, an intelligent system and advanced communication systems such as mobile e-services, wireless application protocol etc. Misono *et al.* (2005) proposed a distributed collaborative decision support system based on semantic web service technology to achieve collaborative goals.

In the UK, the *Advanced Knowledge Technologies* Interdisciplinary Research Collaboration (AKTIRC) was a £7.6 Million project funded by EPSRC to develop knowledge management technologies as part of the e-science initiative on grid computing. Collaborative Advanced Technologies in the Grid (CoAKTinG) aims to support distributed scientific collaboration (Shum *et al.* 2002). Collaboration is the centrepiece of product development processes and involves multidisciplinary teams, functions and heterogeneous tools. Ramesh & Tiwana (1999) viewed new product development as a knowledge sensitive activity and identified problems associated with knowledge management issues of new product development by cross functional collaborative teams. Huang *et al.* (2000) proposed workflow management as a mechanism to facilitate team work in the collaborative product development environment where remote web-based decision support systems (TeleDSS) are extensively used by geographically distributed team members.

Recently, the Decision Support Systems (DSS) journal published a special issue dealing with knowledge management and collaborative work related issues (Li & Lai 2005). However, none of the work discussed above dealt with the knowledge discovery issues to aid the decision making process in a collaborative projects. Scotney & McClean (2004) provided a flexible method of knowledge discovery from semantically heterogeneous data, based on the specification of ontology mapping. Wen *et al.* (2005) used web robots to discover the latest knowledge on the internet for better service of collaborative design. Numata *et al.* (1997) dealt with the knowledge conversion between and within tacit knowledge and explicit knowledge in new product development.

The European Collaborative networked Organizations LEADership initiative (ECOLEAD) was an “integrated project” funded by the European Commission within the 6th Framework Programme and involved 20 partners across 14 European countries. ECOLEAD results claim to provide a set of tools including a Dynamic VO creation assistance tool, a VO collaboration and performance measurement tool, a contract negotiation wizard tool, a VO management e-service tool, a collaborative problem solving support e-services tool, an advanced collaboration platform for professional virtual communities tools etc. (ECOLEAD 2008, ECOLEAD 2005, Ramesh & Tiwana 1999, Camarinha-Matos & Afsarmanesh 2007a, Pechoucek *et al.* 2005). However, so far, none of the published literature as an output of this project deals with the knowledge discovery issues to support collaboration in an industrial context (ECOLEAD 2005). Li & Qiu (2006) reviewed collaborative product development related works and based on the review of 130 papers, they concluded that one of the major issues for future collaborative system development is efficient learning and sharing of knowledge for multiple application domains.

2.1 Moderator Technology

A moderator is a specialist software system that supports collaborative working by raising awareness of the priorities and requirements of other contributors. The moderator concept was first proposed in the MOSES research project (MOSES 1995) as a support tool for design project teams. It addressed the fundamental requirements for provision of support for design teams working in a concurrent engineering environment, by encouraging and facilitating communication between team members, by making them aware of when their activities and decisions may be of interest to, or affect the work of, other team members (Harding & Popplewell 1996, Harding 1996, Harding *et al.* 2003, Lin & Harding 2003, Lin *et al.* 2004). Later the MISSION research project (MISSION Consortium 2001) saw the second phase of the development of the moderator concept where the Manufacturing System Engineering Moderator (MSEM) was designed as an intelligent support system to monitor design decisions, evaluate their significance to individual project members and when necessary communicate details of the identified problems to any affected team members (Harding *et al.* 2003). The third phase of the research on Moderators was carried out to determine how Moderator technology can be extended or modified to benefit extended enterprise and e-supply chain environments (Lin & Harding 2003). An MSE ontology model was proposed to provide a common understanding of manufacturing related terms and thereby enhance the semantic interoperability and reuse of knowledge resources within globally extended manufacturing teams or e-supply chains (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005).

The quality of the support that any Moderator can provide is limited by its knowledge of team members as collected by its Knowledge Acquisition Module (KAM) and stored in its Expert Modules, where each Expert Module is the Moderator's high level model or representation of a team member which includes "item of interest" to that team member and Moderator's knowledge of what need to be done to support that team member and make him aware of potential problems or opportunities which are relevant to him. To date, all the knowledge acquired for the Moderators has been provided by human experts using traditional knowledge acquisition methods. However, huge amounts of experience and expertise lie within the databases' of manufacturing operations. Therefore, knowledge discovery for moderation has been identified as a research gap in the existing research (Choudhary *et al.* 2006, Choudhary 2009).

3 Knowledge Discovery and Data Mining

3.1 Motivation for using KDD

In recent years, knowledge discovery in database (KDD) and data mining (DM) have attracted a great deal of attention in manufacturing enterprises due to wide availability of huge amounts of data and the challenges of converting it into useful and novel information and knowledge. The review section of this paper revealed that the Moderator system requires up to date knowledge and therefore Harding *et al.* (2007) recommended that the Moderators need the capability of ongoing learning. KDD techniques can help by (semi-) automating the time consuming process of knowledge acquisition and reducing the cost of development by decreasing the amount of time needed from experts and knowledge engineers. Implementation of KDD tools and techniques also has the potential to uncover knowledge that might otherwise be overlooked by those involved in the knowledge acquisition process.

3.2 KDD process

KDD refers to the overall process of discovering useful knowledge from data and data mining refers to a particular step in this multi step process. Data mining is the application of specific algorithms for extracting patterns from data (Piatetsky-Shapiro 1991, Fayyad *et al.* 1996). The KDD process includes several pre-processing methods aimed at facilitating the application of the chosen data mining algorithm and post processing methods aimed at refining and improving the discovered knowledge. KDD is interdisciplinary using methods from several research fields including machine learning, statistics, pattern recognition, databases technology, artificial intelligence, knowledge acquisition for expert systems, data visualization and high performance computing. The unifying goal is to extract high level knowledge from low level data within large data sets. The overall KDD process is interactive and iterative involving numerous steps and requiring several decisions to be made by the user

(Fayyad *et al.* 1996, Mitra *et al.* 2002). Data mining uses automated tools and employs sophisticated algorithms to discover hidden patterns, associations, anomalies and/or structure from large amounts of data stored in data warehouses or other information repositories. The goals of data mining can be achieved using a variety of data mining functions such as classification, regression, clustering, summarization, discovering association rule, dependency modelling, sequence analysis. Fayyad *et al.* (1996), Mitra *et al.* (2002), Harding *et al.* (2006), and Choudhary *et al.* (2008) extensively reviewed the application of data mining and identified the several challenges and good practices of data mining.

4 Knowledge discOvery And daTa minING inteGrated (KOATING) frame work for Moderators

All previous Moderators have used a knowledge based approach to capture the relevant knowledge and information by interviewing experts and storing the knowledge in an object oriented database (OODB) based on a Knowledge Representation Model (KRM) (Harding 1996). This manual update has been done by a human operator which is a complex and time consuming task. In addition, every piece of knowledge has a lifespan for its validity and therefore it is necessary to continuously review and update the knowledge contained in any knowledge based system. Figure 1 show the existing structure of a typical Moderator, where the knowledge in any expert module can be manually updated by the KAM.

In the proposed framework, a knowledge discovery module (KDM) with combined functionality of knowledge miners, a knowledge manager and repository provides continuous learning, thus enabling semi-automatic update of its expert modules from time to time. Enterprises continuously generate large amounts of data during their normal operation and this data can be a valuable asset and potentially important source of knowledge. Identification and retrieval of these knowledge assets may be achieved by applying intelligent data analysis approaches to databases with the objectives of identifying patterns, discovering rules and predicting results. Therefore by providing a semi-automatic mean to discover and apply new knowledge, proposed framework will be able to reduce the time intensive activity of knowledge discovery and updating the expert module. In addition, it is also recommended that a check be made on the newly generated knowledge to avoid misunderstandings or contradictory knowledge within the knowledge assets.

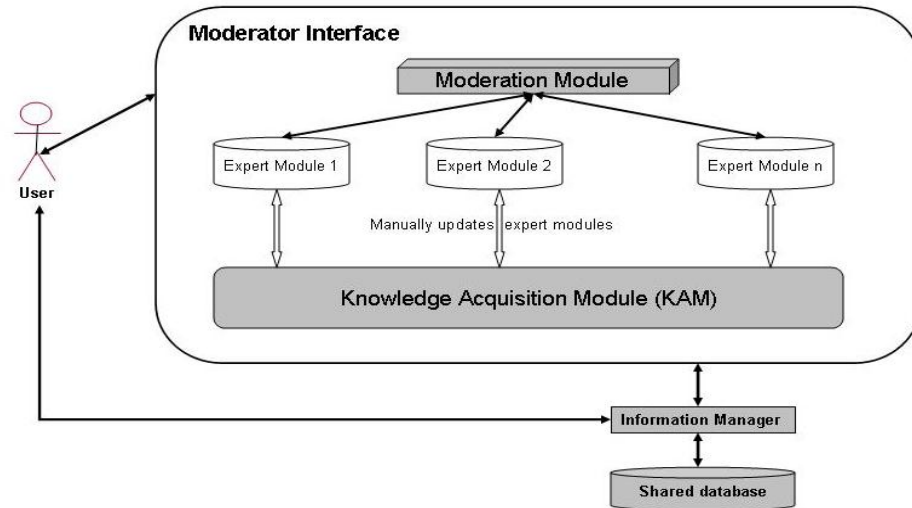


Figure 1: Existing structure of Moderator

4.1 Proposed KOATING Framework

The KOATING framework does not replace the existing KAM but supports it and enables knowledge update. The supporting module to perform this function has been termed as the knowledge discovery module (KDM) which is proposed as an integral part of KAM. The main contribution of this framework is the elements within the knowledge discovery module which can be seen by comparing Figure 1 and Figure 2. Table 2 shows a comparative study of traditional and proposed Moderator system.

Table 1: Comparison of traditional and proposed Moderator system

Traditional Moderator System (Fig 1)	Proposed Moderator system (Fig 2)
1. Knowledge is acquired through Knowledge Acquisition module	Knowledge Discovery Module is integrated with Knowledge Acquisition module for knowledge Acquisition.
2. Manual Update	Semi-Automatic update of knowledge
3. Knowledge is gathered through interview, human experience and traditional knowledge acquisition approaches.	Knowledge is discovered from different kind of databases associated with operation or team of collaborative project.
4. Time consuming process	Reduces the time of conducting interviews, however, knowledge discovered need to be verified by the expert.
5. Doesn't consider knowledge hidden in the operational or team related databases.	Discover hidden knowledge and patterns from databases using methods of knowledge discovery and data mining.
6. Unable to consider up to date knowledge	As the time passes, the knowledge might change with the change in operational data, proposed system uses knowledge miner to discover up to date knowledge with change in operational databases.

In the proposed framework, the project life cycle and operational data can be used to generate structured knowledge which should be at least as good as the knowledge provided by domain experts. Figure 2 shows the proposed KOATING framework which incorporates the features of a knowledge based system designed for individual as well as cooperative learning, knowledge reuse, and corresponding update of expert module knowledge within the Moderator system. The knowledge miners can use many different knowledge discovery and data mining tools to address the challenges of identifying and incorporating new knowledge within the existing expert modules.

As in the original Moderators, each expert module manually stores knowledge about a participating project team member. In the KOATING framework, it also semi-automatically stores the discovered knowledge delivered by the KDM. The bottom part of the framework in figure 2 shows the database of individual project member.

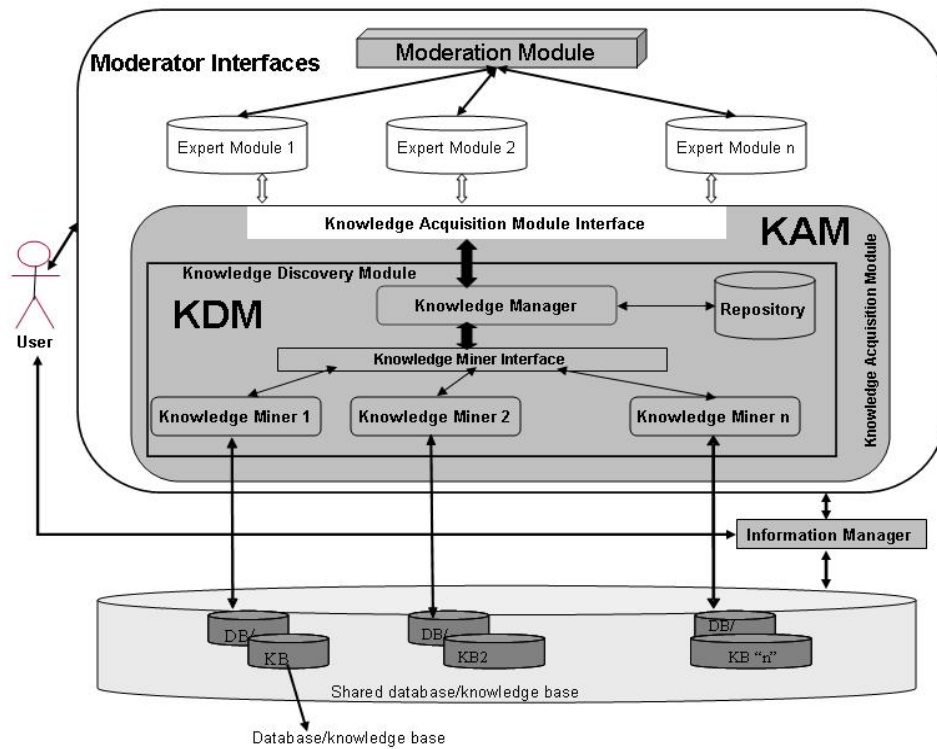


Figure 2: Proposed KOATING framework for Moderator

4.2 Knowledge Discovery Module of KOATING framework

The KDM supports the KAM by providing a semi-automated knowledge acquisition mechanism to identify and retrieve appropriate knowledge from available data sources and store them in a format appropriate for further use by the Moderator. The proposed KDM requires its integration with the Moderator as well as the data source. This research assumes that the data source is accessible for analysis purposes by the KDM. The KDM supports

knowledge discovery through all the period of learning and facilitates the update of the expert module(s) by the KAM. This is demonstrated using an illustrative example of virtual e-supply chain in section 7.

4.2.1 Knowledge Discovery Approach Embedded in KDM

The knowledge discovery process can be achieved through two different approaches namely Data Mining Software Tool Approach (DMSTA) and Data Mining Application System Approach (DMASA) (Holsheimer 1999). DMSTA approach involves the application of data mining software tools on ad hoc data mining projects and requires a significant expertise in data mining methods, databases and/or statistics. The disadvantages associated with this model include the need for several experts to collaborate in a project and poor transferability of results and models. This implies that the results and models derived can be used for reporting, but cannot be directly utilized to integrate with other systems (Holsheimer 1999).

In contrast, the DMASA approach primarily focuses on the requirement of knowledge users and decision makers to enable them to view and exploit data mining models. Models can be presented in a user understandable manner through a user friendly and intuitive GUI using standard and graphical presentation techniques. Knowledge can be discovered by focussing on a specific problem domain covered by areas of analysis with the possibility of repeated analysis at periodic time intervals, or when required by the user, or at a particular milestone such as at the end of projects. Several authors and practitioners have recommended this approach for better integration in the business environment and in decision processes (Holsheimer 1999). Therefore, this research has adopted the DMASA approach.

4.2.2 Process Model of KDM

The process model represents how the knowledge generation and decision making process is supported by the knowledge based system. Determining the process model is one of the key issues for the design of the KDM for Moderators. The CRISP-DM (Cross-Industry Standard Process for Data Mining) process model is a data mining process model developed by the industry leaders in collaboration with data mining experts, users and data mining tool providers (Shearer 2005). The analysis of various other data mining models equivalent to CRISP-DM, identifies CRISP-DM as the most appropriate process model for knowledge based system implementation (Shearer 2005). In the present context, the CRISP-DM process model has been modified in order to make it applicable to the KDM. Unlike the CRISP-DM model, the process model for the KDM has been divided into seven phases and three stages. Each of these phases and stages includes a variety of tasks. The phases include: domain understanding, data understanding, data preparation, modelling, evaluation, deployment and conflict resolution. The sequence of the phases is not strict and moving back and forth

between various phases is always required. Based on the outcome of each phase the next phase or a particular task of a phase that needs to be performed can be decided. The arrows indicate the most important and frequent dependencies between phases. The inner circle shows the cyclical nature of the knowledge discovery process itself. It means that a knowledge discovery process continues even after knowledge is discovered during the deployment phase. The lessons learned and experiences gained during this whole process can benefit the subsequent data mining processes. The elements of the KOATING framework have been shown as actors in Figure 3 with the modified structure of CRISP-DM that has been adapted to the needs of the KDM for three stages:

- the preparation stage
- the knowledge production stage and
- the implementation stage

The preparation stage of the process model prepares the area of analysis for production and implementation uses. This stage focuses on performing the first five phases, i.e. from domain understanding to evaluation in an iterative manner. The major reason why multiple iterations may need to be carried out is to achieve step by step improvement in all the phases. Datasets must be created automatically on a periodic basis, say every night based on the current state of the existing databases, data warehouse and the transactional data. The problems identified in the data preparation phase may demand changes in the data understanding phase. Models may be created and evaluated multiple times in order to fine tune the data mining algorithms. Depending on the results gained by evaluation through multiple iterations, the preparation stage can either reject the area of analysis due to insufficient quality of model or approve it with or without a slight modification to the objectives for domain understanding and this may consequently require changes in other part of the process.

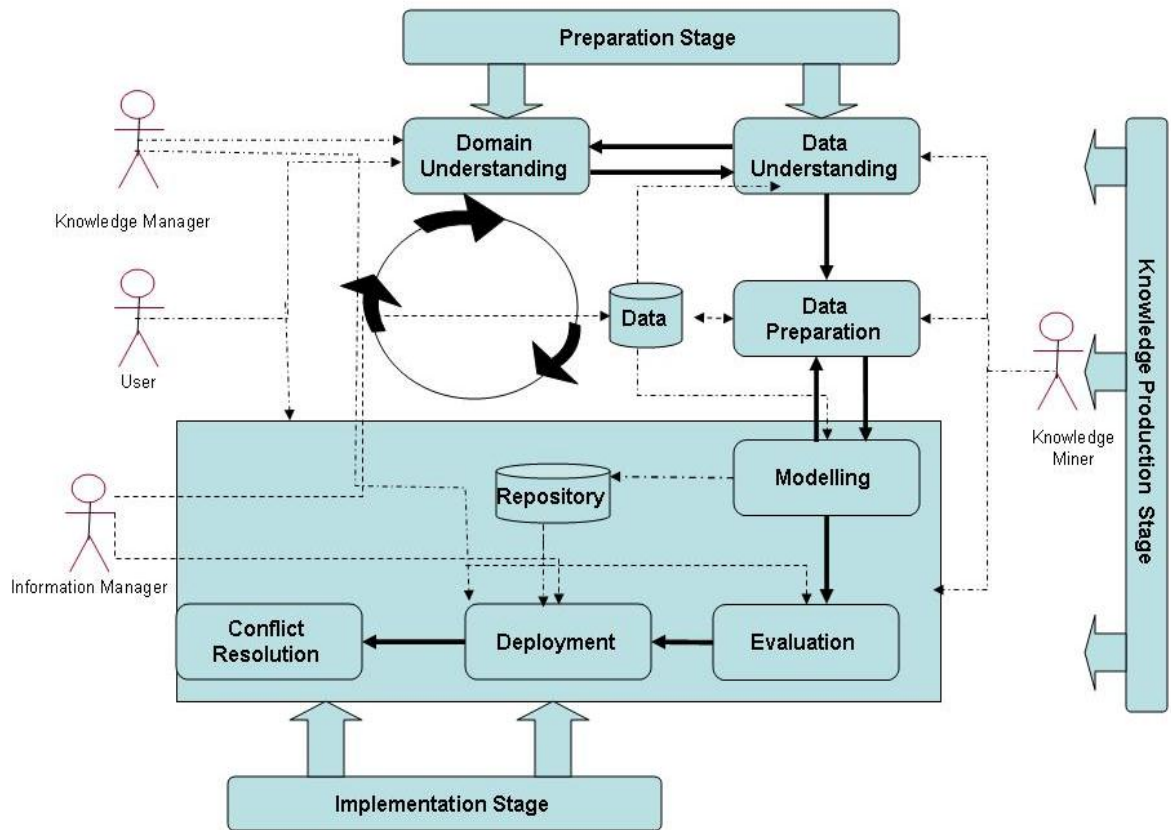


Figure 3: The Process Model of Knowledge Discovery Module

The second stage is called the knowledge production stage, which mainly focuses on modelling, evaluation and deployment. At this stage the models are created and evaluated multiple times to fine-tune the algorithms and parameters used. These functions are performed by knowledge miners with support from knowledge managers and repositories. These are discussed in further detail in the next sections. The third stage, called the implementation stage, updates the fine-tuned knowledge into the expert module and consists of the modelling step through to the conflict resolution step. This stage provides inputs to all the previous stages based on the extracted knowledge, and stage requires interactions from users, knowledge miners, knowledge managers and repositories as shown in Figure 5. Conflict resolution between the existing knowledge and the discovered knowledge requires input from users based on their knowledge of the domain. The development of a process model for the KDM provides a basis for the development of knowledge integrated moderator services. As shown in the KOATING framework in Figure 2 and the process model Figure 3, KDM mainly consists of four components including (1) Knowledge miners (2) Knowledge Manager (3) Information Manager (4) Repository. The next sub-sections discuss the internal structure and the functionality of these components.

4.2.3 Knowledge Miners

The major functionality of a knowledge miner is to extract patterns, relationships and useful knowledge from the operational databases to populate and update the expert modules. In the present context, the performance target of a knowledge miner is to generate knowledge which is as good as or even better than the knowledge associated with human experts in the same situation with the same input datasets. At present, a successful application of data mining generally relies on the experience and expertise of both the data mining expert and the domain expert.

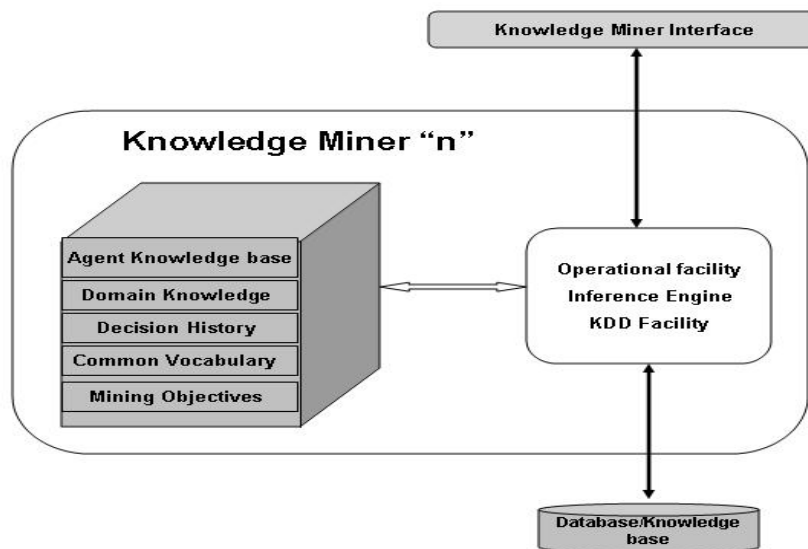


Figure 4: Architecture of the Knowledge Miner

Figure 4 schematically shows the architecture of a knowledge miner, which acts as an intelligent agent. A similar architecture was also developed in the context of shop floor control in manufacturing (Srinivas *et al.* 2008). The knowledge interface manages the communication between the knowledge manager and the knowledge miner. The communication is based on message passing based on a shared ontology; this means that when the knowledge miner receives messages that are represented in a common ontology, the knowledge miner interface converts these messages into local format based on the common ontology. In a similar way, when the knowledge miner sends messages to the knowledge manager, the knowledge interface translates them into a common format first and then sends them to the knowledge manager. A detailed study of ontology models and detailed structure are presented in (Huang & Diao 2008) but the complexity associated with this approach is beyond the scope of this research paper.

The operational facility component is the central control and action part of the knowledge miner. It consists of sub components such as an inference engine, and KDD Facility. The KDD facility is one of the main components of the knowledge miner and it could be implemented in several different ways, e.g. as software code or an expert system. It performs the mining task of discovering causal and interesting relations from the dataset and presents them in a form compatible with the knowledge manager. This component must carry out several functions that are required to carry out the data mining functions discussed in section 3.

The knowledge required to perform the mining task, common vocabulary; knowledge about different users, past decisions, mining objectives and domain knowledge are stored in the agent knowledge base. The data interface component of the system provides a mechanism to extract data from the external data source such as data acquisition systems. In order to perform the mining task, KDD facility uses a variety of tools, techniques and functions. After the knowledge miners complete their tasks, they send the knowledge mining results to the knowledge manager using the knowledge interface and the knowledge miner then terminates.

The functionality of the process model (Figure 3) accomplishes its main objectives of knowledge discovery through the implementation of 4 modules in the KDM. These are described as follows:

1. *Data Acquisition Module*: This module performs three major tasks; firstly it acquires data from the current data acquisition systems/data warehouses of the company. Secondly, it identifies data types such as structured numerical data or unstructured text based data. Thirdly, it selects a subset of the data or focuses on a subset of its attributes based on the objective.
2. *Data Preparation Module*: This module performs all the functions needed to transform the raw data into a form which can be fed into different algorithms. This involves (1) dealing with missing values (2) eliminating noisy data (3) normalizing the database to avoid duplication and eliminating various kinds of logical inconsistencies (4) transforming the data to different values and (5) creating derived attributes in order to reduce the computational burden.
3. *Modeling Module*: based on the goal of data mining, this module performs one or more combinations of the data mining functions such as classification or prediction, etc, using a set of data mining algorithms.
4. *Knowledge Evaluation Module*: this module evaluates the generated knowledge based on certain criteria to make sure that it is novel and useful. It applies statistical techniques to determine the validity of the identified knowledge over the universal set.

The provision of these modules validates the process model discussed earlier as they provide all the functionality required to achieve all the stages mentioned in the process model.

Moderators require two types of update of expert modules (1) regular or periodic update and (2) user requested update. Considering these requirements of the Moderator system, knowledge miners function in two different ways, firstly as periodic knowledge miners and secondly as task oriented knowledge miners. A periodic knowledge miner starts at the beginning of the life cycle of project. Usually a periodic knowledge miner works periodically and generates knowledge based on changes in the database of the corresponding team member. A similar approach was used by (Wang 1997) for group decision making purposes. In contrast, task oriented knowledge miners are activated on a request from the knowledge manager when an expert's role or interest changes or whenever something unusual happens, or the knowledge about an expert is identified to be incorrect. After the knowledge miner has completed the task, the results are sent to the knowledge manager for further processing.

4.2.4 Knowledge Managers

The knowledge manager acts as the heart of the proposed KOATING framework and plays a vital role of manager, mediator and communicator between the different knowledge miners, the information manager, the expert modules and the repository for knowledge sharing. The knowledge manager makes the decisions to create or delete expert modules based on the recommendations from the various constituent elements. The knowledge manager mainly consists of four components: miner interface, knowledge acquisition interface, functional facility and the knowledge manager agent knowledgebase that provides support for localized reasoning. The basic structure is represented in Figure 5.

The knowledge manager checks what special types of knowledge are relevant to specific EMs and consequently which types of databases and files are appropriate to mine in order to update the knowledge content of any particular EM. It activates the responsible knowledge miner to perform the mining task. When mining and update tasks are completed, it stores the knowledge about these activities into the repository for possible future use and transfers the relevant new or updated knowledge into the EMs through the knowledge acquisition interface. The updated EMs can then be enabled, so that they can be used in the ongoing moderation processes. The operational facility provides the mechanisms for knowledge transactions.

The knowledge manager mediates requests from the user and analyzes these requests through its localized knowledge and inference engine and then initiates a knowledge miner to perform the desired task through the miner interface. It communicates the type of knowledge to be

mined i.e. the function that needs to be performed on the data. The information manager, the repository and the knowledge miner are interfaced with the knowledge manager. It also communicates with the knowledge miner to start the task oriented mining process whenever the knowledge about a particular expert is not correct or appropriate.

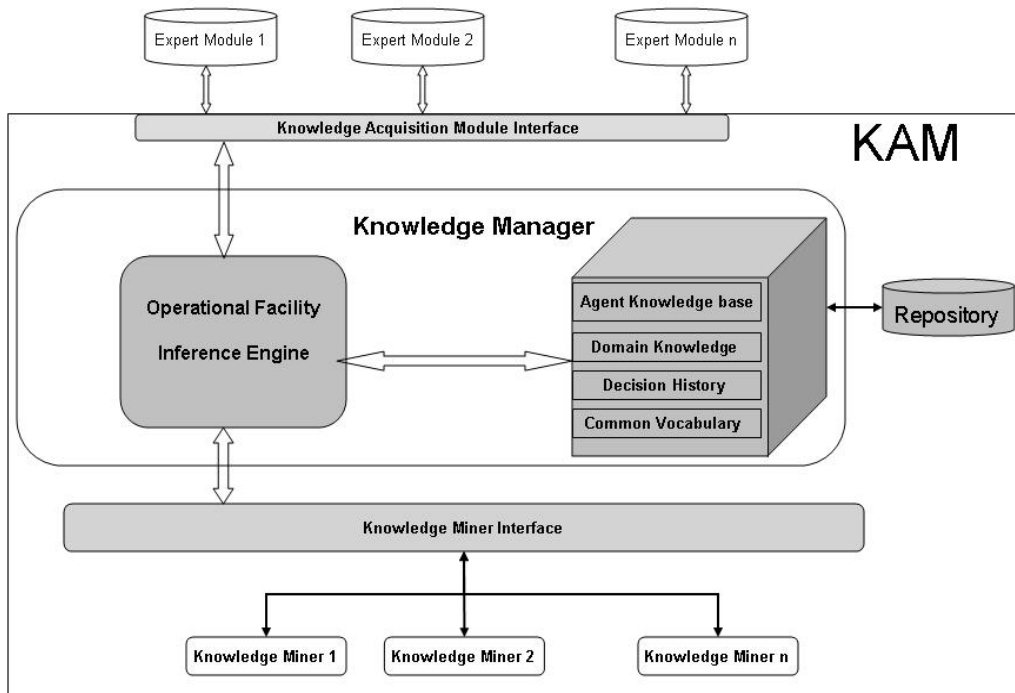


Figure 5: Structure of Knowledge Manager

4.2.5 Knowledge Repository

The repository temporarily stores the mining results and tuning parameters and helps the knowledge manager by providing a set of knowledge required by knowledge miners. When the knowledge manager receives the mining request for information, it first queries the repository to see if relevant knowledge pertaining to the request has already been discovered. If it has not been found, then the knowledge manager initiates the knowledge miner(s) to mine the appropriate knowledge/data bases. In addition, the repository provides the mechanisms for using a common vocabulary. As all the components within the KDM work on the same problem domain and communicate with a set of valid message objects, it is essential for all the components of the KAM to share a common vocabulary. Furthermore, meta-knowledge stored in the repository, such as system configuration, (e.g. various mining parameters of the knowledge miner like number of clusters, similarity criteria, interestingness measures, confidence etc.), can be shared and reused in the future.

4.2.6 Information Manager

The main functionality of the information manager is to share and access the shared data and knowledge bases. It also notifies the knowledge manager about particular contributions or changes to the shared data desired by particular project team members. The information manager also signals the Moderator whenever a change in the project data is recorded in the project database. However this does not affect the knowledge acquisition process, so will not be considered further here. Further study of the information manager are presented in the earlier research by (Harding *et al.* 2003).

4.2.7 Expert Module

Moderators use expert modules to represent each team member and hence the collection of expert modules are populated with all the knowledge about the team members in the collaborative projects. Hence, the collection of EMs provides the moderator with the background knowledge that it requires to support the multidisciplinary team. Based on earlier Moderator concepts (Harding 1996), in the present context, knowledge about individual team members, knowledge of their area of interests, their competencies, and the knowledge about changes that are important to them and actions that need to be taken when such changes occur are stored in the EM. When the Moderator updates an EM, a notification is sent to the relevant team member indicating what knowledge has been updated and also of any conflicting rules found in the EM. In this manner, generated knowledge needs to be verified by the user before it is used in the moderation process. Section 7 provides an illustrative example of how the knowledge can be generated in the form of IF-THEN rules, which can be further updated based on changes in the dataset.

To integrate the proposed framework with the current state-of-art Moderator called Extended Enterprise Manufacturing System Engineering Moderator (EEMSEM) (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005), the next section discusses the Universal Knowledge Moderator (UKM).

5. Knowledge discovery for EEMSEM: Universal Knowledge Moderator (UKM)

The complexity of moderator technology increases when manufacturing projects are large and members are globally distributed in the context of an extended enterprise (EE) or virtual enterprise (VE) (Popplewell & Harding 2004). Manufacturing projects operating within EE and VE environments face additional problems that different information models are likely to be used by different parts of the manufacturing project teams. Supply chain partners inevitably use different vocabularies and terminologies in their work resulting in misunderstandings and confusions. Moreover, the escalating use of web technologies has also accelerated the growth and complexity of manufacturing digital information. The

consequent enormous amounts of heterogeneous data (e.g. structural heterogeneity or semantic heterogeneity) make it increasingly difficult to communicate between different project teams and organizations. In response to this problem and to achieve true information interoperability, (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005) adopted the ontology and semantic web technologies within a Manufacturing System Engineering Moderator (MSEM) to enable semantic interoperability across extended project teams (Poplewell & Harding 2004).

This research builds on previous work on the MSEM and therefore, this section proposes how the MSEM may be extended to provide knowledge discovery for globally distributed and collaborative e-supply chains on the semantic web. The aim of this research is to develop and establish a flexible method for knowledge discovery from semantically heterogeneous data for the moderation of project teams in globally cooperative e-manufacturing chains by integrating the KOATING framework with a state-of-the art Moderator, which is called universal knowledge moderator. This should be able to:

- Analyze and define the specification of a common manufacturing ontology for the manufacturing industry in an ontology server.
- Enable WWW information exchange between partners in cooperative manufacturing chains via common mediated meta-models across different disciplines within engineering project teams through semantic mapping.
- Enable the moderator's KAM to incorporate "learning", updating and reuse elements which exploit knowledge discovery techniques.

Previous research by (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005) has addressed the challenges of different partners within an e-supply chain using different vocabularies and terminologies and therefore the first two of the above objectives are beyond the scope of this paper. This section now sets the KOATING framework into the context of a Moderator to support collaborating working in extended supply chains or similar networks.

5.1 Universal Knowledge Moderator (UKM)

This research integrates the KOATING framework with the functionalities of the MSEM. The previously listed research objectives are discussed in the context of an architecture model for UKM to enable semantic integration of geographically distributed knowledge discovery services. Three main modules have been identified as shown in Figure 6:

- Universal Manufacturing Enterprise Schema (UMES) Module (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005)
- Knowledge Discovery Module

- Moderation Module

5.1.1 Universal Manufacturing Enterprise Schema Module

This architecture of UKM uses a dynamic mediated and shared ontology model for manufacturing enterprises, in order to achieve information interoperation for a UKM within an internationally collaborative environment. The mediated ontology model may involve simple logical reasoning for semantic and syntax mapping. The methods of the UMES are listed and briefly described as follows (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005):

- Analyze and identify the terminology, representation and classification of the manufacturing system for UKM activities in the context of a global high tech industry e.g. PC, IC manufacturing, operating in a globally co-operative e-manufacturing chain.
- Convert the UMES into a web-based ontology language, e.g. Resource Description Framework (RDF) and Web Ontology language (OWL).
- Define a set of semantic mapping rules for automatic reasoning of heterogeneous document structure and data for the UMES in the Metadata Integration Ontology server.

A detailed study of functionalities of UMES and further discussions are presented in (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005) and therefore are not the focus of present research. Therefore only brief details have been included here for completeness.

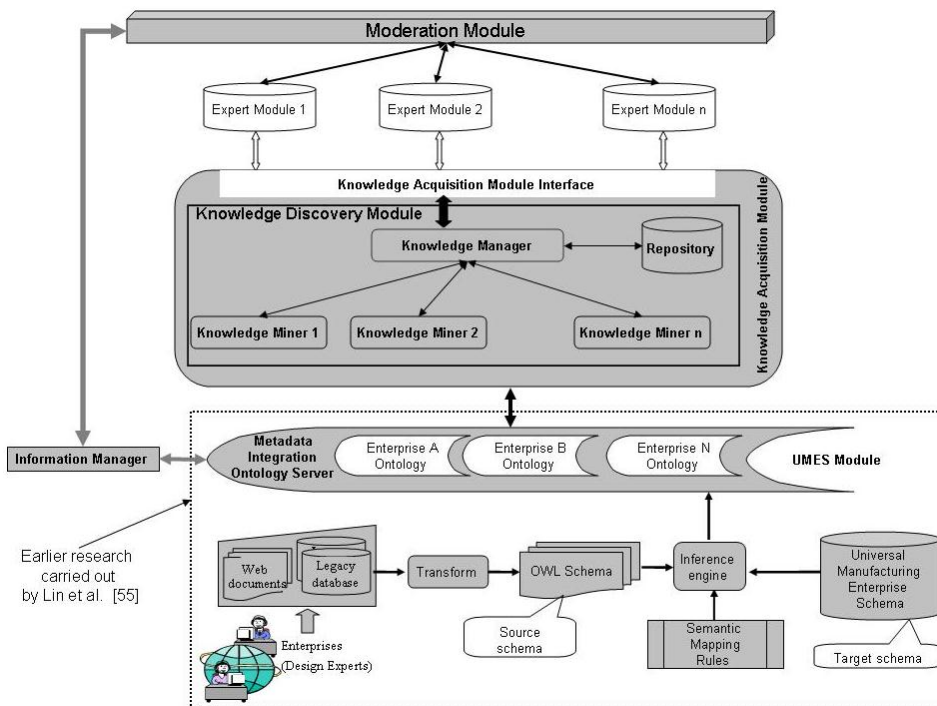


Figure 6: The Architecture of Universal Knowledge Moderator

5.1.2 Knowledge Discovery Module of UKM

The basic structure and functionality of this module remains the same as discussed in section 4. It consists of knowledge miners, knowledge managers and miners interface. This module is interfaced with the UMES module to deal with the semantic heterogeneity of heterogeneous data sources and interoperability issues. Knowledge discovery is based upon the defined and common ontology, so that the KDM generates knowledge in an appropriate language or vocabulary to be used to update the expert module knowledge.

5.1.3 Moderation Module

The major functions of the Moderation Module are to identify when one or more of the partners need to be made aware of a potential opportunities or problems that exist within the project or team. It does so by continuously reviewing the current state of the activities associated with the project and information about recent project decisions, and comparing this with the knowledge it has about team members' interests and requirements as stored within the EMs. The moderation process is activated whenever a project decision is made and this is identified by a change being made to the project information within the shared database. The information manager can notify the UKM of each change. The UKM checks the interests of the team member in its knowledge base by examining the expert modules. If the Moderator finds that one or more team members have an interest in the current type of change, the interested team members are contacted by the UKM and it may remain in dialogue with these team members until conflicts are resolved. Clearly when team members came from different cooperating companies, it is very important that information is understood by different parties and the use of ontologies has previously been covered in detail in (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005).

6. UML Modelling for Knowledge Acquisition Module

This section details the modelling of the KAM using UML. Here the concept of UML grows from analysis to implementation mainly focussing on the knowledge acquisition aspect of the Moderator. In the following section, the use of "system" primarily refers to the KAM. It mainly consists of three stages: system requirement and analysis, system design and system implementation. However, for simplicity of description and considering length of the paper only use case analysis and class diagrams are presented.

6.1. System Requirement and Analysis

This phase of modelling focuses on the set of system requirements, the available resources and the user's desire along with the concept of the system. The behaviour of the KAM (that is

what functionality must be provided by the KAM) is documented in this phase using a use case diagram and use case specification.

A use case diagram illustrates the interaction between the users and system's functions. Here the user is referred to as an actor, which is not part of the system, but represents anything or anyone that interacts or inputs information to or receives information from or both receives and inputs information to and from the system. Use cases represent the functionalities provided by the system. In the context of the Moderator, only the knowledge acquisition aspect has been considered. Therefore in the proposed modelling only those use cases and actors are considered which are essential for understanding the knowledge acquisition aspects of the proposed system. As shown in Figure 7, there are three actors which interact with the system:

- User: represents the person who uses the Moderator system and possesses the domain expertise and has a basic understanding of knowledge discovery.
- Database: represents the past project databases, project summary data and operational data of the enterprises etc., (changes to project information may necessitate changes to the expert module).
- Expert module: is a kind of knowledge base.

Generally, indicating the interactions between actors and the system can help to identify the use cases. The seven use cases that have been identified for the KAM are:

- Create expert module: This use case represents the creation of a new expert module when a new partner joins the collaborative project and agrees with the rules and regulation of the collaboration.
- Edit expert module: this includes the modification or deletion of an expert module.
- Addition of item of interest: Automatic and manual update of an item of interest. Here, the item of interest has been categorized into competencies and objects of interest related to changes.
- Edit item of interest: this can be either the modification or deletion of an item of interest.
- Regular automatic update of rules: this includes the regular update of knowledge in the form of rules in the expert module corresponding to the addition of data to the databases and verification of knowledge from the user.
- User requested update of rules: this identifies a special type of knowledge which is required by the user.

- Manual addition, deletion and modification of rules: this includes the manual addition, deletion or modification of rules in the expert module.

The details of each use case are documented with a flow of events to extract more information from the use case diagram. The flow of events for a use case is a description of events needed to accomplish the required behaviour of the use case. The flow of events for each use case is written in terms of *what* the system should do, not *how* the system does it. The use case specification for the *create expert module* is detailed in Table 2. The specification for the other 3 use cases relating to updating of rules is given in appendix A.

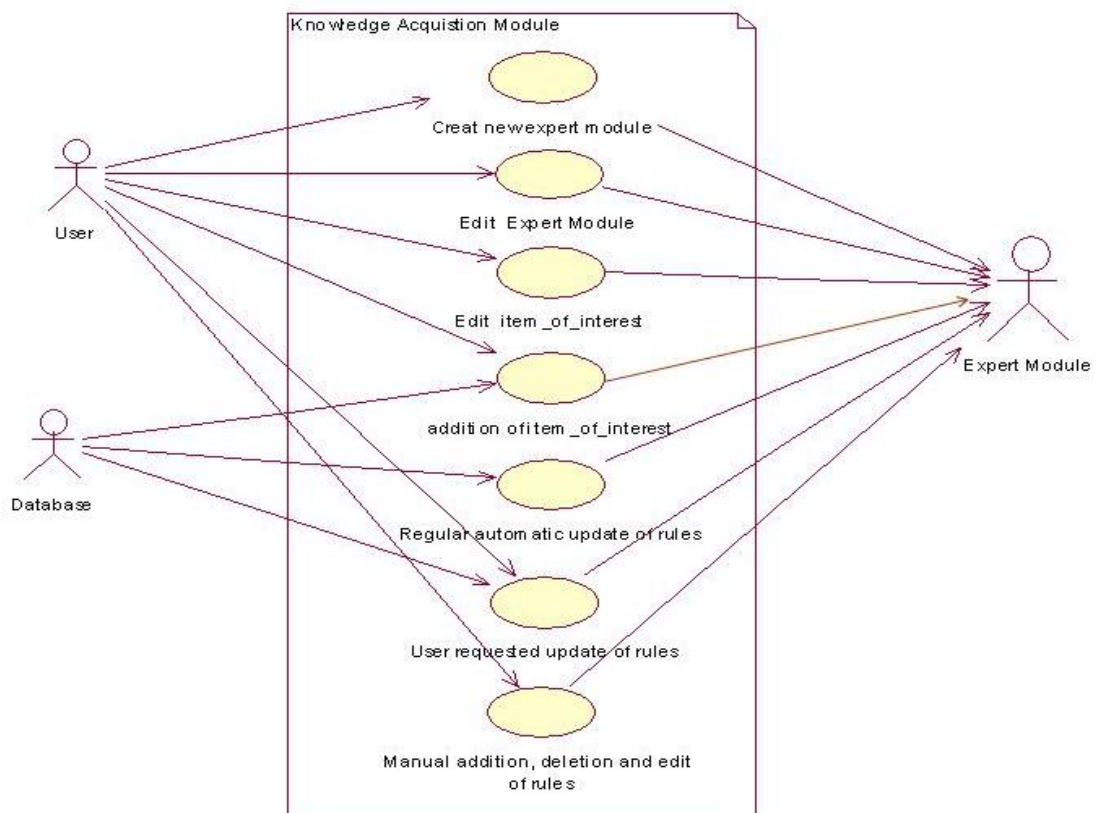


Figure 7: Use case diagram of Knowledge Acquisition Module

Table 2: Use Case specifications of KAM

1. Flow of Event for the Use case <i>Create Expert Module</i> .	
Actors	User and Expert module
Pre-conditions	The collaborative team must validate the creation of an expert module after a new partner joins the collaborative project.
Post-conditions	A partially populated expert module exists. This will require addition of item of interest and regular automatic updates of rules use cases.
Basic Flow	This use case executes when a new team member join the collaboration. User enters his/her password. The system verifies that the password is valid.(E-1.1) and prompts the user to create an expert module's profile(E-1.2). The system prompts the user to enter the

	information required in its profile such as name, contact details and communication method etc., or quit the system.
Alternative flows	E-1.1: An invalid member and wrong password is entered. The user can re-enter its id and password or terminate the use case. E-1.2: An invalid name of expert module is entered or an expert module with that name already exists. The user can re-enter the name or retrieve existing expert module or terminate the use case.

6.2 System Design: Class Diagram

The class diagram is a graphical view of the static structure of the model. A class diagram shows a set of classes, interfaces, collaborations and their relationships (such as dependency, generalization, and association). The UML representation of a class is a rectangle containing

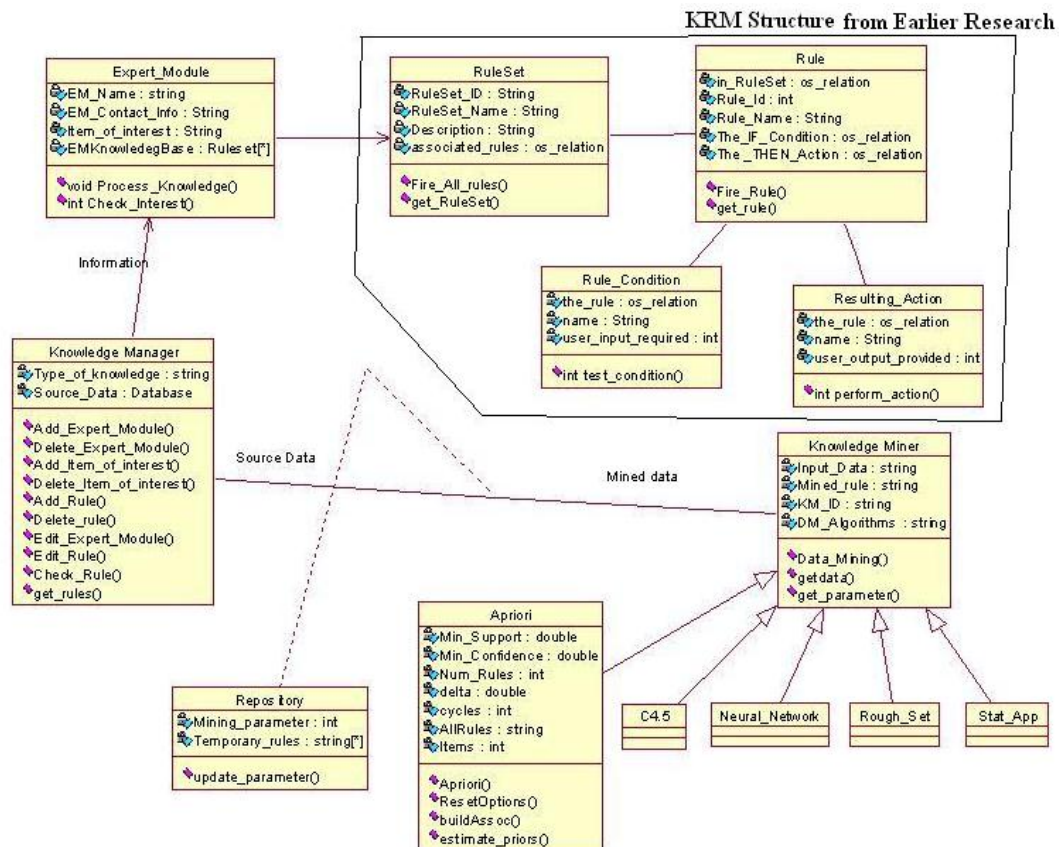


Figure 8: Class diagram of the proposed system.

three compartments stacked vertically, as shown in Figure 8. The top compartment shows the class's name. The middle compartment lists the class's attributes. The bottom compartment lists the class's operations. When drawing a class element on a class diagram, one must use the top compartment, but the bottom two compartments are optional. The class diagram shows that how the different entities of the system relate to each other, their internal structure and their relationships. Figure 8 shows the high level view of the class diagram. It mainly

consists of Expert_Module, RuleSet, Rule, Rule_Condition, Resulting_Action, Knowledge Miner, Knowledge Manager and Repository. Relationships between classes are represented by lines and labels, arrowheads and notation. The upper structure in the boundary line shows the KRM structure used in (Harding 1996, Harding *et al.* 2003, Lin & Harding 2003, Lin *et al.* 2004), therefore, the details are beyond the scope of this paper. The class name attributes and operations are represented in the diagram. The relationship between the Expert_Module class and Knowledge Manager Class is uni-directional association. It shows that two classes are related but only one class knows that relationship exists. The second type of relationship is an association relationship and indicates the long term relationship between the classes. An association relationship exists between Knowledge Manager and Knowledge Miner. The knowledge miner is equipped with a variety of algorithms classes such as Apriori, C4.5, Neural_Network, Rough_Set and Stat_App. These classes are connected with Knowledge_Minor class with generalization relationship. Generalization provides the capability to create a superclass that encapsulates structure and behaviour common to several classes. These classes are examined for commonality of structure and behaviour. The repository class is also associated with the knowledge miner and Knowledge Manager with an association relationship and this is represented by a dotted line.

7. An Illustrative Example

The e-supply chain is the communication and operations backbone of a virtual network that links suppliers, business partners and customers together as one cohesive collaborative entity. A virtually enabled supply chain network is a series of value adding processes or stages owned by one or more enterprises, starting with a material or information supplier and ending with consumers. An open fast communication mechanism is essential for the companies entering into supply chain network activities, allowing its members to jointly forecast, develop, produce, synchronize and deliver their product or services, and anticipate dynamic customer requirements. A typical example of an e-supply chain is schematically shown in Figure 9.

This example assumes that a VE has been created by a leading UK automobile manufacturer (X) which operates in UK and Europe. The automobile enterprise “X” wishes to contract a supplier to produce 2 parts (P_a and P_b) in order to build a new prototype in the UK. However, no local manufacturer can be found with enough resource to meet the demand. The solution adopted was to form a collaborative virtual e-chain with 4 enterprises including suppliers and manufacturers. 4 EMs are created, each containing knowledge about one of the enterprises in the e-supply-chain.

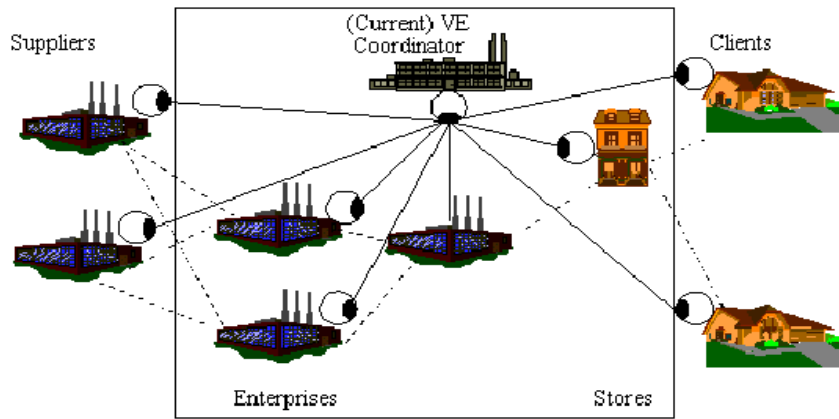


Figure 9: A generic view of Virtual e- Supply Chain

Supply chain operations start when an order for the two products (P_a and P_b) has been sent to the 3 suppliers (S_1 , S_2 , and S_3) through a web interface with requirements such as lead-time, quantity, and product type. Two different parts P_{a1} and P_{a2} , and P_{b1} and P_{b2} are assembled to produce the products P_a and P_b respectively. For product P_a supplier S_1 can only provide P_{a1} , P_{a2} has to be ordered from S_3 . Similarly, for product P_b , S_2 can only provide P_{b1} , P_{b2} has to be ordered from S_3 . On the other hand, S_3 can supply all the parts P_{a1} , P_{a2} , P_{b1} and P_{b2} to produce P_a and P_b . All the suppliers are able to produce different products with different lead times, quantities and prices. All the suppliers use different terminologies for the same context. For example S_1 uses lead time, S_2 uses due date, and S_3 uses delivery time for the same context of delivering the product to the customer. Similarly, S_1 uses quantity, S_2 uses number of products, and S_3 uses number of pieces for the same context. The product information for each supplier is given below in Table 2, which contains the data that the supplier wishes to share with the Coordinator X.

Therefore, a semantic heterogeneity for the product information exists, where different suppliers use different terminologies for the same context. The UMES module of the UKM takes care of this semantic heterogeneity by developing a common and agreed ontology. For example in the present context an ontology can be developed for lead time, due date and delivery time. More details about the development of common/mediated ontology are discussed in (Lin & Harding 2003, Lin *et al.* 2004, Lin & Harding 2007, Lin *et al.* 2005).

There are several combinations of price/quantity/time over several ranges of values which should be considered to ensure that the orders are placed with the most cost effective suppliers. All the EMs must contain the knowledge in the form of rules stating that which combinations of supplier are cheapest. This expert module also contains the items of interest for each supplier such as orders detail, quantity, lead time, due date and price etc. The focus

of this research is limited to the knowledge acquisition aspect therefore this example only shows that how knowledge can be generated for the EM associated with company X from shared database of suppliers. In the present context, the data presented in Table 3 is used as shared data. There is a need to transform these data into knowledge in order to obtain the effective combinations of quantity, price and due date. This knowledge can then be stored in the expert module for further Moderator activity.

Table 3: Supplier details and their capabilities for producing products

Product information for supplier S ₁		
Lead Time	Quantity	Unit price (£)
1-4	1-30	48.5
5-8	1-100	47.5
9-12	1-150	45
Product information for supplier S ₂		
Due date	Number of products	Cost per product
1-3	1-10	49
4-7	1-70	48
8-12	1-150	44.5
Product information for supplier S ₃		
Delivery Time	Number of pieces	Selling price per piece
1-3	1-50	50
4 -8	1-100	46.5
9 – 12	1-150	46

In the traditional system, the knowledge in the form of IF-THEN rules is generated based on human experience and expertise by interviewing different suppliers and looking at their capabilities. In the proposed system, the UKM can help to semi-automatically discover the knowledge in the form of rules providing it has up to date information about current price/quantity/delivery combinations that are best for each supplier. In order to discover the knowledge, knowledge manager instructs the knowledge miner to find patterns, relationships and rules within the shared data associated with supply chain agents considering minimal price/product. The mining engine component of knowledge miner finds rules for minimum price. In the present context, IF-THEN rules were discovered using techniques such as decision tree. For example for the given data set the discovered rule may be as follows:

- 1 IF (LeadTime>8) THEN Select Supplier S₁ & S₃.
- 2 IF (Quantity ≤ 30) AND (LeadTime≤3) THEN Select Supplier S₁ & S₃.
- 3 IF (Quantity ≤ 100) AND (4≤LeadTime≤8) THEN Select Supplier S₃.
- 4 IF (30<Quantity ≤ 50) AND (LeadTime≤8) THEN Select Supplier S₃.
- 5 IF (LeadTime>7) THEN Select Supplier S₂ & S₃.
- 6 IF (Quantity ≤ 10) AND (LeadTime≤3) THEN Select Supplier S₂ & S₃.
- 7 IF (Quantity ≤ 100) AND (4≤LeadTime≤8) THEN Select Supplier S₃.
- 8 IF (10<Quantity ≤ 50) AND (LeadTime≤8) THEN Select Supplier S₃

These rules are stored in the expert module of Company X and need to be checked whenever order is raised. Similarly, expert module of suppliers S_1 , S_2 and S_3 must be populated with knowledge about its area of interests. For example, for S_1 the knowledge in the EM would be:

IF (Lead time ≤ 3) AND (Quantity ≤ 30), THEN notify of order.

IF ($9 \leq$ Lead time ≤ 12) AND (Quantity ≤ 150) THEN notify of order in competition with S_3

Similarly for S_2 , The knowledge in the corresponding EM would be.

IF (Lead time ≤ 3) AND (Quantity ≤ 10), THEN notify of order.

IF ($8 \leq$ Lead time) AND (Quantity ≤ 150) THEN notify of order in competition with S_3

For S_3 , Knowledge in EM would be

IF (Lead time ≥ 4) AND (Quantity ≤ 100), THEN notify of order.

IF ($8 \leq$ Lead time ≤ 12) AND (Quantity ≤ 150) THEN notify of order in competition with S_1 & S_3 .

These are very simple rules and could be generated manually, but, consider a situation where several other qualitative and quantitative attributes such as supplier reputation, quality of product, physical location etc, are considered. In the present context scenario, a decision tree algorithm can be used by the data mining engine of a knowledge miner to generate these rules. These rules need to be updated whenever the data changes. It means that whenever the supplier changes their capability, the dataset will change. For example, after a few orders supplier S_1 may have improved their way of production and be capable of producing more products at reduced cost. Based on this fact, they have changed the information related to the product and the combination of lead-time, quantity and price. Changes in the dataset will trigger the knowledge manager to prompt a message to the knowledge miner to initiate the mining task. The knowledge miner will therefore apply its data mining algorithm to extract new rules, patterns and relationships, and thereby generate and update new knowledge within the expert modules. At this stage, if it finds that there is a conflicting rule, it will trigger a message to the user to resolve this conflict based on its domain knowledge.

Now when the e-supply chain is operating and production is going on, the UKM discovers a delay in the delivery of part P_{a1} by supplier S_1 . Details of each delivery and order are stored in the company's databases so regular updates of UKM knowledge could identify that there is an error in the usual rules as one of the suppliers has been delivering "later" than the quoted date. This delay is critical to the lead time requested by the customer. This delay can be a hindrance to the successful completion of the order by the suppliers involved. Therefore, UKM must notify this delay to relevant supply chain agent responsible for Supplier S_3 and Company X. This alert can be sent in the form of an e-mail about the problem occurrence.

In this case, to overcome this problem UKM may recommend company X to:

- Send the order to an alternative supplier or;
- discuss the quality of deliveries with the existing supplier and negotiate new terms or;

- Increase the working hours of corresponding supplier.

There might be situation where the manufacturing enterprise “X” needs to collaborate with another supplier 4. In this case, chain will be increased based on an agreement of the entire supply chain. In this case, knowledge acquisition module will generate a new expert module corresponding to supplier 4. Above mentioned example is just an instance of several activities involved in the operation of virtual enterprise supply chain.

8. Conclusion and Discussions

In earlier research projects Moderator technology, in the form of knowledge based software support systems, has been successfully demonstrated in both the product and manufacturing system design domains. However, knowledge acquisition, learning and updating of knowledge has not previously been studied fully. Therefore this paper presents a KOATING framework to provide semi- automated knowledge acquisition for moderator technology in collaborative projects to update the expert modules. This enables the reuse of discovered knowledge from operational databases within collaborative projects and facilitates the exploitation of the right knowledge at the right time in the right context.

In addition, a Universal Knowledge Moderator (UKM) system, consisting of a Universal Manufacturing Enterprise Schema Module, Knowledge Discovery Module and Moderation Module, has been proposed to improve the moderation activities. The Universal Manufacturing Enterprise Schema Module enhances the interoperability of the chain on the semantic web. This shows how the KOATING framework can be integrated with the state-of-art Moderator. The proposed KOATING framework will facilitate the decision making and moderation process by incorporating the “learning” and “knowledge reuse” element within moderators. This also facilitates semi-automated knowledge creation, updating and retrieving capability, and if necessary transforming the identified patterns and /or models to alternative representations and resolving conflict or contradictions with previously extracted knowledge. However, the application of Semantic Web technologies and tools require considerable technical expertise, and are thus not well suited for users outside the field of computer science. This makes it hard for domain experts and ontology engineers to work together on e-manufacturing tasks. One of the major challenges for the UKM research is to facilitate interaction and operation for mass collaboration and knowledge sharing.

Appendix A: Use case specification for use cases used in figure 8.

5. Flow of event for the use case automatic regular update of rules	
Actors	Database, user and expert module
Pre-conditions	System has access to relevant database and corresponding expert module exists
Post-conditions	User must be notified to edit, verify and resolve any conflict of rules. The edit of rules use case must execute after the execution of this use case
Basic Flow	This use case begins when new data is added to the data base. The data acquisition system passes this message to KAM. KAM has access to the database (E-5.1). It verifies the type of data and correspondingly activates the knowledge miner within the system to perform the knowledge discovery process (E-5.2). A set of rules are generated as a result of the knowledge discovery process (E-5.3). The rules are compared with the existing rules to check for a conflicting rule (E-5.4). If there is no conflicting rule update the expert module. This use case terminates when the rules have been updated. .
Alternative flows	E-5.1: KAM is unable to access the database; user must be notified of this problem. E-5.2: Unknown type of data identified, user needs to be informed. E-5.3: KAM is unable to capture knowledge in the form of rules; in which case derived knowledge should be notified to user for manual entry of knowledge in the form of rules. E-5.4: There exists a conflicting rule; user should be notified of this conflict and initiate the edit of rules use case.
6. Flow of event for the use case user requested update of rules	
Actors	User, database and expert module
Pre-conditions	System has access to relevant database and corresponding expert module exists
Post-conditions	User must be notified of the discovered rules and edit rule use case executes after this use case.
Basic Flow	This use case is initiated by the user, when s/he needs any special type of knowledge. In this case, it prompts the system to the database and specifies the kind of knowledge to be mined (E-6.1). The system identifies the data type and applies the knowledge discovery process to discover the knowledge in the form of rules (E-6.2). A comparison is made with the existing rules for any conflict (E-6.3). Rules are added to the expert module (E-6.4) This use case terminates after notifying the user of discovered knowledge.
Alternative flows	E-6.1: User selected a wrong database, it can re-select the database or quit. E-6.2: Unknown data type identified, user need to be informed. E-6.3: There exists a conflicting rule; user should be notified of this conflict and initiate the edit of rules use case. E-6.4: System is unable to capture knowledge in the form of rules; in which case derived knowledge should be notified to the user for manual entry of knowledge in the form of rules.
7. Flow of event for the use case manual addition, deletion and modification of rules	
Actors	User and expert module
Pre-conditions	Create expert module and automatic regular update of rules must have executed before this use case executes.
Post-conditions	Expert module has updated list of rules
Basic Flow	This use case begins when the new rules are automatically discovered and need verification from the user for its authenticity. Identified conflict or rules are prompted to users to resolve them (E-7.1). The user resolves the conflict by performing activity such as ADD, DELETE, MODIFY and UPDATE of rules or QUIT the system. If the activity selected is ADD RULE, system adds a new rule to expert module (E-7.2). If the activity selected is DELETE RULE, system deletes the chosen rule from the expert module (E-7.3). If the activity selected is MODIFY RULE, system facilitates the user to modify the rule based on domain knowledge (E-7.4). If the activity selected is UPDATE RULE, system updates the expert module with updated rules.

	If the activity selected is QUIT, the use case ends.
Alternative flows	E-7.1: No conflicting rule found, system updates the expert module E-7.2: An invalid rule is added, the user can delete or modify that rule and re-enter the desired rule or terminate the use case. E-7.3: An invalid rule is selected, the user can re-select the desired rule to delete or terminate the use case. E-7.4: The rule chosen for modification is invalid, the user can re-select the desired rule and modify.

References

- Ahn, H.J., Lee, H.J., Cho, K. & Park, S.J. 2005, "Utilizing knowledge context in virtual collaborative work", *Decision Support Systems*, vol. 39, no. 4, pp. 563-582.
- Browne, J. & Zhang, J. 1999, "Extended and virtual enterprise-similarities and differences", *International Journal of Agile Management Systems*, vol. 1, pp. 30-36.
- Camarinha-Matos, L. & Afsarmanesh, H. 2007a, "A comprehensive modeling framework for collaborative networked organizations", *Journal of Intelligent Manufacturing*, vol. 18, no. 5, pp. 529-542.
- Camarinha-Matos, L.M. & Afsarmanesh, H. 2007b, "A framework for virtual organization creation in a breeding environment", *Annual Reviews in Control*, vol. 31, no. 1, pp. 119-135.
- Choudhary, A.K. 2009, *Knowledge Discovery for Moderating Collaborative Projects. PhD Thesis, Loughborough University, UK.*
- Choudhary, A., Harding, J. & Tiwari, M. 2008, "Data mining in manufacturing: a review based on the kind of knowledge", *Journal of Intelligent Manufacturing*, vol. 20, no. 5, pp. 501-521.
- Choudhary, A.K., Harding, J.A. & Popplewell, K. 2006, "Knowledge Discovery For Moderating Collaborative Projects", *IEEE International Conference on Industrial Informatics*, pp. 519.
- Cil, I., Alpturk, O. & Yazgan, H.R. 2005, "A new collaborative system framework based on a multiple perspective approach: InteliTeam", *Decision Support Systems*, vol. 39, no. 4, pp. 619-641.
- ECOLEAD 2008, June 25, 2008-last update, *European Collaborative Networked Organisations Leadership Initiative*. Available: <http://ecolead.vtt.fi/> [2008, 7/14] .
- ECOLEAD 2005, *ECOLEAD Project Public Deliverable 21.1: Characterization of key components, features, and operating principles of the virtual breeding environment*, .
- Fayyad, U.M., Piatetsky-Shapiro, G. & Smyth, P. 1996, "From data mining to knowledge discovery: an overview" in *Advances in knowledge discovery and data mining* American Association for Artificial Intelligence, , pp. 1-34.
- Frécon, E. and Nöu, A. A. 1998, "Building distributed virtual environments to support collaborative work", *Proceedings of the ACM Symposium on Virtual Reality Software and Technology* ACM, Taipei, Taiwan, pp. 105-113.
- Gaines, B.R., Norrie, D.H. & Lapsley, A.Z. 1995, *Mediator: an intelligent information system supporting the virtual manufacturing enterprise*.
- Harding, J.A. 1996, *A knowledge representation model to support concurrent engineering teamwork.*, Loughborough University, UK.
- Harding, J.A. & Popplewell, K. 1996, "Driving concurrency in a distributed concurrent engineering project team: a specification for an Engineering Moderator", *International Journal of Production Research*, vol. 34, no. 3, pp. 841.
- Harding, J.A., Popplewell, K. & Cook, D. 2003, "Manufacturing systems engineering moderator: an aid for multidiscipline project teams.", *International Journal of Production Research*, vol. 41, no. 9, pp. 1973-1986.
- Harding, J.A., Popplewell, K. & Lin, H.K. 2007, "A Generation of Moderators from Single Product to Global E-Supply" in *Knowledge and Technology Management in Virtual Organizations: Issues, Trends, Opportunities and Solutions*, eds. G.D. Putnik & M.M. Cunha, IGI Publishing, Portugal, pp. Chapter 5.
- Harding, J.A., Shahbaz, M., Srinivas & Kusiak, A. 2006, "Data Mining in Manufacturing: A Review", *American Society of Mechanical Engineers (ASME): Journal of Manufacturing Science and Engineering*, vol. 128, no. 4, pp. 969.
- Hicks, B.J., Culley, S.J., Allen, R.D. & Mullineux, G. 2002, "A framework for the requirements of capturing, storing and reusing information and knowledge in engineering design", *International Journal of Information Management*, vol. 22, no. 4, pp. 263-280.

- Holsheimer, M. 1999, *Data mining by business users: integrating data mining in business processes*, ACM, San Diego, California, United States.
- Huang, G.Q., Huang, J. & Mak, K.L. 2000, "Agent-based workflow management in collaborative product development on the Internet", *Computer-Aided Design*, vol. 32, no. 2, pp. 133-144.
- Huang, N. & Diao, S. 2008, "Ontology-based enterprise knowledge integration", *Robotics and Computer-Integrated Manufacturing*, vol. 24, no. 4, pp. 562-571.
- Ikujiro, N. & Takeuchi, H. 1995, *The Knowledge-Creating Company*, Oxford University Press, New York.
- Katzy, B.R. & Obozinski, V. 1998, "Value System Redesign", *Database and Expert Systems Applications, International Workshop on*, vol. 0, pp. 650.
- Katzy, B.R. & Crowston, K. "Competency rallying for technical innovation—The case of the Virtuelle Fabrik", *Technovation*, Volume 28, Issue 10, October 2008, Pages 679-692.
- Katzy, B.R. & Dissel, M. 2001, "A toolset for building the virtual enterprise", *Journal of Intelligent Manufacturing*, vol. 12, no. 2, pp. 121-131.
- Lastra, J.L.M. & Delamer, M. 2006, "Semantic web services in factory automation: fundamental insights and research roadmap", *Industrial Informatics, IEEE Transactions on*, vol. 2, no. 1, pp. 1-11.
- Li, E.Y. & Lai, H. 2005, "Collaborative work and knowledge management in electronic business", *Decision Support Systems*, vol. 39, no. 4, pp. 545-547.
- Li, W.D. & Qiu, Z.M. 2006, "State-of-the-art technologies and methodologies for collaborative product development systems", *International Journal of Production Research*, vol. 44, no. 13, pp. 2525.
- Lin, H., Harding, J. & Teoh, P. 2005, "An inter-enterprise semantic web system to support information autonomy and conflict moderation", *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 219, no. 12, pp. 903-911.
- Lin, H.K. 2004, *Manufacturing System Engineering Ontology Model for Global Extended Project Team*, Loughborough University.
- Lin, H.K. & Harding, J.A. 2007, "A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration", *Computers in Industry*, vol. 58, no. 5, pp. 428-437.
- Lin, H.K. & Harding, J.A. 2003, "An ontology driven manufacturing system engineering moderator for global virtual enterprise teams", *Advances in manufacturing technology XVI. Proceedings of the International Conference on Manufacturing Research*, eds. Y. Qin & N.P. Juster, Professional Engineering Publication, London, pp. 365.
- Lin, H.-., Harding, J.A. & Shahbaz, M. 2004, "Manufacturing system engineering ontology for semantic interoperability across extended project teams", *International Journal of Production Research*, vol. 42, no. 24, pp. 5099.
- Misono, S., Koide, S., Shimada, N., Kawamura, M. & Nagano, S. 2005, *Distributed collaborative decision support system for rocket launch operation*.
- MISSION Consortium 2001, "Final Report from MISSION Project—Deliverable D24.K" in *Modelling and Simulation Environments for Design, Planning and Operation of Globally Distributed Enterprises (IMS/ESPRIT29656)*, eds. K. Popplewell, J.A. Harding & M. Rabe,.
- Mitra, S., Pal, S.K. & Mitra, P. 2002, "Data mining in soft computing framework: a survey", *Neural Networks, IEEE Transactions on*, vol. 13, no. 1, pp. 3-14.
- MOSES 1995, *Model Oriented Simultaneous Engineering System (EPSRC Project Number GR/H24273)*. Available: <http://leva.leeds.ac.uk/www/amoses/amoses.html> [2008, 7/2] .
- Numata, J., Hane, K., Bangyu Lei & Iwashita, Y. 1997, *Knowledge discovery and sharing in an information system*.
- Panteli, N. & Sockalingam, S. 2005, "Trust and conflict within virtual inter-organizational alliances: a framework for facilitating knowledge sharing", *Decision Support Systems*, vol. 39, no. 4, pp. 599-617.

- Pechoucek, M., Vokrinek, J. & Becvar, P. 2005, "ExPlanTech: multiagent support for manufacturing decision making", *Intelligent Systems, IEEE*, vol. 20, no. 1, pp. 67-74.
- Piatetsky-Shapiro, G. 1991, "Knowledge discovery in real databases: A report on the IJCAI-89 Workshop", *AI Mag.*, vol. 11, no. 5, pp. 68-70.
- Popplewell, K. & Harding, J. 1995, "Engineering Moderation - Supporting Concurrency in Engineering Using Hybrid Knowledge Representation", *Proceedings of IFIP Working Group 5 - 7 Working Conference on Managing Concurrent Manufacturing to Improve Industrial Performance, International Federation for Information Processing, Managing Concurrent Manufacturing to Improve Industrial Performance*, Seattle, USA, pp. 347-357.
- Popplewell, K. & Harding, J.A. 2004, "Impact of Simulation and Moderation through the Virtual Enterprise Life Cycle", *Proceedings of the 11th ASIM Dedicated Conference*, eds. K. Mertins & M. Rabe, IPK Berlin, Germany, pp. 299.
- Ramesh, B. & Tiwana, A. 1999, "Supporting Collaborative Process Knowledge Management in New Product Development Teams", *Decision Support Systems*, vol. 27, no. 1-2, pp. 213-235.
- Sánchez, N.G., Apolinar, D., Zubiaga, G., Atahualpa, J., González, I. & Molina, A. 2005, "Virtual Breeding Environment: A First Approach to Understanding Working and Sharing Principles", *In Proceedings of InterOp-ESA '05* Springer, .
- Scotney, B. & McClean, S. 2004, *Knowledge discovery from databases on the semantic Web*, in the 16th International Conference on Scientific and Statistical Database Management (SSDBM'04), Santorini Island, Greece, June 21-June 23.
- Shafiei, F. & Sundaram, D. 2004, *Multi-enterprise collaborative enterprise resource planning and decision support systems*. HICS, in the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 8 - Volume 8 Page: 80228.
- Shearer, C. 2005, "The CRISP-DM model: the new blueprint for data mining", *Journal of Data Warehousing*, vol. 5, pp. 13-22.
- Shum, S.B., Roure, D.D., Eisenstadt, M., Shadbolt, N. & Tate, A. 2002, "CoAKTinG: Collaborative Advanced Technologies in the Grid", *Proc. Second Workshop on Advanced Collaborative Environments*, Edinburgh, pp. 1-8.
- Slade, A.J. & Bokma, A.F. 2002, *Ontologies within extended enterprises*, in the Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)- Volume 1 - page: 39.1
- Srinivas & Harding, J.A. 2008, "A data mining integrated architecture for shop floor control", *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 222, no. 5, pp. 605-624.
- Skjoett-Larsen, Tage., Thernøe, Christian. and Andresen, Claus. 2003, "Supply chain collaboration: Theoretical perspectives and empirical evidence", *International Journal of Physical Distribution & Logistics Management*, vol. 33, pp. 531-549(19).
- Wang, H. 1997, "Intelligent Agent-Assisted Decision Support System: Integration of Knowledge of Discovery, Knowledge Analysis, and Group Decision Support", *Expert System with Application*, vol. 12, no. 3, pp. 323-335.
- Yue-jie Wen, Ling Tian & Bing-shu Tong 2005, *Knowledge discovery based on Web robots in collaborative design*, in the Proceedings of the The Fifth International Conference on Computer and Information Technology Pages: 1040 – 1044.
- Zhou, L. & Nagi, R. 2002, "Design of distributed information systems for agile manufacturing virtual enterprises using CORBA and STEP standards", *Journal of manufacturing systems*, vol. 21, no. 1, pp. 14-31.